LAFERTGROUP
Member of Sumitomo Drive Technologies

LAFERT
CANOPEN MANUAL

Release 1.12
Date 01/05/2025

Lafert S.p.A.
J.F. Kennedy, 43 – I-30027 San Donà di Piave (Venice), Italy | Phone +39 / 0421 229 611 | Fax +39 / 0421 222 908 | lafert.info@shi-g.com | **www.lafert.com**

This page has been intentionally left blank

Lafert S.p.A.
J.F. Kennedy, 43 – I-30027 San Donà di Piave (Venice), Italy | Phone +39 / 0421 229 611 | Fax +39 / 0421 222 908 | lafert.info@shi-g.com | **www.lafert.com**

INDEX:

Lafert S.p.A.
J.F. Kennedy, 43 – I-30027 San Donà di Piave (Venice), Italy | Phone +39 / 0421 229 611 | Fax +39 / 0421 222 908 | lafert.info@shi-g.com | www.lafert.com

Lafert S.p.A.
J.F. Kennedy, 43 – I-30027 San Donà di Piave (Venice), Italy | Phone +39 / 0421 229 611 | Fax +39 / 0421 222 908 | lafert.info@shi-g.com | **www.lafert.com**

FIGURES:

TABLES:

Lafert S.p.A.
J.F. Kennedy, 43 – I-30027 San Donà di Piave (Venice), Italy | Phone +39 / 0421 229 611 | Fax +39 / 0421 222 908 | lafert.info@shi-g.com | www.lafert.com

REFERENCE DOCUMENTS:

- *Lafert User Guide*
- *CiA 301 (310_1v01010005_cor.pdf)*
- *CiA 402 (CiA® 402 Draft Standard Proposal.pdf)*

TERMS AND ABBREVIATIONS:

CAN	Controller Area Network.
CiA	CAN in Automation.
COB	Communication OBject, transport unit in a CAN network.
COB-ID	Communication OBject Identifier.
DS301	Profile 301 standardizes the CANopen communication profile.
DSP402	Profile 402 standardizes the CANopen device profile for drives.
EDS	Electronic Data Sheet.
EMCY	Emergency Object.
EMC	Electromagnetic compatibility.
FSM	Finite State Machine
HMI	Human Machine Interface.
I/O	Input/output.
LSB	Least significant bit/byte.
LSD	Lafert Servo Drives.
MASTER	It is a device that controls and communicates with drive.
MSB	Most significant bit/byte.
MSM	Macro State Machine of Lafert Servo Drives.
NMT	Network Management.
IdNode	Node address assigned to a device on the network.
OD	Object dictionary.
PDO	Process Data Object.
PDS	Power Drive System.
REG	Register.
RO	Denotes read-only access.
RPDO	Receive (incoming) PDO
RTR	Remote Transmission Request
RW	Denotes read/write access.
RX	Messages sent by Main Control Board and received by Drive.
SDO	Service Data Object.
STO	Safe Torque Off
TX	Messages sent by Drive and received by Main Control Box
TPDO	Transmit (outgoing) PDO

FIRMWARE AND MANUAL RELEASED

This table shows the correlation between firmware and CANopen Manual.

Lafert Servo Drive	Firmware Released	CANopen Manual	Notes
SMARTRIS	2.4.2	1.12	
SMARTRIS COMPACT	2.4.2	1.12	

1. | PURPOSE OF MANUAL

PURPOSE OF THE MANUAL

This operating guide provides information for safe installation and commissioning of the Drive.

Read and follow the instructions to use the Lafert Drive safely and professionally and pay attention to the safety instructions and general warnings.
Always keep this operating guide available with the Drive.

This operating guide provides information for safe installation and commissioning of the Lafert Drive:read carefully the entire guide before installing and using the equipment.

Caution

The operating guide is intended for use by qualified personnel.
THIS MANUAL IS ONLY FOR THE CANOPEN ON LAFERT DRIVE

This guide is delivered subject to the following conditions and restrictions:

- This guide contains proprietary information belonging to Lafert Spa.
- Such information is supplied solely for the purpose of assisting users of Lafert servo drives in implementing CANopen networking.
- The text and graphics included in this manual are for the purpose of illustration and reference only. The specifications on which they are based are subject to change without notice.
- Information in this document is subject to change without notice. Corporate and individual names and data used in examples herein are fictitious unless otherwise noted.

This manual is regularly reviewed and updated. All suggestions for improvement are welcome.

WARNING SAFETY INFORMATION

In order to achieve the optimum, safe operation of the Drive, it is imperative that you implement the safety procedures included in this installation guide. This information is provided to protect you and to keep your work area safe when operating the Drive and accompanying equipment.

Safety Instructions: for the electrical installation, the ESD instructions must be observed.

Caution

- The Systems that are electrically connected must be properly secured so they cannot be switched back on and warnings signs must be put up.
- Before start-up, it must be checked that the wiring is correct and is free of mechanical damages. Only drive with wiring in perfect condition may be enabled to operation.
- Incorrect voltage, reverse polarity and defective wiring can damage the drive.

- Do not connect or disconnect electric cables while the equipment is powered or running.
- The operator is responsible for keeping the safety installations in perfect working order, conforming to prevailing laws and standards.

Please read these chapters carefully before you begin the installation process.

The Lafert Drive contains electrostatic-sensitive components that can be damaged if handled incorrectly. To prevent any electrostatic damage, avoid contact with highly insulating materials, such as plastic film and synthetic fabrics. Place the product on a conductive surface and ground yourself in order to discharge any possible static electricity build-up.
To avoid any potential hazards that may cause severe personal injury or damage to the product during operation, keep all covers and cabinet doors shut.

The following safety symbols are used in this manual:

Warning

This information is needed to avoid a safety hazard, which might cause bodily injury or death as a result of incorrect operation:

- To avoid electric arcing and hazards to personnel and electrical contacts, never connect/disconnect the servo drive while the power source is on.
- Power cables can carry a high voltage, even when the motor is not in motion. Disconnect the Lafert Drive from all voltage sources before servicing.
- After shutting off the power and removing the power source from your equipment, wait at least 1 minute before touching or disconnecting parts of the equipment that are normally loaded with electrical charges (such as capacitors or contacts). Measuring the electrical contact points with a meter, before touching the equipment, is recommended.

Caution

This information is necessary to prevent bodily injury, damage to the product or to other equipment:

- The maximum DC power supply connected to the instrument must comply with the parameters outlined in this guide.
- When connecting the Lafert Drive to an approved control supply, connect it through a line that is separated from hazardous live voltages using reinforced or double insulation in accordance with approved safety standards.
- Before switching on the Drive, verify that all safety precautions have been observed and that the installation procedures in this manual have been followed.
- Make sure that the Safe Torque Off is operational.
- If a fire breaks out, do not direct the water extinguishers near the equipment to put out the flames.

Important

Identifies information that is critical for successful application and understanding of the product.

Safety measures must be taken both for people and machines, in compliance with Standards and local conditions.

APPROVALS

CE Conformity

TheLafert Drive was tested in authorized testing laboratories in accordance with the requirements of this documentation.

The Lafert Drive is in conformity with the following **EC Directives:**

- Low Voltage Directive (*2014/35/EC*)
- Electromagnetic Compatibility (EMC) (*2014/30/EU*)
- RoHS Directive (*2011/65/EU*)
- WEEE Directive (*2012/19/UE*)

Safety

EN 61800-5-1 Adjustable speed electrical power drive systems - Part 5-1: Safety requirements - Electrical, thermal and energy.

EMC Requirements

In terms of emission and immunity, the Lafert Drive fulfills the requirement for the category "second environment" (industrial environment).
EN 61800-3 - Adjustable speed electrical power drive systems - Part 3: EMC requirements and specific test methods.

Safety Conformity (STO) - Where Available

The Lafert Drive provides a two-channel, functionally safe STO function (Safe Torque Off).
The function disables the PWM and the drive can be switched safely to torque OFF.

The circuit design has been tested and subsequently assessed by TÜV Süd. According to that assessment, the circuit design used for the "Safe Torque Off" safety function in the Lafert Drive is suitable for meeting the requirements for .... in accordance with

- ***EN61508*** - Functional safety of electrical/electronic/programmable safety-related systems
- ***EN61800-5-2*** and category ... – Adjustable speed electrical power drive systems – Part 5-2: Safety requirements – Functional
- ***EN ISO 13849-1:2015*** - Safety of machinery — Safety-related parts of control systems — Part 1: General principles for design.

The subsystems (Lafert Drive) are fully described in terms of safety by the following characteristics:

EN 13849-1	EN 61508	PFHD [1/h]
PLe	SIL3	

STARTUP

Startup is prohibited within the scope of the EC directives until it has been determined that the machine/system in which this Lafert Drive is installed corresponds to the regulations within these directives.

Correct Use

The Lafert Drive is intended for operation of permanent magnet synchronous servomotors with compatible feedback systems in stationary machines and systems.
Installation of the Lafert Drive is only approved in industrial environments. For use in residential areas, additional EMC measures are necessary. The user must prepare a hazard analysis of the final product.

Caution

Other uses must first be approved by the manufacturer.

Improper Use

The Lafert Drive is not suitable for operation of motors other than synchronous servo motors or motors with non-compatible feedback systems.

In addition, the following applications are expected from intended use.

The installation of drives in areas at risk, where inflammable substances or combustible vapors or powders are present, could trigger fire outbreaks or explosions. As such, install the drives far away from said areas at risk, even if they are used with motors fit for use under these conditions.

2. | CANOPEN OPERATION

CANopen is a communication protocol and device profile specification for embedded systems used in automation.

The CANopen standard uses an addressing scheme, several small communication protocols and an application layer defined by a device profile.

CANOPEN NETWORK TOPOLOGY OVERVIEW

CANOpen SIGNAL	
SIGNAL	**DESCRIPTION**
GND_CAN	GND reference for CAN
CAN_T	120 Ω Termination resistance CAN (connect to CAN H)
CAN_L	CAN_L Connection
CAN_H	CAN_H Connection

Table 1 - CANOpen Signal

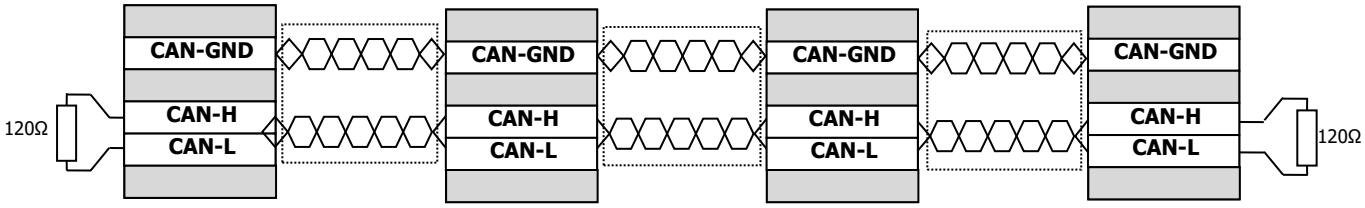


Figure 1- CANopen Network

All nodes of a network are principally connected in series, so that the CAN cable is looped through all controllers. The two ends of the CAN cables have to be terminated by a resistor of 120Ω +/-5%.

For further information refer to the Controller Area Network protocol specification, Ver. 2.0, Robert Bosch GmbH, 1991.

Baud Rate and ID Node

Compliance with the directives CiA DS301 v4.02 and DSP402 v2.0

- Baudrate set by a CANOpen object (default: 1000Kb)
- Id-Node set by software (default value: ID 1)

Caution

When there are more than one drives in the same bus CAN it is mandatory to have different Id-nodes.

information

Referring to "APPENDIX" chapter to know the "How to change Id-Node" and "How to change BaudRate"

CLIENT - SERVER

A CAN master (or client) is a controller that makes requests to nodes to respond to its commands. A CAN slave (or server) responds to the commands issued by the CAN master. The CAN protocol permits both single-master and multiple-master networks.

Information

The Lafert Servo Drive is the SLAVE and the machine controller or PLC is the MASTER.

Every servo drive has a unique ID in the range [1…127]. The network master does not require an ID.
As a slave, the servo drive never sends an unrequested message, other than emergencies. The drive responds only to messages addressed to its ID or to broadcast messages, which have an ID of 0. All messages sent by a servo drive are marked with its own ID.

Caution

If two servo drives have been assigned the same ID, the CAN network may crash.

ELECTRONIC DATASHEET (EDS)

The EDS file is the standardized format for the description of devices.

It contains information about:

- File properties (name, version, release date, …)
- General device information (manufacturer name and code)
- Device name and type, version
- Supported baud rates and boot-up options
- Description of supported objects and attributes

OBJECT DICTIONARY (O.D.)

The most important part of a device profile is the Object Dictionary description. The Object Dictionary is essentially a grouping of objects accessible via the network in an ordered pre-defined fashion. Each object within the dictionary is addressed using a 16-bit index.

The general structure of the object dictionary is as follows:

Index, Sub-Index (HEX)	OBJECT (Symbolic Name)	Name	Type	Attribute	M/O

- **Index, Sub-Index:** The Index column denotes the objects position within the Object Dictionary. This acts as a kind of address to reference the desired data field. The sub-index is not specified here. The sub-index is used to reference data fields within a complex object such as an array or record.
- **Object:** The Object column contains the Object Name and is used to denote what kind of object is at that particular index within the Object Dictionary.
- **Name:** The name column provides a simple textual description of the function of that particular object.
- **Type:** The type column gives information as to the type of the object. E.g.: Boolean, Floating number, Unsigned Integer, Signed Integer etc.
- **Attribute:** The Attribute column defines the access rights for a particular object. E.g.: rw (read and write access), wo (write only), ro (read only), Const (read only and value is constant).
- **M/O:** The M/O column defines whether the object is **M**andatory or **O**ptional

The standard object dictionary is as shown below:

Index (HEX)	Object
0000	Not used
0001-001F	Static data types
0020-003F	Complex data types
0040-005F	Manufacturer specific Complex data types
0060-007F	Device Profile Specific Static Data Types
0080-009F	Device Profile Specific Complex Data Types
00A0-0FFF	Reserved for further us
1000-1FFF	Communication Profile Area
2000-5FFF	Manufacturer Specific Profile Area
6000-9FFF	Standardized Device Profile Area
A000-FFFF	Reserved for further use

SYSTEM DESCRIPTION

information

Compliance with the directives CiA DS301 v4.02 and DSP402 v2.0

- **Identity objects:** Identity including vendor ID, product code, revision number and serial number. BaudRate set by a CANOpen (default: 1000Kb), Id-Node set by CANOpen object (default: Id node is 1)
- **Service Data Object (SDO):** SDO messages are used for reading and writing access to all entries of the object dictionary. SDOs are used for device configuration in the first place.
- **Process Data Object (PDO):** The real-time data transfer of target position, target velocity and definitions input and output is performed by PDO messages. Data is transmitted within four TPDO's (transmit PDO) and each with a maximum 8 byte wide data block. There are a static map with 4 TPDO and 4 RPDO.
- **Network Management (NMT):** The NMT state machine defines the communication behaviour of the CANopen device.
- **Emergency object:** Emergency messages are triggered by the occurrence of a device internal fatal error situation and are transmitted from the application device concerned to the other devices with highest priority. This makes them suitable for interrupting type error alerts.
- **Sync Message:** The SYNC protocol enables synchronous network behaviour.
- **Node-Guard Protocol:** Cyclic querying of the node state by the NMT Master Controller. The NMT Master Controller sends messages to the CANopen slaves that then respond within a defined time.
- **Heartbeat Function Protocol:** Automatic transmission of a heartbeat message by the network nodes. A heartbeat message is sent to the bus in millisecond intervals. Heartbeats are useful for detecting the presence or absence of a node on the network.
- **Event timer:** the event-timer enables a periodical transmission of the PDO. But have in mind, the event-timer is a local timer and not synchronized with the other CANopen device in the network.
- **Store and Restore Parameters:** Parameters save on non volatile memory (communication, manufacturer and device profile).
- **Input/Output:** the digital input and output are defined by object digital input and digital output
- **State machine:** The device control is performed by a state machine according to DSP402**.**
- **Modes of operation:** Different operation modes are available with the CiA 402 profile. Also, the drive supports the manufacture operation mode where the drive is to control with hardware interface

COMMUNICATION CANOPEN OBJECT (COB)

The communication objects are standardized with the DS301 CANopen communication profile.
The objects can be classified into 4 groups according to their tasks.

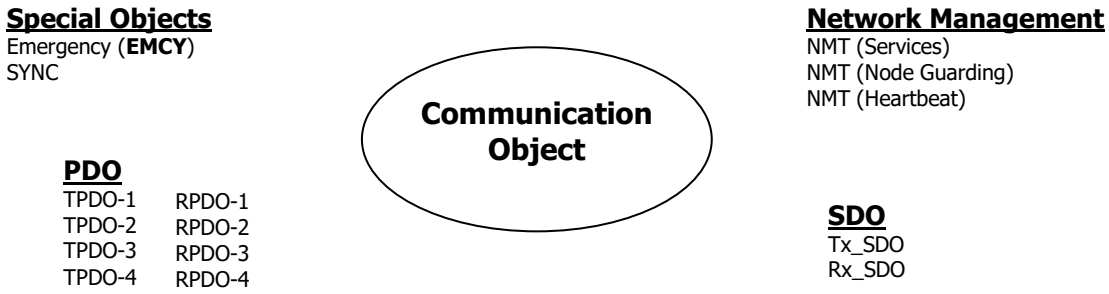


Figure 2 - Communication CANopen Object (COB)

- PDOs (process data objects) for real-time transmission of process data
- SDOs (service data object) for read and write access to the object Dictionary
- Objects for controlling CAN messages:
  - SYNC object (synchronization object) for synchronization of network devices
  - EMCY object (emergency object), for signalling errors of device or its peripherals.
- Network management services:
  - NMT services for initialization and network control (NMT: network management)
  - NMT Node Guarding for monitoring the network devices
  - NMT Heartbeat for monitoring the network devices

For communication between Master Controller and Lafert Servo Drive the following communication objects (COB) are available.

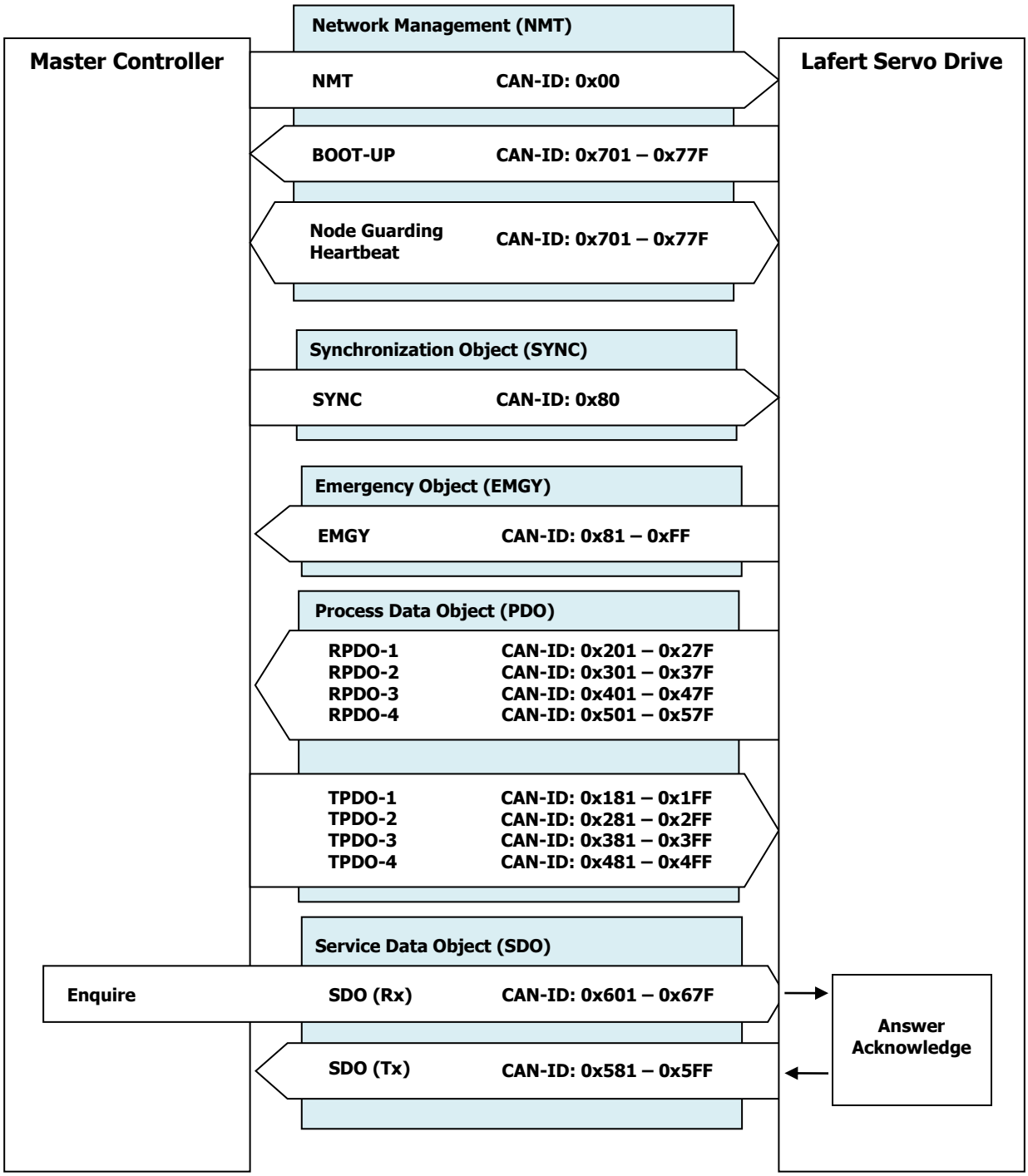


Figure 3 - communication between Master Controller and Drive

information

For Additional Information please refer to CiA DS301 standard.

CANopen makes available a simple and standardised possibility for accessing the parameters of the Lafert Drive (i.e. Target Speed or profile Acceleration). A unique number (index and sub-index) is assigned to each parameter (CANopen object).

The totality of all adjustable parameters is contained in the object directory (OD).

There are 2 methods for accessing CANopen objects via the CAN bus:

- **Access via Service data object (SDO):** confirmed type of access where the Lafert Drive acknowledges every parameter access

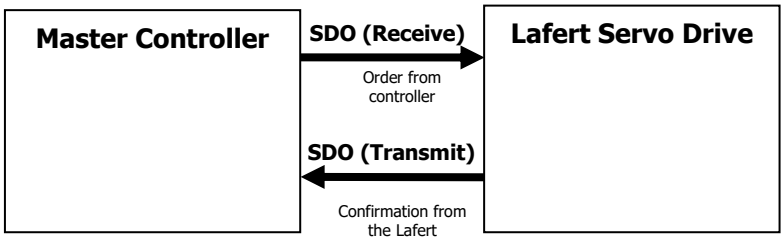


Figure 5 - SDO Communication

- **Access via Process data object (PDO):** unconfirmed type of access for which no acknowledgement takes place

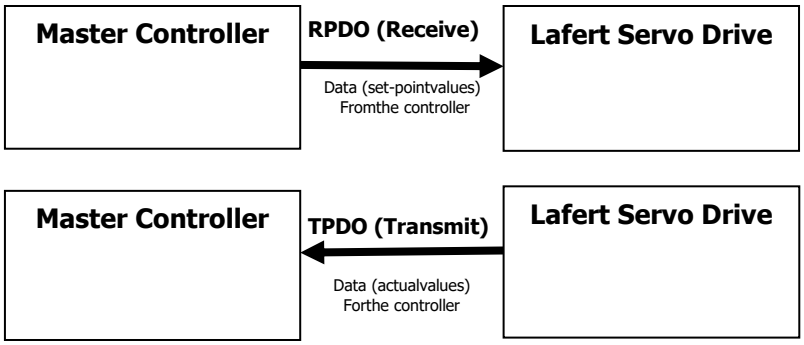


Figure 6 – PDO Communication

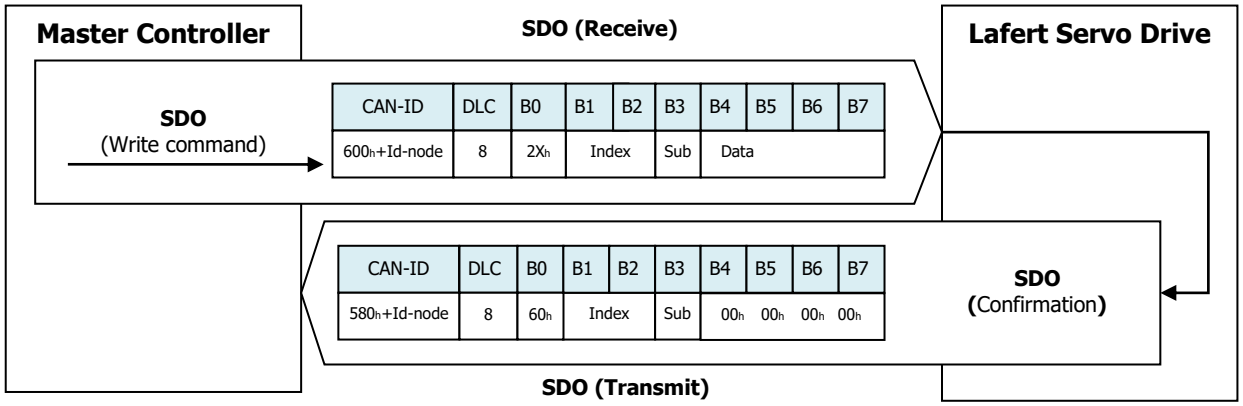
SDO PROTOCOL

The SDO protocol is used for setting and for reading parameters device. The SDOs are used to implement access to the object dictionary. The communication is always initiated by the SDO client.

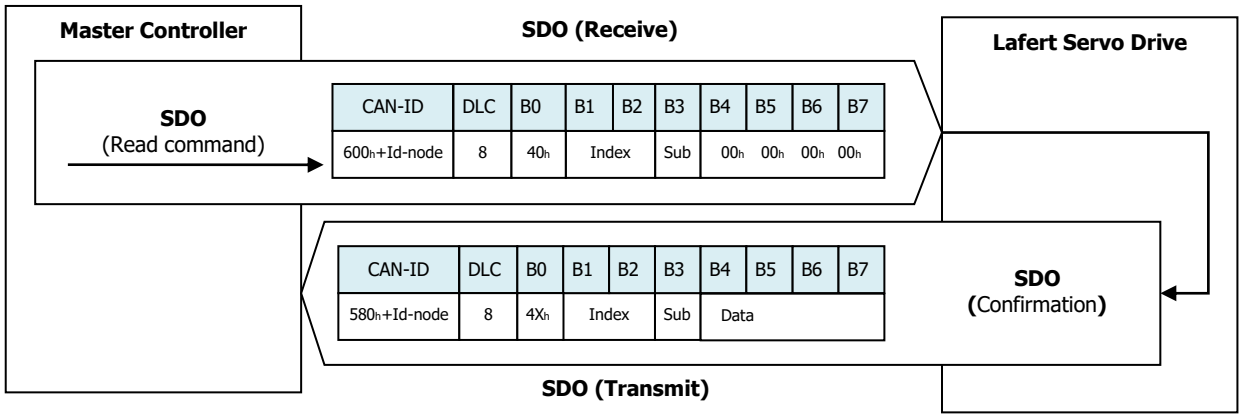
At the request of the client (Master Controller, PC Application, PLC - programmable logic controller) the drive makes data available.

The following communication protocols are supported:

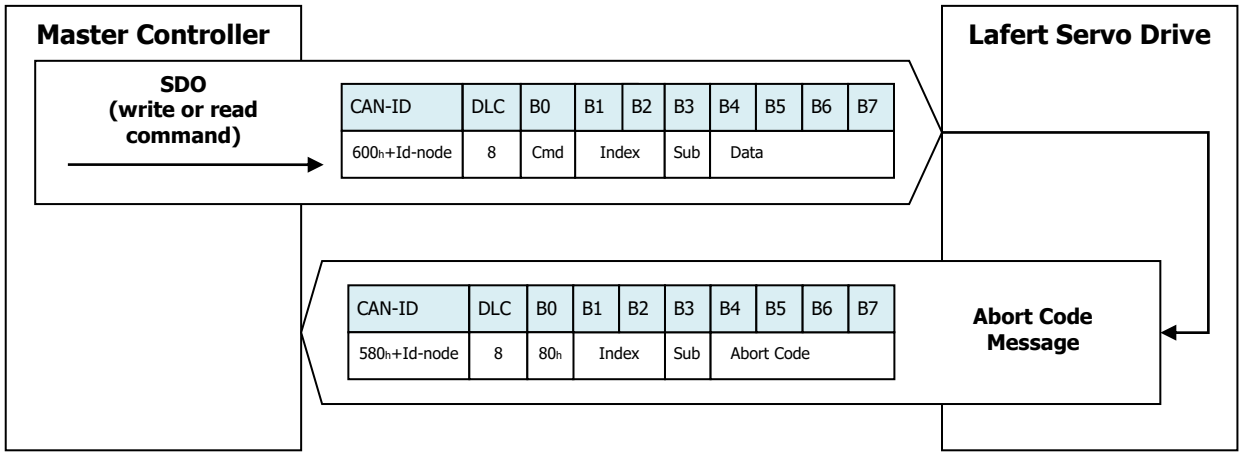
- **SDO Download Protocol**



- **SDO Upload Protocol**



- **SDO Abort Protocol**



SDO Download Protocol (WRITE)

The SDO download service is used to configure the communication, device and manufacturer specific parameters.

SDO Download Message structure:

COB-ID	Request/ Respond	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
600 h +IdNode	Rx	8	2x h	Index		Sub Index	Data LSB	Data	Data	Data MSB
580 h +IdNode	Tx	8	60 h	Index		Sub Index	00 h	00 h	0x00 h	00 h

Table 2- SDO Download Message Structure

SDO Download Message - Data Field:

D0	Description	Number of data bytes
22 h	Write Request (Initiate Domain Download)	-
23 h	Write Request (Initiate Domain Download)	4 bytes
27 h	Write Request (Initiate Domain Download)	3 bytes
2B h	Write Request (Initiate Domain Download)	2 bytes
2F h	Write Request (Initiate Domain Download)	1 byte
60 h	Write Response (Initiate Domain Download)	-

Table 3 - SDO Download Message Data Field

SDO Upload Protocol (READ)

The SDO upload service is used to read the communication, device and manufacturer specific parameters SDO.

SDO Upload Message structure:

COB-ID	Request/ Respond	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
600 h +IdNode	Rx	8	40 h	Index		Sub Index	00 h	00 h	00 h	00 h
580 h +IdNode	Tx	8	4x h	Index		Sub Index	Data LSB	Data	Data	Data MSB

Table 4 - SDO Upload Message Structure

SDO Upload Message - Data Field:

D0	Description	Number of data bytes
40 h	Read Request (Initiate Domain Upload)	-
43 h	Read Response (Initiate Domain Upload)	4 bytes
47 h	Read Response (Initiate Domain Upload)	3 bytes
4B h	Read Response (Initiate Domain Upload)	2 bytes
4F h	Read Response (Initiate Domain Upload)	1 byte

Table 5 - SDO Upload Message Data Field

SDO Abort Code

The SDO Abort service is used to communicate fault by download or upload service.
If the SDO fails then the CANOpen does not respond with the corresponding SDO message, but it uses the SDO abort protocol.

In the Abort message there is the data abort code that recognizes the kind of fault.

SDO Abort Message structure:

COB-ID	Request/ Respond	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
580 h +IdNode	**Tx**	**8**	80 h	Index		Sub Index	Abort Code			

Table 6 - SDO Abort Message Structure

The Abort Code as defined in follow table, It is encoded as UNSIGNED32 value.

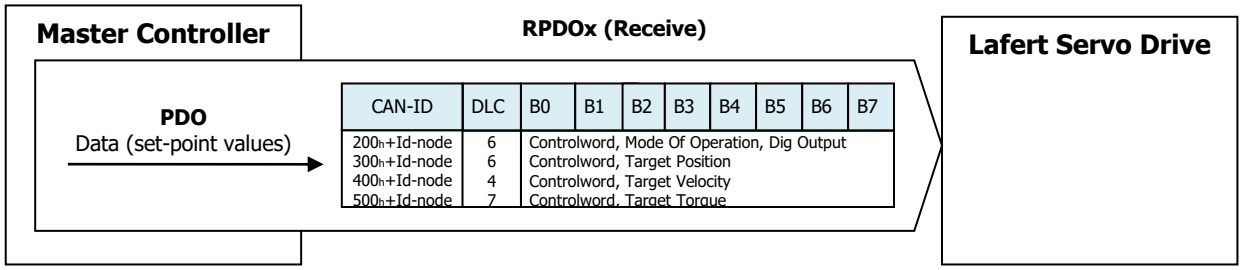
Abort Code	Description
0504 0000h	SDO protocol timed out.
0504 0001h	Client/server command specifier not valid or unknown.
0504 0002h	Invalid block size (block mode only).
0504 0003h	Invalid sequence number (block mode only).
0504 0004h	CRC error (block mode only).
0504 0005h	Out of memory.
0601 0000h	Unsupported access to an object.
0601 0001h	Attempt to read a write only object.
0601 0002h	Attempt to write a read only object.
0602 0000h	Object does not exist in the object dictionary.
0604 0041h	Object cannot be mapped to the PDO.
0604 0042h	The number and length of the objects to be mapped would exceed PDO length.
0604 0043h	General parameter incompatibility reason.
0604 0047h	General internal incompatibility in the device.
0606 0000h	Access failed due to an hardware error.
0607 0010h	Data type does not match, length of service parameter does not match
0607 0012h	Data type does not match, length of service parameter too high
0607 0013h	Data type does not match, length of service parameter too low
0609 0011h	Sub-index does not exist.
0609 0030h	Invalid value for parameter (download only).
0609 0031h	Value of parameter written too high (download only)
0609 0032h	Value of parameter written too low (download only).
0609 0036h	Maximum value is less than minimum value.
060A 0000h	Operation not allowed in this state
060A 0023h	Resource not available: SDO connection
0800 0000h	General error
0800 0020h	Data cannot be transferred or stored to the application.
0800 0021h	Data cannot be transferred or stored to the application because of local control.
0800 0022h	Data cannot be transferred or stored to the application because of the present device state.
0800 0023h	Object dictionary dynamic generation fails or no object dictionary is present (e.g. object dictionary is generated from file and generation fails because of an file error).
0800 0024h	No data available
0800 0030h	Data cannot be written because it need STORE command and a reset or power cycle

Table 7 - SDO Abort Code Description

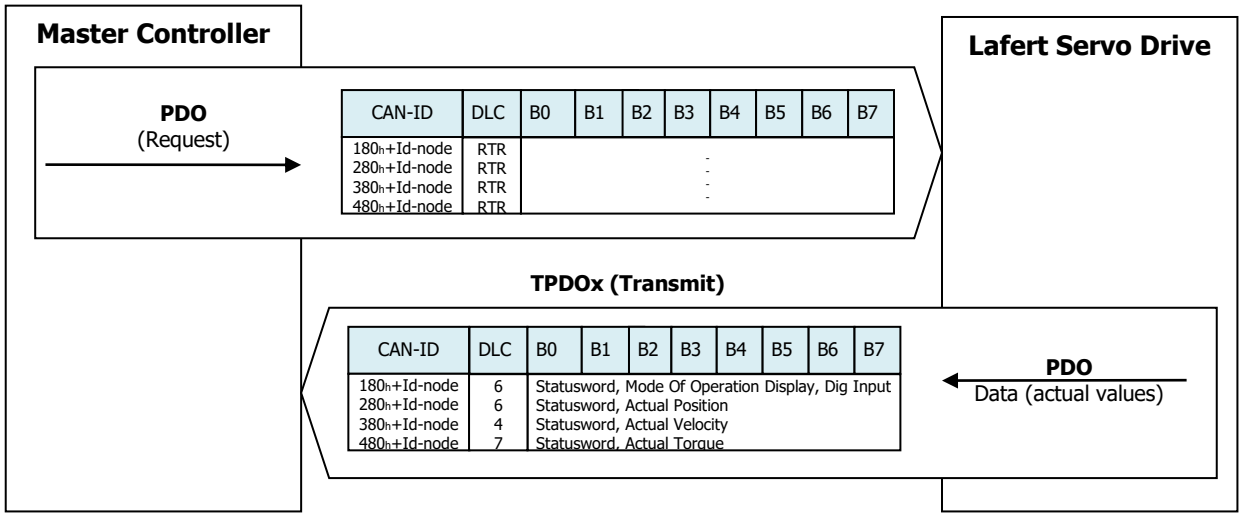
PDO PROTOCOL

The PDO protocol is used to process real time data among various nodes.
This communication Object uses the unconfirmed communication service. Data transferring will be limited to 1 to 8 bytes and there is no hand-shake restriction in PDO communication.

The Receive PDOs default mappings are composed by controlword and set points values:



The Transmit PDOs default mappings are composed by statusword and actual values:



The default PDO configuration can be changed, there are 2 objects groups that are used for each PDO:

- **PDO Communication Parameter Set**: it contains the PDO configuration COB-ID, transmission-type, restriction time and cycle of timer used by PDO (objects 1400_h, 1401_h, 1402_h, 1403_h, 1800_h, 1801_h, 1802_h, 1803_h)

  This communication parameter set is a record:

Sub-Index	Description	Data Type	Description
00_h	Number of Entries	UNSIGNED8	It indicates the sub-parameter with highest number
01_h	COB-ID	UNSIGNED32	It contains the CAN-ID to be used and some additional protocol control bits.
02_h	Transmission Type	UNSIGNED8	It determines the triggering (TPDO) or reception (RPDO) behaviour.
03_h	Inhibit Time	UNSIGNED16	It provides the time in ms, when this TPDO is allowed sending it again.
04_h	Compatibility Entry	UNSIGNED8	Reserved *(Not applicable)*
05_h	Event Timer	UNSIGNED16	It specifies the period in ms of the TPDO transmission respectively the time-out (missing) of a RPDO
06_h	SYNC start value	UNSIGNED8	It needs to match with the value given in the 1-byte SYNC message.

- **PDO Mapping Parameter Set:** it contains a list of objects dictionary. These object are mapped into PDO, include their data length in bits (objects 1600_h, 1601_h, 1602_h, 1603_h, $1A00_h$, $1A01_h$, $1A02_h$, $1A03_h$). Producers and clients must know this mapping.

  The PDO mapping has 2 rules:
  - Each PDO could be mapped into 4 objects
  - The length of each PDO will be no more than 64 bits

The Lafert Servo Drive has a default mapping that it can be changed when the network initializes. The drive supports the dynamic PDO mapping and changing of mapping can be done in the state PRE-OPERATIONAL.

Caution

The "PDO communication Parameter Set" and "PDO Mapping Parameter Set" can't be saved in e^2prom, but they can be changed during initialization.

It is possible to change the mapping in OPERATIONAL state but is not suggested. If the client change the mapping during OPERATIONAL state, then it is responsible for the data consistency.

information

The PDO protocol is available in OPERATIONAL state

COB-ID

The most important communication parameter in PDO is the CAN identifier: also known as the Communication Object Identifier or COB-ID. It is used to identify the data, and it determines their priority for bus access.

For each CAN data telegram there may only one sender node (master controller), although all messages sent in the CAN broadcast procedure can be received by any number of nodes (drives). Thus a node can make its input information available to a number of bus device at the same time-even without transferring them though a logical bus master.

The COB-ID is located in sub-index 1 of the communication parameter. It is coded as 32 bit value It contains some control bits.

The bit 29 defines if the message is a standard message or extended message. In case bit 29 is 0, the following 18 bits are ignored and the remaining 11 bits are regarded as the CAN-ID to be used for the PDO. It is transmitted using the CAN base frame format. If the bit is 1, the following bits are interpreted as a 29-bit ID. This means the corresponding PDO is transmitted in the CAN extended frame format.

The bit 30 is used to indicate if CAN remote frames are allowed or not. This is not supported by all CAN implementations. In some, the remote frame transmission can't be disabled.

The bit 31 enables and disables the PDO transmission respectively the reception function. This means you can switch-off the PDO transmission. The reception of CAN messages can't be switched-off, but the CANopen protocol stack doesn't process the received PDO when you have disabled it.

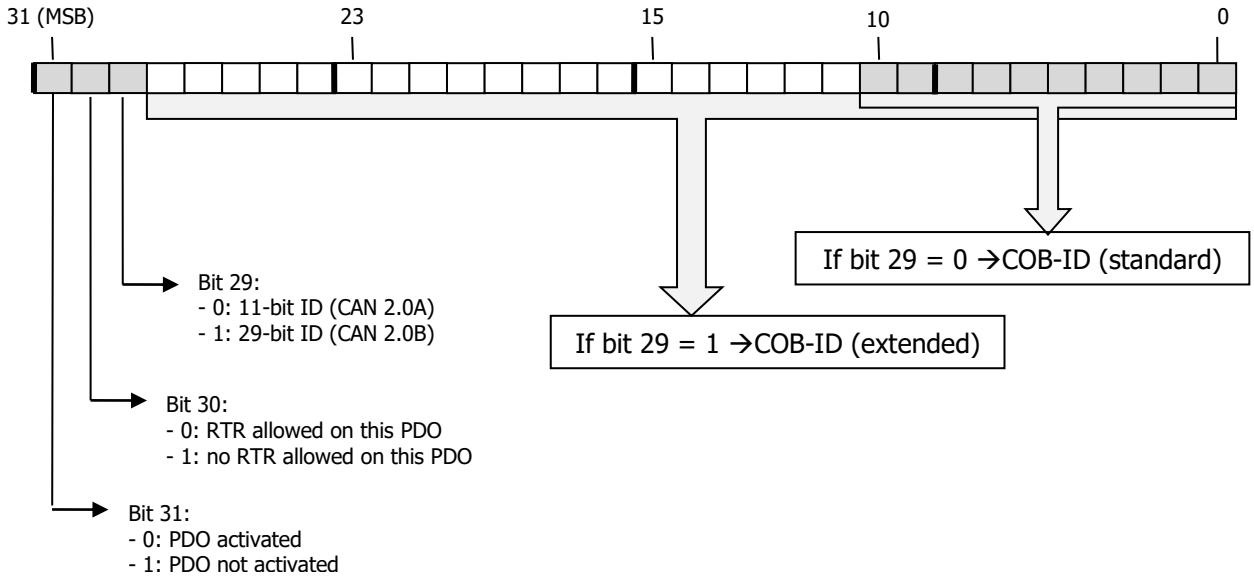


Figure 7 – COB-ID bits

PDO COMMUNICATION

There are multiple ways to transmit PDO:

- Synchronous: PDOs can be triggered cyclic or acyclic with SYNC message
- Asynchronous: PDOs are sent on event (application/profile specific or timer) or a remote request (RTR).

For PDOs different transmission types are distinguished:

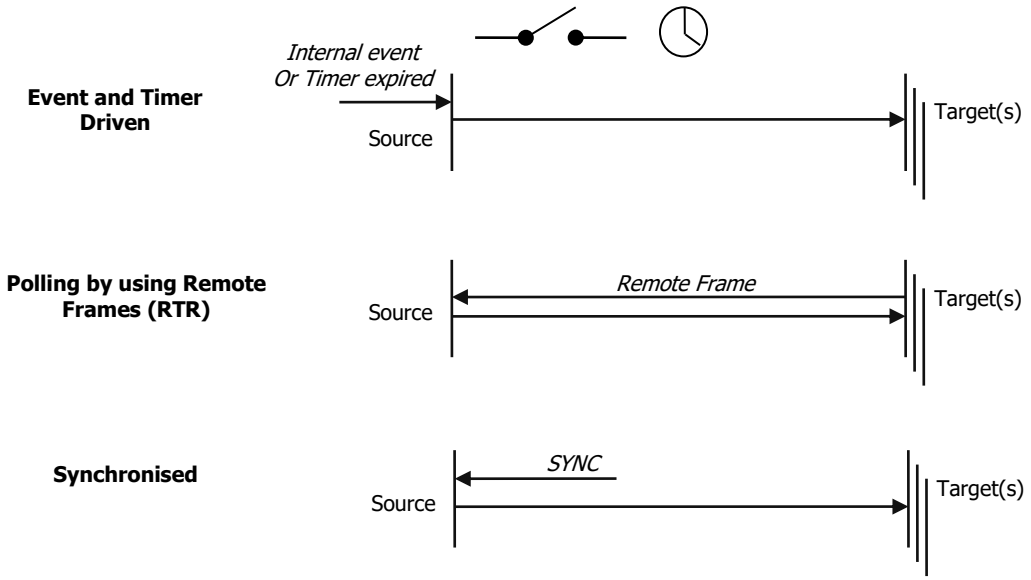


Figure 8 - PDO transmission Type

1) Event and Timer driven:

Message transmission is either triggered by the occurrence of an application-specific event specified in the device profile, application profile or manufacturer-specific, or if a specified time (event-timer) has elapsed without occurrence of an event.

2) Polled Remotely requested:

The PDOs can also be polled by data request telegrams (remote frame or RTR).The transmission of an event-driven PDO is initiated on receipt of a RTR initiated by a PDO consumer.

3) Synchronised:

In order to synchronize CANopen devices a synchronization object (SYNC object) is transmitted periodically by a synchronization application. Message transmission is triggered by the occurrence of the SYNC object. The trigger condition is the number of Sync. The SYNC object is represented by a pre-defined communication object (see chapter "SYNC Protocol").

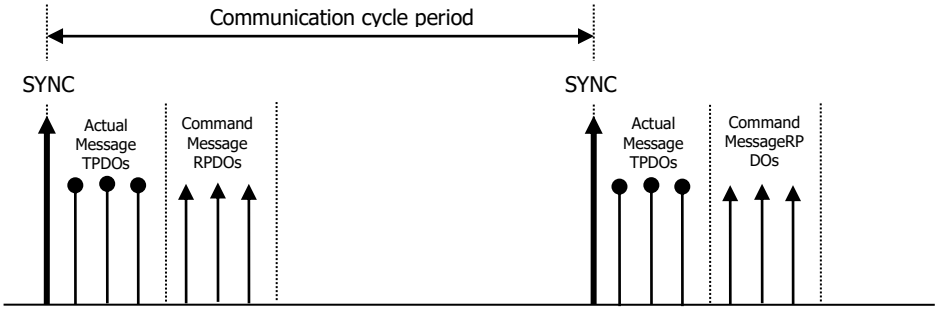


Figure 9 - Sync Transmission Type

The PDO Transmission Type parameter specifies how the transmission of the PDO is triggered or how received PDOs are handled. The transmission Type is located in sub-index 2 of the communication parameter set.

TransmissionType	Cyclical	Acyclical	Synchronous	Asynchronous	Only RTR
0		x	x		
1 - 240	x		x		
241 - 251	*Reserved*				
252			x		X
253				x	X
254 , 255				x	

Table 8- Transmission Type Set

- **Acyclic Synchronous (0):**

PDOs of transmission type 0 function synchronously with SYNC message, but not cyclically.

- RPDO is only evaluated after the next SYNC telegram has been received.
- TPDO acquires its input data when receives the SYNC message and transmits the data current when the event is occurred. Triggered when the SYNC message is received and one of the mapped process data has changed its value after the last transmission.

- **Cyclical Synchronous (1, … , 240):**

PDOs configured for transmission type 1-240 the PDO is transmitted after every SYNC (n=1...240).

- RPDO are always triggered by the following SYNC upon reception of data independent of the transmission type 1-240.
- TPDO is transmitted after the SYNC, the value between 1- 240 means that the PDO is transferred synchronously and cyclically. The value indicates the number of SYNC which are necessary to trigger PDO transmissions.

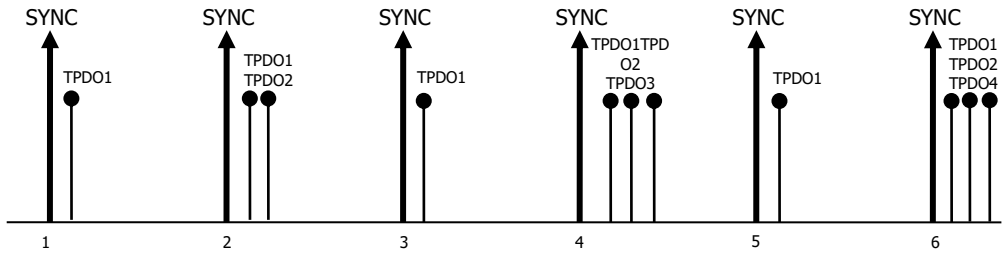


Figure 10 - Example TPDO SYNC (TPDO1 =1, TPDO2=2, TPDO3=4, TPDO4=6)

- **Only RTR (252, 253):**

  Transmission type 252 and 253 is used to transmit exclusively on request by a remote frame (RTR). These values are only possible for TPDOs.

  - 252 is a synchronous: the data is updated (but not sent) immediately after reception of the SYNC object.
    - On a SYNC signal, the contents of the TPDO are stored
    - On request (RTR) the TPDO is sent to the master

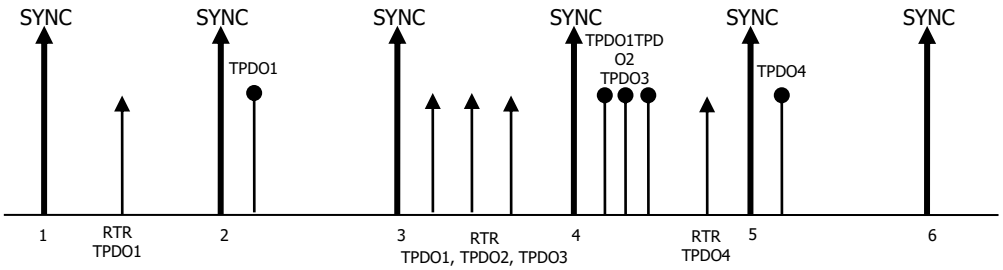


Figure 11 - Transmission Type with RTR synchronous (252)

  - 253 is an asynchronous: the data is acquired continuously and transmitted on request.
    - On request (RTR) the TPDO is sent to the master

- **Asynchronous (254, 255):**

  Asynchronous means that the transmission of the PDO is not related to the SYNC object. Transmission Type 254 and 255 are asynchronous, but also event-driven (timer or event)

  - 254 the event is specific to the manufacturer. Triggered by an internal event (e.g. change-of-state of one of the mapped process data or elapsing of the event-timer or any other event). The device manufacturer specifies the internal event triggering the TPDO transmission.
  - 255 the event is defined in the device profile. As before, but the CiA profile specifies the internal event triggering the TDPO transmission

  The “event” is the alteration of an input value; the data are transmitted immediately after this change.

  The manufacturer-specific event defined is the changing of statusword.

  If the timer is defined and the PDO doesn’t have the event, the PDO transmits when the timer is expired.

Inhibit Timer

The time is the minimum interval for PDO transmission if the asynchronous transmission type is set. The inhibit timer (transmit delay time) specifies the minimum length of time that must be allowed to elapse between the transmission of two of the same telegram. The inhibit timer is located in sub-index 3 of the communication parameter set.

The unit is n*100 µs, where n > 0

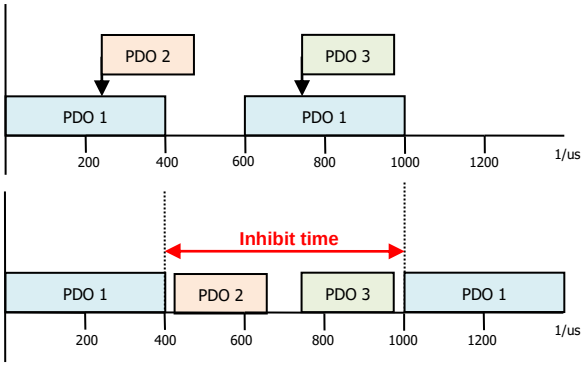


Figure 12 - inhibit time for PDO asynchronous

Event Timer

The event timer is an additional event for the corresponding asynchronous PDO (FE_h and FF_h). When the timer is expired the PDO will be transmitted. If the application event occurs during a timer period, it will also transmit and the timer is reset.

The event timer is located in sub-index 5of the communication parameter set.

The value is defined as multiple of 1 ms, the value of 0 shall disable the event-timer.

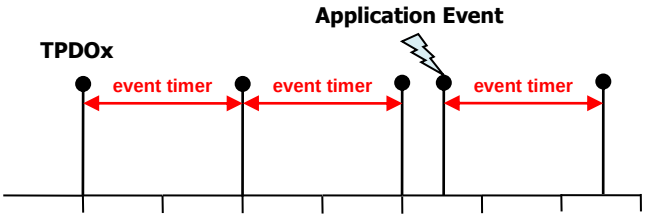


Figure 13 – Event timer for PDO asynchronous

Note: the application event is not available; the event timer can be used to transmit periodically TPDOs.

Caution

The PDOs can be transmitted at minimum interval time 2,5ms.

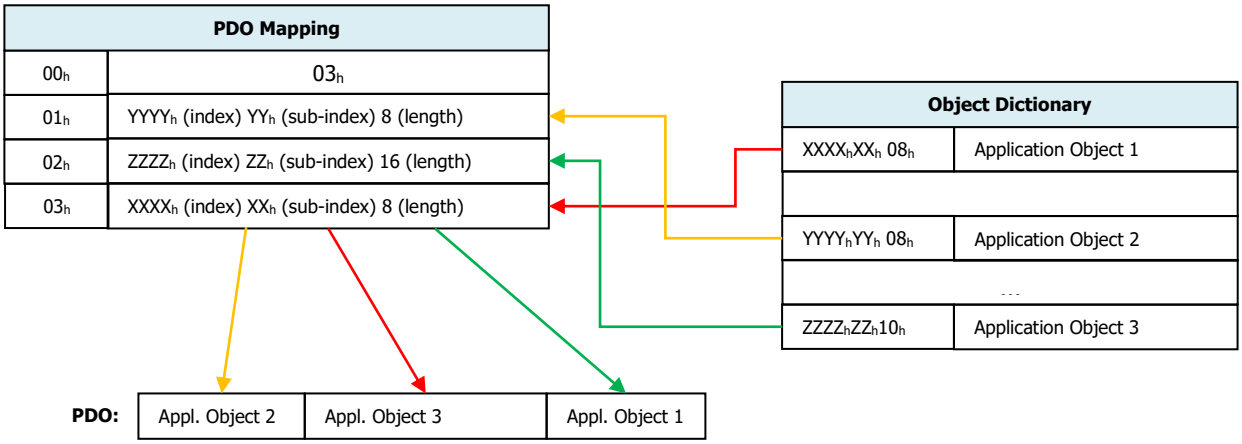
But if the PDO has a controlword, the master must wait 40ms before that the command of controlword is processed.

The user can change the default mapping during initialization (when the drive is in PRE-OPERATIONAL state) because the drive supports the dynamic mapping but the saving in eeprom is not available.

PDO Mapping

PDO mapping refers to mapping of the application object (real time data) that the drive uses. Lafert Servo Drives provides a default mapping but the mapping can be changed during initialization.

A device supports dynamic mapping of PDOs, the mapping must change during the state PRE-OPERATIONAL when the PDOs are disabled.



Receive PDO (RPDO)

1400h – 1403h: RPDO Communication Objects

These objects contain the communication parameters for the PDOs the device is able to receive.

- Sub-index 00h contains the number of valid entries within the communication record
- Sub-index01h is the COB-ID of the PDO, this entry has been defined ad UNSIGNED32 in order to define for 11-bit CAN identifiers (CAN 2.A) as well as for 29 bit identifiers (CAN 2.0B)

31 (MSB)	30	29	28 ... 11	10 ... 0 (LSB)
Valid	Reserved	Frame	0000h	11-bit CAN-ID
			29-bit CAN-ID	

Name	BIT	VALUE	Meaning
Valid	31	0b 1b	PDO exist / is valid PDO does not exist /is not valid
Reserved	30	-	-
frame	29	0b 1b	Message standard 11 bit (CAN 2.0 A) Message extended 29 bit (CAN 2.0 B)
29-bit CAN-ID	11 - 28	X	If Bit 29 = 1 : bits 28-11 of COB-ID message extended
11-bit CAN-ID	0 – 10	X	bits 10-0 of COB-ID

Information

The PDO valid or no allows to select which PDOs are used in the OPERATIONAL state.

- Sub-index 02h is the transmission type; it defines the reception character of the PDO.

  The following table describes the usage entry.

Transmission Type		Meaning
0 - 240	00_h – $F0_h$	synchronous
241 - 253	00_h – FD_h	reserved
254	0xFE	event-driven (manufacturer-specific)
255	0xFF	event-driven (device profile and application profile specific)

- Synchronous means that the CANopen device shall actuate the received data with the reception of the next SYNC.
- Event-driven means that the PDO may be received at any time. The CANopen device will actualize the data immediately.

- Sub-index 03_h contains the inhibit time. The value is defined as multiple of 100 µs. The value of 0 shall disable the inhibit time. It is not allowed to change the value while the PDO exists (bit 31 of sub-index 01_h is set to 0_b). The RPDO may use the time implementation specific.
- Sub-index 04_h is reserved. It shall not be implemented; in this case read or write access leads to the SDO abort transfer service in this case read or write access leads to the SDO abort transfer service (abort code: 0609 0011_h).
- Sub-index 05_h contains the event-timer. The value is defined as multiple of 1 ms. The value of 0 shall disable the event-timer. The RPDO may use the time for deadline monitoring. The deadline monitoring is activated within the next reception of an RPDO after configuring the event-timer. A timeout results in an indication to the local application.
- Sub-index 06_h contains the SYNC start value. This is not used by RPDOs. It shall not be implemented; in this case read or write access shall lead to the SDO abort transfer service (abort-code: 0609 0011_h).***(it is not present)***

It is possible to change the parameters when the drive is in PRE-OPERATIONAL state. Follow the procedure to change a parameter:

- Disable the PDO: set Bit 31 of COB-ID (sub-index 01_h) at "1"
- Write new value in sub-index corresponding
- Enable the PDO writing the COB-ID to configure the PDO enabled.

Example:

Modify the transmission type "1" for RPDO1 (Id-Node 1)

1. **Disable RPDO1:**

   RX: Id 0x601 - 23 00 14 01 00 00 00 80

   Tx: Id 0x581 – 60 00 14 01 00 00 00 00

   Rx: [cmd: 0x23] [Index: 0x1400] [sub-index: 0x01] [Data: 0x80000000] – Disable PDO (bit 31= 1)

   Tx: [resp: 0x60] [Index: 0x1400] [sub-index: 0x01] [Data: 0x00000000] - Successful

2. **Change Tx Type (n.1 Sync):**

   RX: Id 0x601 – 2F 00 14 02 01 00 00 00

   Tx: Id 0x581 – 60 00 14 02 00 00 00 00

   Rx: [cmd: 0x2F] [Index: 1400] [sub-index: 0x02] [Data: 0x00000001] – Tx Type = 1

   Tx:[resp: 0x60] [Index: 1400] [sub-index: 0x02] [Data: 0x00000000] – Successful

3. **Enable PDO And Set COB-ID (0x201):**

   RX: Id 0x601 – 23 00 14 01 01 02 00 00

   Tx: Id 0x581 – 60 00 14 01 00 00 00 00

Rx: [cmd: 0x23] [Index: 0x1400] [sub-index: 0x01] [Data: 0x00000201] – Enable PDO (bit 31=0) and set COBID=0x201

Tx: [resp: 0x60] [Index: 0x1400] [sub-index: 0x01] [Data: 0x00000000] – Successful

Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

1600_h – 1603_h: RPDO Mapping Parameters

These objects contain the mapping for the PDOs device is able to receive.

- Sub-index 00_h contains the number of valid object entries within the mapping record or a specific value. The number of valid object entries shall be the number of the application objects that shall be received with the corresponding RPDO.

Value	Description
00_h	Mapping disabled
01_h	Sub-index 01h valid
02_h	Sub-index 01h and 02h valid
...	...

- Sub-index from 01_h to 40_h contains the information of the mapped application objects. The object describes the content of the PDO by their index, sub-index and length. The length contains the length of the application object in bit. This may be used to verify the mapping.

Byte: MSB LSB

Index (16 bit)	Sub-Index (8bit)	Object Length (8 bit)

If the changing of the PDO mapping cannot be executes (e.g. the PDO length is exceeded or the SDO client attempts to map an object that cannot be mapped) the drive responds with an Abort Code SDO transfer service.

Example:

- controlword: 60400010_h = 6040_h (index) 00_h (sub-index) 10_h (length 2 bytes)
- mode of operation: 60600008_h = 6040_h (index) 00_h (sub-index) 08_h (length1 byte)
- target velocity: $60FF0020_h$ = $60FF_h$ (index) 00_h (sub-index) 20_h (length4 bytes)

Lafert Servo Drive has available 8 record of 32 byte for mapping. The user can map until 8 object (every object is 1 byte) because the total length must be less than or equal 64 bits.

RPDO Mapping Default

RPDOs are CAN frames identified by their 11-bit header.

- RPDO1: 200_h + Node ID
- RPDO2: 300_h + Node ID
- RPDO3: 400_h + Node ID

- RPDO4: 500 h + Node ID

The following tables describe the default mapping for RPDO:

Index	SubIndex	Description	Type	Attr.	Dafault Value	Description
Receive Process Data Object (RPDO1)						
1400 h	0	Receives 1st PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO1	UNSIGNED32	rw	200 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FE h	Asynchronous Man. Spec.
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]
1600 h	0	N receive PDO mapping	UNSIGNED8	rw	3	Number of Entries
	1	1 - application object	UNSIGNED32	rw	6040 0010 h	Control word(2 byte)
	2	2 - application object	UNSIGNED32	rw	6060 0008 h	Mode of operation(1 byte)
	3	3 - application object	UNSIGNED32	rw	60FE 0120 h	Digital Outputs(4 byte)
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-
Receive Process Data Object (RPDO2)						
1401 h	0	Receives 2nd PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO2	UNSIGNED32	rw	300 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FE h	Asynchronous Man. Spec.
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled[ms]
1601 h	0	N receive PDO mapping	UNSIGNED8	rw	2	Number of Entries
	1	1 - application object	UNSIGNED32	rw	6040 0010 h	Control word (2 byte)
	2	2 - application object	UNSIGNED32	rw	607A 0020 h	Target Position (4 byte)
	3	3 - application object	UNSIGNED32	rw	0	-
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-
Receive Process Data Object (RPDO3)						
1402 h	0	Receives 3rd PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO3	UNSIGNED32	rw	400 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FE h	Asynchronous Man. Spec.
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]
1602 h	0	N receive PDO mapping	UNSIGNED8	rw	2	Number of Entries
	1	1 - application object	UNSIGNED32	rw	6040 0010 h	Control word (2 byte)
	2	2 - application object	UNSIGNED32	rw	60FF 0020 h	Target Velocity (4 byte)
	3	3 - application object	UNSIGNED32	rw	0	-
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-
Receive Process Data Object (RPDO4)						
1403 h	0	Receives 4th PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO4	UNSIGNED32	rw	500 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FE h	Asynchronous Man. Spec.
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]
1603h	0	N receive PDO mapping	UNSIGNED8	rw	2	Number of Entries

1	1 - application object	UNSIGNED32	rw	6040 0010 h	Control word (2 byte)
2	2 - application object	UNSIGNED32	rw	6071 0010 h	Target Torque (2 byte)
3	3 - application object	UNSIGNED32	rw	0	-
4	4 - application object	UNSIGNED32	rw	0	-
5	5 - application object	UNSIGNED32	rw	0	-
6	6 - application object	UNSIGNED32	rw	0	-
7	7 - application object	UNSIGNED32	rw	0	-
8	8 - application object	UNSIGNED32	rw	0	-

Table 9 - RPDO Description

Mapping default RPDO 1:Controls state machine – mandatory

Index + subindex	NameReceive PDO 1	Default Value	Size Bytes
1600_h: 0	Number of mapped objects	3	
1600_h: 1	Control word	6040 0010_h	2
1600_h: 2	Mode of operation	6060 0008_h	1
1600_h: 3	Digital Output	60FE 0120_h	4

Table 10 - RPDO1 Mapping

Mapping default RPDO 2:Controls state machine and target position- optional

Index + subindex	NameReceive PDO 1	Default Value	Size Bytes
1601_h: 0	Number of mapped objects	2	
1601_h: 1	Control word	6040 0010_h	2
1601_h: 2	Target Position	607A 0020_h	4

Table 11 - RPDO2 Mapping

Mapping default RPDO 3:Controls state machine and target velocity– optional

Index + subindex	NameReceive PDO 1	Default Value	Size Bytes
1602_h: 0	Number of mapped objects	2	
1602_h: 1	Control word	6040 0010_h	2
1602_h: 2	Target Velocity	60FF 0020_h	4

Table 12 - RPDO3 Mapping

Mapping default RPDO 4: Controls state machine and target torque - optional

Index + subindex	NameReceive PDO 1	Default Value	Size Bytes
1603_h: 0	Number of mapped objects	2	
1603_h: 1	Control word	6040 0010_h	2
1603_h: 2	Target Torque	6071 0010_h	2

Table 13 - RPDO4 Mapping

When the master sends a RPDO must have attention to select the length of PDO.

Example:

RPDO1 has 3 object default, it is composed by

- Controlword 0x6040 (2 bytes length)
- Mode of operation (1 byte length)
- Digital input (4 bytes length)

Total Length is 7 bytes.

The DLC of message RPDO1 must be length 7 bytes. If the PDO Length exceeded the drive sends an emergency message with error code 8220 h.

information

Referring to "APPENDIX – EXAMPLE PROGRAMS" chapter to read the example for "Control via PDO"

Re-Mapping Procedure:

The following procedure shall be used for re-mapping, which may take place during the NMT state Pre-operational:

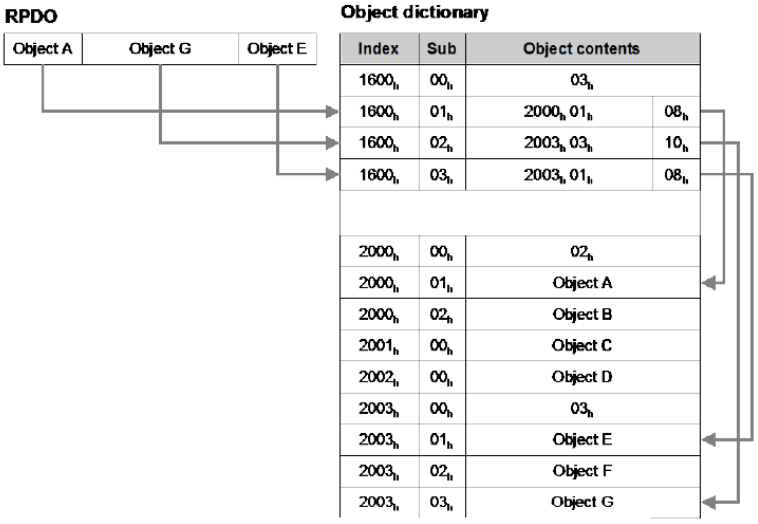
1) For changing the PDO mapping first the PDO has to invalidate the PDO. Destroy the bit *valid* into sub-index 01_h of "RPDO Communication" objects (1400_h, 1401_h, 1402_h e 1403_h). The 31 bit must be set to 1.
2) Disable mapping PDO setting 0 into sub-index $00_h$of "Mapping Parameters" object (1600_h, 1601_h, 1602_h e 1603_h). This will disable PDO.
3) Modify mapping by changing the values of the corresponding sub-indices. Write in sub-index correspondent the description of the object (Index, Sub-Index and Length)
4) Set the sub-index 00_h of PDO coordinated mapping parameter (objects 1600_h, 1601_h, 1602_h e1603_h) as legal number (number of PDO's mapping objects). This will enable new mapping.
5) Create RPDO by setting bit valid to 0of sub-Index 01_h(COB-ID)of "communication object" (objects 1400_h, 1401_h,1402_h e 1403_h) the according RPDO communication parameter.
6) PDO mapping completing

information

Referring to "APPENDIX – EXAMPLE PROGRAMS" chapter to read the example for "Remapping RPDO procedure"

If during step 3 the drive detects that the Index and sub-Index of the mapped objects does not exist or the object cannot be mapped, the device responds with the SDO abort transfer service (abort code: 0602 0000_h or 0604 0041_h).

If during step 4 the drive detects that the RPDO mapping is not valid or not possiblethe CANopen device shall respond with the SDO abort transfer service (abort code: 0602 0000_hor 0604 0042_h).



If the device receives a PDO that is having more data bytes than the number of mapped data bytes (length), then the CANopen device shall use the first data bytes up to the length and may be initiate the EMCY write service, if supported.

Example:

Remap RPDO1 with controlword and target velocity (Id-Node 1)

1. **Disable RPDO1:**

   RX: Id 0x601 - 23 00 14 01 00 00 00 80

   Tx: Id 0x581 – 60 00 14 01 00 00 00 00

   RX: [cmd: 0x23] [Index: 0x1400] [sub-index: 0x01] [Data: 0x80000000] – Disable PDO (bit 31= 1)

   Tx: [resp: 0x60] [Index: 0x1400] [sub-index: 0x01] [Data: 0x00000000] - Successful

2. **Disable Mapping of RPDO1:**

   RX: Id 0x601 – 2F 00 16 00 00 00 00 00

   Tx: Id 0x581 – 60 00 16 00 00 00 00 00

   RX: [cmd: 0x2F] [Index: 0x1600] [sub-index: 0x00] [Data: 0x00000000] – Disable Map PDO (Value = 0)

   Tx: [resp: 0x60] [Index: 0x1600] [sub-index: 0x00] [Data: 0x00000000] - Successful

3. **Map in Pos 1 object controlword (0x6040):**

   RX: Id 0x601 – 23 00 16 01 10 00 40 60

   Tx: Id 0x581 – 60 00 16 01 00 00 00 00

   RX: [cmd: 0x23] [Index: 1600] [sub-index: 0x01] [Data: 0x60400010] – map pos 1 object 0x6040

   Tx: [resp: 0x60] [Index: 1600] [sub-index: 0x01] [Data: 0x00000000] – Successful

4. **Map in Pos 2 object target velocity (0x60FF):**

   RX: Id 0x601 – 23 00 16 02 20 00 FF 60

   Tx: Id 0x581 – 60 00 16 02 00 00 00 00

   RX: [cmd: 0x23] [Index: 1600] [sub-index: 0x02] [Data: 0x60FF0020] – map pos 2 object 0x60FF

   Tx: [resp: 0x60] [Index: 1600] [sub-index: 0x02] [Data: 0x00000000] – Successful

5. **Enable map with number object mapped = 2:**

   RX: Id 0x601 – 2F 00 16 00 02 00 00 00

   Tx: Id 0x581 – 60 00 16 00 00 00 00 00

   RX: [cmd: 0x2F] [Index: 1600] [sub-index: 0x00] [Data: 0x00000002] – num. 2 object mapped

   Tx: [resp: 0x60] [Index: 1600] [sub-index: 0x00] [Data: 0x00000002] – Successful

6. **Enable PDO and Set COB-ID (0x201):**

   RX: Id 0x601 – 23 00 14 01 01 02 00 00

   Tx: Id 0x581 – 60 00 14 01 00 00 00 00

   RX: [cmd: 0x23] [Index: 0x1400] [sub-index: 0x01] [Data: 0x00000201] – Enable PDO (bit 31=0) and set COBID=0x201

   Tx: [resp: 0x60] [Index: 0x1400] [sub-index: 0x01] [Data: 0x00000000] – Successful

Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

Transmit PDO (TPDO)

1800h – 1803h: TPDO Communication Objects

These objects contain the communication parameters for the PDOs the device is able to transmit.

- Sub-index 00h contains the number of valid entries within the communication record
- Sub-index01h is the COB-ID of the PDO, this entry has been defined ad UNSIGNED32 in order to define for 11-bit CAN identifiers (CAN 2.A) as well as for 29 bit identifiers (CAN 2.0B)

31 (MSB)	30	29	28 ... 11	10 ... 0 (LSB)
Valid	RTR	frame	0000h	11-bit CAN-ID
			29-bit CAN-ID	

Name	BIT	VALUE	Meaning
Valid	31	0b 1b	PDO exist / is valid PDO does not exist /is not valid
RTR	30	0b 1b	RTR allowed RTR not allowed
frame	29	0b 1b	Message standard 11 bit (CAN 2.0 A) Message extended 29 bit (CAN 2.0 B)
29-bit CAN-ID	11 - 28	x	If Bit 29 = 1 : bits 28-11 of COB-ID message extended
11-bit CAN-ID	0 – 10	x	bits 10-0 of COB-ID

- Sub-index 02h is the transmission type, it defines the transmitting character of the PDO.

  The following table describes the usage entry.

Transmission Type		Meaning
0	00h	synchronous (acyclic)
1 - 240	01h– F0h	synchronous (cyclic every N sync)
241 - 251	F0h – FBh	reserved
252	FCh	RTR-only (synchronous)
253	FDh	RTR-only (event-driven)
254	FEh	event-driven (manufacturer-specific)
255	FFh	event-driven (device profile and application profile specific)

  - Synchronous means that the PDO is transmitted after the SYNC. The CANopen device will start sampling of the data with the reception of the SYNC. In case it is acyclic the CANopen device internal event is given and with the next SYNC he sampling is started and the PDO is transmitted afterwards. In case it is cyclic the sampling is started with the reception of every SYNC, every 2nd SYNC, every 3rd SYNC, and s.o. depending on the given value and the PDO is transmitted afterwards.

  - RTR-only means that the PDO is not transmitted normally it shall be requested via RTR. In case it is synchronous the CANopen device will start sampling with the reception of every SYNC and then will buffer the PDO. In case it is event-driven the CANopen device will start sampling with the reception of the RTR and will transmit the PDO immediately.

  - Event-driven means that the PDO may be transmitted at any time based on the occurrence of a CANopen device internal event. The definition of the event does not fall into the scope of this specification and may be specified in device profiles and application profiles.

- Sub-index 03h contains the inhibit time. The time is the minimum interval for PDO transmission if the transmission type is set to FEh and FFh. The value is defined as multiple of 100 µs. The value of 0 shall disable the inhibit time. The value shall not be changed while the PDO exists (bit 31 of sub-index 01h is set to 0b)

- Sub-index 04h is reserved. It does shall not be implemented; in this case read or write access leads to the SDO abort transfer service (abort code: 0609 0011h).
- Sub-index 05h contains the event-timer. The time is the maximum interval for PDO transmission if the transmission type is set to FEh and FFh. The value is defined as multiple of 1 ms. The value of 0 shall disable the event-timer.
- Sub-index 06h contains the SYNC start value. The SYNC start value of 0 shall indicate that the counter of the SYNC message shall not be processed for this PDO. The SYNC start value 1 to 240 shall indicate that the counter of the SYNC message shall be processed for this PDO. In case the counter of the SYNC message is not enabled sub-index 06h shall be ignored. The SYNC message of which the counter value equals the SYNC start value shall be regarded as the first received SYNC message. The value shall not be changed while the PDO exists (bit 31 of sub-index 01h is set to 0b). ***(it is not available)***

It is possible to change the parameters when the drive is in PRE-OPERATIONAL state. Follow the procedure to change a parameter:

- Disable the PDO: set Bit 31 of COB-ID (sub-index 01h) at "1"
- Write new value in sub-index corresponding
- Enable the PDO writing the COB-ID to configure the PDO enabled.

Example:

Modify the Event Timer 100ms for TPDO1 (Id-Node 1)

1. **Disable TPDO1:**

   Rx: Id 0x601 - 23 00 18 01 00 00 00 80

   Tx: Id 0x581 – 60 00 18 01 00 00 00 00

   Rx: [cmd: 0x23] [Index: 0x1800] [sub-index: 0x01] [Data: 0x80000000] – Disable PDO (bit 31= 1)

   Tx: [resp: 0x60] [Index: 0x1800] [sub-index: 0x01] [Data: 0x00000000] - Successful

2. **Change event timer:**

   Rx: Id 0x601 – 2B 00 18 05 64 00 00 00

   Tx: Id 0x581 – 60 00 18 05 00 00 00 00

   Rx: [cmd: 0x2B] [Index: 0x1800] [sub-index: 0x05] [Data: 0x00000064] – event timer = 100ms =0x64

   Tx: [resp: 0x60] [Index: 0x1800] [sub-index: 0x05] [Data: 0x00000000] – Successful

3. **Enable PDO And Set COB-ID:**

   Rx: Id 0x601 – 23 00 18 01 81 01 00 00

   Tx: Id 0x581 – 60 00 18 01 00 00 00 00

   Rx: [cmd: 0x23] [Index: 0x1800] [sub-index: 0x01] [Data: 0x00000201] – Enable PDO (bit 31=0) and set COBID=0x181

   Tx: [resp: 0x60] [Index: 0x1800] [sub-index: 0x01] [Data: 0x00000000] – Successful

Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

1A00h – 1A03h: TPDO Mapping Parameters

These objects contain the mapping for the PDOs device is able to transmit.

- Sub-index 00h contains the number of valid object entries within the mapping record or a specific value. The number of valid object entries shall be the number of the application objects that shall be received with the corresponding RPDO.

Value	Description
00h	Mapping disabled
01h	Sub-index 01h valid
02h	Sub-index 01h and 02h valid
…	…

- Sub-index from 01h to 40h contains the information of the mapped application objects. The object describes the content of the PDO by their index, sub-index and length. The length contains the length of the application object in bit. This may be used to verify the mapping.

Byte: MSB LSB

Index (16 bit)	Sub-Index (8bit)	Object Length (8 bit)

Example:

- statusword: 0x60410010 = 6041h (index) 00h (subindex) 10h (length 2 bytes)
- mode of operation display: 0x60610008 = 6041h (index) 00h (subindex) 08h (length 1 byte)
- actual velocity: 0x606C0020 = 60FFh (index) 00h (subindex) 20h (length 4 bytes)

If the change of the PDO mapping cannot be executes (e.g. the PDO length is exceeded or the SDO client attempts to map an object that cannot be mapped) the drive responds with an Abort Code SDO transfer service.

Lafert Servo Drive has available 8 record of 32 byte for mapping. The user can map until 8 object (every object is 1 byte) because the total length must be less than or equal 64 bits.

TPDO Mapping Default

- TPDO1: 180 h + Node ID
- TPDO2: 280 h + Node ID
- TPDO3: 380 h + Node ID
- TPDO4: 480 h + Node ID

The following tables describe the default mapping for TPDO:

Index	SubIndex	Description	Type	Attr.	Dafault Value	Description
Transmit Process Data Object (TPDO1)						
1800 h	0	Transmit 1st PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO1	UNSIGNED32	rw	180 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FF h	Asynchronous
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]

1A00 h	0	N transmit PDO mapping	UNSIGNED8	rw	3	Number of Entries
	1	1 - application object	UNSIGNED32	rw	60410010 h	Status word (2 byte)
	2	2 - application object	UNSIGNED32	rw	60610008 h	Mode Of Operation Display (1 byte)
	3	3 - application object	UNSIGNED32	rw	60FD0020 h	Digitals Inputs
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-
Transmit Process Data Object (TPDO2)						
1801 h	0	Transmit 2nd PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO2	UNSIGNED32	rw	280 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FD h	Asynchronous RTR
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]
1A01 h	0	N transmit PDO mapping	UNSIGNED8	rw	2	Number of Entries
	1	1 - application object	UNSIGNED32	rw	60410010 h	Status word (2 byte)
	2	2 - application object	UNSIGNED32	rw	60640020 h	Position Actual Value (4 byte)
	3	3 - application object	UNSIGNED32	rw	0	-
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-
Transmit Process Data Object (TPDO3)						
1802 h	0	Transmit 3rd PDO	UNSIGNED8	ro	5	Number of Entries
	1	COB ID used by PDO3	UNSIGNED32	rw	380 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FD h	Asynchronous RTR
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]
1A02 h	0	N transmit PDO mapping	UNSIGNED8	rw	2	Number of Entries
	1	1 - application object	UNSIGNED32	rw	60410010 h	Status word (2 byte)
	2	2 - application object	UNSIGNED32	rw	606C0020 h	Velocity Actual Value (4 byte)
	3	3 - application object	UNSIGNED32	rw	0	-
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-
Transmit Process Data Object (TPDO4)						
1803 h	0	Transmit 4th PDO	UNSIGNED8	ro	3	Number of Entries
	1	COB ID used by PDO4	UNSIGNED32	rw	480 h +NodeID	PDO enabled
	2	Transmission Type	UNSIGNED8	rw	FD h	Asynchronous RTR
	3	Inhibit Time	UNSIGNED16	rw	5 h	500 us [n*100us]
	4	Compatibility Entry	UNSIGNED8	rw	0	Not available
	5	Event Timer	UNSIGNED16	rw	0	0 = disabled [ms]
1A03h	0	N transmit PDO mapping	UNSIGNED8	rw	2	Number of Entries
	1	1 - application object	UNSIGNED32	rw	60410010 h	Status word (2 byte)
	2	2 - application object	UNSIGNED32	rw	60770010 h	Torque Actual Value (2 byte)
	3	3 - application object	UNSIGNED32	rw	0	-
	4	4 - application object	UNSIGNED32	rw	0	-
	5	5 - application object	UNSIGNED32	rw	0	-
	6	6 - application object	UNSIGNED32	rw	0	-
	7	7 - application object	UNSIGNED32	rw	0	-
	8	8 - application object	UNSIGNED32	rw	0	-

Table 14 - TPDO Description

Mapping default TPDO 1: status of State machine – mandatory

Index + subindex	Name Transmit PDO 1	Default Value	Size Bytes
1A00h: 0	Number of mapped objects	3	
1A00h: 1	Status word	6041 0010h	2
1A00h: 2	Mode of operation Display	6061 0008h	1
1A00h: 3	Digital Input	60FD 0120h	4

Table 15 - TPDO1 Mapping

Mapping default TPDO 2:status of State machine and current position– optional

Index + subindex	Name Transmit PDO 1	Default Value	Size Bytes
1A01h: 0	Number of mapped objects	2	
1A01h: 1	Status word	6041 0010h	2
1A01h: 2	Position Actual Value	6064 0020h	4

Table 16 - TPDO2 Mapping

Mapping default TPDO 3: status of State machine and current velocity - optional

Index + subindex	Name Transmit PDO 1	Default Value	Size Bytes
1A02h: 0	Number of mapped objects	2	
1A02h: 1	Status word	6041 0010h	2
1A02h: 2	Velocity Actual Value	606C 0020h	4

Table 17 - TPDO3 Mapping

Mapping default TPDO 4: status of State machine and current torque – optional

Index + subindex	Name Transmit PDO 1	Default Value	Size Bytes
1A03h: 0	Number of mapped objects	2	
1A03h: 1	Status word	6041 0010h	2
1A03h: 2	Torque Actual Value	6077 0010h	2

Table 18 - TPDO4 Mapping

information

Referring to "APPENDIX – EXAMPLE PROGRAMS" chapter to read the example for "Control via PDO"

Re-Mapping Procedure:

The following procedure shall be used for re-mapping, which may take place during the NMT state Pre-operational:

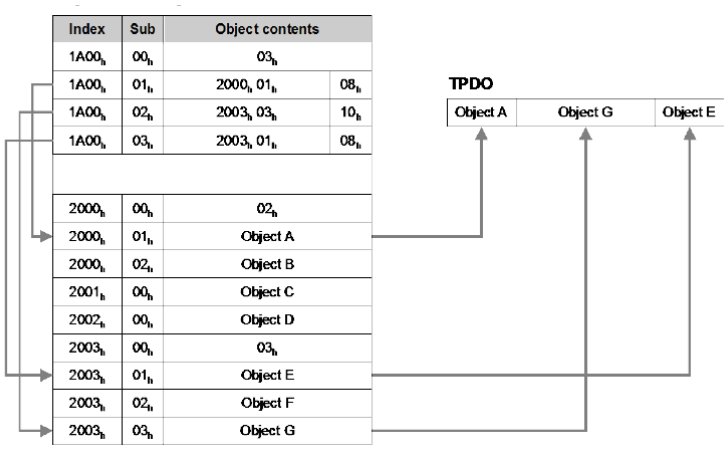
1) For changing the PDO mapping first the PDO has to invalidate the PDO. Destroy the bit valid into sub-index 01h of "TPDO Communication" objects (1800h, 1801h, 1802h and 1803h). The 31 bit must be set to 1.
2) Disable mapping PDO setting 0 into sub-index 00h of "Mapping Parameters" object (1A00h, 1A01h, 1A02h and 1A03h). This will disable PDO.
3) Modify mapping by changing the values of the corresponding sub-indices. Write in sub-index correspondent the description of the object (Index, Sub-Index and Length)
4) Set the sub-index 00h of PDO coordinated mapping parameter (1A00h, 1A01h, 1A02hand 1A03h)as legal number (number of PDO's mapping objects). This will enable new mapping.
5) Create RPDO by setting bit valid to 0 of sub-Index 01h(COB-ID)of "communication object" (1800h, 1801h,1802h and 1803h) the according TPDO communication parameter.
6) PDO mapping completing

information

Referring to "APPENDIX – EXAMPLE PROGRAMS" chapter to read the example for "Remapping TPDO procedure"

If during step 3 the CANopen device detects that index and sub-index of the mapped object does not exist or the object cannot be mapped the CANopen device shall respond with the SDO abort transfer service (abort code: 0602 0000_h or 0604 0041_h).

If during step 4 the CANopen device detects that the RPDO mapping is not valid or not possible the CANopen device shall respond with the SDO abort transfer service (abort code: 0602 0000_h or 0604 0042_h).



Example:

Remap TPDO1 with statusword and actual velocity (Id-Node 1)

1. **Disable TPDO1:**

   Rx: Id 0x601 - 23 00 18 01 00 00 00 80

   Tx: Id 0x581 – 60 00 18 01 00 00 00 00

   Rx: [cmd: 0x23] [Index: 0x1800] [sub-index: 0x01] [Data: 0x80000000] – Disable PDO (bit 31= 1)

   Tx: [resp: 0x60] [Index: 0x1800] [sub-index: 0x01] [Data: 0x00000000] - Successful

2. **Disable Mapping of TPDO1:**

   Rx: Id0x601 – 2F 00 1A 00 00 00 00 00

   Tx: Id 0x581 – 60 00 1A 00 00 00 00 00

   Rx:[cmd: 0x2F] [Index: 0x1A00] [sub-index: 0x00] [Data: 0x00000000] – Disable Map PDO (Value = 0)

   Tx: [resp: 0x60] [Index: 0x1A00] [sub-index: 0x00] [Data: 0x00000000] - Successful

3. **Map in Pos 1 object statusword (0x6041):**

   Rx: Id0x601 – 23 00 1A 01 10 00 41 60

   Tx: Id 0x581 – 60 00 1A 01 00 00 00 00

   Rx:[cmd: 0x23] [Index: 1A00] [sub-index: 0x01] [Data: 0x60410010] – map pos 1 object 0x6041

   Tx: [resp: 0x60] [Index: 1A00] [sub-index: 0x01] [Data: 0x00000000] – Successful

4. **Map in Pos 2 object actual velocity (0x606C):**

Rx: Id0x601 – 23 00 1A 02 20 00 6C 60

Tx: Id 0x581 – 60 00 1A 02 00 00 00 00

Rx:[cmd: 0x23] [Index: 1A00] [sub-index: 0x02] [Data: 0x606C0020] – map pos 2 object 0x606C

Tx: [resp: 0x60] [Index: 1A00] [sub-index: 0x02] [Data: 0x00000000] – Successful

5. **Enable map with number object mapped = 2:**

Rx: Id0x601 – 2F 00 1A 00 02 00 00 00

Tx: Id 0x581 – 60 00 1A 00 00 00 00 00

Rx:[cmd: 0x2F] [Index: 1A00] [sub-index: 0x00] [Data: 0x00000002] – num. 2 object mapped

Tx: [resp: 0x60] [Index: 1A00] [sub-index: 0x00] [Data: 0x00000000] – Successful

4. **Enable PDO and Set COB-ID (0x181):**

Rx: Id0x601 – 23 00 18 01 81 01 00 00

Tx: Id 0x581 – 60 00 18 01 00 00 00 00

Rx:[cmd: 0x23] [Index: 0x1800] [sub-index: 0x01] [Data: 0x00000181] – Enable PDO (bit 31=0) and set COBID=0x181

Tx: [resp: 0x60] [Index: 0x1800] [sub-index: 0x01] [Data: 0x00000000] – Successful

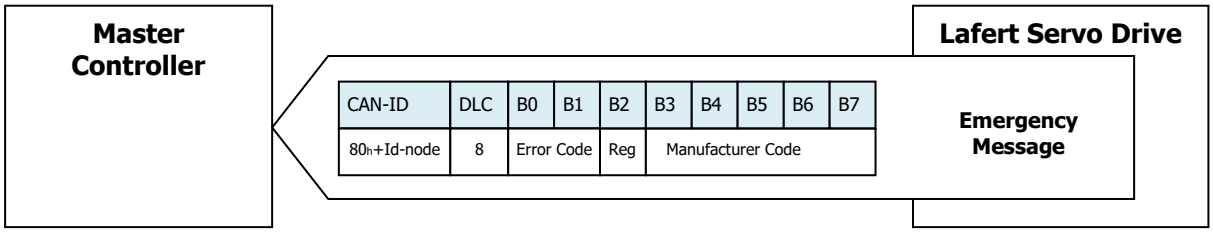
Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

EMERGENCY MESSAGE (EMCY)

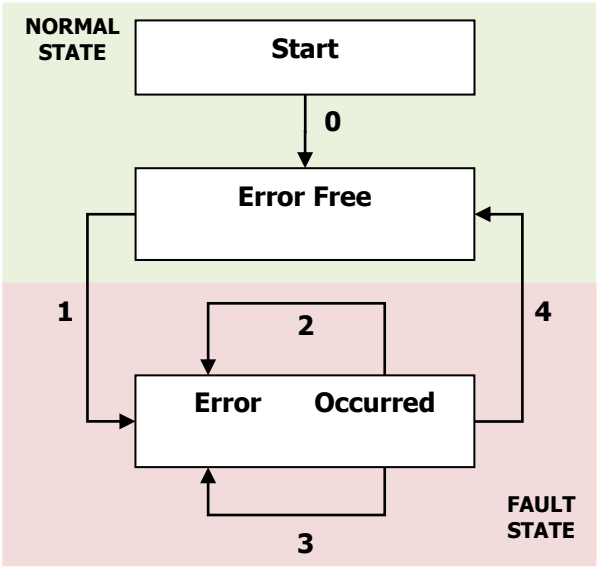
The Lafert Servo Drive monitors the function of internal modules and of the firmware.

Whenever an error occurs, the parameterised error response is initiated and the corresponding EMCY message is transmitted.



The latest error message is always stored here in Error Code object (603Fh:0h).

Also, it is in the highest error memory slot (1003h: 01h), the error memory always saves the 15 most recent error messages that can also be read out.



The following status transitions are possible:

Transition	Cause	Description
0	Initialisation completed	There is no error. The drive sends the error code 0000h (Error reset/No error)
1	Error occurs	No error was present and a new error occurs. The drive goes to Fault State. Verify the diagnostic state and the Emergency message.
2	Error acknowledgment not successful	Not all causes of error have been remedied and an error acknowledgement was performed.
3	New error occurs	There is an error and a new error occurs. An EMCY message with the error code for the new error (1003h: 01h, standard error field 1) is written.
4	Error acknowledgment successful	All causes of error have been remedied and an error acknowledgement was performed. The EMCY message was transmitted with error code 0000h (Error reset/No error).

Emergency objects are triggered by the incident of a CANopen device internal error situation and are transmitted on to the network.

Emergency objects are suitable for error alerts.

Emergency message structure by CanOpen DSP402:

COB-ID	Rx/Tx	DLC	Byte							
			0	1	2	3	4	5	6	7
80 h +IdNode	Tx	8	Error Code		Reg	Manufacturer specific error field				
			E0	E1	R0	M0	M1	M2	M3	M4

Table 19 - Emergency Message Structure

Error Code field standard by DS301:

Error Code	Name	Description
0x0000	NO ERROR	error Reset or No Error
0x1000	GENERIC ERROR	Generic Error
0x2000	CURRENT	Current
0x2100	CURRENT INPUT	Current, device input side
0x2200	CURRENT INSIDE	Current inside the device
0x2300	CURRENT OUTPUT	Current, device output side
0x3000	VOLTAGE	Voltage
0x3100	VOLTAGE MAINS	Mains Voltage
0x3200	VOLTAGE INSIDE	Voltage inside the device
0x3300	VOLTAGE OUTPUT	Output Voltage
0x4000	TEMPERATURE	Temperature
0x4100	TEMP AMBIENT	Ambient Temperature
0x4200	TEMP DEVICE	Device Temperature
0x5000	HARDWARE	Device Hardware
0x6000	SOFTWARE DEVICE	Device Software
0x6100	SOFTWARE INTERNAL	Internal Software
0x6200	SOFTWARE USER	User Software
0x6300	DATA SET	Data Set
0x7000	ADDITIONAL MODULE	Additional Modules
0x8000	MONITORING	Monitoring
0x8100	COMMUNICATION	Communication
0x8200	PROTOCOL ERROR	Protocol Error
0x9000	EXTERNAL ERROR	External Error
0xF000	ADDITIONAL FUNC	Additional Functions
0xFF00	DEVICE SPECIFIC	Device specific

Table 20 - Emergency Error Code

Register field standard by DS301: CANopen device maps internal errors into this object. The bit 0 is the generic error and it is mandatory when error fault is occurred, the others bits specific different type error.

Reg	BIT	NAME	Description
0x00		NO ERROR	none error
0x01	1	REGISTER GENERIC ERROR	generic error
0x02	2	REGISTER CURRENT	current
0x04	3	REGISTER VOLTAGE	voltage
0x08	4	REGISTER TEMPERATURE	temperature
0x10	5	REGISTER COMMUNICATION ERROR	communication error (overrun, error state)
0x20	6	REGISTER DEVICE PROFILE	device profile specific
0x40	7	REGISTER RESERVED	reserved (always 0)
0x80	8	REGISTER MANUFACTURER	manufacturer specific

Table 21- Emergency Register Field

The following table defines the alarms group (Fault / Warning) implemented in Lafert with CANopen code.

The "Led Code" column describes the number of blinking of led.
For example [x, y] = 6,2 means the green Led blinks 6 times, after that, the yellow led blinks 2 times.

"BLINK" code [x]

"BLINK" code [y]

The "Error Code" describes the univocal value of alarm. The last alarm occurred can be read with $603F_h$ object "Error Code".

Some alarms have the sub-codes defined by manufacturer. The column meaning describes the Manufacturer specific error field.

The alarm can be **Fault** (F) or **Warning**(W), if it is a fault the drive will stop.

Error	Error Code	Description	Meaning	F - W	Led Code
NO ERROR	**0x0000**	**No Error**	The Fault Reset command has been executed or there was a reset with power cycle	-	-
	0x0001	**No error After Reset alarm Comand**	The Reset Alarm command has been executed, the drive is working	-	-
		ALARM CURRENT			
SHORT CIRCUIT MOTOR	**0x2340**	**Short circuit (motor-side)**	Alarm Over Current has been occurred	**F**	**3,1**
LOAD LEVEL FAULT	**0x2350**	**Load level fault (I2t, thermal state)**	Alarm Over Current with integral i^2t (Over Load)	**F**	**5,2**
	0x2351	Warning (I2t, thermal state)	Warning Limitation i^2t (Over Load)	**W**	-
	0x2352	Load Level (i2t) not rearmed	Alarm Over Current with integral i^2t (Over Load) not Rearmed (protection i^2t)	**F**	**5,2**
	0x2353	Load Level (i2t) not rearmed Warning	Warning to communicate the protection of maximum number time (Over Load) occurred	**W**	-
		ALARM VOLTAGE			
OVER VOLTAGE	**0x3210**	**DC link over-voltage**	Over Voltage alarm has been occurred	**F**	**4,2**
	0x3211	**Over-voltage no.1**	Over Voltage alarm has been occurred (FAST FUNCTION)	**F**	**4,2**
DC LINK UNDER VOLTAGE	**0x3220**	**DC link under-voltage**	Under Voltage alarm has been occurred	**F**	**4,1**
		ALARM TEMPERATURE			
TEMPERATURE DRIVE	**0x4300**	**Temperature Drive**	Over Temperature Heat Sink	**F**	**1,1**
	0x4301	Warning temperature drive	Warning temperature drive	**W**	-
	0x4310	Excess temperature drive	Heat SinkTemperature too high of maximum Range	**F**	**1,3**
	0x4320	Too low temperature drive	Heat SinkTemperature too low of minimum Range	**F**	**1,3**
TEMPERATURE INTERNAL 1 – BOARD	**0x4500**	**Temperature Logic Board**	Over Temperature Logic Board	**F**	**1,4**
	0x4501	Warning Logic Board temperature	Warning Logic Board temperature	**W**	-
	0x4510	Excess Logic Board temperature	Board Temperature too high of maximum Range	**F**	**1,5**
	0x4520	Too low Logic Board temperature	Board Temperature too low of minimum Range	**F**	**1,5**
TEMPERATURE EXTERNAL 1 - MOTOR	**0x4A00**	**Temperature Motor**	Over Motor Temperature	**F**	**1,10**

	0x4A01	Warning temperature Motor	Warning Motor Temperature	**W**	**-**
	0x4A10	Excess temperature Motor	MotorTemperature too high of maximum Range	**F**	**1,6**
	0x4A20	Too low temperature Motor	Motor Temperature too low of minimum Range	**F**	**1,6**
		ALARM HARDWARE			
INPUT STAGES	**0x5430**	**Input stages**	Generic Input Stages	**-**	**-**
	0x5431	Offset Sensor	Offset Sensor	**F**	**3,10**
HARDWARE MEMORY	**0x5500**	**Hardware Memory**	Generic Hardware Memory	**-**	**-**
	0x5501	HardwareError Write EEprom: Vbus too Low	Write is not possible because the Bus Voltage is too low to guarantee the writing complete	**F**	**5,3**
HARDWARE MEMORY E²PROM - USER	**0x5530**	**E²PROM**	Generic Error E²prom	**-**	**-**
	0x5531 .. 0x556F	E²prom General Error 1-65	Generic Error E²prom Writing (contact Manufacturer)	**F**	**6,1**
HARDWARE MEMORY E²PROM - FACTORY PARAMETERS	**0x5A00**	**E²PROM Data Area Factory Parameters Image**	Generic Error E²prom Data Area Golden Image	**-**	**-**
	0x5A01	Warning Data Factory Parameters	Warning Data Golden Image is free	**W**	**-**
	0x5A02	Error Data Factory Parameters	Data Golden Image is not written	**F**	**8,1**
HARDWARE MEMORY EEPROM - FUNCTION R/W EEPROM	**0x5B00**	**EEPROM Function R/W**	Generic Error E²prom Data Function Read/Write	**-**	**-**
	0x5B01 .. 0x5B19	Error Eeprom Function 1-25	Error Read or Write E²prom (contact Manufacturer)	**F**	**8,3**
		ALARM SOFTWARE			
SOFTWARE DEVICE	**0x6000**	**Software Device**	Generic Error Software Device	**-**	**-**
	0x6001	Update Parameters	Warning Request update by canopen is not permission (ONLY RS232)	**W**	**-**
		ALARM PARAMETERS			
DATA SET	**0x6300**	**Data Set Parameters Table**	Data Set Programming Error	**-**	**-**
	0x6301 ... 0x630F	Data record no. 1 - 15	Programming Error Data Set (contact Manufacturer)	**F**	**7,1**
	0x6401	Data record no. 16	Programming Error Data Set (contact Manufacturer) Attention Manufacture Code: number 1 - 19	**F**	**7,1**
	0x6402 ... 0x6403	Data record no. 17-18	Programming Error Data Set (contact Manufacturer)	**F**	**7,1**
	0x6404	Data record no. 19	Programming Error Data Set (contact Manufacturer) Attention Manufacture Code: number 1 - 20	**F**	**7,1**
	0x6405 ... 0x6423	Data record no. 20 - 50	Programming Error Data Set (contact Manufacturer)	**F**	**7,1**
PARAMETER ERROR	**0x6320**	**Parameter Error**	Generic Parameter Error	**-**	**-**
	0x6321 ... 0x6326	Incongruity Data Configuration 1-6	Configuration Error (contact Manufacturer)	**F**	**6,4**
		ALARM ADDITIONAL MODULE			
ENCODER SINCOS	**0x7350**	**Encoder SinCos**	Error Generic Encoder SinCos	**F**	**2,6**
	0x7351	Rx Error	Error Message Received	**F**	**2,6**
	0x7352	Tx Error	Error Message Transmitted	**F**	**2,6**
	0x7353	Comand Read Position Error	Error Read Position	**F**	**2,6**

	0x7354	Comand Status Error	Error Status Encoder SinCos	**F**	**2,6**
	0x7355	Comand Type Error	Error Type Encoder SinCos	**F**	**2,6**
	0x7356	Comand Init Timeout	Error Timeout during Initialization SinCos	**F**	**2,6**
CONVERTER SINCOS	**0X7360**	**Converter Sin/Cos**	Error Generic Converter Sin/Cos	**F**	**6,3**
	0x7361	E^2prom Ext	First programming E^2prom external, we must reset the driver	**F**	**6,3**
	0x7362	Nerr Signal Amp	Fault has been occurred: Amplitude Error	**F**	**6,3**
	0x7363	Nerr Signal Freq	Fault has been occurred: Frequency Error	**F**	**6,3**
	0x7364	Nerr Signal Other	Fault has been occurred: configuration or Under voltage or System Error	**F**	**6,3**
	0x7365	Error Gen	General Error	**F**	**6,3**
RESOLVER	**0x7370**	**Resolver**	General Error	**-**	**-**
	0x7373	Resolver not in phasing	Alignment Fault Initial of resolver during read	**F**	**2,4**
	0x7374	Resolver Initialization	Resolver Fault Initialization has been occurred	**F**	**2,4**
	0x7375	Resolver Hardware Fault LOS (Loss of Signal)	Manufacturer specific value describes the cause of the triggering of the fault detection output pins (value of fault register chip resolver).	**F**	**2,10**
	0x7376	Resolver Hardware Fault DOS (Degradation of Signal)	Manufacturer specific value describes the cause of the triggering of the fault detection output pins (value of fault register chip resolver).	**F**	**2,10**
	0x7377	Resolver Hardware Fault LOT (Loss of Tracking)	Manufacturer specific value describes the cause of the triggering of the fault detection output pins (value of fault register chip resolver).	**F**	**2,10**
	0x7378	Resolver Hardware Fault LOS, DOS, LOT during phasing initialisation	Manufacturer specific value describes the cause of the triggering of the fault detection output pins (value of fault register chip resolver).	**F**	**2,10**
INCREMENTAL ENCODER	**0x7390**	**Incremental Encoder**	Error Generic Incremental Encoder	**F**	**2, 5**
	0x7391	Encoder error init	Encoder has initialization error due to sequence Hall or value null	**F**	**2, 1**
	0x7392	Encoder error congruence	Encoder has congruence error between Hall	**F**	**2, 2**
	0x7393	Encoder error phasing	Encoder has phasing error	**F**	**2, 3**
	0x7394	Encoder error Distance	Encoder Error Distance Hall	**F**	**2, 4**
COMMUNICATION	**0x7500**	**Communication**			
	0x7520	**Serial interface no. 2 - ModBus**	ModBus Error Generic	**F**	**9,1**
	0x7521	Communication Lost	ModBus Communication Lost	**F**	**9,1**
	0x7530	**FieldBus no.1 - CANopen**	CANopen Error Generic	**-**	**-**
	0x7531	CANopen Protocol – Init Error	Initialization Error	**W**	**-**
	0x7532	CANopen Protocol – Hardware Error	hardware Error	**F**	**5,4**
		ALARM MONITORING			
COMUNICATION CANOPEN	**0x8100**	**Comunication Canopen**	**communication error**	**F**	**6,2**
	0x8110	Can Overrun	CAN Controller RX buffer hardware overrun (Overflow)	**F**	**6,2**
	0x8111	Tx Buffer Overflow	TX software buffer overflow	**F**	**6,2**
	0x8112	Rx Buffer Overflow	RX software buffer overflow	**F**	**6,2**

	0x8120	Can Passive	CAN in error passive	**F**	**6,2**
	0x8121	Can BusOff	CAN in error BUSOFF	**F**	**6,2**
	0x8130	Heartbeat/Node Guarding	Heartbeat or Life Node Guarding	**F**	**6,2**
	0x8131	Error Node Guarding slave misses msg	Error Node Guarding: slave misses guarding message	**F**	**6,2**
	0x8132	Error Node Guarding lost connection	Error Node Guarding: lost connection life time elapsed for node	**F**	**6,2**
	0x8133	Error Node Guarding lost at least one msg	Error Node Guarding: slave misses guarding at least one msg	**W**	**-**
	0x8135	NMT Stopped Error	Abort Communication caused node in Stopped Error in NMT protocol	**F**	**6,2**
	0x8136	Reset Application	Abort Communication caused Reset Application command in NMT protocol	**F**	**6,2**
	0x8137	Reset communication	Abort Communication caused Reset Communication command in NMT protocol	**F**	**6,2**
	0x8140	Bus Off Recovered	CAN recovered from bus-off	**W**	**-**
	0x8150	Can Id Collision	CAN-ID collision	**W**	**-**
	0x8160	State CAN Init	Drive communicates State Message :CANopen is in INIT state	**W**	**-**
	0x8170	State CAN Active	Drive communicates State Message :CANopen is in ACTIVE state	**W**	**-**
	0x8180	State CAN Busoff	Drive communicates State Message :CANopen is in BUSOFF state	**W**	**-**
	0x8190	State CAN Error Passive	Drive communicates State Message :CANopen is in PASSIVE state	**W**	**-**
ALARM PROTOCOL					
PROTOCOL ERROR	**0x8200**	**Protocol Error**	Error Protocol general	**W**	**-**
	0x8210	Protocol PDO – Error Length	PDO not processed due the length	**W**	**-**
	0x8220	Protocol PDO – Error PDO Length Exc	PDO length exceeded	**W**	**-**
	0x8230	Protocol PDO – Error MPDO	DAM MPDO not processed, destination object not available	**W**	**-**
	0x8240	Protocol PDO – Error Sync Data Length	Unexpected SYNC data length	**W**	**-**
	0x8250	Protocol PDO – Error RPDO Timeout	RPDO timeout	**W**	**-**
TORQUE PROFILE CONTROL	**0x8300**	**Torque control**	**General Error for Profile Torque Controller**	**F**	**6,6**
	0x8341	Torque Type	Error type selected is not managed	**F**	**6,6**
	0x8351	Torque Dynamic Stop	Error Dynamic Stopis not implemented	**F**	**6,6**
VELOCITY SPEED CONTROLLER	**0x8400**	**Velocity speed controller**	**General Error for Velocity Speed Controller**	**F**	**6,7**
	0x8410	Following error Current Mode	The difference between the velocity command and the actual velocity is greater than the value that is set in maximum velocity error. The drive is in Torque Profile	**F**	**5,10**
	0x8412	Over Speed	Actual speed exceeds the velocity over speed value	**F**	**5,8**
POSITION CONTROLLER	**0x8500**	**Position controller (N/A)**	**General Error for Profile Positioner Controller**	**F**	**-**
	0x8501	Warning - Limit disabled (N/A)	Warning: Limit of Position are disabled	**W**	-
	0x8502	Warning - Limitation (N/A)	Warning: Target Position is in limitation	**W**	-
	0x8551	Configuration Error (N/A)	Error setting configuration of position controller: Attention Manufacture Code: number 1,2,3,4	**F**	**10,10**
	0x8552	Finite State Machine Position (N/A)	Error Finite State Machine of Generator trajectory of Positioner	**F**	**10,10**

	0x8553	Calculation Settings (N/A)	Error during Settings Calculation (Profile Acceleration, Deceleration and Velocity)	**F**	**10,10**
POSITIONING CONTROLLER	**0x8600**	**Positioning controller (N/A)**	**General Error for Profile Positioning Controller**	**F**	**10,10**
HOMING CONTROLLER	0x8651	Method Error (N/A)	Error setting configuration of position controller: Attention Manufacture Code: number 1,2,3,4	**F**	**6,8**
	0x8652	Finite State Machine Homing (N/A)	Error Finite State Machine of Generator trajetor of homing	**F**	**6,8**
	0x8653	Calculation Settings (N/A)	Error during Settings Calculation (Profile Acceleration, Deceleration and Velocity)	**F**	**6,8**
CANOPEN EEPROM	**0x8B00**	**Store and Restore Process**	**General Error for Store and Restore Process**	**F**	**8,2**
	0x8B01	Warning Store/Restore/ Load Parameters	Warning: command store/restore/load are disabled because the drive isn't in "ready to switch on" or "disabled"	**W**	-
	0x8B02	Store Parameters Error	Error management object Store Parameters 1010h	**F**	**8,2**
	0x8B03	Memory Store Eeprom Full	Error Memory Store full for CAN object parameters	**F**	**8,2**
	0x8B04	Restore Par Eeprom	Error management object ReStore Parameters 1011h	**F**	**8,2**
	0x8B05	Memory Restore Eeprom Full	Error Memory Restore full for CAN object parameters	**F**	**8,2**
	0x8B10	**Init values P402 from Eeprom**	**Error Initialization from Eeprom.**	**F**	**8,2**
	0x8B11	Init Object CANopen 0x6081	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B12	Init Object CANopen 0x6082	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B13	Init Object CANopen 0x6083	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B14	Init Object CANopen 0x6084	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B15	Init Object CANopen 0x60C5	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B16	Init Object CANopen 0x60C6	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B17	Init Object CANopen 0x607F	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B18	Init Object CANopen 0x6088	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B19	Init Object CANopen 0x6096	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B1A	Init Object CANopen 0x6097	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B1B	Init Object CANopen 0x606D	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B1C	Init Object CANopen 0x606E	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B1D	Init Object CANopen 0x606F	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B1E	Init Object CANopen 0x6070	Error Initialization value from Eeprom	**F**	**8,2**
	0x8B1F	Init Object CANopen 0x6075	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B20	Init Object CANopen 0x6076	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B21	Init Object CANopen 0x6072	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B22	Init Object CANopen 0x6073	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B23	Init Object CANopen 0x60E0	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B24	Init Object CANopen 0x60E1	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B25	Init Object CANopen 0x6087	Error Initialization value from Eeprom	**F**	**8,2**

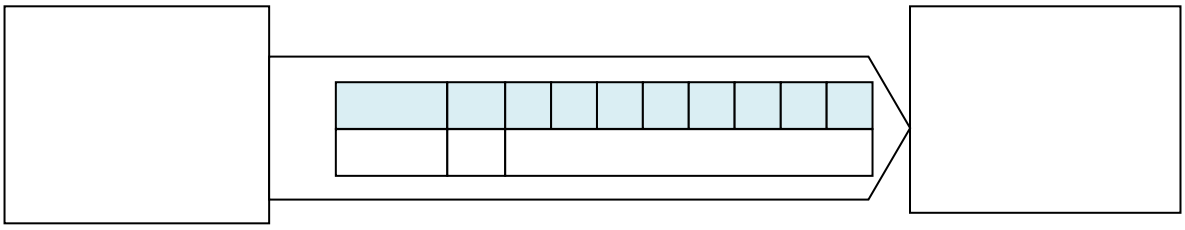
	0X8B26	Init Object CANopen 0x6086	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B27	Init Object CANopen 0x607B	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B28	Init Object CANopen 0x607D	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B29	Init Object CANopen 0x6099	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B2A	Init Object CANopen 0x609A	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B2B	Init Object CANopen 0x607C	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B2C	Init Object CANopen 0x6065	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B2D	Init Object CANopen 0x6066	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B2E	Init Object CANopen 0x6067	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B2F	Init Object CANopen 0x6068	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B30	Init Object CANopen 0x60F2	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B41	Init Object CANopen 0x60A2	Error Initialization value from Eeprom	**F**	**8,2**
	0X8B42	Init Object CANopen 0x60A4	Error Initialization value from Eeprom	**F**	**8,2**
DSP402 FSM	**0x8C00**	**Profile 402 Finite State Machine**	**General Error Profile 402 Finite State Machine**	**F**	**6,5**
	0x8C01	Mode Of Operation Error	Mode Of Operation (6060h) has been written when the drive is in "operation enabled" state	**F**	**6,5**
	0x8C02	Type Profile	Type Profile not defined	**F**	**6,5**
	0x8C03	Profile Error	Profile Selected not managed	**F**	**6,5**
	0x8C04	None Profile	Run State and No Profile selected	**F**	**6,5**
	0x8C06	Protocol Type	Protocol Type Warning (contact Manufacturer)	**W**	**-**
	0x8D00	Analog Profile	Analog Profile Condition Error	**F**	**6,5**
ALARM DEVICE SPECIFIC					
ALARM DEVICE SPECIFIC	**0xFF00**	**Manufacturer Specific**	**Alarm Manufacturer Specific**	**-**	-
	0xFF01	Safety Fault *(* not available)*	STO Occurred	**F**	**3,5**
LIMITATION	**0xFF10**	**LIMITATION**	**General Error Limitation**	**-**	**-**
	0xFF11	Warning - ECL Dynamic Tab	Warning - ECL Dynamic Tab	**W**	**-**
STO CIRCUIT (* Only Smartris Compact)	**0xFF20**	**STO Circuit**	**General Error STO circuit**	**-**	**-**
	0xFF21	STO - start control - 12V	Reserved (contact Manufacturer)	**F**	**3,4**
	0xFF22	STO - Run time - Level	Reserved (contact Manufacturer)	**F**	**3,4**
	0xFF23	STO - Run time - Congruence	Reserved (contact Manufacturer)	**F**	**3,4**
	0xFF24	STO - Run time - Feedback	Reserved (contact Manufacturer)	**F**	**3,4**

Table 22 - Emergency Description

SYNC Protocol

Several devices can be synchronised with each other. To that purpose one of the devices (in general the master controller) periodically sends a SYNC message.

The SYNC object is a network wide system clock and it is the trigger for synchronous message. The SYNC has a very high priority and contains no data in order to guarantee a minimum of jitter.



The SYNC object is used to trigger synchronous PDOs; all connected devices receive these messages and use them for the treatment of the PDO.

The identifier of the SYNC objects is set in the object dictionary under the index 1005_h (COB-ID of the SYNC Message), the default value is 0x80.

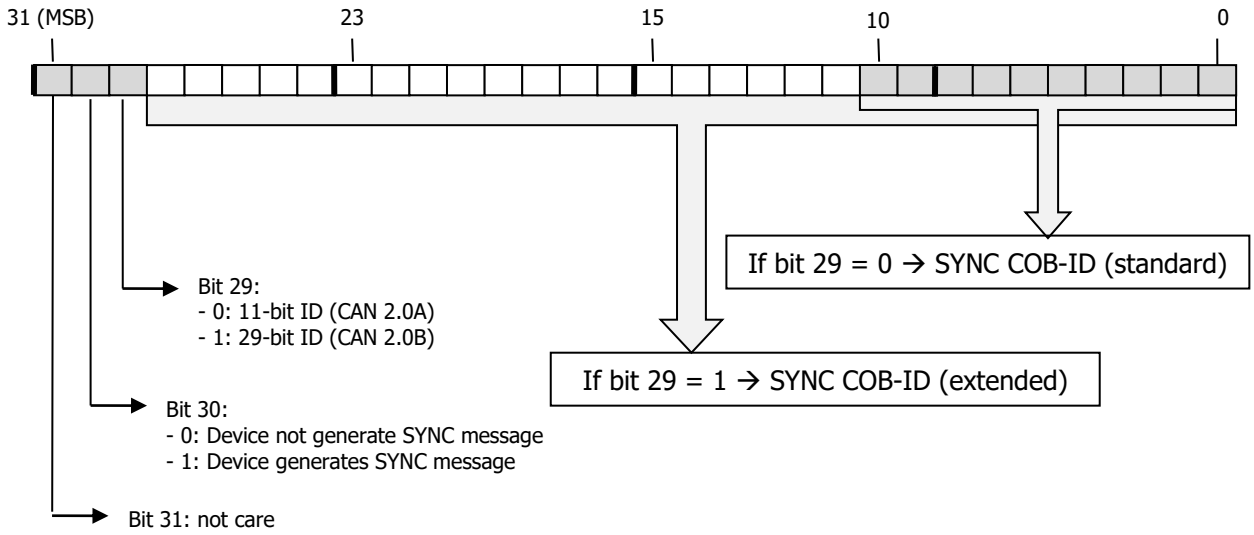


Figure 14 – COB-ID SYNC

information

The drive does not support SYNC message generation and only supports ID of bit 11. Therefore, bits 11 through 30 are always 0.

The drive can't send a SYNC message because the drive is only consumer. The master controller must send a SYNC message because it is producer.

The structure of SYNC message is:

COB-ID	Rx/Tx	DLC	Byte 0	1	2	3	4	5	6	7
0x80	**Tx**	**0**	No user data							

Table 23 - SYNC message Structure

Triggering synchronous PDOs:

- Synchronous RPDO: The command transmitted with the PDO is not executed until a SYNC object is received.
- Synchronous TPDO: The PDO with the current data is not sent until a SYNC object is received.

ERROR CONTROL PROTOCOLS

There are 2 error control protocols enable the monitoring of a CANopen network.

1. HeartBeat
2. Node/Life-Guarding

The Heartbeat protocol is used to verify that all network participants are still available in a CANopen network and that they are still in their intended NMT state. In old-fashioned CANopen systems, the CAN remote frame-based Node/Life-guarding protocol is used for this purpose, instead of the Heartbeat protocol.

All error control protocols are based on the same CAN message with the CAN-ID 700 $_h$ + Node-ID of the CANopen device that are to be monitored.

Caution

IT IS IMPORTANT AND MANDATORY TO HAVE AT LEAST ONE OF MONITOR CONTROL FOR CAN COMMUNICATION SAFETY.

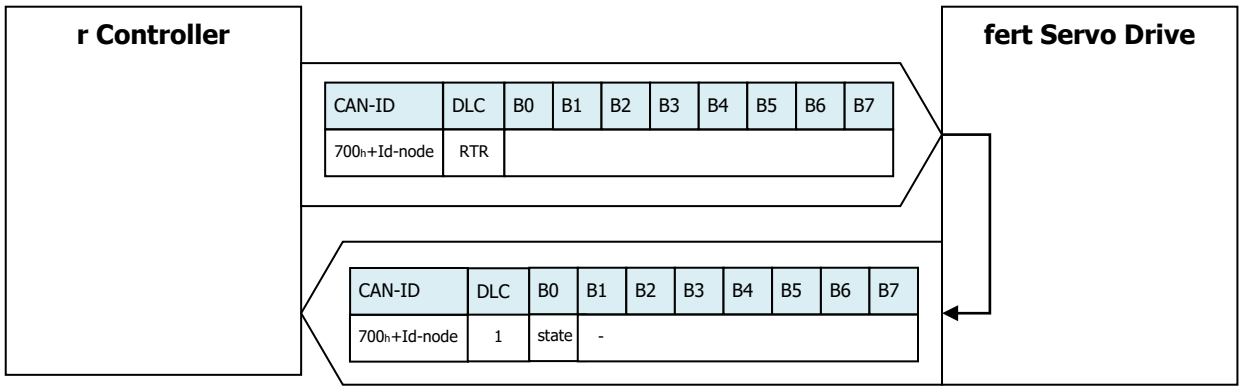
Caution

CAN in Automation (CiA) recommends using heartbeat in the new implementations due to the sensible nature of Guarding using the RTR frames. CiA no longer recommends using CAN remote frame-based services.

The default error control protocol is HEARTBEAT with timing of 200ms

Node Guarding Protocol

This service is based that the Master Controller sends an RTR message with the identifier 700$_h$+IdNode to the respective slave. The slave must send a message as response: this message is structured as follows.



Bit 7 alternates here on each transfer, this bit determines if a message was lost.

Bit 6 to 0 define the current NMT status of the slave.

COB-ID	Rx/Tx	DLC	Byte 0	1	2	3	4	5	6	7
700 $_h$ + IdNode	**Tx**	**1**	7Bit toggle +NMT State	-						

Table 24 - Node Guarding Message Structure

To configure the node guarding use three time intervals

- Guard time: the time between two RTR messages. This can be different for each CAN node and is stored in the slave in object $100C_h$"Guard Time" (unit ms).
- Live time factor: a multiplier for the guard time, this is stored in the CAN slave in object $100D_h$ "Life Time Factor" and it can be different for each slave on the CAN bus.
- Possible live time: the time produced by multiplying guard time and live time factor

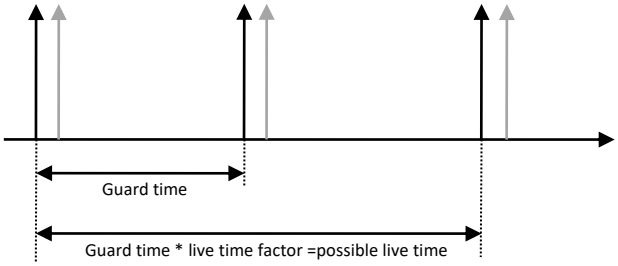


Figure 15 - Node Guarding time message

The following conditions are checked during node guarding:

- The NMT Master Controller must send the RTR requests within "possible live time"
- The slave must send the response to the RTR request within the "possible live time"
- The slave must respond with its NMT state. In addition the "toggle bit" must be set correctly

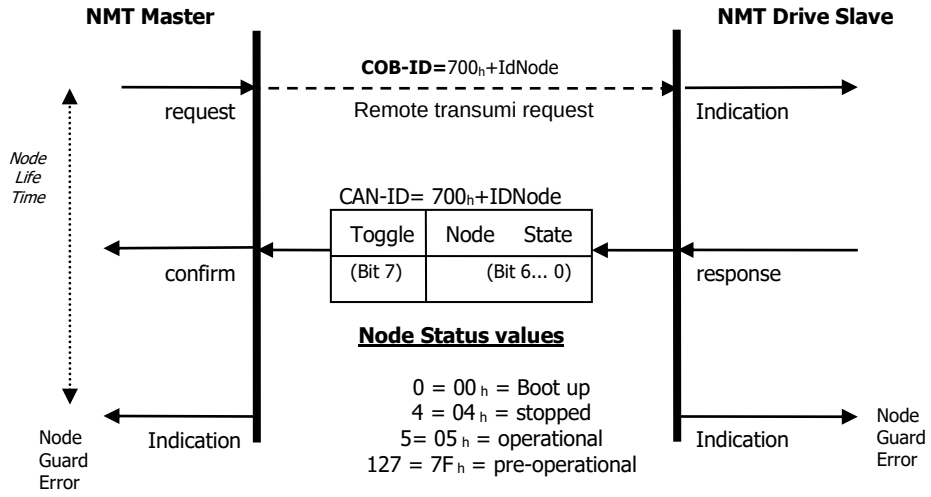
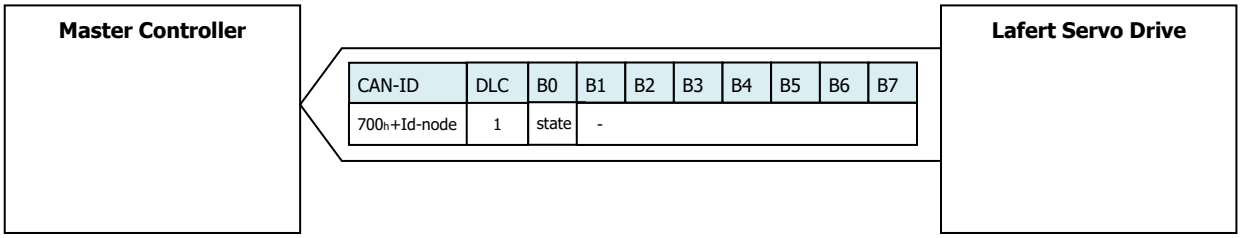


Figure 16 - Node Guarding timeframe message

Heartbeat Protocol

Heartbeat is the message to monitor the communication between drive and Master Controller.



The drive cyclically sends a message to the master controller. The master controller can check if it cyclically receives the heartbeat and initiate appropriate reactions if not. The heartbeat message will be sent with the identifier 700h +Id-Node. It is only composed of 1 Byte, containing the NMT state of the servo.

COB-ID	Rx/Tx	DLC	Byte							
			0	1	2	3	4	5	6	7
700 h + IdNode	**Tx**	**1**	NMT State	-						

Table 25 - HeartBeat Message Structure

This object indicates what action shall be performed when one of the following events occurs:

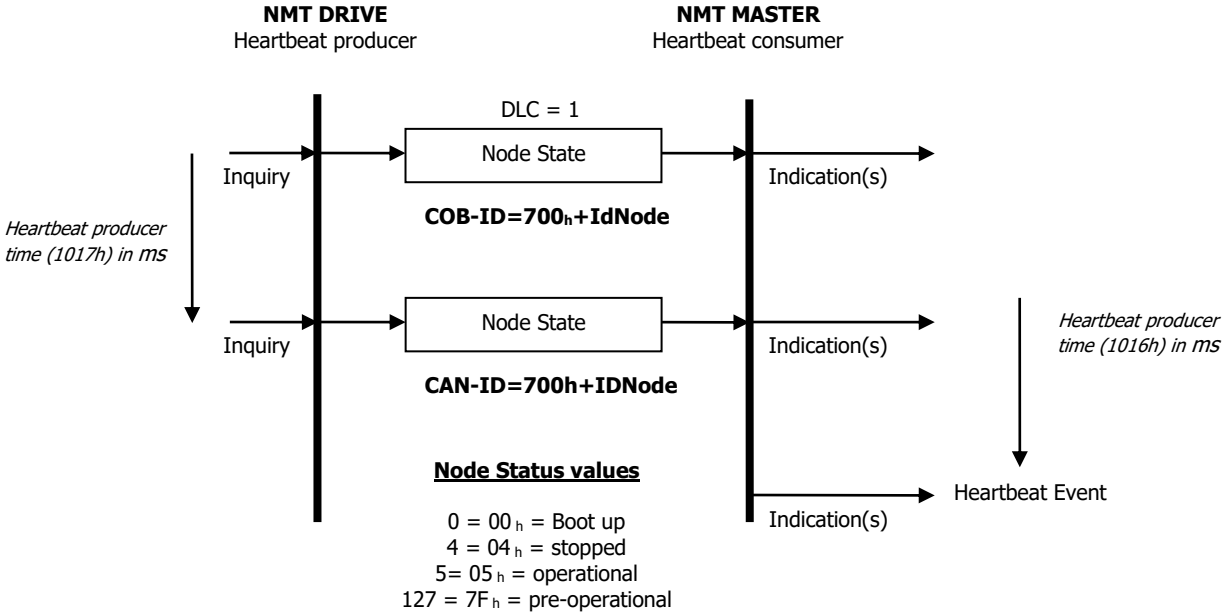


Figure 17 - Heartbeat timeframe

This service is enabled when the value of Producer heartbeat time (1017h) object is not zero.

The relationship between producer and consumer can be configured with objects. If a consumer does not receive a signal within the period of time set with Consumer heartbeat time (1016h) it generates an error message (heartbeat event).

If the consumer heartbeat time (1016h) object equal 0 then the monitoring by a consumer.

information

Referring to "APPENDIX" chapter to read "Heartbeat Mechanism"

CAN ERROR COMMUNICATION

It sends an emergency message to describe the cause of the communication error.

The drive in default configuration goes in FAULT state when it detects the BUS-OFF and Node Guarding (if it is activated) in other cases the drive it does not do action. To change the behaviour it can set the object 0x6007 "Abort Connection Option Code".

The drive in default configuration goes in FAULT after time when the state of CAN peripheral goes in ERROR PASSIVE and it the time defined in object 4102_h Settings Communication CAN subindex 2 is expired. This configuration is called High sensibility "CAN Error Communication" handling. To disable this configuration contact the manufacturer.

When the drive is configured by analog mode the drive draws attention with warning message.

To monitor the communication CAN network read the objects:

- 4100_h State Communication CAN
- 4101_h Counter Communication CAN
- 4102_h Settings Communication CAN
- 4103_h Error Communication CAN

The State of line can be read in "Actual State CAN driver" of 4100h: 5 "State Communication CAN" object:

- CAN Controller is INIT state: 0001_h
- CAN Controller is ACTIVE state: 0002_h
- CAN Controller is BUSOFF state: 0004_h
- CAN Controller is PASSIVE state: 0008_h

To know the counter of TEC and REC register read the sub-index 1 and 2 of "State Communication CAN" object.

CAN overrun (object lost)

If the messages are lost the controller sends the emergency message 8110_h.
The "Actual Flag Error" of 4100_h: 3 "State Communication CAN" object is set bit "error overrun" (BIT 4).

CAN in Error Passive Mode

If the CAN module of the drive is set the Error-Passive State the emergency message 8120_h is sent.
The "Actual Flag Error" of 4100_h: 3 "State Communication CAN" object is set bit "Error Passive" (BIT 7).

CAN in BUS-Off

If the CAN module of the drive is set the Bus-Off State the emergency message 8180_h is sent.
The "Actual Flag Error" of 4100_h: 3 "State Communication CAN" object is set bit "bus-off" (BIT 8).
It is possible to read the counter of BUS-OFF is occurred reading the "Counter Communication CAN" 4101_h: 1 "Counter Bus-off".

Recovered from BUS-Off

If the CAN module of the drive receives a valid message while set the BUSOFF state, the emergency message 8140_h is sent.
The "Actual Flag Error" of 4100_h: 3 "State Communication CAN" object is set bit "Recovery Bus-off" (BIT 9).
It is possible to read the counter of recovery BUS-OFF reading the "Counter Communication CAN" 4101_h: 2 "Counter recovery BUSOFF".

Abort Communication

From firmware 242 is possible to set the different behaviour in response to a failure/interruption in CAN bus communication defined "abort connection".

The failure/interruption can be occurred for bus-off, heartbeat (only if device is master), life guarding, NMT stopped state entered, reset application, and reset communication.

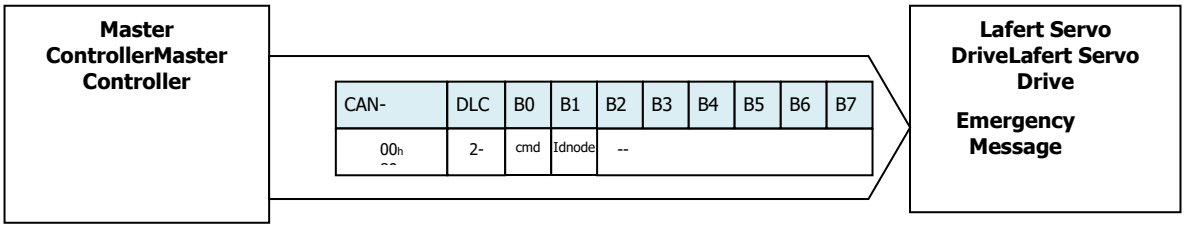
The object that define this feature is 0x6007 "Abort Connection Option Code".

NETWORK MANAGEMENT (NMT)

The Network Management (NMT) is one of the service elements of the application layer.

The NMT serves to configure, initialise, and handle errors in a CAN network. NMT commands are used to control the communication state of the servo drive and to broadcast manufacturer messages to all other connected servo drives.

An NMT Slave is uniquely identified in the network by its Node-ID, a value in the range of [1 to127].



CANopen devices enter the NMT state Pre-operational directly after finishing the CANopen devices initialization. During this NMT state CANopen device parameterization and CAN-ID-allocation via SDO (e.g. using a configuration tool) is possible. Then the CANopen devices may be switched directly into the NMT state Operational.

The Network Management is node oriented and follows a master-slave structure. NMT objects are used for executing NMT services. Through NMT services, nodes are initialised, started, monitored, reset or stopped. All nodes are regarded as NMT slaves.

NMT requires that one device in the network fulfils the function of the NMT Master.

NMT Services:

- **Module Control Services:** Through Module Control Services, the NMT master controls the state of the NMT slaves. The state attribute is one of the values (STOPPED, PRE-OPERATIONAL, OPERATIONAL and INITIALISING). The Module Control Services can be performed with a certain node or with all nodes simultaneously.

- **Error Control Service:** Through Error control services the NMT detects failures in a CAN-based Network. Local errors in a node may e.g. lead to a reset or change of state. Error Control services are achieved principally through periodically transmitting of messages by a device. There exist two possibilities to perform Error Control i.e. Node Guard and Heart Beat Error Control.

- **Boot-up Service:** Through this service, the NMT slave indicates that a local state transition occurred from the state INITIALISING to the state PRE-OPERATIONAL.

NMT state machine

CANopen devices enter the NMT state Pre-operational directly after finishing the CANopen devices initialization. During this NMT state CANopen device parameterization and CAN-ID-allocation via SDO possible. Then the CANopen devices may be switched directly into the NMT state Operational.

The NMT state machine determines the behaviour of the communication function unit.

The coupling of the application state machine to the NMT state machine is CANopen device dependent and falls into the scope of device profiles and application profiles.

The following picture shows the NMT state diagram of a CANopen device is specified.

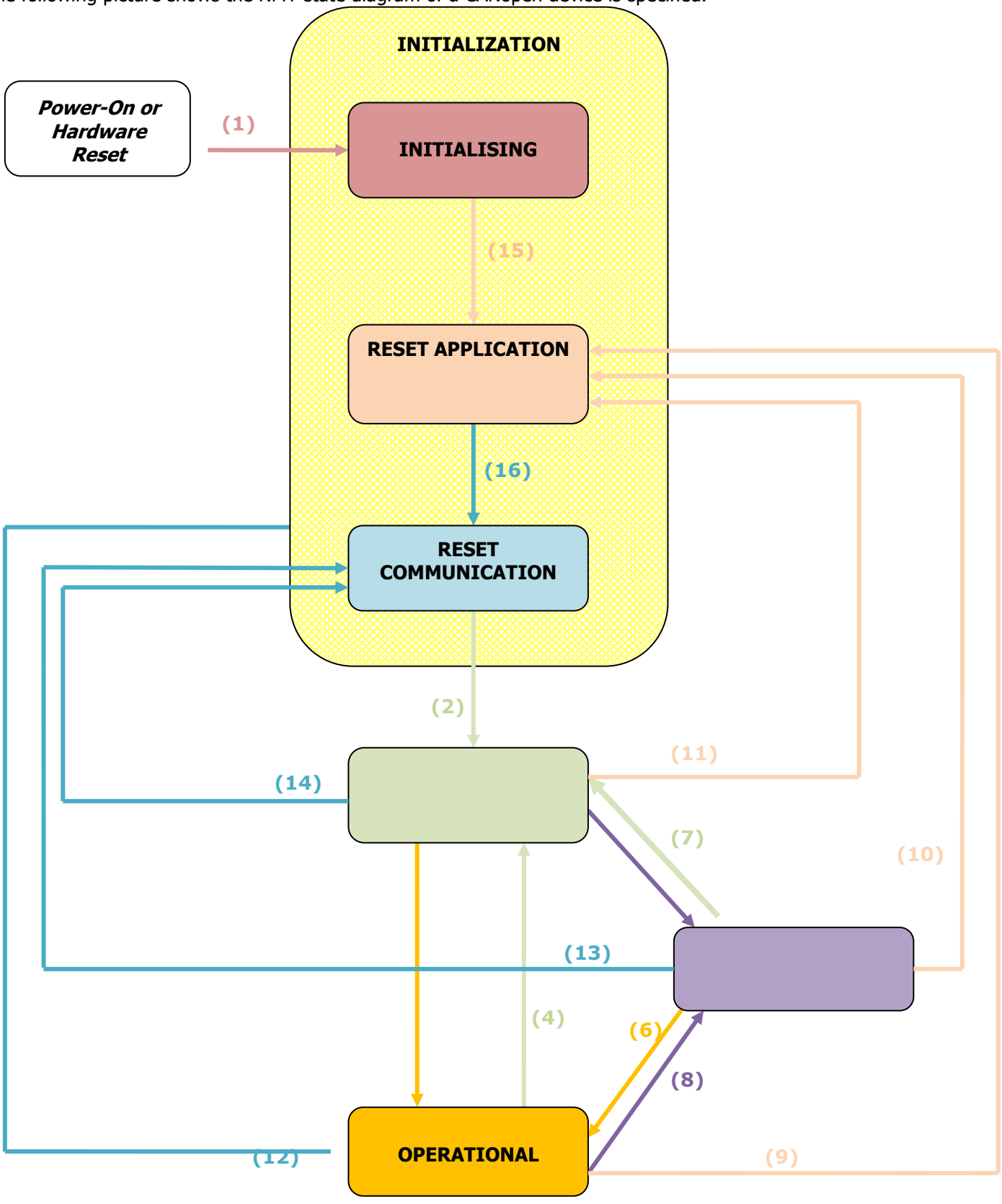


Figure 18 - NMT state machine

The following Table describes the transitions.

Transition	Description
(1), (15), (16)	At Power on the NMT state initialization is entered autonomously
(2)	NMT state initialization finished – enter NMT state Pre-operational automatically
(3)	NMT service start remote node indication or by local control
(4), (7)	NMT service enter pre-operational indication
(5), (8)	NMT service stop remote node indication
(6)	NMT service start remote node indication
(9), (10), (11)	NMT service reset node indication
(12), (13), (14)	NMT service reset communication indication

Table 26 -NMT Network Management

The following network communication states are supported, with the following communication type.

State	Description	SDO	PDO	NMT	SYNC
Initialization	**Drive is not ready, or it is booting. Drive will not respond to communication and will not transmit anything.**	-	-	-	-
Pre-operational	**Drive boot sequence is complete, but no command has been received to enter operational mode. The servo drive will respond to SDO and NMT messages, but not to PDOs.**	x	-	x	x
Operational	Drive is fully operational, responding to PDO, SDO and NMT messages.	x	x	x	x
Stopped	Servo drive can respond only to NMT objects (including heartbeats).	-	-	x	-

Table 27 -NMT Network Management

Network Initialization:

When powering the drive enter in the state machine Network Management (NMT). The first state after an internal reset or a power cycle is the NMT initialization state.
In this state the drive loads all parameters from the non-volatile memory into the RAM. After finishing the NMT initialisation state the drive enters the pre-operational State. During this state transition the CANOpen drive sends its boot-up message.

The NMT state INITIALIZATION shall be divided into three NMT sub-states in order to enable a complete or partial reset of a CANopen device.

- **Initialising**: This is the first NMT sub-state the CANopen device enters after power-on or hardware reset. After finishing the basic CANopen device initialisation the CANopen device enters autonomously into the NMT sub-state reset application.
- **Reset application**: In this NMT sub-state the parameters of the manufacturer-specific profile area and of the standardized device profile area are set to their power-on values. After setting of the power-on values the NMT sub-state reset communication is entered autonomously.
- **Reset communication**: In this NMT sub-state the parameters of the communication profile area are set to their power-on values. After this the NMT state Initialisation is finished and the CANopen device executes the NMT service boot-up write and enters the NMT state Pre-operational.

Power-on values are the last stored parameters. If storing has not been executed or if the reset was preceded by the command restore defaults, the power-on values are the default values according to the communication and device profile specifications.

Network Pre-Operational state:

In the pre-operational state communication via SDOs is possible, while (PDO) communication is not allowed. Configuration of PDOs and device parameters may be performed. Also the emergency objects and error control service like the CANopen sensors "heartbeat message" occur in this state. The node will be switched into the operational state directly by sending a NMT "start remote node".

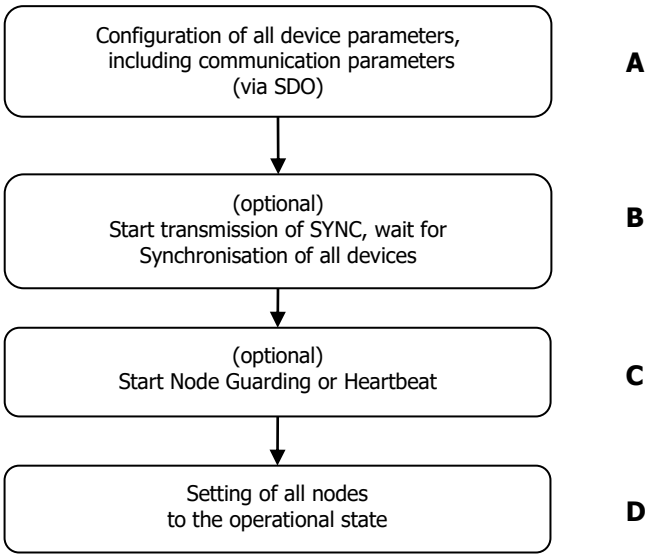
Network Operational State:

In the operational state all communication objects – including PDO handling – are active. Object dictionary access via SDO is possible.

Network Stopped State:

By switching a device into the stopped state it is forced to stop the communication, except node guarding and heartbeat, if active.

Network Initialisation Process:



STEP A): the device is in the node state PRE-OPERATIONAL which is entered automatically after power-on. In this state the devices are accessible via their default-SDO, the configuration of SDOs settings and optionally the setting of COB-IDs may be performed via SDO objects. In many cases a configuration is not even necessary as default values are defined for all application and communication parameters.

STEP B): If the application requires the synchronisation of all or some nodes in the network, the appropriate mechanisms can be initiated in the optional Step B. It can be used to ensure that all nodes are synchronised by the SYNC object before entering the node state OPERATIONAL in step D. The first transmission of SYNC object starts within 1 sync cycle after entering the PRE-OPERATIONAL state.

STEP C): In this step the Node guarding or Heartbeat can be activated using the guarding parameters configured in step A.

STEP D): Now the master controller has to move the drive in OPERATIONAL state. With step D all nodes are enabled to communicate via their PDO objects.

NMT Message

The NMT message contains only 2 data byte, with the following format:

COB-ID	Rx/Tx	DLC	Byte							
			0	1	2	3	4	5	6	7
00 h	**Rx**	**2**	Command	Id Node	-					

Table 28 - NMT Message Structure

With the following commands the NMT state can be changed.

Byte 0 value Definition:

Command	Meaning	Description	Transition	Target-State
01 h	Start Remote Node	Through this service the NMT master sets the state of the selected NMT slave(s) to "operational".	(3) (6)	**OPERATIONAL**
02 h	Stop Remote Node	Through this service the NMT master sets the state of the selected NMT slave(s) to "stopped".	(5) (8)	**STOPPED**
80 h	Enter Pre-Operational State	Through this service the NMT master sets the state of the selected NMT slave(s) to "pre-operational".	(4) (7)	**PRE-OPERATIONAL**
81 h	Reset Application	Through this service the NMT master sets the state of the selected NMT slave(s) from any state to the "reset application" sub-state.	(9) (10) (11)	**RESET APPLICATION**
82 h	Reset Communication	Through this service the NMT master sets the state of the selected NMT slave(s) from any state to the "reset communication" sub-state. After completion of the service, the state of the selected remote nodes will reset communication.	(12) (13) (14)	**RESET COMMUNICATION**

Byte 1 value Definition:

Id Node	Description
ID Node	set 0x00 for all devices (global mode) set Id-Node (0x01…0x7F) for a specific device

Table 29 - NMT Description Field

Example:

START REMOTE NODE: move All nodes in "Operational state"

Rx: Id 0x00 – 01 00

Rx: [cmd: 0x01] [Id-Node: 0x00] - Master sends value 01 00 at COB-ID 0x00

RESET APPLICATION: reset application ONLY id-Node number 1:

Rx: Id 0x00 – 81 01

Rx: [cmd: 0x81] [Id-Node: 0x01] - Master sends value 81 01 at COB-ID 0x00

Boot-up Message

After power-on or after reset, the LSD controller reports through a Boot-up message that the initialising has been finished. Next this message the LSD is in the NMT state preoperational.

COB-ID	Rx/Tx	DLC	Byte							
			0	1	2	3	4	5	6	7
700 h + IdNode	**Tx**	**1**	0x00	-						

Table 30 - BOOTUP Message Structure

Example:

Boot-up Node-Id 1:

Tx: Id 0x701 – 00

Tx: [cmd: 0x00] – drive sends a boot-up message

Value of NMT State:

- 0 = 00 h = Boot up
- 4 = 04 h = stopped
- 5= 05 h = operational
- 127 = 7F h = pre-operational

STORE AND RESTORE

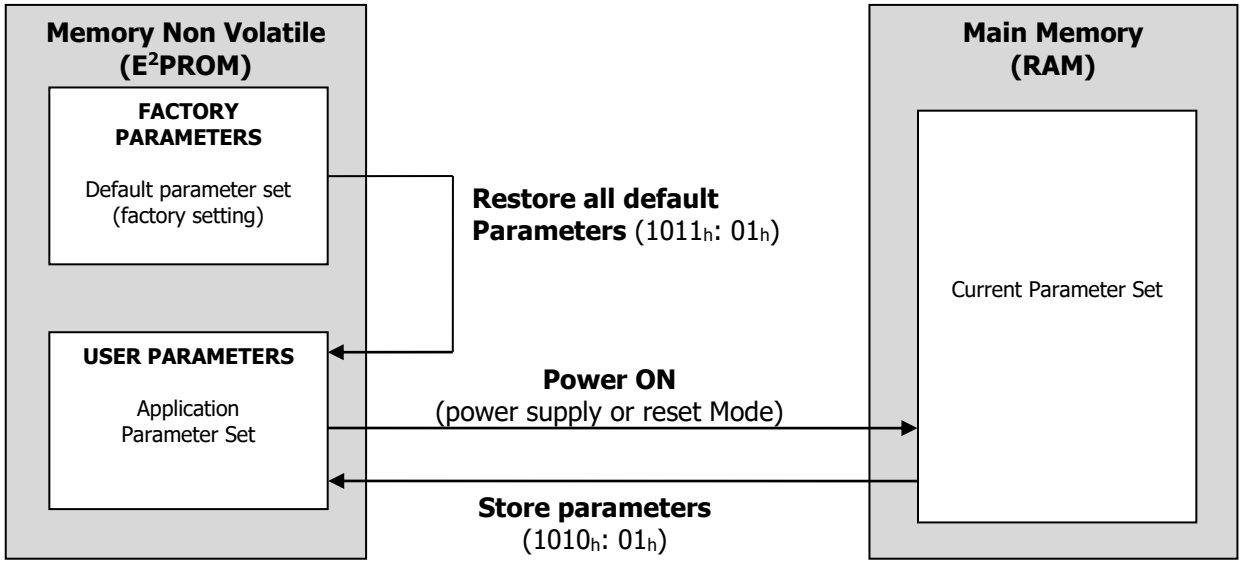


Figure 19 - Store and Restore

The CiA CANopen protocol specification defines two objects to store and restore parameters:

- Object 1010_h – Store Parameters
- Object 1011_h – Restore Parameters

Store:

In order to save all parameters the master writes in the SDO 1010_h index the value 'SAVE' to one of the subentries of the object. This procedure causes the corresponding set of parameters to be written to non-volatile memory. After the NMT reset node or the NMT reset communication the parameters will be loaded in object dictionary automatically (see object 1011h: Store Parameters).

Caution

To store ALL parameters the master must be send via SDO the command 'SAVE' in "Save All Parameters" object 1010h: 1h after that it must RESET the drive or it does power cycle.

To save different area of parameters the master can send the command 'STORE' in different sub-index:

- Sub-Index 1: Save all Parameters
- Sub-Index 2: Save Communication Parameters
- Sub-Index 3: Save Application Parameters
- Sub-Index 4: Save Manufacturer Parameters

Procedure:

The procedure to save the parameters is:

- The drive must be power supply with the VDC-Link. The SAVE command does not work if the drive is supplied with only an auxiliary voltage (+24V).
- The drive mustn't be in "Operational enabled" state or "Quick Stop Active" state. In these states the drive has torque applied.
- Write the new value in SDO corresponded. E.g. to change ID with new value '3' using command 'write' in 2000_h:0 object (drive Node-Id 1):
  - **Rx: 0x601 – 2F 00 20 00 03 00 00 00 -** [cmd: 0x2F] [Index: 0x2000] [sub-index: 0x00] [Data: 0x00000003]
  - **Tx: 0x581 – 60 00 20 00 00 00 00 00 -** [resp: 0x60] [Index: 0x2000] [sub-index: 0x00] [Data: 0x00000000]
- Write signature "SAVE" in Store object. E.g. Send command SAVE in 1010_h:0 object (pay attention on the processing time)
  - **Rx: 0x601 - 23 10 10 01 73 61 76 65 -** [cmd: 0x23] [Index: 0x1010] [sub-index: 0x01] [Data: 0x65766173]
  - **Tx: 0x581 – 60 10 10 01 00 00 00 00 -** [resp: 0x60] [Index: 0x1010] [sub-index: 0x01] [Data: 0x00000000]
- NMT Reset Node
  - **Rx: 0x00 - 81 01 -** [cmd: 0x81] [IdNode: 0x01]

Area Parameters:

The following Objects can be changed and stored in E^2prom by writing in object 1010_h: 2_h (Save Communication Parameters).

- $100C_h$: Guard Time (*)
- $100D_h$: Life Time Factor (*)
- 1017_h: Producer Heartbeat Time (*)

The following Objects can be changed and stored in E^2prom by writing in object 1010_h: 3_h (Save Application Parameters).

- 6007_h: Abort Connection Option Code (*)
- $605B_h$: Shutdown option code (*)
- $605C_h$: Disable operation option code (*)
- $605E_h$: Fault reaction option code (*)
- 6065_h: Following error window *(not implemented)*
- 6066_h: Following error time out *(not implemented)*
- 6067_h: Position window *(not implemented)*
- $606D_h$: Velocity Window
- $606E_h$: Velocity Window Time

- 606F$_h$: Velocity Threshold
- 6070$_h$: Velocity Threshold Time
- 6068$_h$: Position window time ***(not implemented)***
- 6072$_h$: Max torque ***(not implemented)***
- 6073$_h$: Max Current
- 6075$_h$: Motor Rated Current
- 6076$_h$: Motor Rated torque ***(not implemented)***
- 607B$_h$: Position range limit ***(not implemented)***
- 607C$_h$: Home offset ***(not implemented)***
- 607D$_h$: Software position limit ***(not implemented)***
- 607E$_h$: Polarity (*)
- 607F$_h$: Max Profile Velocity
- 6080$_h$: Max Motor Speed
- 6081$_h$: Profile Velocity (***not implemented)***
- 6082$_h$: End Velocity (***not implemented)***
- 6083$_h$: Profile Acceleration (*)
- 6084$_h$: Profile Deceleration (*)
- 6086$_h$: Motion profile type
- 6087$_h$: Torque Slope (*)
- 6088$_h$: Torque Profile Type
- 6096$_h$: Velocity Factor
- 6097$_h$: Acceleration Factor
- 6099$_h$: Homing speeds ***(not implemented)***
- 609A$_h$: Homing acceleration ***(not implemented)***
- 60A2$_h$: Jerk Factor
- 60A4$_h$: Profile Jerk
- 60C5$_h$: Max Acceleration
- 60C6$_h$: Max Deceleration
- 60F2$_h$: Positioning option code ***(not implemented)***

The following Objects can be changed and stored in E^{2}prom by writing in object 1010$_h$: 4$_h$ (Save Manufacturer Parameters).

- 2000$_h$: Id Node
- 2001$_h$: Baudrate
- 3002$_h$: Brake Parameters (*)
- 3004$_h$: Feedback Parameters (only sub-index 0A$_h$)

- 3005_h: Filter Parameters (*) (only sub-index 5_h and 6_h)
- 3007_h: Dynamic Stop Parameters (*)
- 3008_h: Emergency Input Enable (*)
- 3050_h: Analog Output (*)
- 3100_h: Configuration 1 - Statusword (*)
- 3101_h: Configuration 2 - Reserved
- 3102_h: Configuration 3 - STO (*)
- 3103_h: Configuration 4 - Safety FSM (*)
- 3104_h: Configuration 5 - Warning (*)
- 3200_h: Pid Current (*)
- 3201_h: Pid Velocity (*)
- 3202_h: Pid Positioner (*) ***(not implemented)***
- 3300_h: Velocity Full Scale
- 3900_h: Warning Settings (*)
- 4102_h: Settings Communication CAN
- 4400_h : Reserved
- 4401_h : Reserved
- 4402_h : Reserved

The objects marked with (*) can be changed in run time. If the drive is disconnected the value modified are lost.

information

The command to save parameters must be send with +Vdc-Link actives. When the VBus is turn-off and the drive is supplied with ONLY auxiliary voltage (+24V voltage for Logic Board) the store command has an Abort Code and the store command is failed.

Example: the Store command without +Vdc-Link -> The answer is "abort Code"

Rx: Id 0x601 - 23 10 10 01 73 61 76 65

Tx: Id 0x581 – 80 10 10 01 00 00 06 06

Rx: [cmd: 0x23] [Index: 0x1010] [sub-index: 0x01] [Data: 0x65766173] – SAVE all parameters
Tx: [resp: 0x80] [Index: 0x1010] [sub-index: 0x01] [Data: 0x00000606] – Abort Code

Example: the Store command with +Vdc-Link -> The answer is correct, value "60" is correct value.

Rx: Id 0x601 - 23 10 10 01 73 61 76 65

Tx: Id 0x581 – 60 10 10 01 00 00 00 00

Rx: [cmd: 0x23] [Index: 0x1010] [sub-index: 0x01] [Data: 0x65766173] – SAVE all parameters
Tx: [resp: 0x60] [Index: 0x1010] [sub-index: 0x01] [Data: 0x00000000] – SUCCESSFULL!!!

Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

All parameters can be stored in E^2prom, the changes are not accepted until either the voltage supply is briefly disconnected or the CANopen message RESET COMM (NMT) is sent to the motor.

Restore:

In order to avoid the restoring of default parameters by mistake, it is possible loading the factory parameters. The master sends the SDO 1011_h and writes the signature 'LOAD' to one of sub-index (see object 1011h: Retore default Parameters).

Function restore factory parameters:

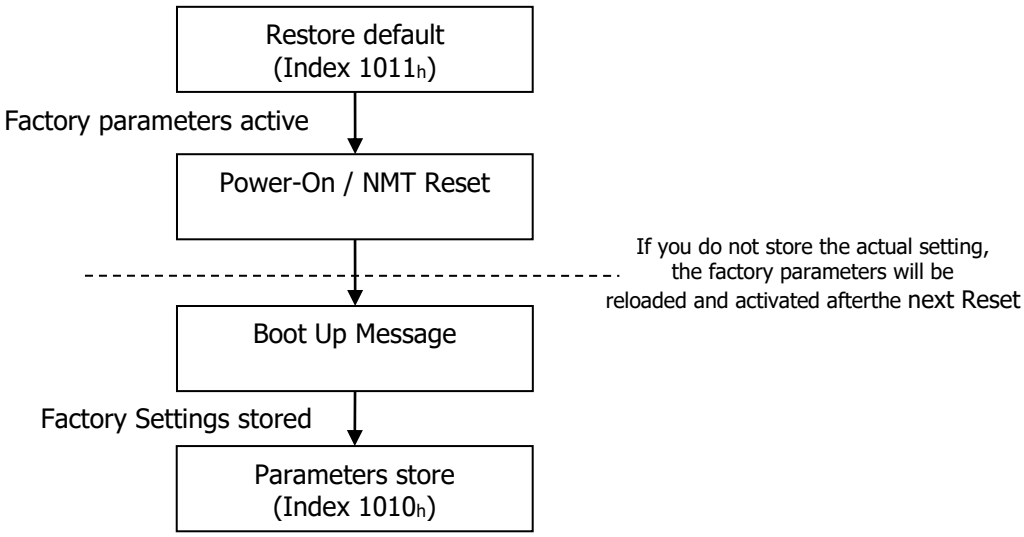


Figure 20 - Restore Flow Chart

Default Communication Parameters:

NAME	Index	SubIndex	Value Field	Default Parameters
P301 DEV TYPE	0x1000	0	Device Type	0xFF7A0192
P301 ERR REG	0x1001	0	Error Register	0
P301 MANUF STATUS REG	0x1002	0	Manufacturer Status Register	0
P301 PREDEF ERR FIELD	0x1003	0	Number of Errors	15
		1	history[1]	0
		2	history[2]	0
		3	history[3]	0
		4	history[4]	0
		5	history[5]	0
		6	history[6]	0
		7	history[7]	0
		8	history[8]	0
		9	history[9]	0
		10	history[10]	0
		11	history[11]	0
		12	history[12]	0
		13	history[13]	0
		14	history[14]	0
		15	history[15]	0
P301 COBID SYNC	0x1005	0	COB-ID SYNC	COB-ID = 80000080h +Id
P301 GUARD TIME	0x100C	0	Guard Time	0 = Disabled

P301 LIFETIME FACTOR	0x100D	0	Life Time Factor	0 = Disabled
P301 COBID EMERGENCY	0x1014	0	COB-ID EMCY	COB-ID = 80h+ID
P301 PRODUCER HB TIME	0x1017	0	Producer Heartbeat Time	200 ms
P301 IDENTITY OBJECT	0x1018	0	number entries	4
		1	Vendor Id	0x01FB
		2	Product Code	0
		3	Revision number	0
		4	Serial number	0
P301 ERR BEHAVIOR	0x1029	0	Number of Entries	1
		1	Communication Error	0
P301 RXPDO 1 PARAM	0x1400	0	Number of Entries	5
		1	COB-ID	COB-ID = 200h+ID, Receive PDO enabled
		2	Transmission Type	0xFE =Asynchronous
		3	Inhibit Time	0x5 = 500us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 RXPDO 2 PARAM	0x1401	0	Number of Entries	5
		1	COB-ID	COB-ID = 300h+ID, Receive PDO enabled
		2	Transmission Type	0xFE =Asynchronous
		3	Inhibit Time	0x5 = 100us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 RXPDO 3 PARAM	0x1402	0	Number of Entries	5
		1	COB-ID	COB-ID = 400h+ID, Receive PDO enabled
		2	Transmission Type	0xFE =Asynchronous
		3	Inhibit Time	0x5 = 500us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 RXPDO 4 PARAM	0x1403	0	Number of Entries	5
		1	COB-ID	COB-ID = 500h+ID, Receive PDO enabled
		2	Transmission Type	0xFE =Asynchronous
		3	Inhibit Time	0x5 = 500us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 RXPDO 1 MAPPING	0x1600	0	Number of Entries	8
		1	Mapping Entry 1	0x60400010 = Controlword
		2	Mapping Entry 2	0x60600008 = Mode of operation
		3	Mapping Entry 3	0x60FE0120 = Digital output
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 RXPDO 2 MAPPING	0x1601	0	Number of Entries	8
		1	Mapping Entry 1	0x60400010 = Controlword
		2	Mapping Entry 2	0x607A0020 = Target Position
		3	Mapping Entry 3	0
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 RXPDO 3 MAPPING	0x1602	0	Number of Entries	8
		1	Mapping Entry 1	0x60400010 = Controlword
		2	Mapping Entry 2	0x60FF0020 = Target Velocity
		3	Mapping Entry 3	0
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 RXPDO 4 MAPPING	0x1603	0	Number of Entries	8
		1	Mapping Entry 1	0x60400010 = Controlword
		2	Mapping Entry 2	0x60710010 = Target Torque
		3	Mapping Entry 3	0
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0

		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 TXPDO 1 PARAM	0x1800	0	Number of Entries	5
		1	COB-ID	COB-ID = 180h+ID, Receive PDO enabled
		2	Transmission Type	0xFD = Asynchronous – RTR only
		3	Inhibit Time	0x5 = 100us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 TXPDO 2 PARAM	0x1801	0	Number of Entries	5
		1	COB-ID	COB-ID = 280h+ID, Receive PDO enabled
		2	Transmission Type	0xFD = Asynchronous – RTR only
		3	Inhibit Time	0x5 = 100us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 TXPDO 3 PARAM	0x1802	0	Number of Entries	5
		1	COB-ID	COB-ID = 380h+ID, Receive PDO enabled
		2	Transmission Type	0xFD = Asynchronous – RTR only
		3	Inhibit Time	0x5 = 100us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 TXPDO 4 PARAM	0x1803	0	Number of Entries	5
		1	COB-ID	COB-ID = 480h+ID, Receive PDO enabled
		2	Transmission Type	0xFD = Asynchronous – RTR only
		3	Inhibit Time	0x5 = 100us
		4	Compatibility entry	0 = not available
		5	Event timer	0 = disabled [ms]
P301 TXPDO 1 MAPPING	0x1A00	0	Number of Entries	8
		1	Mapping Entry 1	0x60410010 = Statusword
		2	Mapping Entry 2	0x60610008 = Mode of operation display
		3	Mapping Entry 3	0x60FD0020 = Digital input
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 TXPDO 2 MAPPING	0x1A01	0	Number of Entries	8
		1	Mapping Entry 1	0x60410010 = Statusword
		2	Mapping Entry 2	0x60640020 =Position Actual Value
		3	Mapping Entry 3	0
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 TXPDO 3 MAPPING	0x1A02	0	Number of Entries	3
		1	Mapping Entry 1	0x60410010 = Statusword
		2	Mapping Entry 2	0x606C0020 = Velocity Actual Value
		3	Mapping Entry 3	0
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0
P301 TXPDO 4 MAPPING	0x1A03	0	Number of Entries	8
		1	Mapping Entry 1	0x60410010 = Statusword
		2	Mapping Entry 2	0x60770010 = Torque Actual Value
		3	Mapping Entry 3	0
		4	Mapping Entry 4	0
		5	Mapping Entry 5	0
		6	Mapping Entry 6	0
		7	Mapping Entry 7	0
		8	Mapping Entry 8	0

Table 31 - Communication Parameters

TABLE OF IDENTIFIERS

The following table gives a survey of the used identifiers.

Object-Type	Identifier (hexdecimal)
SDO (MASTER to drive)	600 h + IdNode
SDO (drive to MASTER)	580 h + IdNode
TPDO1	180 h + IdNode
TPDO2	280 h + IdNode
TPDO3	380 h + IdNode
TPDO4	480 h + IdNode
RPDO1	200 h + IdNode
RPDO2	300 h + IdNode
RPDO3	400 h + IdNode
RPDO4	500 h + IdNode
SYNC	80 h
EMCY	80 h + IdNode
HEARTBEAT	700 h + IdNode
BOOTUP	700 h + IdNode
NMT	00 h

Table 32 - Table of Identifiers

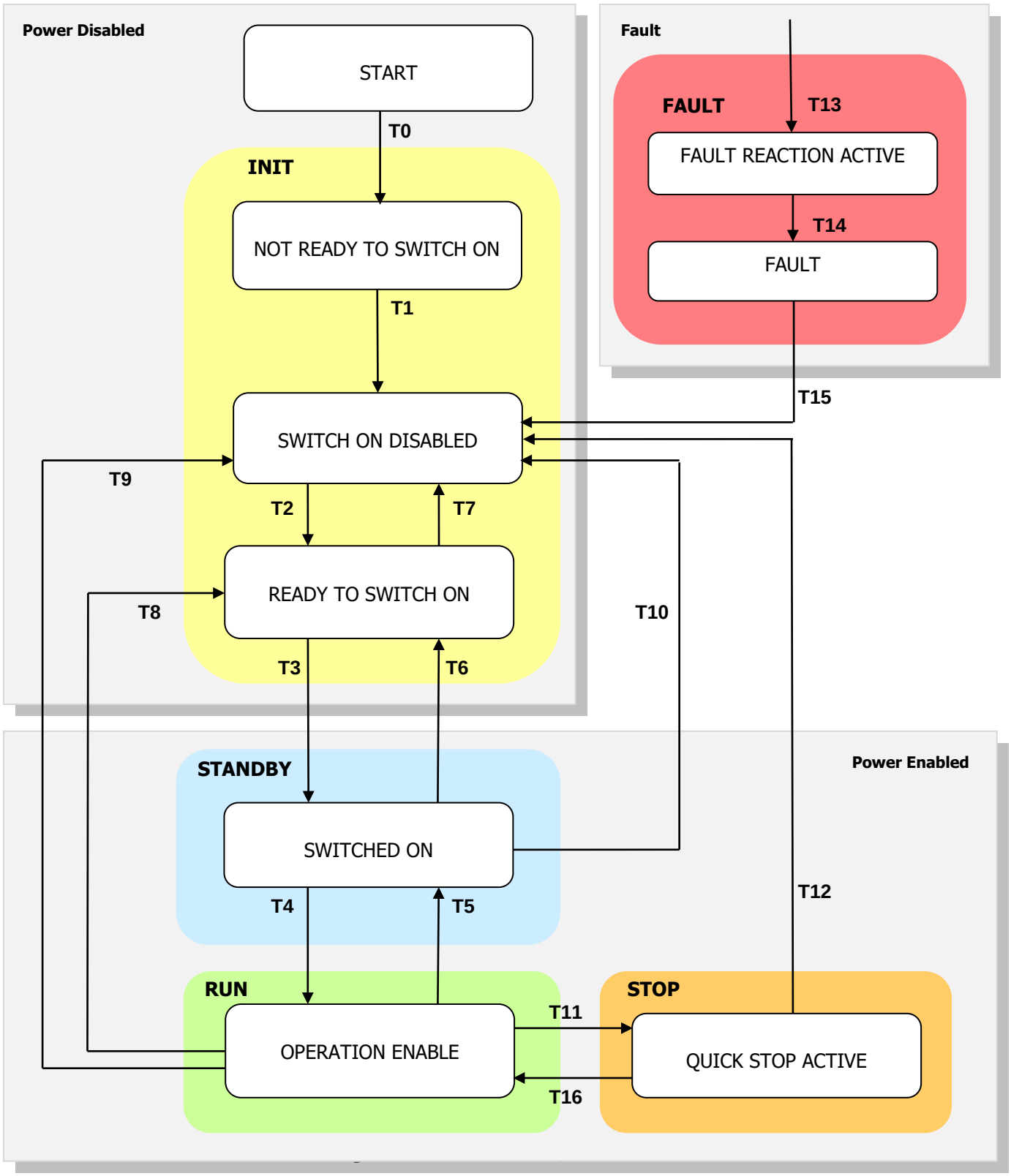
PROFILE DSP402

information

For Additional Information please refer to CiA DS402 standard.

State Machine Profile DSP402

The drive is checked and controlled by a state machine according t DSP402.



State changes are triggered by internal events such as the occurrence of an error or external demand by means of Controlword (6040_h). The object Statusword (6041_h) gives feedback about the actual state.

After power-up and initialisation, the drive switches to the state "Switch On Disabled" automatically. In this state the device waits a controlword command. In the state "Operation Enabled" the drive is fully operational.

SAFETY state is not implemented in DSP402. The state "SAFETY" has been added to protect and to define the drive when the emergency is applied. (See chapter "SAFETY")

Actual State may be read by statusword, with standard coding (defined by CiA DSP402):

STATUS	Description
NOT READY TO SWITCH ON	**INIT STATE:** The drive is being initialized and is running the self test. The drive function is disabled. This state is an internal state in which communication is enabled only at the end. The user can neither retrieve nor monitor this state.
SWITCH ON DISABLED	**INIT STATE:** No power applied. Drive initialization is completed. The drive parameters have been set up. Drive parameters may be changed. The drive function is disabled. The parameters can be saved in E^2prom. *SWITCH ON DISABLED is the minimum state to which a user may switch.*
READY TO SWITCH ON	**INIT STATE:** No power applied. The drive parameters have been set up. Drive parameters may be changed. The drive function is disabled. The parameters can be saved in E^2prom.
SWITCHED ON	**STANDBY STATE:** No power applied. The power stage is ready to RUN condition (to "operation enable"). The drive parameters may be changed. The drive function is disabled. The parameters can be saved in E^2prom.
OPERATION ENABLE	**RUN STATE:** *(This corresponds to normal operation of the drive)* No faults have been detected. Power applied to the motor. The drive function is enabled. The drive parameters may be changed. If automatically stop is enabled than it is released, in according to the brake parameter timing. ***The drive parameters can't be saved and restored in E^2prom.***

QUICK STOP ACTIVE	**STOP STATE:** No faults have been detected. Power applied to the motor. The drive function is enabled. The drive parameters may be changed. The drive stops the motion and either stays in quick stop with torque applied. ***The drive parameters can't be saved and restored in E^2prom.***
FAULT REACTION ACTIVE	**FAULT STATE:** The drive parameters may be changed. A fault has occurred in the drive. The fault reaction function is being executed. The drive function is disabled. This parameter cannot be retrieved by the user. The parameters can be saved in E^2prom.
FAULT	**FAULT STATE:** The drive parameters may be changed. A fault has occurred in the drive. The drive function is disabled. The parameters can be saved in E^2prom. The drive accepts the "Reset Fault Alarm".

Table 33 - Status Word

The follow table shows the Led Codes referring the status Drive of "Lafert Servo Drive" and the correspondent state DSP402 state machine.

MACRO DRIVE STATE	CANOpen STATE	STATUS 1 LED GREEN	STATUS 2 LED YELLOW	LED VIEW SMARTRIS STANDARD	LED VIEW SMARTRIS COMPACT
INIT	Not Ready To Switch On	"BLINK" simultaneously	"BLINK" simultaneously	1 simultaneously 2 simultaneously	2 simultaneously 3 simultaneously 4 OFF
	Switch On Disabled Ready to Switch On	"BLINK" alternately	"BLINK" alternately	1 alternately 2 alternately	2 alternately 3 alternately 4 OFF
STANDBY	Switched On	"BLINK"	OFF	1 BLINK 50% 2 OFF	2 BLINK 50% 3 OFF 4 OFF
FAULT	Fault Fault reaction fault	"BLINK" [x]	"BLINK" [y]	1 see fault 2 chapter	2 see fault 3 chapter 4 OFF
RUN (RUNV or RUNC)	Operation Enabled	ON	OFF	1 ON 2 OFF	2 ON 3 OFF 4 ON
STOP	Quick Stop Active	ON	ON	1 ON 2 ON	2 ON 3 ON 4 ON
SAFETY	-	OFF	"BLINK"	1 OFF 2 BLINK	2 OFF 3 BLINK 4 OFF
COMMUN ERROR	-	OFF	ON	1 OFF 2 ON	2 OFF 3 ON 4 OFF

Table 34 - Drive Status

MODE OF OPERATION

Different operation modes are available with the CiA 402 profile:

- **Profile position mode**: the feedback such as an encoder or resolver ensures that the actual position is equal to the commanded position. The servo drive's position loop compensates for position errors by applying more torque to push the system back into position.
- **Profile velocity mode**: Reference velocity assignment by a controller. The drive calculates the necessary motion profiles independently. The movement profile is defined by velocity and acceleration/decelerations commands.
- **Profile torque mode**: The profile torque mode allows control device to transmit the target torque value, which is processed via the trajectory generator. The torque slope and torque profile type parameters are required.
- **Homing Mode**: The homing is the method by which a drive seeks the home position (also called, the datum, reference point or zero point). There are various methods of achieving this using limit switches at the ends of travel or a home switch (zero point switch) in mid-travel, most of the methods also use the index (zero) pulse train from an incremental encoder.

CANOpen Run Sequence Velocity Mode

See picture below to the flow chart of running sequence

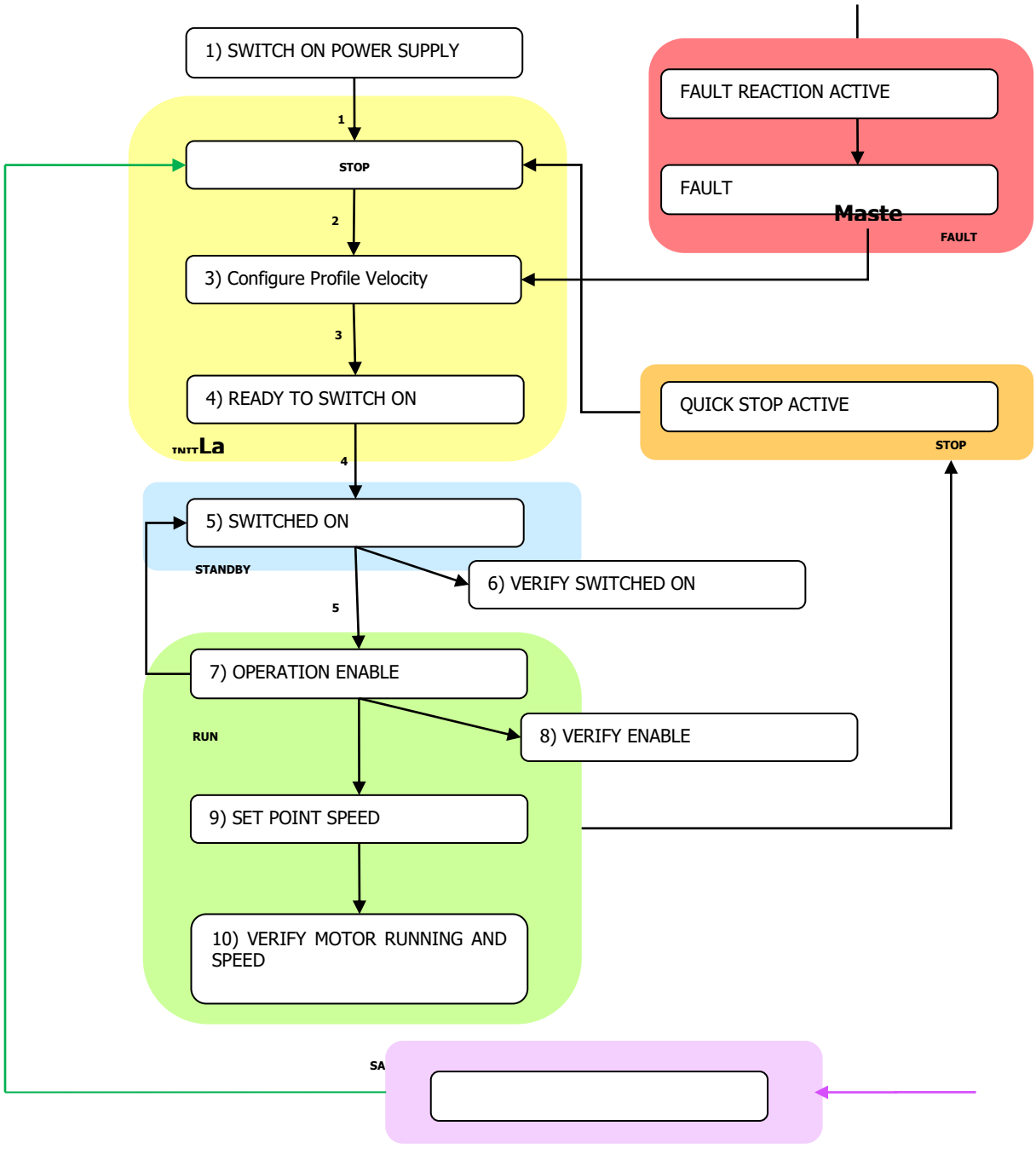


Figure 22 - CANOpen Run Sequence Velocity Mode

NOTE:

- The STO (SAFETY) command may can stop the running command immediately
- The STOP Command can stop the running command immediately
- A FAULT (see table in Diagnostic) can stop the running command immediately

- Switch ON Power Supply
- Verify LED Status 1/2 in INIT Mode
- Configure Profile Velocity 0x6060 → 0x03
- Set **READY TO SWITCH ON** State: write Control Word 0x6040 → 0x06
- Set **SWITCHED ON** State: write Control Word 0x6040 → 0x07
- Verify that the Drive is in SWITCHED ON: read Status Word 0x6041
- Set **OPERATION ENABLED** State: write Control Word 0x6040 → 0x0F
- Verify LED STATUS ENABLED
- Verify that the Drive is in ENABLED: read Status Word 0x6041
- Verify that the brake is released
- Write Speed Set Point: 0x60FF → 0x03E8 (for example set 1000 rpm)
- Verify if the motor is running
- Verify the motor speed (after ramp): read 0x606C→ 0x03E8 (for example set 1000 rpm)

TRANSITION STOPPING TO EXIT FROM "OPERATION ENABLED" STATE (RUN)

Various motion reactions can be programmed upon exiting the "Operation Enabled" state. It can set the different STOPPING mode during different transition.

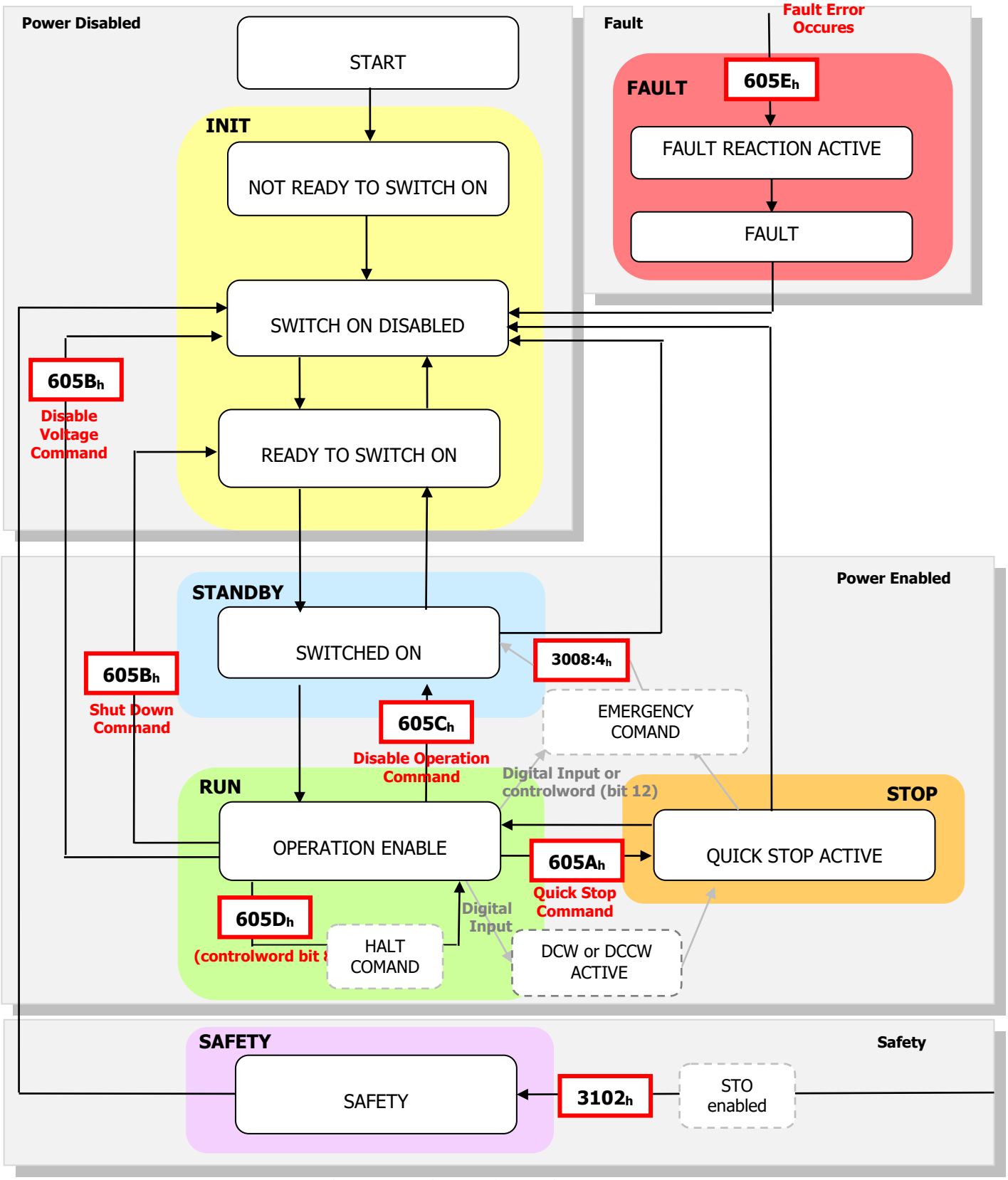


Figure 23 - Reactions to Exit Operation Enabled state

The objects that describe the transitions are:

- **$605A_h:0_h$ - Quick stop option code:** from "Operation Enable" (RUN) State to go to "Quick Stop" (STOP) State with CONTROLWORD command or Limit Switch clockwise and counter-clockwise
- **$605B_h:0_h$ - Shutdown option code:** from "Operation Enable" (RUN) State to go to "Ready To Switch On" or "Switch on Disabled" (INIT) State with CONTROLWORD command
- **$605C_h:0_h$ - Disable operation option code:** from "Operation Enable" (RUN) State to go to "Switched On" (STANDBY) State with CONTROLWORD command
- **$605D_h:0_h$ - Halt option code:** from "Operation Enable" (RUN) State to stay to "Operation Enable" (RUN) State with CONTROLWORD command
- **$605E_h:0_h$ - Fault reaction option code:** from all States to go to "Fault Reaction Active" (FAULT) State
- **$3008_h:4_h$ - Emergency option code:** from "Operation Enable" (RUN) State to go to "Switched On" (STANDBY) State with EMERGENCY command
- **$3102_h:0_h$ - STO option code:** from all States to go to "Safety" (SAFETY) State with STO hardware switched

Before firmware 236 some objects were read-only, after firmware 236 some objects were added and some became read/write.

Following table describes these objects:

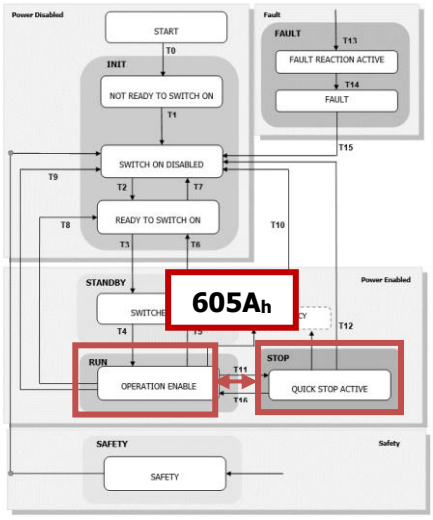
OBJECTS	BEFORE FW 236	AFTER FW 236
Braking Mode	**Old Management**	**New Management**
$605A_h:0_h$ - Quick stop option code	Read only	Read only
$605B_h:0_h$ - Shutdown option code	Read only	Read/write
$605C_h:0_h$ - Disable operation option code	Read only	Read/write
$605D_h:0_h$ - Halt option code	Read only	Read only
$605E_h:0_h$ - Fault reaction option code	Read only	Read/write
$3008_h:4_h$ - Emergency option code (*)	-	Read/write
$3102_h:0_h$ - Configuration STO Option Code (*)	-	Read/write

(*) add object from firmware 236

information

To use new management write '1' the "Option Code Braking Mode" object 0x3007:7

Transition to STOP "Quick Stop Active" with controlword:



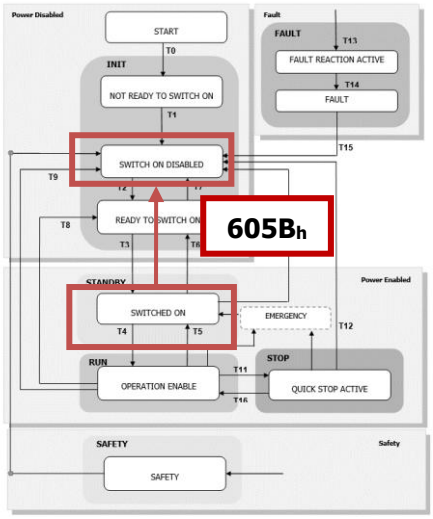
This transition is described in object 605A$_h$ "Quick stop option code" (read only object).

To exit from "operation enabled" (RUN) to "quick stop active" (STOP) it needs to set the controlword command "Quick Stop".

In case of Limit Switch Clockwise (DCW) or Limit Switch Counter Clockwise (DCCW) programmed and enabled the drive will go in "Quick Stop Active" (STOP) after the digital input correspondent was triggered.

Description Object 605A$_h$ "Quick stop option code"	
	This object is read only. • -1 = The drive will go immediately in stop (without ramp), in this state the drive will be in holding torque

Transition to INIT "Switch On Disabled" with controlword:



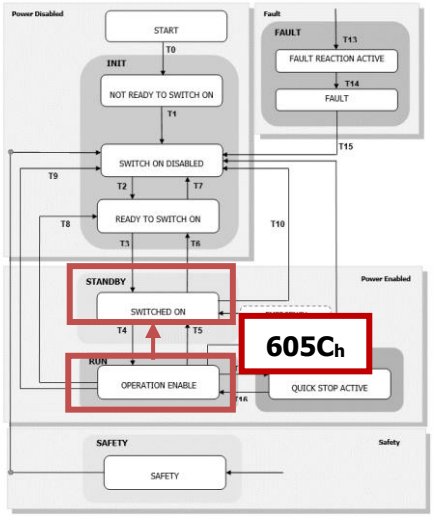
This transition is described in object 605B$_h$ "Shutdown option code".

- To go in "Ready to switch on" (INIT) the drive needs receive a "shutdown" command of controlword.
- To go in "Switch On Disabled" (INIT) the drive needs receive a "Disable Voltage" command of controlword.

It's possible to exit from "operation enabled" (RUN) to "Ready To Switch On" or "Switch On Disabled" (INIT) using specific command of controlword.

Description Object 605B$_h$ "Shutdown option code"	
Before FW 236:	This object was read only. • 0 = The drive switch-off the drive power stage and the motor will stop by inertia (if dynamic brake is not configured, object "Dynamic Stop Parameters" 3007$_h$:1 = '0'). • 1 = The drive switches-off and the motor will stop with a deceleration with a specific ramp that can be configured by the user (if dynamic brake is configured, object "Dynamic Stop Parameters" 3007$_h$:1 = '1').
After FW 236:	This object is read/write. This transition can be configured writing value: • 0 = The drive switch-off the drive power stage and the motor will stop by inertia • 1 = The drive switches-off and the motor will stop with a deceleration with a specific ramp that can be configured by the user (object 3007$_h$4$_h$). • -5 = The drive switches-off and the motor will stop with a deceleration with a S-Ramp. This operation mode grants a deceleration with a specific ramp that can be configured by the user. (Enable from FW 240 with S-Ramp profile) All parameters of Dynamic Ramp are described in object "Dynamic Stop Parameters" 3007$_h$. • The deceleration Ramp could be configured in Object 3007$_h$: 4. • The deceleration S-Ramp could be configured in Object 3007$_h$: 8.

Transition to STANDBY "Switched On" with controlword:

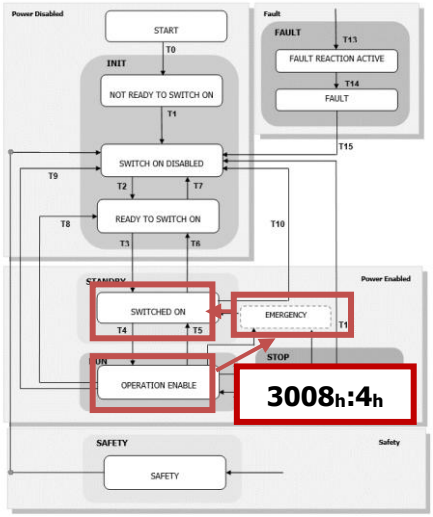


This transition is described in object 605Ch "Disable Operation Option Code" .

It's possible to exit from "operation enabled" (RUN) to "switched on" (STANDBY) using "*Disable Operation*" command of controlword.

Description object 605Ch "Disable Operation Option Code"	
Before FW 236:	It was possible to exit from "operation enabled" (RUN) to "switched on" (STANDBY) with a dynamic ramp programmed. This operation mode grants a deceleration with a specific ramp that can be configured by the user. All parameters of Dynamic Stop are described in object "Dynamic Stop Parameters" 3007h. The deceleration ramp could be configured in Object 3007h: 4. In case that the application needs a deceleration ramp to exit from "operation Enabled" state the value "option dynamic stop" must be enabled (object "Dynamic Stop Parameters" 3007h:1 = '1'). ***The default configuration is enabled.*** In case Dynamic Stop enabled to exit from "Operation enabled" state (RUN): Use Command "Disable Operation" of controlword: while motor is running, put the drive in "Switched On" (STANDBY)using the command in controlword "Disable Operation" and in this case motor will be stopped with a "Dynamic Stop" ramp. At the end of the deceleration drive will be in "Switched On" (STANDBY). In case dynamic stop disabled the transition from "Operation enabled" (RUN) to "Switched On" (STANDBY) will be done immediately and the motor will stop by inertia.
After FW 236:	This object is read/write. This transition can be configured writing value: • 0 = The drive switch-off the drive power stage and the motor will stop by inertia. • 1 = The drive switches-off and the motor will stop with a deceleration with a dynamic ramp. This operation mode grants a deceleration with a specific ramp that can be configured by the user. • -1 = The drive will stop immediately with setpoint set to 0 • -2 = The drive switch-off the drive power stage and the motor will stop by inertia. When the motor is not running the drive remains stopped with torque for a time delay • -5 = The drive switches-off and the motor will stop with a deceleration with a S-Ramp. This operation mode grants a deceleration with a specific ramp that can be configured by the user. (Enable from FW 240 with S-Ramp profile) All parameters of Dynamic Ramp are described in object "Dynamic Stop Parameters" 3007h. • The deceleration Ramp could be configured in Object 3007h: 4. • The deceleration S-Ramp could be configured in Object 3007h: 8.

Transition to STANDBY "Switched On" with Emergency command



This transition is described in object 3008_h:4_h "Emergency Option Code" .

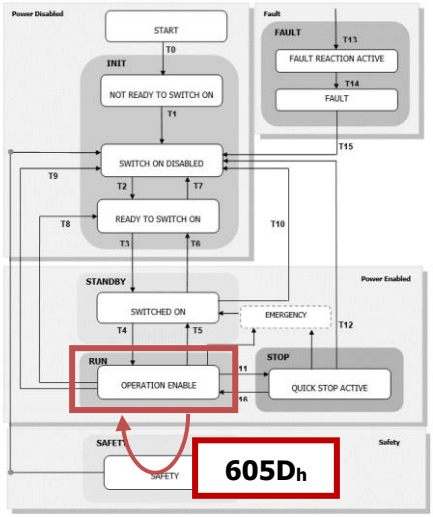
It's possible to exit from "operation enabled" (RUN) to "switched on" (STANDBY) consider these different ways with "emergency command":

- Use "Emergency command" if a digital input is programmed as "EMERGENCY" function. This is a hardware physical input at 24V. In case of a transition level (0 to 24V or 24V to 0 depending on configuration) the drive will go in "Switched On" (STANDBY).
- Use "Emergency command" if the bit 12 of controlword is set. While motor is running, the master controller can move the drive in "Switched On" (STANDBY) writing '1' in bit 12 of controlword (command EMERGENCY).

If the drive has torque applied means that the drive is in "operation enabled" (RUN) state or "quick stop active" (STOP) state, if one digital input is configured as "Emergency Input" the customer can use this digital to move the drive in "switched on" (STANDBY) state.

Description object 3008_h:4_h "Emergency Option Code"	
Before FW 236:	When the digital is triggered then the drive will go in "Switched On" (STANDBY), the motor move could be with a dynamic ramp or without ramp considering configuration "Dynamic Stop parameters" object (3007_h:4). Read the "Emergency Input Enable Parameters" object (3008_h) to configure this feature.
After FW 236:	This object is read/write. This transition can be configured writing value: • 0 = The drive switch-off the drive power stage and the motor will stop by inertia. • 1 = The drive switches-off and the motor will stop with a deceleration with a dynamic ramp. This operation mode grants a deceleration with a specific ramp that can be configured by the user. • -1 = The drive will stop immediately with setpoint set to 0 • -2 = The drive switch-off the drive power stage and the motor will stop by inertia. When the motor is not running the drive remains stopped with torque for a time delay • -3 = The drive will stop immediately with setpoint set to 0 and the motor brake is locked immediately • -4 = The drive switch-off the drive power stage and the motor brake is locked immediately • -5 = The drive switches-off and the motor will stop with a deceleration with a S-Ramp. This operation mode grants a deceleration with a specific ramp that can be configured by the user. (Enable from FW 240 with S-Ramp profile) All parameters of Dynamic Ramp are described in object "Dynamic Stop Parameters" 3007_h. • The deceleration Ramp could be configured in Object 3007_h: 4. • The deceleration S-Ramp could be configured in Object 3007_h: 8.

Transition to Stay In "Operation Enabled":



This transition is described in object 605D$_h$ "Halt option code" (read only object).

It is possible to remain in the state with the motor with torque applied using HALT command bit 8 of controlword.

When switching modes in "operation enabled" (RUN), the transition proceeds as if bit 8 (Halt bit Command) in the controlword has been set.

While motor is running, in case bit 8 in controlword has been set (Halt bit Command), motor will stop immediately (without ramp) maintaining "operation enabled" (RUN).
When motor will read 0 speed, torque still be present (and maintained) at motor shaft.

This condition is used in the following modes:

- Profile Velocity
- Profile Torque

If the master controller switch the controlword bit 8 (Halt bit Command)

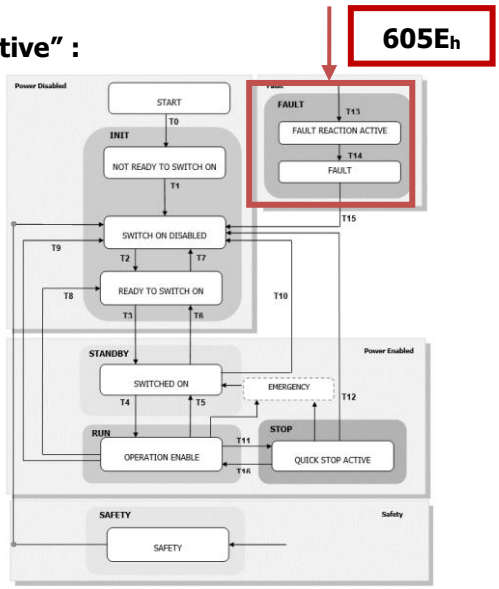
- 0 → 1 the trajectory generator ramps its control output down to zero
- 1 → 0 the trajectory generator ramps its control output up to the target.

In both cases the trajectory generator takes the profile acceleration and velocity type (profile velocity) or the torque slope and torque type (profile torque) into consideration. To have more information it needs read the chapter relatives "Velocity Profile" and "Torque Profile" Halt bit Command description (bit 8 of controlword).

To stay in "operation enabled" (RUN) with motor in STOP is possible also setting the Set Point equal 0: the Customer has to fix at 0 rpm the speed set point. In this case drive will perform a stop with a deceleration ramp configured in object 6084$_h$ "Profile deceleration".
Motor will be ramped to 0 rpm and the status of "Operation Enabled"(RUN).

Description Object 605D$_h$ "Halt option code"	
	This object was read only. -1 = the drive remains in state "operation enabled" with torque without ramp, in this state the drive will be in holding torque

Transition To "Fault Reaction Active" :



This transition is described in object 605Eh "Fault Reaction Option Code"

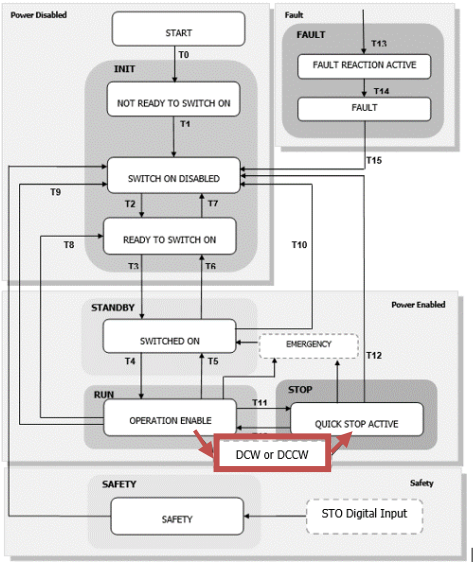
When the fault occurred the drive exits from "operation enabled" (RUN) (and from all states) considering the value of object 605Eh: it defines the configuration of transition.

Some alarms are not configurable with dynamic stop and the drive will stop (turn off the PWM immediately) and it will go in fault state without dynamic stop.

These alarms are: Over Current, Over Voltage, Over Speed, Feedback Resolver Error.

Description object 605Eh "Fault Reaction Option Code"	
Before FW 236:	This object is only read. When the fault occurred, the drive exits from "operation enabled" (RUN) (and from all states) considering the dynamic stop configuration with a proper ramp (if fault type allows a deceleration ramp). If the Dynamic Stop option is enabled the drive will go in "Fault Reaction Active" (FAULT) state with ramp controlled, if it is disabled it will go immediately (without ramp).
After FW 236:	This object is read/write. This transition can be configured writing value. The value can be: • 0 = the drive will go in "Fault Reaction Active" (FAULT) will go immediately (without ramp) and the motor will stop by inertia. • 1 = the drive will go in "Fault Reaction Active" (FAULT) state with ramp controlled. • -1 = the drive will go in "Fault Reaction Active" (FAULT) and it will stop immediately with setpoint set to 0 • -2 = the drive will go in "Fault Reaction Active" (FAULT) and it switch-off the drive power stage and the motor will stop by inertia. When the motor is not running the drive remains stopped with torque for a time delay • -3 = the drive will go in "Fault Reaction Active" (FAULT) and it will stop immediately with setpoint set to 0 and the motor brake is locked immediately • -4 = the drive will go in "Fault Reaction Active" (FAULT) and it switch-off the drive power stage and the motor brake is locked immediately • -5 = the drive will go in "Fault Reaction Active" (FAULT), switches-off and the motor will stop with a deceleration with a S-Ramp. This operation mode grants a deceleration with a specific ramp that can be configured by the user. (Enable from FW 240 with S-Ramp profile) All parameters of Dynamic Ramp are described in object "Dynamic Stop Parameters" 3007h. • The deceleration Ramp could be configured in Object 3007h: 4. • The deceleration S-Ramp could be configured in Object 3007h: 8.

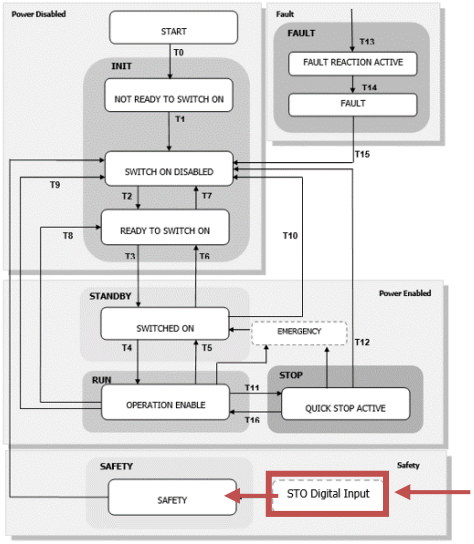
Transition to STOP "Quick Stop Active" with Dig. Input (DCW or DCCW)



If the digital inputs are configured like Digital Limit Switch Clockwise (DCW) and Digital Limit Switch Counterclockwise (DCCW) means that when these inputs are triggered and the motor speed is not zero (positive velocity and negative velocity correspondent), the drive will go in "Quick Stop Active" immediately with set point at 0.

To exit for this condition (and from state "quick stop active") the master controller has to send an opposite set point, or the limit switch must be disabled.

Transition To "Safety" state with STO



This transition is described in object 3102_h:0_h "STO option code"

From all states, also from "operation enabled" (RUN), it is possible to enable STO if it is available.
When STO will be triggered the drive switches-off and the motor will stop by inertia. Refer to chapter "Safety" to have more information.

Description object 3102_h "Configuration - STO Option Code"	
Before FW 236:	This object didn't exist and there were only one configuration: when the STO hardware is applied the drive immediately goes to safety and the drive disables the PWM. The brake is not activated and the motor will stop with inertia.
After FW 236:	This object is read/write. This transition can be configured writing value. The value can be: • 0 = the drive goes immediately in safety and it disables the PWM. The brake is not activated and the motor will stop with inertia. • 1 = the drive goes immediately in safety and it disables the PWM and it locks immediately the brake.

information

To have more information about safety read chapter "SAFETY"

3. | MEASURING UNIT CONVERSION

The Lafert Drive is used in different applications. For setting parameters easily in different applications, our clients could use the internal measuring unit conversion module to converse any users parameters into drive's internal unit.

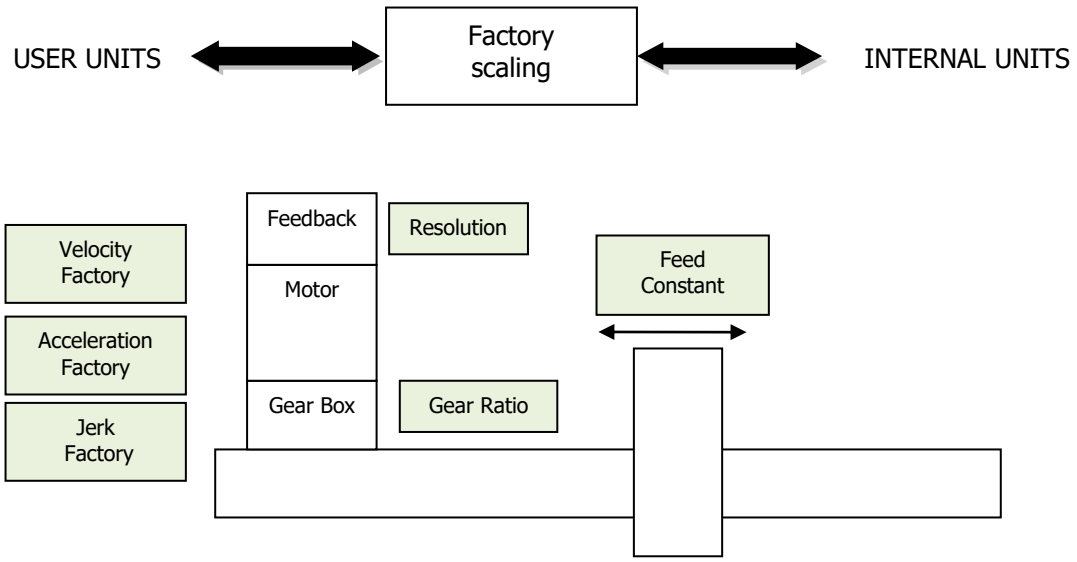


Figure 24 - Factory group

The objects of the Factor Group are used to convert internal position values, speed values and acceleration values into user-defined units.
Internal position values are entered in increments and are dependent on the resolution of the encoder used.
User-defined units depend on the encoder resolution and on attached linear reduction.

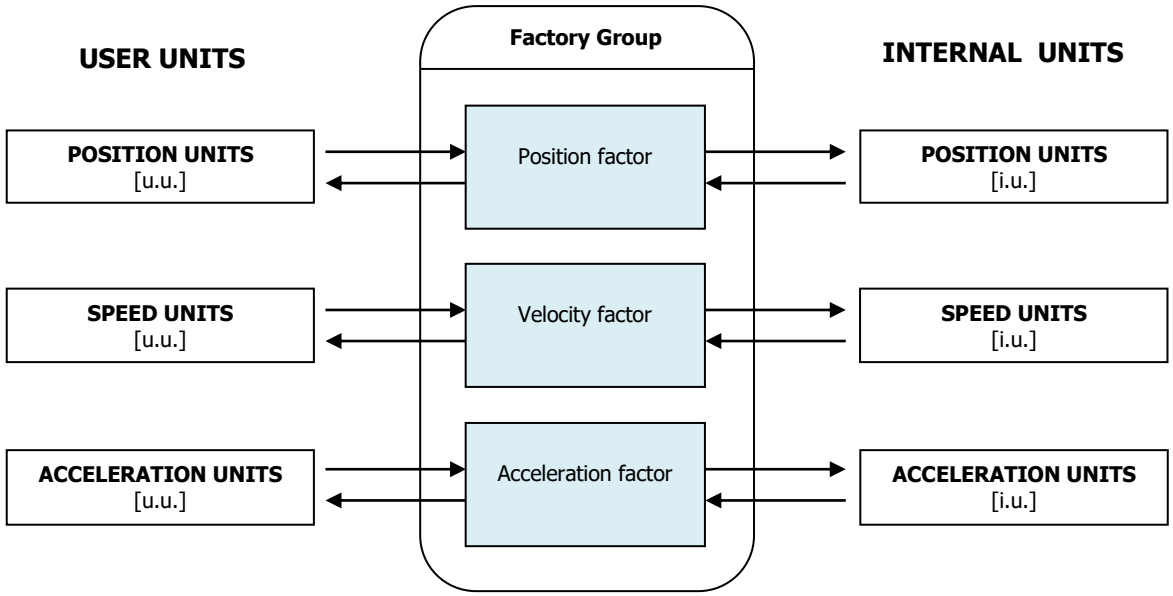


Figure 25 - Factory group units

All parameters are storage in internal units and parameters can be converted in user units using value of factor group.
The default values are following:

Object	Name	Internal Unit	Default User Unit
Length	Position Units	inc	inc
Speed	Speed Units	inc/s	inc/s
Acceleration	Acceleration Units	inc/s^2	inc/s^2
Jerk	Jerk Units	inc/s^3	inc/s^3

The factors defined in the factor group set up a relationship between device-internal units (increments) and physical units.

It defines [***u.u.***] as *user unit* and [***i.u.***] as *internal units*.

Caution

The drive has as a default unit the increments. If you want to change the units see the procedure. After the units changing It is mandatory modify also some objects (for example profile acceleration/deceleration, max acceleration/deceleration ...)

information

Referring to "APPENDIX" chapter to know the "How to change the user units"

MEASURING UNIT CONVERSION PARAMETER:

The factors are the result of the calculation of two parameters called dimension index and notation index.

Index	Name	Object Code	Data Type	Attr.	
$608F_h$	Position encoder resolution	ARRAY	UNSIGNED32	rw	not used
6090_h	Velocity Encoder Resolution	ARRAY	UNSIGNED32	rw	not used
6091_h	Gear Ratio	ARRAY	UNSIGNED32	rw	not used
6092_h	Feed Constant	ARRAY	UNSIGNED32	rw	not used
6096_h	Velocity Factor	ARRAY	UNSIGNED32	rw	used
6097_h	Acceleration Factor	ARRAY	UNSIGNED32	rw	used
$60A2_h$	Jerk Factor	ARRAY	UNSIGNED32	rw	used

Object 6096_h: Velocity Factor

This object can be used to match the velocity units to the user-defined velocity units.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6096_h	Velocity Factor	Array	UNSIGNED32	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Value default
00_h	Highest sub-index supported	ro	no	2	2
01_h	Numerator	rw	no	[1 ... 2147483647]	1
02_h	Divisor	rw	no	[1 ... 2147483647]	1

Numerator and divisor of the Velocity Factor has to be entered separately.

$$VelocityFactor = \frac{Numerator}{Divisor}$$

Internal velocity in internal unit is calculated from velocity in user unit using Numerator and Divisor according to the following formula:

$$Velocity[i.u.] = Velocity[u.u.] \times (\frac{Numerator}{Divisor})$$

The default value of user unit is the same of internal unit [inc/s]: then the numerator and divisor are both set to 1. If the user wishes to use a different unit of measurement, then he must change the Numerator and Divisor values.

Example 1:

- Internal Unit: [inc/s]
- User Unit: [rpm]

$$Velocity[inc/sec] = Velocity[rpm] \times \left(\frac{Resolution}{60}\right)$$

then Numerator = Resolution (see object 3004_h:2, resolution round of position transducer) and Divisor = 60

Example 2:

- Internal Unit: [inc/s]
- User Unit: [drpm] dozen of rpm

$$Velocity[inc/sec] = Velocity[drpm] \times \left(\frac{10 \times Resolution}{60}\right)$$

then Numerator = 10xResolution (see object 3004_h:2, resolution round of position transducer) and Divisor = 60

Example 3:

- Internal Unit: [inc/s]
- User Unit: [RPH] round per hour

$$Velocity[inc/sec] = Velocity[RPH] \times \left(\frac{Resolution}{3600}\right)$$

then Numerator = Resolution (see object 3004_h:2, resolution round of position transducer) and Divisor = 3600

Example 4:

- Internal Unit: [inc/s]
- User Unit: [m/s] meter for second

Suppose we have a pinion and a toothed belt connected to motor crankshaft, where one motor revolution corresponds to 30 linear cm.

$$Velocity[inc/sec] = Velocity[m/s] \times \left(\frac{Resolution}{0{,}3}\right)$$

then Numerator = Resolution (see object 3004_h:2, resolution round of position transducer) and Divisor = 0,3

Note: Data type of numerator and divisor is integer number, then you have a range of [-2147483648....+2147483647], you cannot represent a number with decimal. In this case you need to multiply both Numerator and Divisor to have integer number maintaining ratio value unchanged
Numerator = 10 * Resolution, Divisor = 3

The factor group used for the following objects:

- 60FF$_h$: Target Velocity
- 606D$_h$: Velocity Window
- 606F$_h$: Velocity Threshold
- 6081$_h$: Profile Velocity (for Profile Positioner Mode) (N/A)
- 6082$_h$: End Velocity (for Profile Positioner Mode) (N/A)

Caution

If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B19

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)

This object can be changed, to use the new value it must be saved in the memory of e2prom.

E^{2}prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6096$_h$
- Write signature "SAVE" in Store Parameters 1010$_h$ object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "APPENDIX" chapter to know the "How to change the user units"

Object 6097$_h$: Accelerator Factor

This object can be used to match the acceleration units to the user-defined acceleration units.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6097$_h$	Acceleration Factor	Array	UNSIGNED32	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Value default
00$_h$	Highest sub-index supported	ro	no	2	2
01$_h$	Numerator	rw	no	[1 ... 2147483647]	1
02$_h$	Divisor	rw	no	[1 ... 2147483647]	1

Numerator and divisor of the Acceleration Factor has to be entered separately.

$$Acceleration\ Factor = \frac{Numerator}{Divisor}$$

Internal acceleration in internal unit is calculated from acceleration in user unit using Numerator and Divisor according to the following formula:

$$Acceleration[i.u.] = Acceleration\ [u.u.] \times \left(\frac{Numerator}{Divisor}\right)$$

The default value of user unit is the same of internal unit [inc/s^2]: then the numerator and divisor are both set to 1.If the user wishes to use a different unit of measurement, then he must change the Numerator and Divisor values.

Example 1:

- Internal Unit: [inc/s^2]
- User Unit: [rpm/s]

$$Acceleration[inc/sec^2] = Acceleration[rpm/sec] \times \left(\frac{Resolution}{60}\right)$$

then Numerator = Resolution (see object 3004$_h$:2, resolution round of position transducer) and Divisor = 60

Example 2:

- Internal Unit: [inc/s^2]
- User Unit: [RPS/s] round per second at second

$$Acceleration[inc/sec^2] = Acceleration[RPS/sec] \times \left(\frac{Resolution}{1}\right)$$

then Numerator = 10xResolution (see object 3004$_h$:2, resolution round of position transducer) and Divisor = 1

The factor group used for the following objects:

- 6083$_h$: Profile Acceleration
- 6084$_h$:Profile Deceleration
- 60C5$_h$: Max acceleration
- 60C6$_h$: Max deceleration

Caution

If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B1A

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)

This object can be changed, to use the new value it must be saved in the memory of e2prom.

E^{2}prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6096$_h$
- Write signature "SAVE" in Store Parameters 1010$_h$ object (pay attention on the processing time)

- NMT Reset Node

information

Referring to "APPENDIX" chapter to know the "How to change the user units"

Object 60A2h: Jerk Factor

This object represents the conversion factor from internal to user-defined (physical) jerk units.
Object Description:

Index	Name EDS	Object Code	Data Type	Category
60A2h	Jerk Factor	Array	UNSIGNED32	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Value default
00h	Highest sub-index supported	ro	no	2	2
01h	Numerator	rw	no	[1 ... 2147483647]	1
02h	Divisor	rw	no	[1 ... 2147483647]	1

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)

This object can be changed, to use the new value it must be saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 60A2h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "APPENDIX" chapter to know the "SRAMP Profile"

4. | SAFETY

The drive moves in SAFETY state from all states (Except Fault State).

In this condition the drive is safe certificated.

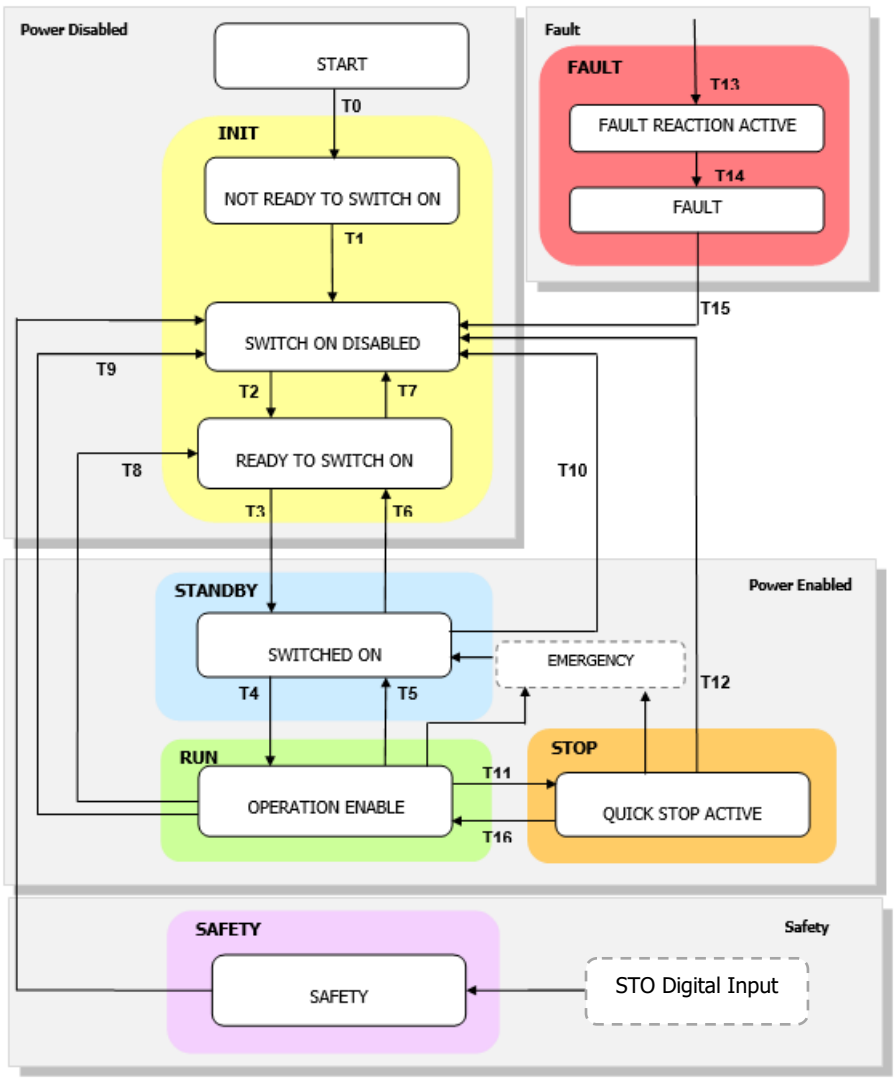
The Safety Circuit provides drive integrated safety function according to IEC 61800-5-2:2016.

STO (Safe Torque Off) Module prevents the creation of torque in the motor and it corresponds to a "*category 0*" an uncontrolled stop in accordance with stop of IEC 60204-1.

This means that the motor will coast to a stop when the "STO" function is activated, this method of stopping should be confirmed as being acceptable to the system the motor is driving.

SAFETY MANAGEMENT

The following picture shows the safety state. This state is added in the state machine DSP402.



Follow STO event (Digital Input - Safety input) the drive will go in SAFETY state of Finite State Machine (FSM). This state is added in Finite State Machine standard of DSP402.

The safety management can be configured by object 0x3103:0 "Configuration – Safety FSM". There is a default value (0) called "Standard Lafert", but it is possible to change with other configurations (it is possible change in run time and saved in e^2prom).

SAFETY F.S.M. MANAGEMENT:

When the STO event occurs there are 3 different cases:

- **STANDARD LAFERT:** (object 0x3103:0 = 0)

  Following an STO event, the drive switches to the "SAFETY" state of FSM.

  The drive is in a certified safe condition until the STO is triggered.

  It remains in this state even when the STO is disarmed.

  In this case bit 14 of the status word (SAFETY bit) is again set to '1'.

  When STO is removed, the drive is no longer safe condition but remains in the "SAFETY" state of FSM.

  To exit the "SAFETY" state of FSM and clear bit 14 of the statusword, it is necessary to send the "Disable Voltage" command of controlword (value 0x00).

  To know the level of the STO digital input, it is necessary to read the "STO statusword" (object 0x2080:0).

  The bits 0, 1, 2, 3, 5, 6 of statusword object define the last condition where the drive was when the STO occurred. To know if the drive is in safety state use bit 14 of the status word (SAFETY bit).

Example: command to exit from safety state:

Rx: Id 0x601 – 2b 40 60 00 00 00 00 00
Tx: Id 0x581 – 60 40 60 00 00 00 00 00

Rx: [cmd: 0x2b] [Index: 0x6040] [sub-index: 0x00] [Data: 0x00000000] – Command "disable Voltage"
Tx: [resp: 0x60] [Index: 0x6040] [sub-index: 0x00] [Data: 0x00000000] – Successful

The command "Disable Voltage" must have bit 1 and 7 with value 0 (see chapter Object description controlword 6040_h)

- **SAFETY DIRECT:** (object 0x3103:0 = 1)

  After an STO event, the drive goes into the "SAFETY" state of FSM but automatically it moves to the "SWITCH ON DISABLED" state of FSM.

  The drive is in a certified safe condition until the STO is triggered.

  Bit 14 of the status word (SAFETY bit) is set to '1' until STO is set.

  Once the STO is removed, the drive is still in the "SWITCH ON DISABLED" state of FSM.

  To move the drive to the FSM, it is necessary to write the controlword with the value 0x06 "shutdown" command.

To know the level of the STO digital input, it is necessary to read the "STO statusword" (object 0x2080:0).

The bits 0, 1, 2, 3, 5, 6 of statusword object defines the condition of drive in the "SWITCH ON DISABLED" state of FSM. To know if the drive is in safety state use bit 14 of the status word (SAFETY bit).

- **SAFETY FAULT:** (object 0x3103:0 = 2) *(it is not available)*

  After an STO event the drive produces a fault (error code in emergency protocol 0xFF01).

  After that, the drive goes in "FAULT" state of FSM.

  The master can send a "Reset" bit command by controlword and the drive moves in the "SWITCH ON DISABLED" state of FSM.

  To move the drive into FSM it is necessary to write the controlword command "shutdown" 0x06 and the STO must be removed.

  ➔ TO USE THIS FEATURE CONTACT MANUFACTURER

BRAKE MANAGEMENT:

Brake management during SAFETY is handled by CAN object (0x3102:0) called "Configuration – STO Option Code". The values are:

- 0 = brake released, default value (it will then be automatically closed in "switched-on")
- 1 = brake closed immediately and held before switching to "operation enabled"

Caution

- **EMERGENCY COMMAND (NOT SAFETY CERTIFIED):**

  If It is given an *Emergency Command* (Digital Input/Control Word - **not Safety input**) when the drive is in RUN/STOP state then the drive goes to standby (T11/T11a). If you need to give a run command (or stop command) with controlword is necessary to clear *Emergency Command* (Digital Input) and send a controlword to move in RUN/STOP state.

  The transition correct is

  - RUN/STOP→ STANDBY → RUN/STOP

- **STO COMMAND (SAFETY CERTIFIED):**

  If is given an *STO Command* (Digital Input - **Safety input**) the drive state goes to SAFETY state. To exit from SAFETY state depend by SAFETY FSM configuration object 0x3103:0

  The transition correct is

  - Configuration 0, 1: ALL STATES → SAFETY → INIT→ STANDBY → RUN/STOP
  - Configuration 3: ALL STATES → SAFETY → FAULT → INIT→ STANDBY → RUN/STOP

SAFETY OBJECT

The object concerning safety are:

- Object 6041_h:0 bit 14 of "statusword" – it defines if the drive in "SAFETY" state of FSM
- Object 2080_h:0 "STO statusword" – it communicates the level state of digital inputs

- Object 3102h:0 "Configuration STO Option Code" – it defines the brake management after STO event
- Object 3103h:0 "Configuration Safety FSM" – it defines the safety management on Finite State Machine after STO event
- Object 4000h:1 "Safety State" – it communicates if the drive has read the STO level to move the drive in safety
- Object 4000h:2 "STO Function" – it defines the level of STO digital input after counter validation

SOFTWARE SPECIFICS

Correct procedure to SAFETY is: RUN → STANDBY → SAFETY.

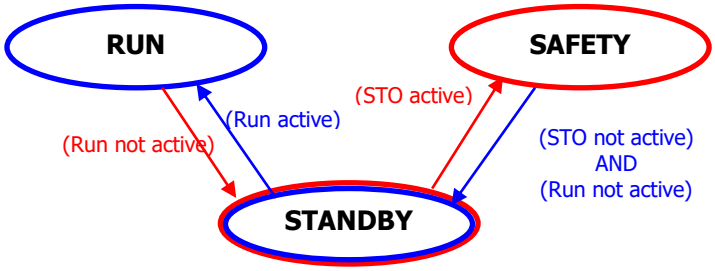


Figure 26 - STO Machine

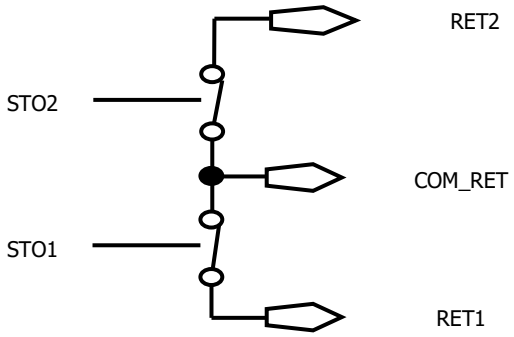
Caution

The Drive cannot hold the load with the STO function activated because the motor no longer supplies any torque.

- If the STO function is activated during operation, the drive will stop in an uncontrolled manner.
- If the drive has the Safety Torque OFF (STO), verify that this circuit is correctly supplied before all operation functions.

STO features SMARTRIS STANDARD:

STO circuit concept uses a two channel architecture. This architecture is shown in the system block diagram below.



The two isolated differential STO inputs have to be connected at 24V voltage to allow that the motor operates. The STO digital inputs status are written in the object digital input 60FDh.

Input 1	Input 2	Output 1	Output 2	STO	Output SW
STO1	STO2	RET1	RET2	statusword	STATUS
0	0	CLOSE	CLOSE	SAFE STO	SAFETY
24V	0	OPEN	CLOSE	SAFE STO 1	SAFETY
0	24V	CLOSE	OPEN	SAFE STO 2	SAFETY
24V	24V	OPEN	OPEN	NO STO	OPERATION MODE

In the Safety State, the drive will not produce torque or force in the motor. The STO function achieves and maintains a safe state by disabling the ability of the attached motor to produce torque/force.

This both halts any drive induced acceleration already in process and prevents initiation of motion.
The expectation is that an inability of the motor to produce torque/force translates into a reduction of risk of hazardous motion for the larger system.

STO features SMARTRIS COMPACT:
STO circuit concept uses a two channel architecture. This architecture is shown example schematic application.

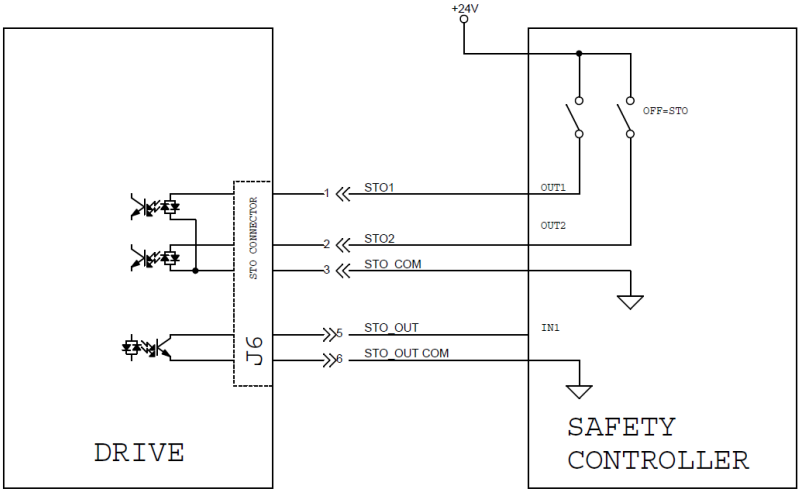


Figure 27 - example STO circuit for Smartris compact application

Safety controller can read "STO_OUT" digital output that corresponding to STO STATUS OUTPUT.

The Drive continuously check the correct operation of STO Internal Circuits, during start-up and during safety and operating states.

If one of these checks fails, the drive is placed in an FAULT state, operation is inhibited and the type of error is indicated.

In this state of STO circuit error (drive in FAULT state) "DRIVE OK" digital output and "STO_OUT" digital output are OPEN.

Verify that STO1 and STO2 behavior is congruent as in table below in every states.

Input 1	Input 2	Output	STO	Output SW
STO1	STO2	STO OUT	statusword	STATUS
0	0	CLOSE	SAFE STO	SAFETY
0	24V	CLOSE	SAFE STO 1	SAFETY
24V	0	CLOSE	SAFE STO 2	SAFETY
24V	24V	OPEN	NO STO	OPERATION MODE

5. | CANOPEN OBJECT DICTIONARY

GENERAL OBJECTS (DS301)

Object 1000h: Device Type

The device type specifies the kind of device. The lower 16 bit contain the device profile number and the upper 16 bit an additional information.

Object Description:

Index	Name	Object Code	Data Type	Category
1000h	Device Type	VAR	U32	O

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Device Type	ro	no	0xFF7A0192

Value Definition:

Bit MSB 31	Bit LSB 0
Additional Information	Device Profile Number

Default value for Lafert Drive is **0xFF7A0192**, the number 0192h means the device uses the profile 402.

Object 1002h: Manufacturer status register

This object shall provide a common status register for manufacturer-specific purposes. In this specification only the size and the location of this object are defined.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1002h	Manufacturer Status Register	VARIABLE	U32	O

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Manufacturer Status Register	ro	no	-

Object 1005h: COB-ID SYNC

COB-ID of the synchronization object.
The device generates a SYNC message if bit 30 is set. The meaning of other bits is equal to the other communication objects.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1005h	COB-ID SYNC	VARIABLE	UNSIGNED32	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	COB-ID SYNC	rw	no	80h

Bit Coded information:

BIT	VALUE	Meaning
31	x	reserved
30	0_b 1_b	Device does not generate SYNC message Device generates SYNC message (not supported)
29	0_b 1_b	11-bit ID (CAN 2.0A) 29-bit ID (CAN 2.0B)
28 - 11	0 X	If bit 29 = 0 If bit 29 = 1: bits 28 - 11 of 29-bit SYNC COB-ID
10 - 0	X	bits 10-0 of COB-ID

information

The drive does not support SYNC message generation and only supports ID of bit 11-bit ID (CAN 2.0A). Therefore, bits 11 through 30 are always 0. The only COBID admissible is 0x80.

The drive will sent the follow abort codes:

- 0x06090030 = Invalid value for parameter (download only). If the value is different to 0x80

Object 1008_h: Manufacturer Device Name

This object contains the device name.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1008_h	Manufacturer Device Name	VARIABLE	STRING (4 char)	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Manufacturer device name	cost	no	-

Example:

The product device named Smartris the string will be "SMA".

Object 1009_h: Manufacturer Hardware Version

This object contains the device hardware version.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1009_h	Manufacturer Hardware Name	VARIABLE	STRING (4 char)	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Manufacturer Hardware name	cost	no	-

Example:

The hardware version number 1.1 (it means the first version with STO), the string will be "11".

Object 100Ah: Manufacturer Software Version

This object contains the device software version.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
100Ah	Manufacturer Software Name	VARIABLE	STRING (4 char)	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Manufacturer Software name	cost	no	-

Example:

The software version number 1.0.8 then the string will be "108".

Object 100Ch: Guard Time

This entry contains the guard time in milliseconds. The value 0 switches node guarding off.
The guard time multiplied with the life time factor object 100Dh gives the life time for the life guarding protocol.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
100Ch	Guard Time	VARIABLE	U16	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Guard Time	rw	no	[0- 65535]	0

Caution

The Heartbeat protocol has a higher priority than Node-Guarding. If both protocols are activated simultaneously, the Node Guarding Timer is suppressed, but no EMCY message is sent either.

Object 100Dh: Life Time Factor

The Life Time Factor multiplied by the Guard Time Object 100Ch gives the Life Time for the Node Guarding. The value 0 switches the Node Guarding off.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
100Ah	Life Time Factor	VARIABLE	U8	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Life Time Factor	rw	no	[0-255]	0

Object 1017h: Producer Heartbeat Time

The producer heartbeat time defines the cycle time of the heartbeat. If the time is 0 it is not used. The time has to be a multiple of 1 msec.

Object Description:

Index	Name	Object Code	Data Type	Category
1017_h	Producer Heartbeat Time	VAR	U16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Producer Heartbeat Time	rw	no	[0 – 65535]	0

Data Byte for NMT state evaluation of the HeartBeat producer:

- 0 (00_h): "Boot-Up"
- 4 (04_h): "Stopped"
- 5 (05_h): "Operational"
- 127 ($7F_h$) "Pre-operational

Object 1010_h: Store Parameters Field

This object supports the saving of parameters in non-volatile memory. By read access the device provides information about its saving capabilities.

Several parameter groups are distinguished.

- *Sub index 0:* contains the largest Sub-Index that is supported.
- *Sub index 1:* refers to all parameters that can be stored on the device.
- *Sub index 2:* refers to communication related parameters (Index 1000h - 1FFFh manufacturer specific communication parameters).
- *Sub index 3:* refers to application related parameters (Index 6000h - 9FFFh manufacturer specific application parameters).
- *Sub index 4 - 127:* manufacturers may store their choice of parameters individually.
- *Sub-Index 128 – 254:* are reserved for future use.

This command can only be carried out if the module isn't in "Operation Enabled" or "Quick Stop Active". If the command can't be processed then the drive sends a warning message and set a '1' the third bit of warning object (see **Errore. L'origine riferimento non è stata trovata.** Flag)

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1010_h	Store Parameter Field	ARRAY	U32	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Entries	c	no	5
01_h	Save all Parameters	rw	no	0
02_h	Save Communication Parameters	rw	no	0
03_h	Save Application Parameters	rw	no	0
04_h	Save Manufacturer Parameters	rw	no	0
05_h	Reserved	rw	no	0

In order to avoid storage of parameters by mistake, storage is only executed when a specific signature is written to the appropriate Sub-Index.

The signature is "save": 0x65766173.

Signature ISO 8859 ("ASCII")	e	v	a	s
hex	65h	76h	61h	73h

On reception of the correct signature in the appropriate sub-index the CANopen device shall store the parameter and then it shall confirm the SDO transmission (SDO download initiate response).

Example: Command to save:

Rx: Id 0x601 - 23 10 10 01 73 61 76 65
Tx: Id 0x581 – 60 10 10 01 00 00 00 00

Rx: [cmd: 0x23] [Index: 0x1010] [sub-index: 0x01] [Data: 0x65766173] – SAVE all parameters
Tx: [resp: 0x60] [Index: 0x1010] [sub-index: 0x01] [Data: 0x00000000] - Successful

The drive will sent the follow abort codes:

- 0x06060000 = If the storing failed, the CANopen device shall respond with the SDO abort transfer service.
- 0x08000002 = If a wrong signature is written, the CANopen device shall refuse to store and it shall respond with the SDO abort transfer service.

Caution

The Vdc-Link must be present to SAVE parameters.

If the drive is connect only with an auxiliary Voltage (+24V) the drive answers with an ABORT CODE.

Example: Command to save without Vdc-Link:

Rx: Id 0x601 - 23 10 10 01 73 61 76 65
Tx: Id 0x581 – 80 10 10 01 00 00 06 06

Rx: [cmd: 0x23] [Index: 0x1010] [sub-index: 0x01] [Data: 0x65766173] – SAVE all parameters
Tx: [resp: 0x80] [Index: 0x1010] [sub-index: 0x01] [Data: 0x06060000] – Abort Code

Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

information

After command STORE the drive MUST be reset to have the new parameters.

On read access to the appropriate sub-index the CANopen device shall provide information about its storage functionality with the following format:

Bit MSB		Bit LSB
31 2	1	0
Reserved	Auto	Cmd

Structure of read access:

Bit	Field	Configuration	Definition
0	Cmd	0_b 1_b	CANopen device does not save parameters on command CANopen device saves parameters on command
1	Auto	0_b 1_b	CANopen device does not save parameters autonomously CANopen device saves parameters autonomously

Object 1011_h: Restore default parameters

This entry supports restoring of default parameters. With a read access the device provides information about its capabilities to restore these values.
Several parameter groups are distinguished.

- *Sub index 0:* contains the largest Sub-Index that is supported
- *Sub index 1:* Restore all factory settings
- *Sub index 2:* Restore all factory settings for communications parameters (0x0000 to 0x1FFF)
- *Sub index 3:* Restore all factory settings for application parameters (from 0x2000)
- *Sub index 4- 127:* manufacturer defined parameters

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1010_h	Store Parameter Field	ARRAY	U32	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Entries	c	no	5
01_h	Restore all Default Parameters	rw	no	0
02_h	Restore Communication Default Parameters (N/A)	rw	no	0
03_h	Restore Application Default Parameters (N/A)	rw	no	0
04_h	Restore Manufacturer Parameters (N/A)	rw	no	0
05_h	Reserved	rw	no	0

The object "Restore Default Parameters" loads the standard configuration parameters. The standard configuration parameters are either that as delivered or those last saved. Read access supplies information about the restore options. For restoring the signature "load" (**0x64616f6c**) must be written.

"Load" signature:

Signature ISO 8859 ("ASCII")	d	a	o	l
hex	64h	61h	6Fh	6Ch

On reception of the correct signature in the appropriate sub-index the CANopen device shall restore the default parameters and then it shall confirm the SDO transmission (SDO download CANopen application layer and communication profile initiate response).

Example: Command to restore:

Rx: Id 0x601 - 23 11 10 01 6C 6F 61 64
Tx: Id 0x581 – 60 11 10 01 00 00 00 00

Rx: [cmd: 0x23] [Index: 0x1011] [sub-index: 0x01] [Data: 0x64616F6C] – RESTORE factory parameters
Tx: [resp: 0x60] [Index: 0x1011] [sub-index: 0x01] [Data: 0x00000000] - Successful

The drive will sent the follow abort codes:

- 0x06060000 = If the storing failed, the CANopen device shall respond with the SDO abort transfer service.
- 0x08000002 = If a wrong signature is written, the CANopen device shall refuse to store and it shall respond withthe SDO abort transfer service.

The default values shall be set valid after the CANopen device is reset (NMT service reset node for sub-index from 01_h to $7F_h$, NMT service reset communication for sub-index 02_h) or power cycled.

information

After command RESTORE the drive MUST be reset to have the factory parameters.

On read access to the appropriate sub-index the CANopen device shall provide information about its default parameter restoring capability with the following format:

Bit MSB 31 ... 1	Bit LSB 0
Reserved	CMD

Structure of read access:

Bit	Field	Configuration	Definition
0	Cmd	0_b 1_b	CANopen device does not restore default parameters CANopen device restores parameters

Object 1014h: COB-ID Emergency Message

Object Description:

Index	Name	Object Code	Data Type	Category
1010_h	COB-ID EMCY	VAR	U32	optional

Entry Description:

Sub-Index	Access	PDO mapping	Default Value
00_h	ro	no	0x80 + IdNode

Object 1018_h: Identity object

This object shall provide general identification information of the CANopen device.

- **Sub-index 01_h**: shall contain the unique value1 that is allocated uniquely to each vendor of a CANopen device.The value 0000 0000_h shall indicate an invalid vendor-ID.
- **Sub-index 02_h:**shall contain the unique value that identifies a specific type of CANopen devices.The value of 0000 0000_h shall be reserved.
- **Sub-index 03_h:** shall contain the major revision number and the minor revision number of therevision of the CANopen device. The major revision number shall identify a specific CANopen behaviour. That means if the CANopen functionality is different, the major revision number shall be incremented. The minor revision number shall identify different versions of CANopen device with the same CANopen behaviour. The value of 0000 0000_h shall be reserved.
- **Sub-index 04_h:** shall contain the serial number that identifies uniquely a CANopen device within a product group and a specific revision. The value of 0000 0000_h shall be reserved.

Object Description:

Index	Name	Object Code	Data Type	Category
1018h	Identity object	RECORD	Identity	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Highest sub-index supported	ro	no	4
01h	Vendor-ID	ro	no	000001FBh
02h	Product code	ro	no	reserved
03h	Revision number	ro	no	reserved
04h	Serial number	ro	no	reserved

Object 1400h: Receive PDO1 Communication Parameter

It contains the communication parameters of the current PDO1 the device is able to receive.

Object Description:

Index	Name	Object Code	Data Type	Category
1400h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1401h: Receive PDO2 Communication Parameter

It contains the communication parameters of the current PDO2 the device is able to receive.

Object Description:

Index	Name	Object Code	Data Type	Category
1401h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1402h: Receive PDO3 Communication Parameter

It contains the communication parameters of the current PDO3 the device is able to receive.

Object Description:

Index	Name	Object Code	Data Type	Category
1402h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1403h: Receive PDO4 Communication Parameter

It contains the communication parameters of the current PDO4 the device is able to receive.

Object Description:

Index	Name	Object Code	Data Type	Category
1403h	Receive PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1600h: Receive PDO1 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1600h	Receive PDO1 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	rw	no	UNSIGNED8	0..8
01h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1601h: Receive PDO2 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1601h	Receive PDO2 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00_h	Number of Entries	rw	no	UNSIGNED8	0..8
01_h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02_h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03_h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04_h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05_h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06_h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07_h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08_h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1602_h: Receive PDO3 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1602_h	Receive PDO3 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00_h	Number of Entries	rw	no	UNSIGNED8	0..8
01_h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02_h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03_h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04_h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05_h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06_h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07_h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08_h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1603_h: Receive PDO4 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1603_h	Receive PDO4 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00_h	Number of Entries	rw	no	UNSIGNED8	0..8
01_h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02_h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03_h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04_h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05_h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06_h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07_h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08_h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1800h: Transmit PDO1 Communication Parameter

It contains the communication parameters of the current PDOn the device is able to transmit.

Object Description:

Index	Name	Object Code	Data Type	Category
1800h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1801h: Transmit PDO2 Communication Parameter

It contains the communication parameters of the current PDOn the device is able to transmit.

Object Description:

Index	Name	Object Code	Data Type	Category
1801h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1802h: Transmit PDO3 Communication Parameter

It contains the communication parameters of the current PDOn the device is able to transmit.

Object Description:

Index	Name	Object Code	Data Type	Category
1802h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1803h: Transmit PDO4 Communication Parameter

It contains the communication parameters of the current PDOn the device is able to transmit.

Object Description:

Index	Name	Object Code	Data Type	Category
1803h	Transmit PDO Communication Parameter	PDO_COMM_PAR	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	ro	no	UNSIGNED8	5
01h	COB-ID	ro	no	UNSIGNED32	0..0xFFFFFFFF
02h	Transmission Type	ro	no	UNSIGNED8	0..0xFF
03h	Inhibit Time	ro	no	UNSIGNED16	0..0xFFFF
04h	Compatibility Entry	ro	no	UNSIGNED8	0..0xFF
05h	Event Timer	ro	no	UNSIGNED16	0..0xFFFF

Object 1A00h: Transmit PDO1 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1A00h	Transmit PDO1 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	rw	no	UNSIGNED8	0..8
01h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1A01h: Transmit PDO2 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1A01h	Transmit PDO2 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	rw	no	UNSIGNED8	0..8
01h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF

05h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1A02h: Transmit PDO3 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1A02h	Transmit PDO3 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	rw	no	UNSIGNED8	0..8
01h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

Object 1A03h: Transmit PDO4 Mapping Parameter

The sub-indices contain the indices, the sub-indices and the lengths of the mapped sub-indices.

Their structure is as follows: Value: index (16bit) + sub-index (8bit) + length (8bit)

Object Description:

Index	Name	Object Code	Data Type	Category
1A03h	Transmit PDO4 Mapping Parameter	PDO_MAPPING	RECORD	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Number of Entries	rw	no	UNSIGNED8	0..8
01h	Mapping Entry 1	rw	no	UNSIGNED32	0..0xFFFFFFFF
02h	Mapping Entry 2	rw	no	UNSIGNED32	0..0xFFFFFFFF
03h	Mapping Entry 3	rw	no	UNSIGNED32	0..0xFFFFFFFF
04h	Mapping Entry 4	rw	no	UNSIGNED32	0..0xFFFFFFFF
05h	Mapping Entry 5	rw	no	UNSIGNED32	0..0xFFFFFFFF
06h	Mapping Entry 6	rw	no	UNSIGNED32	0..0xFFFFFFFF
07h	Mapping Entry 7	rw	no	UNSIGNED32	0..0xFFFFFFFF
08h	Mapping Entry 8	rw	no	UNSIGNED32	0..0xFFFFFFFF

MANUFACTURER OBJECTS- SETTINGS PARAMETERS

Object 2000h: Id-Node

The object allows the user to set the CAN Id-Node of the Node, the change takes effect at next power cycle.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2000h	IdNode	VAR	UNSIGNED8	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Value Range
00h	CAN IdNode	rw	no	1 ... 127	1

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)

This object can be changed, to use the new value it must be saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 2000h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "APPENDIX" chapter to know "How to change Id-Node"

Example: Drive with Id-Node 1, command to change Id-Node '3' :

Rx: Id 0x601 – 2F 00 20 00 03 00 00 00
Tx: Id 0x581 – 60 00 20 00 00 00 00 00

Rx: [cmd: 0x2F] [Index: 0x2000] [sub-index: 0x00] [Data: 0x00000003] – new Id-Node value '3'
Tx: [resp: 0x60] [Index: 0x2000] [sub-index: 0x00] [Data: 0x00000000] – Success

After that the new value must be saved in e²prom with command store (1010h) and it needs at power cycle or reset.

Description Message:

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
cmd	Index		Sub-Index	Data Value			

Object 2001h: CAN Baud Rate

The object allows the user to set the CAN bit rate of the Node, the change takes effect at next power cycle.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2001h	CAN Baudrate	VAR	UNSIGNED16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	CAN Baudrate	rw	no	See table	0x03E8

Valid entries:

BaudRate	Entry	Lafert Servo Drive
10 kBit/s	0x000A	Available
20 kbit/s	0x0014	Available
50 kbit/s	0x0032	Available
100 kbit/s	0x0064	Available
125 kbit/s	0x007D	Available
250 kbit/s	0x00FA	Available
500 kbit/s	0x01F4	Available
800 kbit/s	0x0320	Available
1000 kbit/s	0x03E8	Available

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)

This object can be changed, to use the new value it must be saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 2001h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "APPENDIX" chapter to know "How to change BaudRate"

Example: command to change new Baudrate value 500 Kbit :

Rx: Id 0x601 – 2B 01 20 00 F4 01 00 00
Tx: Id 0x581 – 60 01 20 00 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x2001] [sub-index: 0x00] [Data: 0x00000003] – new Baudrate 500K = 0x01F4
Tx: [resp: 0x60] [Index: 0x2001] [sub-index: 0x00] [Data: 0x00000000] – Success

After that the new value must be saved in e²prom with command store (1010h) and it needs at power cycle or reset.

Object 3001h: Absolute Limits Parameters

This object describes the Absolute Limits. These parameters are only in reading because they are set by manufacturer.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3001h	Absolute Limits Parameters	ARRAY	UNSIGNED32	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	-	5
1	Velocity ABS	ro	no	[0 - 2147483647] rpm	defined by application
2	Acceleration ABS	ro	no	[0 - 2147483647] rpm/s	defined by application
3	Limit Profile Velocity	ro	no	[0 - 65535] rpm	defined by application
4	free	ro	no	-	-
5	free	ro	no	-	-

Value Definition:

Sub-Index	Field	Configuration	Definition
1	Velocity ABS	[rpm]	It is maximum absolute value of Velocity profile. It is a limit for 607Fh(Max Velocity Profile).
2	Acceleration ABS	[rpm /s]	It is maximum absolute value of acceleration profile. It is a limit for 60C5h (Max Acceleration) and 60C6h (Max Deceleration).
3	Limit Profile Velocity (Min Value)	[rpm]	It is a Limit Velocity for Profile Mode. It is a minimum between 607Fh (Max Velocity Profile) and 3300h (Full Scale Velocity)

These parameters are the maximum rating of drive and they are only reading.

- The velocity parameters have to be lower than the "velocity ABS" object (3001h:1).

Example:

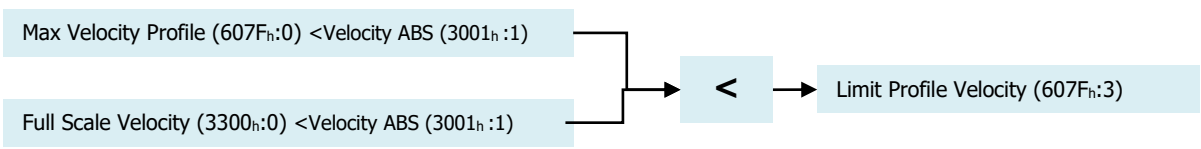
If the "Velocity ABS" is 4500 rpm then the "Max Velocity Profile" (607Fh:0) and "Velocity Full Scale" (3300h:0) must be smaller or equal 4500 rpm. Remember that these objects are saved in user units, pay attention at the conversion.

- The acceleration parameters have to be lower than the acceleration "ABS object" (3001h:2).

Example:

If the "Acceleration ABS" is 2228 rpm/s then the "Max Acceleration" (60C5h:0) and the "Max Deceleration" (60C6h:0) must be smaller or equal 2228 rpm/s. Remember that these objects are saved in user units, pay attention at the conversion.

- The "Limit Profile Velocity" (607Fh:3) is the limit value of profile velocity, in fact this object is the minimum between 607Fh (Max Velocity Profile) and 3300h (Full Scale Velocity). Therefore, the "Target Velocity" (60FFh:0) will be limited by the "Limit Profile Velocity" (607Fh:3).



The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

Object 3002h: Motor Brake Parameters

This object describes the parameters of Electronic Brake Configuration.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3002h	Brake Parameters	ARRAY	INTEGER16	Mandatory IF

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	-	8
01h	Motor Brake Option	rw	no	[0,1]	defined by application
02h	Motor Brake Delay	rw	no	[1... 32767]	defined by application
03h	Unlock Motor Brake	rw	no	[1 ... 32767]	defined by application
04h	Motor Brake timeout	rw	no	[1 ... 32767]	defined by application
05h	Automatic/Manual Mode Configuration	rw	no	[0,1]	defined by application
06h	Motor Brake Status	ro	no	[0,1]	-
07h	Motor Brake Type	ro	no	[1,2]	defined by application
08h	reserved	rw	no	[0,1]	defined by application

Value Definition:

Sub-Index	Field	Configuration	Definition
01h	Motor Brake Option (*)	0b 1b	Motor Brake disabled or Motor Brake is not present Motor Brake enabled
02h	Motor Brake Delay	[ms * 10]	Delay open command. This timeout is the delay between STBY Status and unlock brake.
03h	Unlock Motor Brake time	[ms * 10]	Delay between STOP and RUN mode before unlock Brake. This timeout depends by kind of motor brake.
04h	Max Motor Brake Timeout	[ms * 10]	Max time programmed for natural Inertia deceleration. At the end of this timeout the brake is locked and drive will be in STBY status.
05h	Automatic/Manual Mode Configuration	0b 1b	Automatic Mode Activated Manual Mode Activated
06h	Motor Brake Status	0b 1b	Brake Status: activated → Motor is locked Brake Status: released → Motor is not locked
07h	Motor Brake Type	1 2	Magnetic Brake Spring Brake
08h	reserved	0 1	reserved

Caution

(*) If the motor does not have the brake, this value has 0 as default value.

The user CAN'T enable because it is not present.

The Motor Brake Management is an output who can drive and supply power directly to a motor brake.

Example: Enable Motor Brake

To enable this function, you must write 1 in "motor Brake Option" object 3002h: 1.

Rx: Id 0x601 - 2B 02 30 01 01 00 00 00
Tx: Id 0x581 – 60 02 30 01 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x3002] [sub-index: 0x01] [Data: 0x00000001] – Enable Brake
Tx: [resp: 0x60] [Index: 0x3002] [sub-index: 0x01] [Data: 0x00000000] – Success

After that the new value must be saved in e^2prom with command store (1010_h) and it needs at power cycle.

The type of brake available is

- Magnetic Brake
- Spring Brake

It is possible to know what brake is used reading the index object 3002_h: 7.

Example: Read Type Motor Brake

To Read type Motor Brake, you must read object 3002h:7.

Rx: Id 0x601 - 40 02 30 07 00 00 00 00
Tx: Id 0x581 – 4B 02 30 07 01 00 00 00

Rx: [cmd: 0x40] [Index: 0x3002] [sub-index: 0x07] [Data: 0x00000000] – Read type Motor Brake
Tx: [resp: 0x4B] [Index: 0x3002] [sub-index: 0x07] [Data: 0x00000001] – Success

AUTOMATIC/MANUAL BRAKE:

The Motor Brake can be configured in Automatic Mode or in Manual Mode.

- **Automatic Mode:** the brake will be released (Brake+ = 24V) automatically when the Drive is set in "Operation Enabled" (RUN STATE) and is activated automatically in other states.
- **Manual Mode:** the user can be released the brake using a dedicated command in the object "Digital Outputs" $60FE_h$: 1.

To set the Motor Brake in Manual Mode write 1 the "Automatic/Manual Mode Configuration" object 3002_h: 5 and saving the new value in e2prom (see procedure E2prom Store).

The brake is configured as default in automatic mode.

Example: command to change in Manual Mode :

To enable Motor Brake in Manual Mode, you must write "1" in object 3002_h5_h.

Rx: Id 0x601 – 2B 02 30 05 01 00 00 00
Tx: Id 0x581 – 60 02 30 05 00 00 00 00

Rx: [cmd: 0x40] [Index: 0x3002] [sub-index: 0x05] [Data: 0x00000001] – 1 = Manual configuration
Tx: [resp: 0x60] [Index: 0x3002] [sub-index: 0x05] [Data: 0x00000000] – Success

After that the new value must be saved in e^2prom with command store (1010_h) and it needs at power cycle.

Manual Brake Mode

If the drive is set in manual brake the master controller has completely the brake control.

To change the state of brake using the object "Digital Output" object $60FE_h$ bit 1.

- Bit 1 = 0 : Brake Activated → Motor Locked
- Bit 1 = 1 : Brake Released → Motor Free

Example:

1. Command to Released Brake Motor in Manual Mode:

   Rx: Id 0x601 – 23 FE 60 01 01 00 00 00
   Tx: Id 0x581 – 60 FE 60 01 00 00 00 00

Rx: [cmd: 0x23] [Index: 0x60FE] [sub-index: 0x01] [Data: 0x00000001] – 1 = Released
Tx: [resp: 0x60] [Index: 0x60FE] [sub-index: 0x01] [Data: 0x00000000] – Success

2. Command to Activated Brake Motor in Manual Mode:

   Rx: Id 0x601 – 23 FE 60 01 00 00 00 00
   Tx: Id 0x581 – 60 FE 60 01 00 00 00 00

Rx: [cmd: 0x23] [Index: 0x60FE] [sub-index: 0x01] [Data: 0x00000001] – 0 = Activated
Tx: [resp: 0x60] [Index: 0x60FE] [sub-index: 0x01] [Data: 0x00000000] – Success

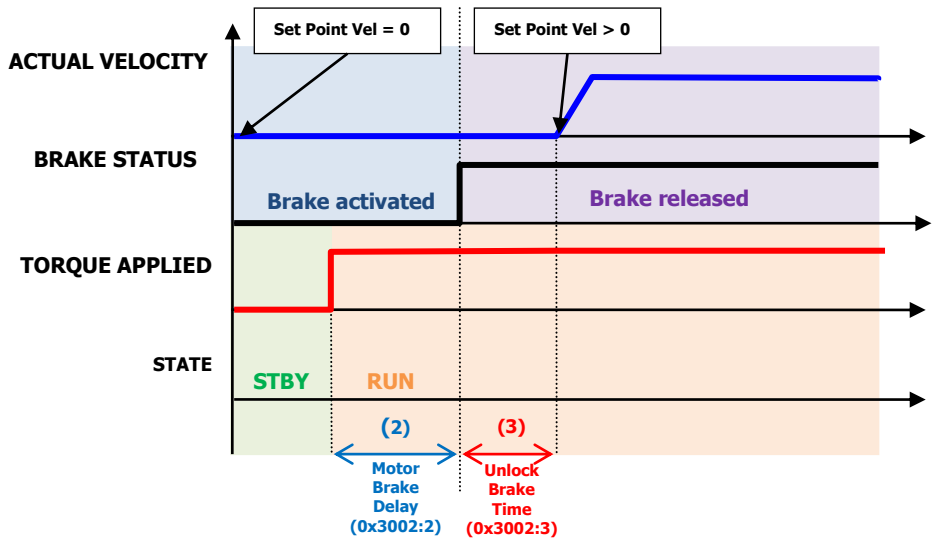
Automatic Brake Mode

When the drive is configured in Automatic Mode the brake will involve the status following the graphs.

Switched-On" (STANDBY) → "Operation Enabled" (RUN)

The follow graph describes the timing of the brake when the drive moves from "Switched-On" (STANDBY state) to "Operation Enabled" (RUN state). This transition can be set with controlword ('enable operation' command).

When the drive goes in "operation enabled" the brake is activated for "motor brake delay" (3002h: 2) timer, after that the brake will release. After time "Unlock brake time" (3002h: 3) the actual velocity increases.



"Operation Enabled" (RUN) → "Switched-On" (STANDBY)
NORMAL CASE

The follow graph describes the timing of the brake when the drive moves from "Operation Enabled" (RUN state) to "Switched-On" (STANDBY state). This transition can be set with controlword ('enable operation' command) or emergency command (via controlword or with digital input)

The motor movement can have dynamic stop enabled or not. In the end when the motor is not running (rotational speed is less than +/- 5 RPM) the brake will be activated.

If the dynamic stop is activated then the braking off is controlled and the motor will stop with a ramp.
If the dynamic stop is not activated then the motor will stop with different mode depending by configuration.

WITH ELECTRONIC DYNAMIC STOP

If Dynamic Stop is activated the braking off is controlled and the motor will stop with a ramp defined by configuration parameters.

Between "Operation Enabled" (RUN state) and "Switched-On" (STANDBY state) the drive will go in "Dynamic Stop" state.

Referring to "FUNCTIONS" chapter to know the different "Braking Mode"

WITHOUT ELECTRONIC DYNAMIC STOP

If the dynamic stop is not activated the motor will stop with different mode (see chapter dynamic Stop Parameters)

Referring to "FUNCTIONS" chapter to know the different "Braking Mode"

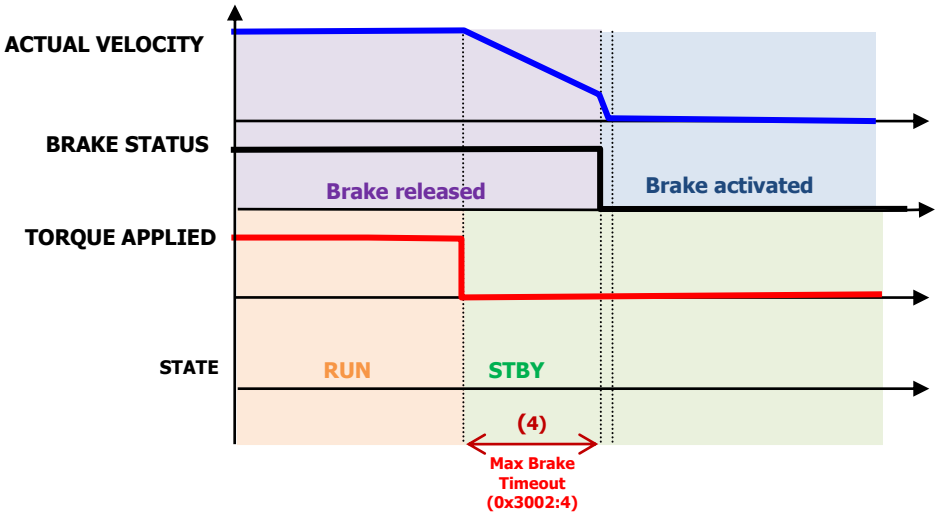
"Operation Enabled" (RUN) → "Switched-On" (STANDBY)
SPECIAL CASE

The follow graph describes the timing of brake when the drive moves from "Operation Enabled" (RUN state) to "Switched-On" (STANDBY state) and if the time of braking-off is greater than "Brake Timeout" (object 3002h: 4h).

This transition can be set with controlword ('enable operation' command) or emergency command (via controlword or with digital input)

The "Brake Timeout" is expired before the motor stops, the motor brake will be locked

WITH ELECTRONIC DYNAMIC STOP

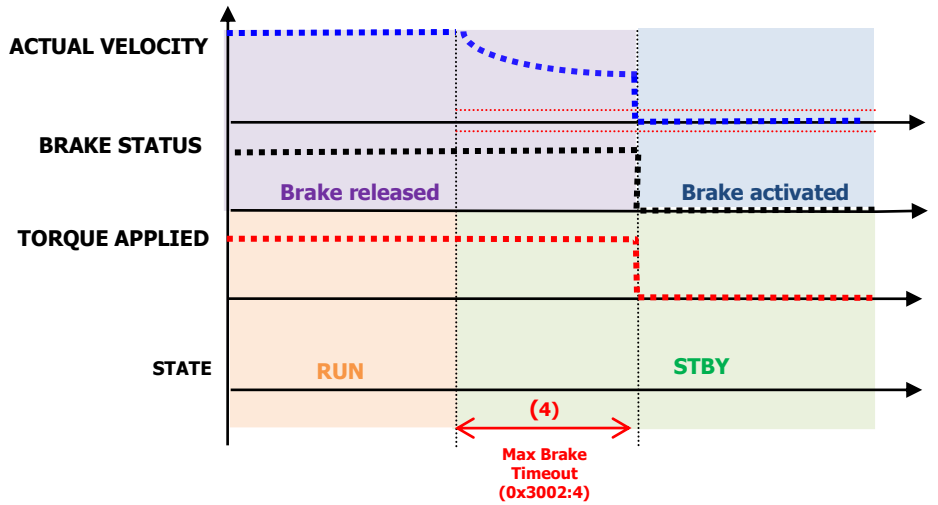


WITHOUT ELECTRONIC DYNAMIC STOP

If the dynamic stop is not activated the motor will stop with different mode (see chapter dynamic Stop Parameters)

Referring to "FUNCTIONS" chapter to know the "Braking Mode".

The default behaviour without electronic dynamic stop is:



"Operation Enabled" (RUN) → Fault Condition (FAULT)
The follows graphs describe the timing of brake when the drive has an fault error. The drive will move from all states to "Fault Reaction Active" (FAULT state). There are different mode to enter in FAULT state depending dynamic Stop is activated or not. Some alarms will stop immediately (turn off the PWM immediately) and it will go in fault state without dynamic brake. These alarms are: ○ Over Current ○ Over Voltage ○ Over Speed ○ Feedback Resolver Error Referring to "FUNCTIONS" chapter to know the different "Braking Mode".
WITH ELECTRONIC DYNAMIC STOP The drive waits the actual velocity will be 0 to activate the brake Referring to "FUNCTIONS" chapter to know the different "Braking Mode"
WITHOUT ELECTRONIC DYNAMIC STOP The default behaviour without electronic dynamic stop is then the brake is immediately activated. Referring to "FUNCTIONS" chapter to know the different "Braking Mode"

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06010002 = written is not permitted because the value is only READ (for object 0x3002:6 and 0x3002:7)
- 0x06090011 = sub-index does not exist

It is possible to change the Brake Parameters in run time.
This object can be changed, to use the new value it must be saved in the memory of e2prom

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3002_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "FUNCTIONS" chapter to know the management "Braking Mode"

Object 3007h: Electronic Dynamic Stop Parameters

This object describes the parameters of Electronic Dynamic Stop function. This function describes a specific ramp controlled when the master controller moves the drive from RUN state "Operation Enabled" to STANDBY "switched on".

The drive has two different mode to braking the motor set by object 0x3007:1 "Dynamic Stop Enabled" :

- (0) for inertia, the motor will be free to run until the stop.
- (1) with ramp controlled enabling Electronic Dynamic Stop (DEFAULT)

The master controller can enable and disable the Electronic "Dynamic Stop Enabled" (mode 0 and 1) setting 0x3007:1 and It can be saved in e^2prom.

NOTE FROM FW 236:

Since firmware 236 they are added different braking modes to stop the motor. To activate them it needs set the new option code of braking "Option Code Braking Mode" of object 0x3007:7 at '1' .

When the "Option Code Braking Mode" of object 0x3007:7 = '1' the value of "Dynamic Stop Enabled" of object 0x3007:1 is not considered and the braking modes is relative at the transition defined by objects 0x605C, 0x3008:4, 0x605B, 0x605E according to the event occurred and the goal state.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3007h	Electronic Dynamic Stop Parameters	ARRAY	INTEGER16	Mandatory IF

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	-	7
01h	Dynamic Stop Enabled	rw	no	[0,1]	1
02h	Holding Torque Time	rw	no	[1... 32767]	defined by application
03h	Dynamic Stop Status	ro	no	[0,1]	defined by application
04h	Decrement step ramp	rw	no	[1... 8191]	defined by application
05h	Max Timeout Dynamic Stop	rw	no	[1... 32767]	defined by application
06h	reserved	-	-		
07h	Option Code Braking Mode	rw	no	[0,1]	defined by application
08h	Decrement step ramp - SRAMP	rw	no	[0, 32767]	defined by application

Value Definition:

Sub-Index	Field	Configuration	Definition
1	Dynamic Stop Enabled	0b 1b	Dynamic Stop Mode Deactivated Dynamic Stop Mode Activated
2	Holding Torque Time	[ms * 10]	This time is the delay between STOP Status and unlock brake, at the end of deceleration ramp, before to stay in STBY status.
3	Dynamic Stop Status	0b 1b	Drive is not in Dynamic Stop Drive is in Dynamic Stop
4	Decrement step ramp	[rpm*100/sec]	This number is the step to decrement the Set Point during the transition from Run to Standby with Dynamic Stop activated
5	Max Timeout Dynamic Brake	[ms * 10]	Max Dynamic stop Timeout is the maximum time to exit from condition dynamic brake. It must be higher than "Decrement Step Ramp"
6	reserved	-	-

7	Option Code Braking Mode	0_b 1_b	Old Braking Mode New Braking Mode
8	Decrement step ramp - SRAMP	[rpm/sec]	This number is the step to decrement the Set Point during the transition from RUN (operation enabled) to STANDBY (Switch-on Disabled). The motor will stop with programmable ramp that it follows s-curve.

Since firmware greater 236 the new braking mode are set in the following objects:

- 0x605C:0 - "Disable Operation Option Code" is object for transition from "Operation Enable" (RUN state) to "Switched On" (STANDBY state) with command controlword
- 0x3008:4 - "Emergency Operation Option Code" is object for transition from "Operation Enable" (RUN state) to "Switched On" (STANDBY state) with digital input emergency
- 0x605B:0 – "Shutdown Option Code" is object for transition from "Operation Enable" (RUN state) to "Switch On Disabled" (STANDBY state) with command controlword
- 0x605E:0 – "Fault Reaction Option Code" is object for transition from all states to "Fault Reaction Active" (FAULT state) with fault occurred

The mode of transition defined on the objects 0x605C, 0x3008:4, 0x605B, 0x605E:

Option Code	Description
1	Electronic Dynamic Stop is ENABLED It slows down with ramp programmable defined by object 0x3007:4. When the velocity is on the range ±5 RPM it disables the drive. Stop with ramp programmed (RAMP TRAPEZOIDAL).
0	Electronic Dynamic Stop is DISABLED. Disable drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.
-1	Electronic Dynamic Stop is DISABLED. It stops immediately and set the Velocity to 0. When the velocity is on the range ±5 RPM it actives the electronic brake motor.
-2	Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive function until the time delay defined by object 0x3007:2 is expired.
-3	Electronic Dynamic Stop is DISABLED. It stops immediately and It locks immediately the brake motor, and set the Velocity to 0
-4	Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. and lock immediately the brake motor.
-5 (1)	Electronic Dynamic Stop is ENABLED: It slows down with down S-ramp defined by object 0x3007:8 Stop with ramp programmed (S-RAMP).

(1) it is enable from firmware 240

The master controller has these possibilities:

BEFORE FW 236:

The object 0x3007:7 "Option Code Braking Mode" must be 0 , it corresponds "Old Braking Mode".
The master controller could chose the different option code [0, 1].
It had write the object 0x3007:1 "Dynamic Stop Enabled" to use 0 or 1 configuration of braking mode.

- 1 = Dynamic Stop Mode Activated (DEFAULT) = The drive switches-off and the motor will stop with a deceleration with a specific ramp defined by "Decrement Step Ramp" (subindex 4)

- 0= Dynamic Stop Mode Deactivated = The drive switches-off and the motor will stop with a deceleration with a specific ramp that can be configured by the user

AFTER FW 236:

The object 0x3007:7 "Option Code Braking Mode" must be 1, it corresponds "New Braking Mode".
The master controller can chose the different option code [-4, -3, -2, -1, 0, 1].
It must write value [-4,-3,-2,-1,0,1] into object 0x605C, 0x3008:4, 0x605B, 0x605E to define the braking mode for the different transitions in the FSM.

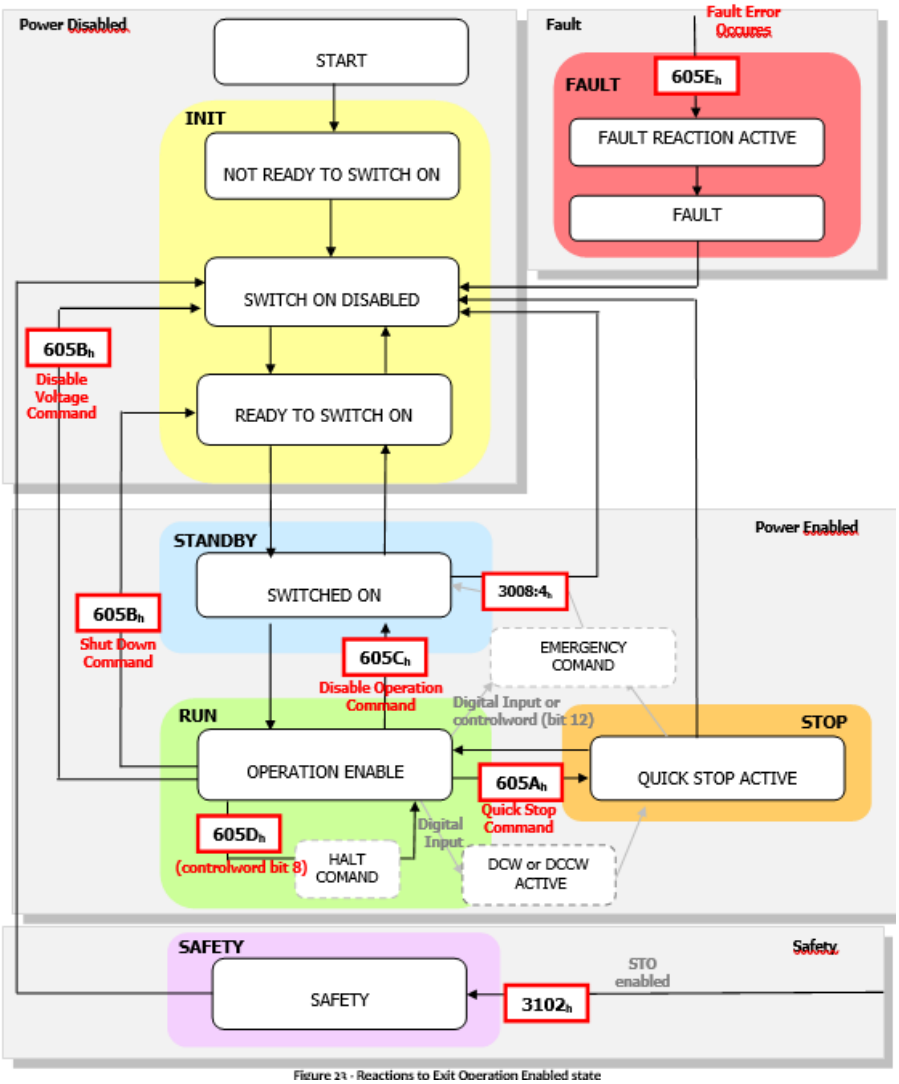


Figure 23 - Reactions to Exit Operation Enabled state

Calculation of Controlled Ramp (Electronic Dynamic Stop)

If Electronic Dynamic stop is activated (1) the drive switches-off and the motor will stop with a deceleration with a specific ramp defined by "Decrement Step Ramp" (subindex 4)

Example:

If Max Velocity defined by 0x3001:3 "Limit Velocity" is 4500 rpm and "Decrement Step Ramp" 0x3007:4 is 45.

- With Target Velocity 1000 rpm → the motor will stop 0,222 sec.
- With Target Velocity 4500 rpm → the motor will stop 1 sec.

For some transition modes, if the deceleration speed is greater than the delay time in "Brake Timeout" (3002_h:4_h), the drive will lock the brake.

information

For some transition modes, if the deceleration speed is greater than the delay time in "Brake Timeout" (3002_h:4_h), the drive will lock the brake.

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06010002 = written is not permitted because the value is only READ (for object 0x3002:6 and 0x3002:7)
- 0x06090011 = sub-index does not exist

It is possible to change the Brake Parameters in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3002_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

information

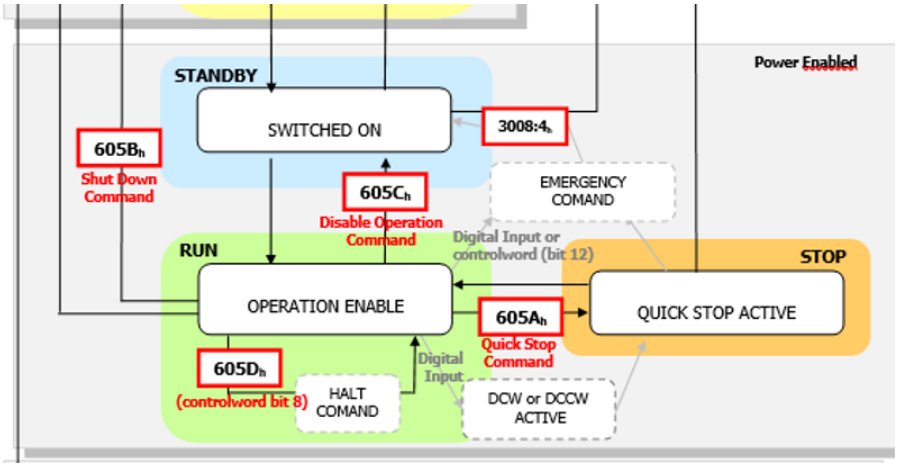
Referring to "FUNCTIONS" chapter to know the management "Stopping Mode"

Object 3008h: Emergency Input Parameters
This object describes the parameters of function "Emergency".

Master Control can stop the drive in emergency enabling the Digital Input via hardware or via SDO writing controlword (bit 12).

It means that it is possible to move the drive using a Emergency Command:

- from "Operation Enable" state [RUN] to "Switched On" state [STANDBY]
- from "Quick Stop Active" state [STOP] to "Switched On" state [STANDBY]



The Emergency Command can be

- Digital Input if it is connected and configured
- Bit 12 of controlword

The drive can't return to RUN state ("Operation Enabled") if the digital input continues to stay High Level or if the bit 12 of controlword is set.

To use this function it must be configured It must be enabled (object 0x3008:1).

Caution

EMERGENCY INPUT IS NOT SAFETY CERTIFICATE!!

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3008h	Emergency Enabling Input Parameters	ARRAY	INTEGER16	Mandatory IF

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	-	7
01h	Emergency Enable Option	rw	no	[0,1]	1
02h	Emergency Input Neg	rw	no	[0,1]	0
03h	Emergency Input Status	ro	no	[0,1]	0
04h	Emergency Option Code	rw	No	[-4 … 1]	0

Value Definition:

Sub-Index	Field	Configuration	Definition
1	Emergency Enable	0_b 1_b	Digital Input 3 is configured as General Purpose Digital Input 3 is configured as Emergency Enable
2	Emergency Input Neg	0_b 1_b	None inversion of input emergency level Inversion of input emergency level
3	Emergency Status	0_b 1_b	Status Low Level: Emergency not active Status High Level: Emergency active
4	Option Code (*)	[-5 … 1]	See Table

BEFORE FW 236:

The master controller can chose the mode of braking writing [0,1] into object 0x3007:1 "Dynamic Stop Enabled" option code [0,1].

AFTER FW 236:

The master controller can chose the different mode of braking writing [-4, -3, -2, -1, 0, 1] into object 0x3008:4 "Option Code".

Option Code description:

Option Code	Description
1	Electronic Dynamic Stop is ENABLED. It slows down with ramp programmable defined by object 0x3007:4. When the velocity is on the range ±5 RPM it disables the drive. Stop with ramp programmed (RAMP TRAPEZOIDAL).
0	Electronic Dynamic Stop is DISABLED. Disable drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.
-1	Electronic Dynamic Stop is DISABLED. It stops immediately and set the Velocity to 0. When the velocity is on the range ±5 RPM it actives the electronic brake motor.
-2	Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive function until the time delay defined by object 0x3007:2 is expired.
-3	Electronic Dynamic Stop is DISABLED. It stops immediately and It locks immediately the brake motor, and set the Velocity to 0
-4	Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. and lock immediately the brake motor.
-5 (1)	Electronic Dynamic Stop is ENABLED: It slows down with down S-ramp defined by object 0x3007:8 Stop with ramp programmed (S-RAMP).

(1) it is enable from firmware 240

Caution

It can be considered as emergency signal but it isn't safety certificated.
For disabling the power in safety certificated condition referred to STO chapter of Drive User Guide.

COMMAND EMERGENCY

Emergency Function must be enabled.

Example:

Enable Emergency Input:

Rx: Id 0x601 – 2b 08 30 01 01 00 00 00
Tx: Id 0x581 – 60 08 30 01 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x3008] [sub-index: 0x01] [Data: 0x00000001] – Write Enable
Tx: [resp: 0x60] [Index: 0x3008] [sub-index: 0x01] [Data: 0x00000000] – Success

1) **With digital input**:

To use Emergency Function, the master must configure in one digital Input the EMERGENCY configuration and via hardware it must set the digital input (see chapter digital input).

2) **With controlword bit 12**:

The master must write bit 12 of controlword

Example:

Master Controller writes bit 12 of controlword to enable emergency function

Rx: Id 0x601 – 2B 40 60 00 0F 10 00 00
Tx: Id 0x581 – 60 40 60 00 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x6040] [sub-index: 0x00] [Data: 0x0000100F] – Write BIT12
Tx: [resp: 0x60] [Index: 0x6040] [sub-index: 0x00] [Data: 0x00000000] – Success

It is possible to change the level. Write the object Index 0x3008 sub-index 2

Example:

Rx: Id 0x601 – 2b 08 30 02 01 00 00 00
Tx: Id 0x581 – 60 08 30 02 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x3008] [sub-index: 0x02] [Data: 0x00000001] – WRITE
Tx: [resp: 0x60] [Index: 0x3008] [sub-index: 0x02] [Data: 0x00000000] – Success

It is possible verify the Function Emergency actives.

Example:

Master Controller can read the status of the drive object 0x6041: xxxx xxx**1** x01x 0011b

Rx: Id 0x601 – 40 41 60 00 00 00 00 00
Tx: Id 0x581 – 4b 41 60 00 33 15 00 00

Rx: [cmd: 0x40] [Index: 0x6041] [sub-index: 0x00] [Data: 0x00000000] – Read SDO Status Word
Tx: [resp: 0x4B] [Index: 0x6041] [sub-index: 0x00] [Data: 0x00001533] – Success

Read the status of "Emergency Input" via SDO (index 0x3008 and sub-index 3)

Rx: Id 0x601 – 40 08 30 03 00 00 00 00
Tx: Id 0x581 – 4b 08 30 03 01 00 00 00

Rx: [cmd: 0x40] [Index: 0x3008] [sub-index: 0x03] [Data: 0x00000000] – Read
Tx: [resp: 0x4b] [Index: 0x3008] [sub-index: 0x03] [Data: 0x00000001] – Success

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06010002 = written is not permitted because the value is only READ (for object 0x3007:3)
- 0x06090011 = sub-index does not exist

This object can be changed, to use the new value it must be saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3008_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "FUNCTIONS" chapter to know the management "Emergency Input"

Object 3040_h: Analog Input

This object describes the analog input parameters. This object is ONLY read. If it needs to change the configuration contact the manufacturer.
In "Analog mode" the analog input is used to reference to control the motor.

The analog input can be configured:

- -10V … +10V
- 0 … 10V
- -5V … +5V
- 0 … 5V
- Threshold … 10V
- Threshold … 5V

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3040_h	Analog Input	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	3	3
01_h	Configuration	ro	no	[0 … 5]	0
02_h	Digital Value [12bit]	ro	no	[0 … 4095]	-
03_h	Digital Value [16bit]	ro	no	[-32768 … 32768]	-
04_h	Thrs for Cfg 4 and 5	ro	no	[-32768 … 32768]	-

Value Definition:

Sub-Index	Field	Value	Definition
01_h	Configuration	0 … 7	0) -10V … +10V 1) 0 … 10V 2) -5V … +5V 3) 0 … +5V 4) Threshold … 10V 5) Threshold … 5V 6) 0 … 10V 7) 0 … +5V
02_h	Digital Value [12 bit]	0 … 4095	Analog To Digital conversion: see the table below
03_h	Digital Value [16 bit]	-32768 … 32768	This value is the ADC Reference [12 bit] converted in 16 bit
04_h	Threshold for Configuration 4 and 5	-32768 … 32768	This value is the threshold to start the control in RUN for configuration 4 and 5

It is the ADC Reference raw:

Analog	Digital
-10V	247
-5V	1147
0V	2048
+5V	2948
+10V	3848

To read Analog Input using Voltage Reference object 2043_h: 0 (see object).

Object 3050_h: Analog Output 1

This object describes the analog output parameters. The analog output capability of the drive is [0÷10]V corresponds [0÷4095] bit.

The analog output configuration can be set by user:

- 0 = analog output disabled
- 1 = analog output configured as "general purpose". The analog output gets a value from 0 to 10V following [0 ÷ 4095] bit. The digital input value can be written via CANopen (sub-index 2).

Example: Analog Output configured General Purpose

1) **Write configuration in objet 3050_h: 1**

   Rx: Id 0x601 – 2B 50 30 01 01 00 00 00
   Tx: Id 0x581 – 60 50 30 01 00 00 00 00

   Rx: [cmd: 0x2B] [Index: 0x3050] [sub-index: 0x01] [Data: 0x00000001] – Configuration value '1'
   Tx: [resp: 0x60] [Index: 0x3050] [sub-index: 0x01] [Data: 0x00000000] – Success

2) **Write digital value in objet 3050_h: 2. E.g. 2048 bit (0x8000)**

   Rx: Id 0x601 – 2B 50 30 02 00 08 00 00
   Tx: Id 0x581 – 60 50 30 02 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x3050] [sub-index: 0x02] [Data: 0x00008000] – value 2048 = 0x8000
Tx: [resp: 0x60] [Index: 0x3050] [sub-index: 0x02] [Data: 0x00000000] – Success

3) Read digital value in object 3050h: 3

Rx: Id 0x601 – 40 50 30 03 00 00 00 00
Tx: Id 0x581 – 4B 50 30 03 00 08 00 00

Rx: [cmd: 0x2B] [Index: 0x3050] [sub-index: 0x03] [Data: 0x00000000] – read command
Tx: [resp: 0x4B] [Index: 0x3050] [sub-index: 0x03] [Data: 0x00008000] – Success value 0x8000

In the analog output the voltage is 5V (2048 bit).

- 2 = analog output configured as "Velocity monitoring". The analog output gets a proportional signal voltage of speed monitoring. The output signal is [0 – 10] V and it matches with the value range [-Velocity Full Scale ÷ +Velocity Full Scale] rpm (object 3300h).

Example: Analog Output configured Velocity Monitoring

Write configuration in object 3050h:1
Rx: Id 0x601 – 2B 50 30 01 02 00 00 00
Tx: Id 0x581 – 60 50 30 01 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x3050] [sub-index: 0x01] [Data: 0x00000002] – Configuration value '2'
Tx: [resp: 0x60] [Index: 0x3050] [sub-index: 0x01] [Data: 0x00000000] – Success

If the drive has Velocity Full Scale (object 3300h:0) = 3000 RPM then:
- 0V = - 3000 rpm
- 5V = 0 rpm
- 10V = 3000 rpm

- 3 = analog output configured as "Current monitoring". The analog output gets a proportional signal voltage of absolute current absorbed. The output signal is [0 – 10] V and it matches with the value range [0 ÷ Peak Current] (object 3303h: 2).

Example: Analog Output configured current Monitoring

Write configuration in objet 3050h:1
Rx: Id 0x601 – 2B 50 30 01 03 00 00 00
Tx: Id 0x581 – 60 50 30 01 00 00 00 00

Rx: [cmd: 0x2B] [Index: 0x3050] [sub-index: 0x01] [Data: 0x00000003] – Configuration value '3'
Tx: [resp: 0x60] [Index: 0x3050] [sub-index: 0x01] [Data: 0x00000000] – Success

If the drive is configured as SMALL size and it has Current Peak (object 3303h:0) = 41 Arms then:
- 0V = 0 Arms
- 3V = 12,5 Arms (rated current in SMALL size)
- 10V = 41 Arms

Object 3100h: Configuration 1 – Statusword

This object changes the configuration of Manufacturer bits of statusword (bit 8, 14 and 15).

The user can change this object during initialization or saved in eeprom to have a different combination of manufacturer bits of statusword. Default Value is 0 and it is a STANDARD bits configuration.

There are 2 different configurations available.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3100_h	Configuration 1 – Statusword	VAR	INTEGER16	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Configuration 1 – Statusword	rw	no	[0 … 32767]	0

Configuration:

Value	Bits of statusword		
	Bit 8	Bit 14	Bit 15
0	Emergency In. Enable	Safety State	Fault State
1	Emergency In. Enable	Safety State	Ready
2	Emergency In. Enable	S-Ramp Error Calc Jerk	S-Ramp Error Calc Limit
3	Emergency In. Enable	Safety State	Abort Communication

Meaning of bits:

Name	Meaning	Description
Emergency Input Enable	It shows if the input Emergency is active	0 = not active 1= active
Safety State	It shows if the drive is in SAFETY state	0 = drive not in SAFETY state 1= drive in SAFETY state
Fault State	It shows if the drive is in FAULT state	0 = drive not in FAULT state 1= drive in FAULT state
Ready State	It shows if the drive is ready to go in RUN state (operational enabled)	0 = drive not ready 1= drive is ready: Voltage Enabled, STO digital input not active, no fault, emergency input enable not active
S-Ramp Error Calc Jerk	It shows if the calculation of S-Ramp with new target is correct	0 = S-Ramp calculation with new target is within the range of maximum allowed Jerk 1 = S-Ramp calculation with new target is outside the range of maximum allowed Jerk
S-Ramp Error Calc Limit	It shows if the calculation of S-Ramp with new target is correct	0 = S-Ramp calculation with the new target is within the range of maximum allowed acceleration 1 = S-Ramp calculation with the new target is outside the range of maximum allowed acceleration
Abort Communication	It shows if there has been an abort communication	0 = Nothing abort communication is occurred 1 = At least abort communication is occurred

The drive will send the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)

It is possible to change the configuration also in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3100_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3101_h: Configuration 2 (RESERVED)

Contact Manufacturer for additional information

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3100_h	Configuration 2 – Reserved	VAR	INTEGER16	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Configuration 2 – Reserved	rw	no	[0 … 32767]	1

Object 3102_h: Configuration 3 – STO option Code

The object changes the configuration of STO Management.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3102_h	Configuration 3 – STO option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Configuration 3 – STO Management	rw	no	[0 … 32767]	0

Configuration:

Value	Name	Description
0	**STANDARD**	See chapter "SAFETY"
1	**BRAKE IMMEDIATELY**	See chapter "SAFETY"

information

Referring to "FUNCTIONS" chapter to know "SAFETY"

The drive will send the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)

It is possible to change the configuration also in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3102_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3103_h: Configuration 4 – Safety FSM

The object defines the different transition following STO event into Finite State Machine.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3103_h	Config 4 - Safety FSM	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Config 4 - Safety FSM	rw	no	[0 … 32767]	0

Configuration:

Value	Name	Description
0	**STANDARD LAFERT**	See chapter "SAFETY"
1	**SAFETY DIRECT**	See chapter "SAFETY"
2	**SAFETY FAULT**	See chapter "SAFETY" (*)

(*) it is not available, please contact manufacturer

The drive will send the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)

It is possible to change the configuration also in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3103_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3104_h: Configuration 5 – Warning Cfg

The object defines the different configuration of warning.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3104h	Config 5 – Warning Cfg	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Config 5 – Warning Cfg	rw	no	[0 … 32767]	0

Configuration:

Value	Name	Description
0	**STANDARD LAFERT**	See chapter "DIAGNOSTIC" part Warning
1	**AUTOMATICALLY VALIDATION**	See chapter "DIAGNOSTIC" part Warning

The drive will send the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)

It is possible to change the configuration also in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3104h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

Object 3200h: Current PID (RESERVED)

The object controls equivalent of PID current parameters.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3200h	Current Pid	ARRAY	INTEGER16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	6	6
01h	PidCur Kp (reserved)	rw	no	[0 … 32767]	(reserved)
02h	PidCur K (reserved)	rw	no	[0 … 32767]	(reserved)
03h	PidCur Kv (reserved)	rw	no	[0 … 32767]	(reserved)
04h	PidCur Kd (reserved)	ro	no	[0 … 32767]	(reserved)
05h	PidCur N (reserved)	ro	no	[0 … 32767]	(reserved)
06h	PidCur FF (reserved)	ro	no	[0 … 32767]	(reserved)

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06090011 = sub-index does not exist

Object 3201h: Speed PID

The object controls equivalent of PID speed parameters.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3201h	Speed PID	ARRAY	INTEGER16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	6	6
01h	PidVel Kp	rw	no	[0 ... 32767]	defined by application
02h	PidVel Ki	rw	no	[0 ... 32767]	defined by application
03h	PidVel Kv	rw	no	[0 ... 32767]	defined by application
04h	PidVel Kd (reserved)	ro	no	[0 ... 32767]	(reserved)
05h	PidVel N (reserved)	ro	no	[0 ... 32767]	(reserved)
06h	PidVel FF (reserved)	ro	no	[0 ... 32767]	(reserved)

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06090011 = sub-index does not exist

It is possible to change the Speed Pid in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3201h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

Object 3202h: Position PID (RESERVED)

The object controls equivalent of PID position parameters.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3202h	Position Pid	ARRAY	INTEGER16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	ro	no	9	9
01h	PidPos Kp (reserved)	rw	no	[0 ... 32767]	(reserved)
02h	PidPos Ki (reserved)	rw	no	[0 ... 32767]	(reserved)
03h	PidPos Kv (reserved)	rw	no	[0 ... 32767]	(reserved)
04h	PidPos FF Ra V (reserved)	ro	no	[0 ... 32767]	(reserved)
05h	PidPos FF Ra A (reserved)	ro	no	[0 ... 32767]	(reserved)
06h	PidPos FF Vr V (reserved)	ro	no	[0 ... 32767]	(reserved)

07h	PidPos FF Rd A (reserved)	ro	no	[0 … 32767]	(reserved)
08h	PidPos FF Rd V (reserved)	ro	no	[0 … 32767]	(reserved)
09h	PidPos Tc (reserved)	ro	no	[0 … 32767]	(reserved)

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06090011 = sub-index does not exist

Object 3300h: Velocity Full Scale

That is value is the Full Scale of Velocity.
In analog mode it is the maximum value of reference voltage for Speed Set Point.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3300h	Velocity Full Scale	VARIABLE	UNSIGNED16	Mandatory

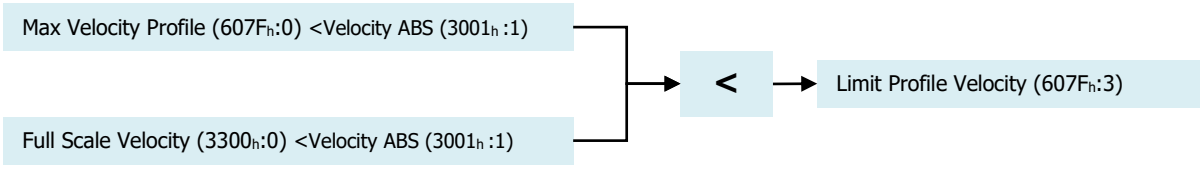
Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number of Entries	rw	no	[0-32767]	Defined by Application

In Analog Mode this object defines the range to set the target of velocity. The analog set point is the input 0 to 10V where the range is defined by the +/- "Velocity Full scale" (3300h).

You should program the "Velocity Full Scale" to be smaller or equal than "Velocity Absolute Maximum Rating" (3001h: 1).

In Profile Velocity Mode via CAN this object, together the "Max Profile Velocity" (607Fh), defines the limit of Speed. The scheme to set the limit is the follow:



The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06090031 = Value of parameter written too high (it must be smaller than 0x3001:1)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3300h

- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3401_h: Warning Emergency Protocol

This object clear all error code of warning from emergency protocol.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3401h	Warning Emergency Protocol	VAR	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value
00h	Warning Emergency Protocol	rw	NO	0

Configuration:

Value	Name	Description
0	STANDARD LAFERT	The error code of warning is displayed in emergency protocol (CAN BUS the error code can be read with COBID 0x80+IdNode
1	CLEAR WARNING CODE IN EMERGENCY PROTOCOL	All error code of warning are cleaned in emergency protocol

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3401_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3900_h: Warning Settings

This object configure the warning.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3900_h	Warning Settings	ARRAY	INT32	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Warning Enable User	rw	no	1081343
02_h	Warning Select Output	rw	no	1081343

03h	free	rw	no	-
04h	free	rw	no	-

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3900h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

MANUFACTURER OBJECTS – RUNTIME MONITORING DATA

Object 2002h: Drive Control State

This object communicates the drive's state. This object defines exactly the PWM control motor.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2002h	Drive Control State	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range
00h	Drive Control State	ro	YES	See table

Valid entries:

VALUE	BIT	Name	Description
0x0001	0	Run Velocity	The motor runs in velocity control mode.
0x0002	1	Standby	The drive is in stand-by. The PWM is OFF.
0x0004	2	Stop	The drive is in stop. It is stationary with torque applied.
0x0008	3	Off	*Not used*
0x0010	4	Alarm	The drive has detected an alarm.
0x0020	5	Run Current	The motor runs in Current control mode.
0x0040	6	Init	The drive is in initialization state. The PWM is OFF.
0x0080	7	Safe	The drive is in safe with STO applied. The PWM is OFF.
0x0100	8	reserved	reserved

Object 2004h: State Lafert Servo Drive Machine

This object describes exactly the drive state. The drive follows a finite state machine proprietary Lafert Serve Drive that it is compliant with profile DSP402.

State Value Definition:

- 0 = Lafert Servo Drive state INIT – p402 state not ready to switch on
- 1 = Lafert Servo Drive state SAFETY
- 2 = Lafert Servo Drive state STOP – p402 state quick stop active
- 3 = Lafert Servo Drive state RUN – p402 state operation enabled
- 4 = Lafert Servo Drive state STANDBY – p402 state switched on
- 5 = Lafert Servo Drive state DYNAMIC TOP
- 6 = Lafert Servo Drive state Reserved
- 7 = Lafert Servo Drive state Reserved
- 10= Lafert Servo Drive state INIT – p402 state switch on disabled
- 11= Lafert Servo Drive state INIT – p402 state ready to switch on
- 16 = Lafert Servo Drive state FAULT – p402 state fault reaction active
- 17 = Lafert Servo Drive state FAULT – p402 state fault
- 18 = Lafert Servo Drive state FAULT – p402 state error

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2004h	Drive Status LSD	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range
00_h	Drive State Lafert Servo Drive	ro	YES	See table

Valid entries:

Value	Lafert Servo Drive State	State DSP402
0	**INIT**	**not ready to switch on**
1	**SAFETY**	**-**
2	**STOP**	**quick stop active**
3	**RUN**	**operation enabled**
4	**STANDBY**	**switched on**
5	**DYNAMIC STOP**	**-**
6	Reserved	-
7	Reserved	-
8	-	-
9	-	-
10	**INIT**	**state switch on disabled**
11	**INIT**	**ready to switch on**
12	-	-
13	-	-
14	-	-
15	-	-
16	**FAULT**	**fault reaction active**
17	**FAULT**	**fault**
18	**FAULT**	**error**

Object 2030_h: Temperature Drive

This object communicates the temperature internal of drive (Logic Board).

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2030_h	Temperature Drive	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Drive temperature	ro	YES	[-150 … 1250]	[°C /10]

Object 2031h: Temperature Motor

This object communicates the motor temperature.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2031h	Temperature Motor	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Motor temperature	ro	YES	[-400 … 1300]	[°C /10]

Object 2032h: Temperature Heat Sink

This object communicates the temperature internal of drive Heat Sink.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2032h	Temperature Heat Sink	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Heat Sink temperature	ro	YES	[-400 … 1300]	[°C /10]

Object 2041h: Voltage Bus

This object communicates the value of voltage Bus.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2041h	Voltage Bus	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Voltage Bus	ro	YES	[0 … 11000]	[V/100]

Object 2043h: Voltage Reference

This object communicates the value of voltage reference in mV of Analog Input signal.
It is used in Analog Mode to read the Voltage Analog Reference (speed set point).
In CAN protocol can be used to read analog input value.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2043h	Voltage Reference	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Voltage Reference	ro	YES	[-32767… 32767]	[mV]

Object 2050h: Torque Current

This object communicates the value of Torque Current.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2050h	Torque Current	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Torque Current	ro	YES	[-32767… 32767]	[Arms/100]

Object 2053h: Velocity Filtered

This object communicates the value of Velocity filtered.
The velocity filter can be modified using object Filter Parameters (3005h:5) it is a one pole for velocity.
The unit is [rpm/4].

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2053h	Velocity Filtered	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Velocity Filtered	ro	YES	[-32767… 32767]	[rpm/4]

Object 2054h: Energy i²t

This object describes the percent of energy of i²t.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2054h	Energy i²t	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Energy i²t	ro	YES	[0… 65536]	[%]

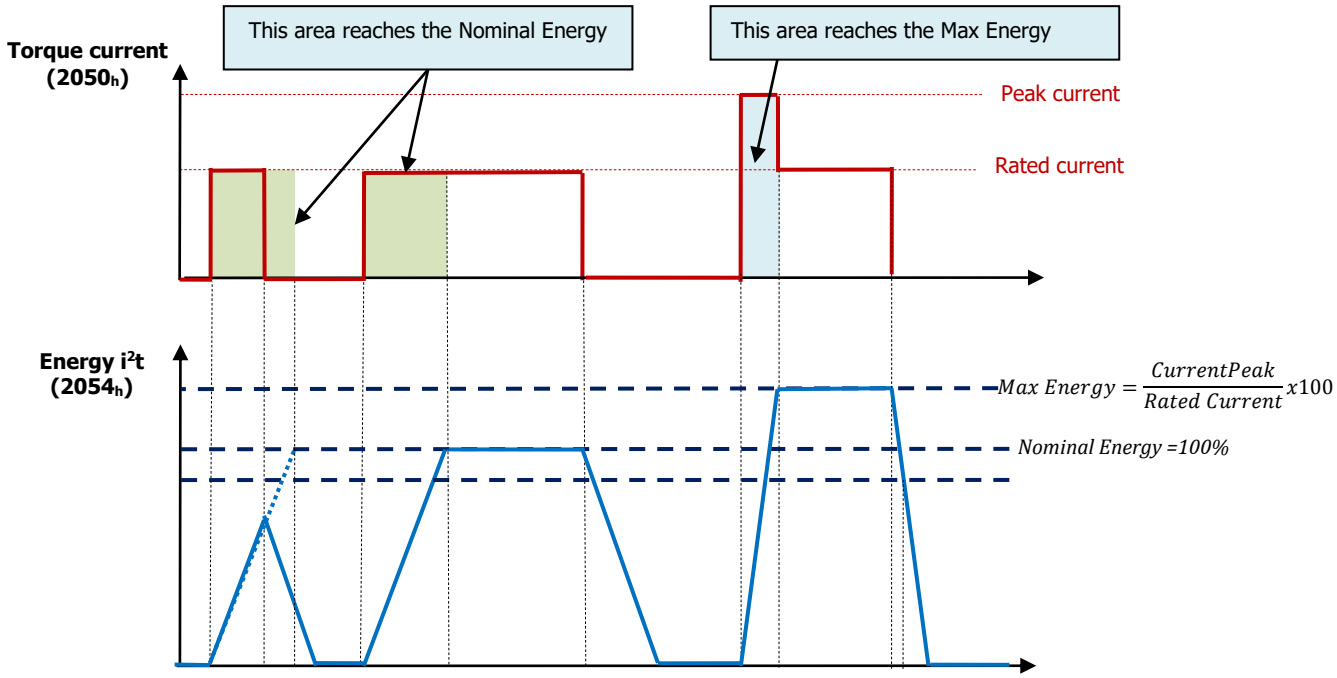


Figure 28 - Diagram i2t monitoring

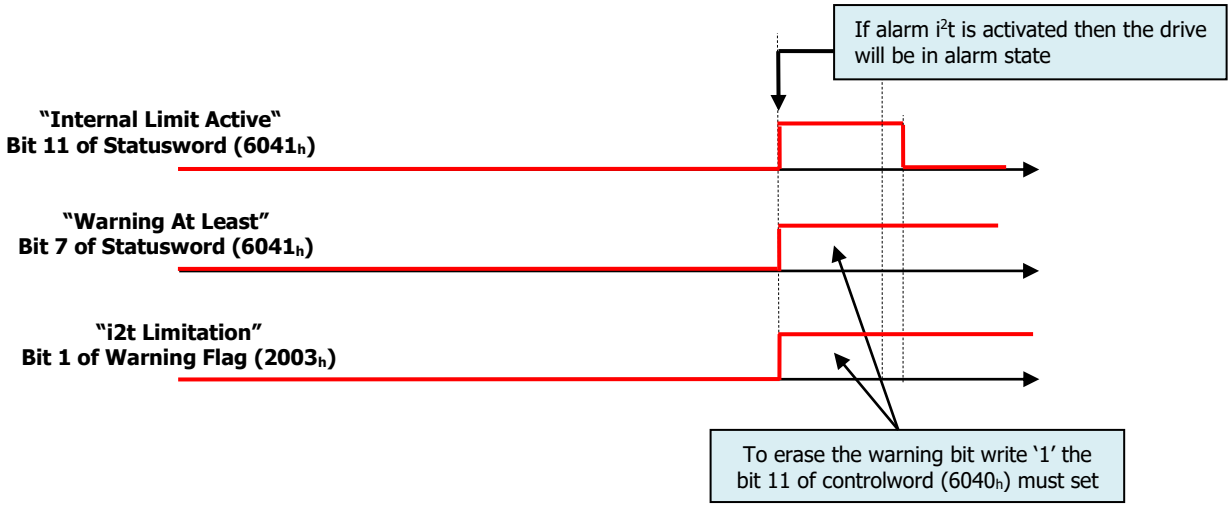


Figure 29 - Diagram i2t monitoring (2)

Object 2060h: Impulse

This object is the electrical angle (it depends on number of motor pole pairs) with increment units, the max value is the feedback's resolution.

The resolution can be read with the Feedback object (3004h: 2).

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2060h	Impulse	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit

00h	Impulse	ro	YES	[-32767… 32767]	[0 –Max Resolution]

Object 2070h: Torque Current Windowed

This object communicates the value of Torque Current with a floating window.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2070h	Torque Current Windowed	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	Impulse	ro	YES	[-32767… 32767]	[Arms/100]

Object 2080h: STO Statusword

This object describes STO statusword

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2080h	STO Statusword	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Unit
00h	STO Statusword	ro	no	[-32767… 32767]	[-]

Configuration – SMARTRIS STANDARD:

Value	Description
0	Drive not in safety
1	Drive in safety – STO activated
2	Drive in safety – STO activated
3	Drive in safety – STO activated

Configuration – SMARTRIS COMPACT:

Value	Description
0	Drive not in safety
1	Drive in safety – STO activated
2	Drive in safety – STO activated
3	Drive in safety – STO activated
10	STO Circuit FAULT

Object 3003h: Drive Size Parameters

This object defines the Current Parameters of drive, they are decrypted by datasheet.
This is only READ.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3003h	Impulse	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	7	2

01h	Current Sense Full Scale	ro	no	[0 ... 32767]	defined by application
02h	Peak Current	ro	no	[0 ... 32767]	defined by application
03h	Rated Current	ro	no	[0 ... 32767]	defined by application
04h	I^2t	ro	no	[0 ... 32767]	defined by application
05h	reserved	ro	no	[0 ... 32767]	defined by application
06h	reserved	ro	no	[0 ... 32767]	defined by application
07h	reserved	ro	no	[0 ... 32767]	defined by application

Value Definition:

Sub-Index	Field	Configuration	Definition
01h	Current Sense Full Scale	[Arms]	Maximum scale of Current sense
02h	Peak Current	[Arms/100]	Peak current of drive's size.
03h	Rated Current	[Arms/100]	Rated current of drive's size.
04h	I^2t timeout	[sec]	Timeout for occur alarm over load.
05h	reserved	-	-
06h	reserved	-	-
07h	reserved	-	-

Object 3004h: Feedback Parameters

This object defines the specifics characteristics of Feedback. It is only READ.
The feedbacks used are:

- Resolver
- Incremental Encoder
- Sin/Cos Encoder (single and multiturn)

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3004h	FeedBack Parameters	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	15	15
01h	Feedback Type	ro	no	[0 ... 32767]	defined by applic
02h	Resolution	ro	no	[0 ... 32767]	defined by applic
03h	Absolute position L (initial) - 16 bit LSB (*)	ro	no	[0 ... 32767]	defined by applic
04h	Absolute position H (initial) – 16 bit MSB (*)	ro	no	[0 ... 32767]	defined by applic
05h	Absolute position L (actual) - 16 bit LSB (*)	ro	no	[0 ... 32767]	defined by applic
06h	Absolute position H (actual) – 16 bit MSB (*)	ro	no	[0 ... 32767]	defined by applic
07h	Encoder Type (*)	ro	no	[0 ... 32767]	defined by applic
08h	Reserved	ro	no	[0 ... 32767]	-
09h	Reserved	ro	no	[0 ... 32767]	-
0Ah	Encoder Command (*)	rw	no	[0 ... 32767]	defined by applic
0Bh	Initial gear number (*)	ro	no	[0 ... 32767]	defined by applic

$0C_h$	Relative position (initial) (*)	ro	no	[0 … 32767]	defined by applic
$0D_h$	Reserved	ro	no	[0 … 32767]	-
$0E_h$	Reserved	ro	no	[0 … 32767]	-
$0F_h$	Reserved	ro	no	[0 … 32767]	-

(*) only for SinCos Encoder

Value Definition:

Sub-Index	Field	Configuration	Definition
01_h	Feedback Type	[0 … 2]	0 = Resolver 1 = Incremental Encoder 2= Sin/Cos Encoder
02_h	Feedback Resolution	[0 … 32767]	This object shall indicate the configured rate of change of torque. The value shall be given in units of per thousand of rated torque per second.
03_h	Absolute position L (initial) - 16 bit LSB (*)	[0 … 32767]	Example: pos = 0x297421 value = 0x7421
04_h	Absolute position H (initial) – 16 bit MSB (*)	[0 … 32767]	Example: pos = 0x297421 value = 0x0029
05_h	Absolute position L (actual) - 16 bit LSB (*)	[0 … 32767]	Absolute position is 24 bit for multi turn / 12 bit for single turn (1)
06_h	Absolute position H (actual) – 16 bit MSB (*)	[0 … 32767]	Absolute position is 24 bit for multi turn / 12 bit for single turn (1)
07_h	Encoder Type (*)	0x32, 0x37	Singleturn = 0x32, Multiturn = 0x37
08_h	Reserved	-	-
09_h	Reserved	-	-
$0A_h$	Encoder Command (3) (*)	[0 … 4]	0.none 1.READ ENCODER STATUS (encoder status) 2.READ OUT NAME PLATE (Encoder Type) 3.Reserved 4.READ POSITION (actual position)
$0B_h$	Initial gear number (*)	[0 … 32767]	(12 bit) (4)
$0C_h$	Relative position (initial) (*)	[0 … 32767]	(12 bit) (4)
$0D_h$	Reserved	-	-
$0E_h$	Reserved	-	-
$0F_h$	Reserved	-	-

Note: (*) Only Encoder Sin/Cos (single and multiturn)

1) At the beginning initial and actual position are the same. Actual position is updated only after a "read position" command number 4 of "Encoder Command" (subindx $0A_h$) see (3).
2) In the 8 bit LSB there is internal error code
3) The drive sents an encoder command, after that the master can be read a new value
4) Example: pos = 0x297421 N° of gear = 0x0421

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

Object 3005_h: Filter Parameters

This object defines the Filter Parameters of drive. This is only READ.

It can be modified only sub-index 05_h and 06_h.

- Sub-index 05_h: filter pole for "Velocity Filtered" object 2053_h
- Sub-index 06_h: filter pole for "Actual Velocity" object $606C_h$

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3005_h	Filter Parameters	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	10	10
01_h	Filter Pole 1 Set Point	ro	no	[0 …10000]	defined by application
02_h	Free	ro	no	[0 …10000]	defined by application
03_h	Filter Pole 1 Velocity	ro	no	[0 …10000]	defined by application
04_h	Filter Pole 2 Velocity	ro	no	[0 …10000]	defined by application
05_h	Filter Pole 1 Velocity Filtered (2053_h)	rw	no	[0 …10000]	50
06_h	Filter Pole 1 Actual Velocity ($606C_h$)	rw	no	[0 …10000]	50
07_h	Filter Pole 1 Pid	ro	no	[0 …10000]	defined by application
08_h	Filter Pole 2 Pid	ro	no	[0 …10000]	defined by application
09_h	Free	ro	no	[0 …10000]	defined by application
10_h	Free	ro	no	[0 …10000]	defined by application

Value Definition:

Sub-Index	Field	Configuration	Definition
01_h	Filter Pole 1 Set Point	[0 … 10000] Hz	Reserved – This value cannot be modified
02_h	Free	[0 … 10000] Hz	-
03_h	Filter Pole 1 Velocity	[0 … 10000] Hz	Reserved – This value cannot be modified
04_h	Filter Pole 2 Velocity	[0 … 10000] Hz	Reserved – This value cannot be modified
05_h	Filter Pole 1 Velocity Filtered (2053_h)	[0 … 10000] Hz	This value can be modified to change the output velocity filtered of 2053_h object. With 0 the filter is not applied. The units are [Hz] (default value is 50Hz).
06_h	Filter Pole 1 Actual Velocity ($606C_h$)	[0 … 10000] Hz	This value can be modified to change the signal "Actual Velocity" $606C_h$ object. With 0 the filter is not applied. The units are [Hz] (default value is 50Hz).
07_h	Filter Pole 1 Pid	[0 … 10000] Hz	Reserved – This value cannot be modified
08_h	Filter Pole 2 Pid	[0 … 10000] Hz	Reserved – This value cannot be modified
09_h	Free	[0 … 10000] Hz	-
10_h	Free	[0 … 10000] Hz	-

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

The sub-index number 05_h and 06_h can be modified and then This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3005_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3006_h: Motor Specific Settings

This object defines the specifics characteristics of motor. It is only READ.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3006_h	Motor Specific Settings	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	3	3
01_h	Motor Part Number	ro	no	[0 – 32767]	defined by application
02_h	Max Motor Speed	ro	no	[0 – 32767]	defined by application
03_h	N Pole	ro	no	[0 – 32767]	defined by application
04_h	Motor Kt	ro	no	[0 – 32767]	***Not available yet***
05_h	Motor Sense Type	ro	no	[0 – 32767]	defined by application

Motor Sense Type can be the following values:

1. KTY83
2. PT1000
3. KTY84
4. PTC ON/OFF

Object 3020h: Digital Input Function

This object describes what is the configuration of digital Inputs.
This object is only read, to change the function of digital input write specific object of digital input: 0x3021 (dig.In.1), 0x3022 (dig.In.2), 0x3023 (dig.In.3), 0x3024 (dig.In.4).

The digital input configuration available are:

- FUNCTION 1 – "RUN": in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication) this input is the command to move the drive in RUN state.
- FUNCTION 2 – "STOP": in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication) this input is the command to move the drive in STOP state.
- FUNCTION 3 – "EMERGENCY INPUT". When the option of digital is defined "Emergency Input" this input is the command to move the drive in STANDBY state.
- FUNCTION 4 – "RESET": When this function is enabled the input configured can put the drive in reset (it is an hardware reset). If the digital input is configured as reset, the reset has a filter with 100ms.
- FUNCTION 5 – "DCW": This function is to configure the input with an actuator for clockwise. If the input state is 1 the drive goes in STOP state.
- FUNCTION 6 – "DCCW": This function is to configure the input with an actuator for counter clockwise. If the input state is 1 the drive goes in STOP state
- FUNCTION 7 – "SETVEL1": it is used in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication). This function is to configure the Set Point value defined by "File Parameters". If the digital inputs were set with only this function, then the Set Points available are 2.
- FUNCTION 8 – "SETVEL2": it is used in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication). This function is to configure the Set Point value defined by "File Parameters". If the digital inputs were set with the functions "SETVEL1" and "SETVEL2", then the Set Points available are 4.
- FUNCTION 9 – "SETVEL3": it is used in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication). This function is to configure the Set Point value defined by "File Parameters". If the digital inputs were set with the functions "SETVEL1", "SETVEL2" and "SETVEL3", then the Set Points available are 8.
- FUNCTION 10 – "SETVEL4": it is used in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication). This function is to configure the Set Point value defined by "File Parameters". If the digital inputs were set with the functions "SETVEL1", "SETVEL2", "SETVEL3" and "SETVEL4", then the Set Points available are 15. If this function is set then all digital inputs are used. When all digital inputs are 0 the drive is in STANDBY, to have the RUN function it needs to have at least one digital input as 1.
- FUNCTION 11 – "VEL/CUR": it is used in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication). If the digital Input is selected with this function then the drive change the mode operation (Velocity Profile and Torque Profile), the function can be used only in standby state.
- FUNCTION 12 – "DIR": it is used in Analog Mode (where Mode of Operation set as "Manufacturer Mode" without CANopen communication). This function changes the direction of the motor. The set point changes the polarity.
- FUNCTION 13 – "ZERO POS": when the digital input is configured as "zero pos", if it is set the actual position will be 0, It does not move the motor in ZERO position, but it reset the actual position.

Every bit of subindex 1 "Configuration Enable" defined which function is enabled.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3020h	Digital Input Function	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	6	6

01_h	Configuration Enable	ro	no	[0, 32767]	defined by application
02_h	State Function	ro	no	[0, 32767]	defined by application
03_h	Level Function	ro	no	[0, 32767]	defined by application
04_h	Digital Raw	ro	no	[0, 32767]	defined by application
05_h	Configuration Set Point	ro	no	[0, 32767]	defined by application
06_h	Direction	ro	no	[0, 32767]	defined by application

Every bit of value is the function that the digital input can be used.

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FUNCTION	F16	F15	F14	F13	F12	F11	F10	F9	F8	F7	F6	F5	F4	F3	F2	F1
	-	-	-	ZERO POS	DIR	VEL / CUR	SET VEL4	SET VEL3	SET VEL2	SET VEL1	DCCW	DCW	RST	EMGY	STOP	RUN

Value Definition:

Sub-Index	Field	Configuration	Definition
1	Function Configuration Enable	Every bit defines the function and it can be '0' or '1'	Every bit is the configuration function: 0 = the function is not enabled 1 = the function is enabled
2	Function State	Every bit defines the function and it can be '0' or '1'	Every bit is the state of the function: 0 = the function actives 1 = the function doesn't active
3	Function Level	Every bit defines the function and it can be '0' or '1'	Every bit is the level of the input function: 0 = positive edge (standard configuration) 1 = negative configuration
4	Digital Raw	Dig In 1 – Bit 0 = '0' or '1' Dig In 2 – Bit 1 = '0' or '1' Dig In 3 – Bit 2 = '0' or '1' Dig In 4 – Bit 3 = '0' or '1'	Every Bit indicates the Raw Level of digital Input 0 = positive edge 1 = negative edge
5	Configuration Set Point	[0 … 4]	0 = Set Point defined by Reference (Analog) o Target Velocity (CAN) 1 = SetVel1→ 2 Set Points 1 = SetVel1 + SetVel2→ 4 Set Points 3 = SetVel1 + SetVel2+ SetVel3 → 8 Set Points 4 = SetVel1 + SetVel2+ SetVel3 + SetVel4 → 15 Set Points
6	Direction	[-1, 0, 1]	The motor runs -1 = counter clockwise 0 = stopped 1 = clockwise

Example:

1. Digital Input **DEFAULT** configuration:
   - Digital Input 1 = RUN (command Analog Mode) → FUNCTION 1
   - Digital Input 2 = STOP (command Analog Mode) → FUNCTION 2
   - Digital Input 3 = EMERGENCY INPUT ENABLE → FUNCTION 3
   - Digital Input 4 = RESET → FUNCTION 4

   The value of "Function Configuration Enable" (0x3020:1) will be 0x000F (0000 0000 0000 1111_b).

   If the value "Function State" (0x3020:2) is 0x0005 (0000 0000 0000 0101_b) means that the Input 1 (RUN, function 1) and input 3 (EMERGENCY INPUT ENABLE, function 3) are activated.

   Then the "Digital Raw" (0x3020:4) will be 0x0005 (0000 0000 0000 0101_b).

   If the "Function Level" (0x3020:3) is 0x0000 then all edge are positive level.

2. Digital Input **OTHER** configuration (example)
   - Digital Input 1 = DCW → FUNCTION 5
   - Digital Input 2 = DCCW → FUNCTION 6
   - Digital Input 3 = EMERGENCY INPUT ENABLE → FUNCTION 3
   - Digital Input 4 = RESET → FUNCTION 4

The value of "Function Configuration Enable" (0x3020:1) will be 0x003C (0000 0000 0011 1100 $_b$).

If the value "Function State" (0x3020:2) is 0x0024 (0000 0000 0010 0100$_b$) means that the Input 2 (DCCW, function 6) and input 3 (EMERGENCY INPUT ENABLE, function 3) are activated.

Then the "Digital Raw" (0x3020:4) will be 0x0005 (0000 0000 0000 0101$_b$).

If the "Function Level" is 0x0030 means that the Function DCW and DCCW have negative level.

information

To change Digital Input **DEFAULT** configuration contact manufacturer

Caution

- RUN, STOP, SETVEL1, SETVEL2, SETVEL3, SETVEL4, VEL/CUR, DIR are digital Input function ONLY for Analog Mode.
- EMERGENCY INPUT is the function to move the drive in "Switched On" state of DSP402 (or "STANDBY" state of the manufacturer State Machine of LSD) from the "Operation Enabled" state of DSP402 (or "RUN" state of the Macro State Machine of LSD). This command is used for emergency stop. See object 3008$_h$ "Emergency Enable Parameters"
- In Analog Mode it is mandatory to have one digital input configured in "Run" function. Only the configuration SETVEL1, SETVEL2, SETVEL3 and SETVEL4 the RUN function can't be set.

Object 3021$_h$: Digital Input 1

This object describes the digital Input 1 configuration.
It is possible to change the configuration ad the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3021$_h$	Digital Input 1	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00$_h$	Number Of Entries	ro	no	5	5
01$_h$	Configuration	rw	no	[0..13]	defined by application
02$_h$	State	ro	no	[0,1]	defined by application
03$_h$	Level	rw	no	[0,1]	defined by application
04$_h$	free				
05$_h$	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 13]	0 = none function 1 = Digital Input configured as "RUN" function 2 = Digital Input configured as "STOP" function 3 = Digital Input configured as "EMERGENCY" function 4 = Digital Input configured as "RESET" function 5 = Digital Input configured as "DCW" function 6 = Digital Input configured as "DCCW" function 7 = Digital Input configured as "SETVEL1" function

			8 = Digital Input configured as "SETVEL2" function 9 = Digital Input configured as "SETVEL3" function 10 = Digital Input configured as "SETVEL4" function 11 = Digital Input configured as "VEL/CUR" function 12 = Digital Input configured as "DIR" function 13 = Digital Input configured as "ZEROPOS" function
2	State	[0,1]	0 = Low Level Digital Input Not active 1 = High Level Digital Input Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3021_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3022_h: Digital Input 2

This object describes the digital Input 2 configuration.
It is possible to change the configuration ad the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3022_h	Digital Input 2	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0 .. 13]	defined by application
02_h	State	ro	no	[0,1]	defined by application
03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 13]	0 = none function 1 = Digital Input configured as "RUN" function 2 = Digital Input configured as "STOP" function 3 = Digital Input configured as "EMERGENCY" function 4 = Digital Input configured as "RESET" function 5 = Digital Input configured as "DCW" function 6 = Digital Input configured as "DCCW" function

			7 = Digital Input configured as "SETVEL1" function 8 = Digital Input configured as "SETVEL2" function 9 = Digital Input configured as "SETVEL3" function 10 = Digital Input configured as "SETVEL4" function 11 = Digital Input configured as "VEL/CUR" function 12 = Digital Input configured as "DIR" function 13 = Digital Input configured as "ZEROPOS" function
2	State	[0,1]	0 = Low Level Digital Input Not active 1 = High Level Digital Input Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3022_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3023_h: Digital Input 3

This object describes the digital Input 3 configuration.
It is possible to change the configuration ad the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3023_h	Digital Input 3	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0 .. 13]	defined by application
02_h	State	ro	no	[0,1]	defined by application
03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 13]	0 = none function 1 = Digital Input configured as "RUN" function 2 = Digital Input configured as "STOP" function 3 = Digital Input configured as "EMERGENCY" function 4 = Digital Input configured as "RESET" function 5 = Digital Input configured as "DCW" function

			6 = Digital Input configured as "DCCW" function 7 = Digital Input configured as "SETVEL1" function 8 = Digital Input configured as "SETVEL2" function 9 = Digital Input configured as "SETVEL3" function 10 = Digital Input configured as "SETVEL4" function 11 = Digital Input configured as "VEL/CUR" function 12 = Digital Input configured as "DIR" function 13 = Digital Input configured as "ZEROPOS" function
2	State	[0,1]	0 = Low Level Digital Input Not active 1 = High Level Digital Input Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3023_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3024_h: Digital Input 4

This object describes the digital Input 4 configuration.
It is possible to change the configuration ad the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3024_h	Digital Input 4	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0 .. 13]	defined by application
02_h	State	ro	no	[0,1]	defined by application
03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 13]	0 = none function 1 = Digital Input configured as "RUN" function 2 = Digital Input configured as "STOP" function 3 = Digital Input configured as "EMERGENCY" function 4 = Digital Input configured as "RESET" function

			5 = Digital Input configured as "DCW" function 6 = Digital Input configured as "DCCW" function 7 = Digital Input configured as "SETVEL1" function 8 = Digital Input configured as "SETVEL2" function 9 = Digital Input configured as "SETVEL3" function 10 = Digital Input configured as "SETVEL4" function 11 = Digital Input configured as "VEL/CUR" function 12 = Digital Input configured as "DIR" function 13 = Digital Input configured as "ZEROPOS" function
2	State	[0,1]	0 = Low Level Digital Input Not active 1 = High Level Digital Input Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3024_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3030_h: Digital Output Function

This object describes the function of Digital Outputs.
This object is only read, to change the function of digital output write specific object of digital output: 0x3031 (dig. Out.1), 0x3032 (dig. Out.2), 0x3033 (dig. Out.3), 0x3034 (dig. Out.4).

The digital output configurations available are:

- BIT 0 – FUNCTION 1 - "DRIVE OK": the level defines if the drive is in alarm or not. If the level is '1' the drive does not have alarm and it is in normal condition.
- BIT 1 – FUNCTION 2 - WARNING": the level defines if there was at least warning. To know what is the warning occurred it is necessary to read the warning object 0x2003.
- BIT 2 – FUNCTION 3 - "TARGET REACHED": reserved
- BIT 3 – FUNCTION 4 - "BRAKE STATUS": the level defines the state of brake.
- BIT 4 – FUNCTION 5 - "READY": verifies if the drive is ready to go in run
- BIT 5 – FUNCTION 6 - "MOTOR MOVE": it shows the motor moving. The threshold is object $606F_h$:0 "Velocity Threshold" with timer 6070_h: 0 "Velocity Threshold Timer". If the velocity exceeds this threshold the bit "MOTOR MOVE" will be '1'. This control is valid for all profiles.
- BIT 6 – FUNCTION 7 - "LIMIT ACTIVE": it shows if the drive is in i2t limitation.

Every bit of subindex 1 "Configuration Enable" defined which function is enabled.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3030_h	Digital Outputs Function	ARRAY	INTEGER16	Optional

Every bit of value is the function that the digital output can be used.

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FUNCTION	F16	F15	F14	F13	F12	F11	F10	F9	F8	F7	F6	F5	F4	F3	F2	F1
	-	-	-	-	-	-	-	-	-	LIMIT ACTIVE	MOTOR MOVE	READY	BRAKE STATUS	TARGET REACHED	WARNING	DRIVE OK

Value Definition:

Sub-Index	Field	Configuration	Definition
1	Function Configuration Enable	Every bit defines the function enabled and It can be '0' or '1'	Every bit is the function configuration: 0 = the function is not enabled 1 = the function is enabled
2	Function State	Every bit defines the function and It can be '0' or '1'	Every bit is the state of the function: 0 = the function actives 1 = the function doesn't active
3	Function Level	Every bit defines the function and It can be '0' or '1'	Every bit is the level of the output function: 0 = positive edge (standard configuration) 1 = negative configuration
4	Digital Raw	Dig Out 1 - Bit 0 = '0' or '1' Dig Out 2 - Bit 1 = '0' or '1' Dig Out 3 - Bit 2 = '0' or '1' Dig Out 4 - Bit 3 = '0' or '1'	Every Bit indicates the Raw Level of digital Output 0 = positive edge 1 = negative edge

Example:

1. Digital Output **DEFAULT** configuration:
   - Digital Output 1 = DRIVE OK → FUNCTION 1
   - Digital Output 2 = WARNING → FUNCTION 2
   - Digital Output 3 = READY → FUNCTION 5
   - Digital Output 4 = BRAKE STATUS → FUNCTION 4

   The value of "Function Configuration Enable" (0x3030:1) will be 0x002B (0000 0000 0001 1011_b).

   If the value "Function State" (0x3030:2) is 0x0002 (0000 0000 0000 0010_b) means that the output 2 (WARNING, FUNCTION 2) is activated, and the drive is not ok then it has an alarm occurred.

   The "Digital Raw" (0x3030:4) will be 0x0002 (0000 0000 0000 0010_b).

   If the "Function Level" (0x3030:3) is 0x0000 then all edge are positive level.

information

To change Digital Output **DEFAULT** configuration contact manufacturer

Object 3031_h: Digital Output 1

This object describes the digital Output 1 configuration.
It is possible to change the configuration and the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3031_h	Digital Output 1	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0..7]	defined by application
02_h	State	ro	no	[0,1]	defined by application
03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 7]	0 = none function 1 = Digital Output configured as "DRIVE OK" function 2 = Digital Output configured as "WARNING" function 3 = Digital Output configured as "TARGET REACHED" function 4 = Digital Output configured as "BRAKE STATUS" function 5 = Digital Output configured as "READY" function 6 = Digital Output configured as "MOTOR MOVE" function 7 = Digital Output configured as "LIMIT ACTIVE" function
2	State	[0,1]	0 = Low Level Digital Output Not active 1 = High Level Digital Output Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3031_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3032_h: Digital Output 2

This object describes the digital Output 2 configuration.
It is possible to change the configuration and the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3032_h	Digital Output 2	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value

00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0..7]	defined by application
02_h	State	ro	no	[0,1]	defined by application
03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 7]	0 = none function 1 = Digital Output configured as "DRIVE OK" function 2 = Digital Output configured as "WARNING" function 3 = Digital Output configured as "TARGET REACHED" function 4 = Digital Output configured as "BRAKE STATUS" function 5 = Digital Output configured as "READY" function 6 = Digital Output configured as "MOTOR MOVE" function 7 = Digital Output configured as "LIMIT ACTIVE" function
2	State	[0,1]	0 = Low Level Digital Output Not active 1 = High Level Digital Output Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3032_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3033_h: Digital Output 3

This object describes the digital Output 3 configuration.
It is possible to change the configuration and the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3033_h	Digital Output 3	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0..7]	defined by application
02_h	State	ro	no	[0,1]	defined by application

03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 7]	0 = none function 1 = Digital Output configured as "DRIVE OK" function 2 = Digital Output configured as "WARNING" function 3 = Digital Output configured as "TARGET REACHED" function 4 = Digital Output configured as "BRAKE STATUS" function 5 = Digital Output configured as "READY" function 6 = Digital Output configured as "MOTOR MOVE" function 7 = Digital Output configured as "LIMIT ACTIVE" function
2	State	[0,1]	0 = Low Level Digital Output Not active 1 = High Level Digital Output Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3033_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 3034_h: Digital Output 4

This object describes the digital Output 4 configuration.
It is possible to change the configuration ad the Level.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3034_h	Digital Output 4	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Configuration	rw	no	[0..7]	defined by application
02_h	State	ro	no	[0,1]	defined by application
03_h	Level	rw	no	[0,1]	defined by application
04_h	free				
05_h	free				

Value Definition:

Sub-Index	Field	value	Definition
1	Configuration	[0 .. 7]	0 = none function 1 = Digital Output configured as "DRIVE OK" function 2 = Digital Output configured as "WARNING" function 3 = Digital Output configured as "TARGET REACHED" function 4 = Digital Output configured as "BRAKE STATUS" function 5 = Digital Output configured as "READY" function 6 = Digital Output configured as "MOTOR MOVE" function 7 = Digital Output configured as "LIMIT ACTIVE" function
2	State	[0,1]	0 = Low Level Digital Output Not active 1 = High Level Digital Output Active
3	Level	[0,1]	0 = positive edge (standard configuration) 1 = negative edge

The drive will sent the follow abort codes:

- 0x06090011 = sub-index does not exist

To use new configuration this object must be changed and saved in the memory of e2prom.

E^2prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 3034_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 4000_h: Safety State

This object is used to communicate the state Safety of the drive. It is only read.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
4000h	Safety State	ARRAY	UNSIGNED16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
0	Number Of Entries	ro	no	-	2
1	Safety State	ro	no	[0,1]	-
2	STO Function	ro	no	[0,1]	-

Value Definition:

Sub-Index	Field	Configuration	Definition
1	Safety State	0_b 1_b	Drive isn't in safety state. Drive in safety state
2	STO Function	0_b 1_b	STO Safety State is not happened. STO Safety State is happened

information

Referring to "FUNCTIONS" chapter to know the "SAFETY" function.

Example: To know the status of drive in "SAFETY" :

Read the "Status Word" Object, Index 0x6041 and sub-index 0: xx1x xxxx xxxx xxxxb

Rx: Id 0x601 – 40 41 60 00 00 00 00 00
Tx: Id 0x581 – 4b 41 60 00 23 40 00 00

Rx: [cmd: 0x40] [Index: 0x6041] [sub-index: 0x00] [Data: 0x00000000] – Read SDO Status Word
Tx: [resp: 0x4B] [Index: 0x6041] [sub-index: 0x00] [Data: 0x00004023] – Bit 14: 0 = No Safety, 1 = Safety

Or read the "Safety State" Object, Index 0x4000 and sub-index 1:

Rx: Id 0x601 – 40 00 40 01 00 00 00 00
Tx: Id 0x581 – 4b 00 40 01 01 00 00 00

Rx: [cmd: 0x40] [Index: 0x4000] [sub-index: 0x01] [Data: 0x00000000] – Read "Safety State"
Tx: [resp: 0x4B] [Index: 0x4000] [sub-index: 0x01] [Data: 0x00000001] – `0=no Safety, 1=Safety`

Read the "Drive Mode" Object, Index 0x2002 and sub-index 0:

Rx: Id 0x601 – 40 02 20 00 00 00 00 00
Tx: Id 0x581 – 4b 02 20 00 80 00 00 00

Rx: [cmd: 0x40] [Index: 0x2002] [sub-index: 0x00] [Data: 0x00000000] – Read State "Drive Control State
Tx: [resp: 0x4B] [Index: 0x2002] [sub-index: 0x00] [Data: 0x00000080] – Value = 0x80 means the drive is in SAFETY state with STO applied

Object 4100h: Status Communication CAN

This object describes the status of Communication of Network. Also it provides the status of the CAN controller status register.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
4100h	Status Communication CAN	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	5	5
01h	TEC Register	ro	no	[0, 255]	-
02h	REC Register	ro	no	[0,255]	-
03h	Actual Flag Error	ro	no	[0, 32767]	-
04h	Actual State Machine	ro	no	[0, 32767]	-
05h	Actual State Drive	ro	no	[0, 32767]	-

Value Definition:

Sub-Index	Field	value	Definition
1	TEC Register	[0 , 255]	CAN transmit error counter

2	REC Register	[0 , 255]	CAN Receive error counter
3	Actual Flag Error	[0, 32767]	Indicate the occurence of errors on the CAN driver
4	Actual State Communication Machine	[0 , 7]	The value indicates the state of Communication State Machine of Lafert Servo Drive: • 0 = IDLE STATE (first state) • 1 = INIT STATE (initialization CAN Controller) • 2 = TELL STATE (the CAN protocol isn't selected to drive the motor, The CAN communication is used only read the object) • 3 = INACTIVE STATE (when the CAN drive is in sleep) (***not implemented)*** • 4 = STATE OK (the communication is ok, the drive transmit and receive correctly) • 5 = WARNING STATE (when the network has the error passive) • 6 = FAULT STATE (the network is in bus off , the drive will go in Fault State to Communication Error) • 7 = RESET STATE
5	Actual State Drive	[0 , 8]	Every BIT indicates the state of CAN Controller: • BIT 0 : STATE INIT - CAN controller is init • BIT 1 : STATE ACTIVE - CAN Controller is active • BIT 2 : STATE BUSOFF - CAN Controller is busoff • BIT 3 : STATE PASSIVE - CAN Controller is passive

Bits definitions of subindex 3 "Actual Flag Error":

BIT	Meaning	Description	Remarks
0 ... 4	Reserved	Not used	
5	Error OverFlow	CAN-controller overrun error	1: Data overrun detected 0: No data overrun
6	Error Tx Buffer Overflow	Transmit buffer overflow	1: Tx buffer in overflow 0: No Tx Buffer in overflow
7	Error Rx Buffer Overflow	Receive buffer overflow	1: Rx buffer in overflow 0: No Rx Buffer in overflow
8	Error Passive	CAN-controller in error Passive	1: Error passive detected 0: No error passive
9	Error Busoff	CAN-controller in BUS OFF	1: Error bus off detected 0: No bus off
10 ... 15	Reserved	Not used	

Object 4101h: Counter Communication CAN

This object describes the counter of Communication CAN.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
4101h	Counter Communication CAN	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	5	5
01h	Counter BusOff	ro	no	[0, 32767]	0
02h	Counter Recovery Busoff	ro	no	[0, 32767]	0
03h	Counter Reset Communication CAN	ro	no	[0, 32767]	0
04h	Counter Abort	ro	no	[0, 32767]	0
05h	free	ro	no	[0, 32767]	-

Value Definition:

Sub-Index	Field	value	Definition

01h	Counter BusOff	[0 ... 32767]	This counter defines the number of times that the drive was in BUS OFF State
02h	Counter Recovery BusOff	[0 ... 32767]	This counter defined the number of times that the drive can tried to exit by BUS OFF State
03h	Counter Reset Communication CAN	[0 ... 32767]	This counter defined the number of times that the drive reset the CAN-drive after fault communication CAN occurred
04h	Counter Abort Communication	[0 ... 32767]	This counter defines the number of times the drive detects an interruption in communication.

Object 4102h: Settings Communication CAN

This object describes the settings of Communication CAN.

- The drive will be in alarm Fault when the drive is in Bus-Off State.
- The drive can try to recovery from Bus-Off State.
- The user can to set the different sensibility to manage the communication error. If the sensibility is set to HIGH then the drive will go in ALARM also with the Error Passive State. The timeout to enter in fault can be set in this subject.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
4102h	Settings Communication CAN	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	5	5
01h	Sensibility Error CAN	rw	no	[0, 32767]	1
02h	Timeout Error Passive	rw	no	[0, 32767]	100
03h	Clear Abort Communication Flag	rw	no	[0, 32767]	0
04h	free	ro	no	[0, 32767]	-
05h	free	ro	no	[0, 32767]	-

Value Definition:

Sub-Index	Field	value	Definition
01h	Sensibility Error CAN	[0 ,1]	0 = High sensibility Fault Mode - The drive will go in alarm state with "Error Passive" after timeout (4102h:2) 1 = Low Sensibility Fault Mode - The drive will go in alarm state ONLY "BusOff Error"(the error Passive can be read in the Warning Flags)
02h	Timeout Error Passive	[0 ... 32767]	Timeout to enter in FAULT state with Error Passive. This Timeout is used only the sensibility Error CAN i set in HIGH. The units is [time *10ms] = 100 is 1second.
03h	Clear Abort Communication Flag	[0 ,1]	This value clears the abort communication bit set in the statusword when a communication interruption occurs (configuration statusword must be set value 3 in object 0x3100:0)

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state

- Write the new value in SDO object 3033_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 4103_h: Error Communication CAN

This object describes the errors occurred of Communication CAN

Object Description:

Index	Name EDS	Object Code	Data Type	Category
4103_h	Error Communication CAN	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00_h	Number Of Entries	ro	no	5	5
01_h	Last Error Flag	ro	no	[0, 32767]	0
02_h	All Flags Error	ro	no	[0, 32767]	0
03_h	Flag Abort Communication	ro	no	[0, 32767]	0
04_h	First Abort Code Occurred	ro	no	[0, 32767]	0
05_h	free	ro	no	[0, 32767]	-

Sub-Index	Field	value	Definition
01_h	Last Error Flag	[0 ... 32767]	The value describe the last error flag occurred: • 0x0010 = Error OverFlow • 0x0020 = Error Tx Buffer Overflow • 0x0040 = Error Rx Buffer Overflow • 0x0080 = Error Passive • 0x0100 = Error Busoff
02_h	All Flags Error	[0 ... 32767]	The bits selected describe the error occurred • BIT 4 = Error OverFlow • BIT 5 = Error Tx Buffer Overflow • BIT 6 = Error Rx Buffer Overflow • BIT 7 = Error Passive • BIT 8 = Error Busoff
03_h	Flag Abort Communication	[0 ... 32767]	The bits selected describe the abort code communication occurred • Bit 0: none • Bit 1: Bus-Off • Bit 2: Life guarding • Bit 3: HeartBeat • Bit 4: NMT Stopped state entered • Bit 5: Reset application • Bit 6: Reset communication • Bit 7: Error Passive
03_h	First Abort Code Occurred	[0 ... 32767]	The value describes the first abort code communication that it is occurred: 1 = Bus-Of 2 = Life guarding 3 = HeartBeat 4 = NMT Stopped state entered 5 = Reset application 6 = Reset communication

Object 6402_h: Motor Type

This object indicates the type of motor attached to and driven by the drive device.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6402h	Motor Type	VARIABLE	UNSIGNED16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Motor Type	rw	no	0 – 0xFFFF	-

Value Definition:

Sub-Index	Field	Definition
0000h	non-standard motor	-
0001h	phase modulated DC motor	-
0002h	frequency controlled DC motor	-
0003h	PM synchronous motor	-
0004h	FC synchronous motor	AC synchronous sinewave wound field
0005h	switched reluctance motor	AC synchronous reluctance switched
0006h	wound rotor induction motor	AC asynchronous induction polyphase wound rotor
0007h	squirrel cage induction motor	AC asynchronous induction squirrel cage
0008h	stepper motor	AC synchronous step
0009h	micro-step stepper motor	-
000Ah	sinusoidal PM BL motor	AC synchronous sinusoidal PM
000Bh	trapezoidal PM BL motor	AC synchronous brushless PM trapezoidal
000Ch	AC synchronous reluctance sync	-
000Dh	DC commutator PM	-
000Eh	DC commutator wound field series	-
000Fh	DC commutator wound field shunt	-
0010h	DC commutator wound field compound	-
0011h to 7FFEh	Reserved	
7FFFh	no motor type assigned	-
8000h-FFFFh	manufacturer-specific	-

Object 6403h: Motor Catalogue Number

This object indicates the motor catalogue number (nameplate number) provided by the motor manufacturer. If the number is not assigned yet, this object shall indicate this by /0 (empty string).

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6403h	Motor Catalogue Number	VARIABLE	STRING	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Motor Catalogue Number	rw	no	4	'000'

Valid entries:

Value	Size
'-'	
'S'	Small
'M	Medium
'L'	Large
'C1'	Custom 1
'C2'	Custom 2
'C3'	Custom 3
'C4'	Custom 4

Object 6404h: Motor Manufacturer

This object indicates the name of the motor manufacturer. If the name is not assigned yet, this object shall indicate this by /0 (empty string).

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6404h	Motor Manufacturer	VARIABLE	STRING	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Motor Manufacturer	rw	no	4	'000'

Valid entries:

Value string	Size	Motor product code
'NaN'	None	Not defined
'B40'	Medium	B40E4J – C1078
'B63'	Large	B6304K – H32mm – 48Vdc
'B71'	Small	B7108Q – H40mm – 48Vdc

Object 6502h: Supported Drive Modes

This object provides information on the supported drive modes.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6502h	Supported Drive Modes	VARIABLE	U32	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Supported Drive Modes	ro	no	[0 – 32767]	4

Valid entries:

	Value	Lafert Servo Drive
bit 0	profile position mode	
bit 1	velocity mode	
bit 2	profile velocity mode	SUPPORTED
bit 3	profile torque mode	SUPPORTED
bit 4	reserved	
bit 5	homing mode	
bit 6	interpolated position mode	
bit 7	cyclic synchronous position mode	
bit 8	cyclic synchronous velocity mode	
bit 9	cyclic synchronous torque mode	
bit 10-15	reserved	
bit 16	manufacturer-specific – Analog Mode	SUPPORTED
bit 17	manufacturer-specific – Test Mode (reserved)	SUPPORTED
bit 18-31	manufacturer-specific	

PROFILE OBJECTS DSP402

Object 6040h: Controlword

This object is used to control the CiA-402 FSA, CiA-402 modes and manufacturer-specific entities.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6040h	Controlword	VAR	UNSIGNED16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Control word	rw	YES (default)	See table	-

This object is organized bit-wise. The bits have the following meaning.

Bits	Name	Description
bit 0	Switch ON	Bit command to move into state machine DSP402
bit 1	Enable Voltage	Bit command to move into state machine DSP402
bit 2	Quick Stop	Bit command to move into state machine DSP402
bit 3	Enable Operation	Bit command to move into state machine DSP402
bit 4	Operation mode specific *	These bits are different meaning as profile mode selected
bit 5	Operation mode specific *	These bits are different meaning as profile mode selected
bit 6	Operation mode specific *	These bits are different meaning as profile mode selected
bit 7	Fault Reset	If 1 then it clears the fault and to restart the drive without error fault
bit 8	Halt	0 = The commanded motion shall be continued if possible 1 = The commanded motion shall be interrupted
bit 9	Operation mode specific *	These bits are different meaning as profile mode selected
bit 10	reserved	-
bit 11	Man. specific – Warning Ack	If 1 then It clears the warning bit in the status word
bit 12	Man. specific – Emergency Cmd	If 1 then the drive set an Emergency Command (not safety certified)
bit 13	Man. specific	free
bit 14	Man. specific	free
Bit 15	Man. specific	free

(*)These bits are different meaning as profile mode selected

To move into the states of State Machine DSP402 the master must be send the controlword with the bit 0, 1, 2, 3 and 7 written as command. The meaning of commands is decrypted in the following table:

Command	Bit of the controlword					Transitions
	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	
	Fault Reset	Enable Operation	Quick Stop	Enable Voltage	Switch On	
Shutdown	0	X	1	1	0	T2, T6, T8
Switch On	0	0	1	1	1	T3
Switch ON	0	1	1	1	1	T3 (note 2)
Disable Voltage	0	X	X	0	X	T7, T9, T10, T12
Quick Stop	0	X	0	1	X	T7, T10, T11
Disable Operation	0	0	1	1	1	T5
Enable Operation	0	1	1	1	1	T4, T16
Fault Reset(note 1)		X	X	X	X	T15

NOTE

- (note 1) Reset Fault occurred to exit from FAULT state
- (note 2) Automatic transition to enable operation state after executing switched on state functionality - **Not Available**

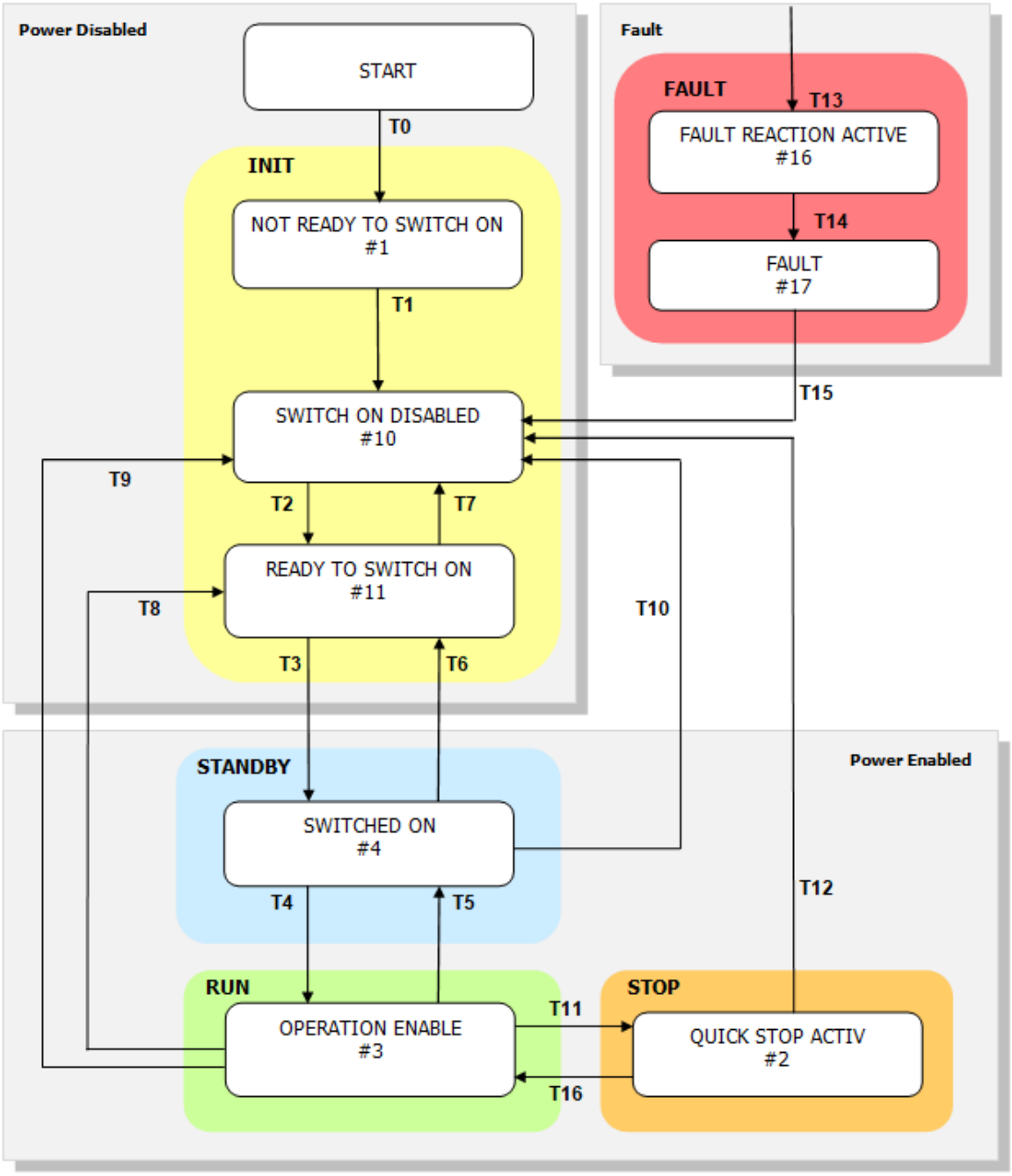


Figure 30 - state machine DSP402 transition controlword

Caution

Between 2 transitions command you wait at least 40ms.

The state machine change from state "switched on" to "operation enable" a time 40ms. Then you are sure that the drive processes correctly the controlword you have to send via PDO or via SDO with the command control word after 40ms or change the value after 40ms.

The controlword is used to move in the state machine of DSP402. The table shows how the master must write the bit to use the command:

Command	Transition	15	14	13	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Shutdown	T2, T6, T8	x	x	x	x	x	x	x	x	x	0	x	x	x	x	1	1	0
Switch-On	T3	x	x	x	x	x	x	x	x	x	0	x	x	x	0	1	1	1
Disable Voltage	T7, T9, T10, T12	x	x	x	x	x	x	x	x	x	0	x	x	x	x	x	0	x
Quick Stop	T7, T10, T11	x	x	x	x	x	x	x	x	x	0	x	x	x	x	0	1	x
Disable Operation	T5	x	x	x	x	x	x	x	x	x	0	x	x	x	0	1	1	1
Enable Operation	T4, T16	x	x	x	x	x	x	x	x	x	0	x	x	x	1	1	1	1
Fault Reset	T15	x	x	x	x	x	x	x	x	x	↑	x	x	x	x	x	x	x

Transition machine state DSP402:

START STATE		Transition	Command	GOAL STATE	
INIT	Switch on Disable (#10)	T2	SHUTDOWN	Ready To Switch ON (#11)	INIT
INIT	Ready To Switch ON (#11)	T3	SWITCH-ON	Switch ON (#4)	STANDBY
INIT	Ready To Switch ON (#11)	T7	QUICK STOP	Switch on Disable (#10)	INIT
STANDBY	Switch ON (#4)	T4	ENABLE OPERATION	Operation Enabled (#3)	RUN
STANDBY	Switch ON (#4)	T6	SHUTDOWN	Ready To Switch ON (#11)	INIT
STANDBY	Switch ON (#4)	T10	DISABLE VOLTAGE or QUICK STOP	Switch on Disable (#10)	INIT
RUN	Operation Enabled (#3)	T5	DISABLE OPERATION	Switch ON (#4)	STANDBY
RUN	Operation Enabled (#3)	T8	SHUTDOWN	Ready To Switch ON (#11)	INIT
RUN	Operation Enabled (#3)	T9	DISABLE VOLTAGE	Switch on Disable (#10)	INIT
RUN	Operation Enabled (#3)	T11	QUICK STOP	Quick Stop Active (#2)	STOP
STOP	Quick Stop Active (#2)	T16	ENABLE OPERATION	Operation Enabled (#3)	RUN
STOP	Quick Stop Active (#2)	T12	DISABLE VOLTAGE	Switch on Disable (#10)	INIT

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x05040001 = operation mode set not available

Object 6041h: Statusword

This object is used to indicate the current state of the FSA, the operation mode and manufacturer-specific entities.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6041h	Statusword	VAR	UNSIGNED16	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Data Type	Value Range
00h	Status word	ro	YES (default)	See table	-

This object is organized bit-wise. The bits have the following meaning:

Bits	Name	Description
bit 0	Ready to switch On	Bit statusword of state machine DSP402
bit 1	Switched On	Bit statusword of state machine DSP402
bit 2	Operation Enabled	Bit statusword of state machine DSP402
bit 3	Fault	Bit statusword of state machine DSP402
bit 4	Voltage Enabled	0 = Vdc-Link is smaller than Value of "under Voltage" 1 = Vdc-Link is greater than Value of "under Voltage"
bit 5	Quick Stop	Bit statusword of state machine DSP402. This shall indicate that the drive is reacting on a quick stop request
bit 6	Switch on disabled	Bit statusword of state machine DSP402
bit 7	Warning	0 = No warning is present (Warning is not an error or fault) 1 = At least warning is occurred (To refer at warning list in object 2003h to know the warning occurred)
bit 8	Manufacturer specific	See table
bit 9	Remote	If 1 the controlword is processed
bit 10	Target reached	0 = The set-point has not been reached yet. 1 = The drive has reached the set-point.
bit 11	Internal Limit Active	0 = Indicate that an i2T limit is not active 1 = Indicate that an i2T limit is active
bit 12	Operation mode specific	- Profile Position mode: Set-point acknowledge - Profile Velocity mode: Speed - Profile Torque mode: reserved - Homing mode: Homing Attained
bit 13	Operation mode specific	- Profile Position mode: Following error - Profile Velocity mode: reserved - Profile Torque mode: reserved - Homing mode: Homing Error
bit 14	Manufacturer specific	See table
Bit 15	Manufacturer specific	See table

Bits 0, 1, 2, 3, 5, 6 statusword description:

Value (binary)	statusword
xxxx xxxx x0xx 0000	Not ready to switch on
xxxx xxxx x1xx 0000	Switch on disabled
xxxx xxxx x01x 0001	Ready to switch on
xxxx xxxx x01x 0011	Switch on
xxxx xxxx x01x 0111	Operation enabled
xxxx xxxx x00x 0111	Quick stop active
xxxx xxxx x0xx 1111	Fault reaction active
xxxx xxxx x0xx 1000	Fault

Table 35 - Bits "Statusword"

To know which state of the State Machine DSP402 the drive is, the master can read the statusword (bit 0, 1, 2, 3, 5, 6).

The meaning of commands is decrypted in the following table:

STATE		Number State	statusword																
			15	14	13	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INIT	Not Ready to switch On	#1	x	x	x	x	x	x	x	x	x	x	0	x	x	0	0	0	0
INIT	Switch On disabled	#10	x	x	x	x	x	x	x	x	x	x	1	x	x	0	0	0	0
INIT	Ready to switch on	#11	x	x	x	x	x	x	x	x	x	x	0	1	x	0	0	0	1
STANDBY	Switched on	#4	x	x	x	x	x	x	x	x	x	x	0	1	x	0	0	1	1
RUN	Operation enabled	#3	x	x	x	x	x	x	x	x	x	x	0	1	x	0	1	1	1
STOP	Quick Stop Active	#2	x	x	x	x	x	x	x	x	x	x	0	0	x	0	1	1	1
FAULT	Fault Reaction Active	#16	x	x	x	x	x	x	x	x	x	x	0	x	x	1	1	1	1
FAULT	Fault	#17	x	x	x	x	x	x	x	x	x	x	0	x	x	1	0	0	0

Table 36 – statusword to know state of state machine

Bits 12, 13 "operation mode spec" description:

BIT	Operation Mode			
	Profile Position mode	Profile Velocity Mode	Profile Torque Mode	Homing Mode
12	Set-point acknowledge	Speed	Reserved	Homing Attained
13	Following error	Reserved	Reserved	Homing Error

Table 37 - bits "operation mode specific" of statusword

Bits 8, 14, 15 are "Manufacturer Specific": these bits have a default configuration if the object 3100_h: 0 ("Configuration 1 - statusword") is 0.

Configuration 0 (object 3100_h: 0 = 0) – DEFAULT:

BIT	Manufacturer Specific		
	Name	VALUE	Description
8	Emergency Input Enable	0_b 1_b	Input Emergency Function is not enabled Input Emergency Function is enabled
14	Safety State [1]	0_b 1_b	Drive is not in SAFETY state Drive is in SAFETY state
15	Fault State	0_b 1_b	Drive is not in safe SAFETY and it is not FAULT, normal condition Drive is in FAULT, one alarm is detected

Table 38 – Bits "Manufacturer Specific" - Configuration Default

Configuration 1 (object 3100_h: 0 = 1):

BIT	Manufacturer Specific		
	Name	VALUE	Description
8	Emergency Input Enable	0_b 1_b	Input Emergency Function is not enabled Input Emergency Function is enabled
14	Safety State [1]	0_b 1_b	Drive is not in SAFETY state Drive is in SAFETY state
15	Ready [2]	0_b 1_b	Drive is not READY to start Drive is READY to start

Table 39 - Bits "Manufacturer Specific" - Configuration n.1

[1] SAFETY is a state of state machine. The drive can be in SAFETY state but the STO relay was rearmed, the drive is waiting a command "disable voltage" (0x00) of controlword to exit by SAFETY State.

[2] READY verifies if the drive is ready to go in run. The different conditions must be verified: Vdc-Link must be enabled (see bit 4 of status word), STO dig input must be not activated, there must be no fault and if it is present one input configured by "Emergency Input Enable", it must be disabled.

Configuration 1 (object 3100_h: 0 = 2):

BIT	Manufacturer Specific		
	Name	VALUE	Description
8	Emergency Input Enable	0_b 1_b	Input Emergency Function is not enabled Input Emergency Function is enabled
14	S-Ramp Error Calc Jerk [3]	0_b 1_b	S-Ramp calculation with new target is within the range of maximum allowed Jerk S-Ramp calculation with new target is outside the range of maximum allowed Jerk
15	S-Ramp Error Calc Limit [4]	0_b 1_b	S-Ramp calculation with the new target is within the range of maximum allowed acceleration S-Ramp calculation with the new target is outside the range of maximum allowed acceleration

Table 40 - Bits "Manufacturer Specific" - Configuration n.2

[3] S-Ramp Error Calc Jerk: the calculation is performed when the speed is processed. (This are available only S-Ramp profile)

[4] S-Ramp Error Calc Limit: the calculation is performed when the speed is processed.
The maximum acceleration allowed is = Max Acceleration *1,57. (This are available only S-Ramp profile)

Configuration 1 (object 3100_h: 0 = 3):

BIT	Manufacturer Specific		
	Name	VALUE	Description
8	Emergency Input Enable	0_b 1_b	Input Emergency Function is not enabled Input Emergency Function is enabled
14	Safety State [1]	0_b 1_b	Drive is not in SAFETY state Drive is in SAFETY state
15	Abort Communication [5]	0_b 1_b	Nothing abort communication is occurred At least abort communication is occurred

Table 41 - Bits "Manufacturer Specific" - Configuration n.3

[5] Abort Communication: this flag can become 1 if at least one cause of communication abort has occurred: bus-off, life guarding, entering NMT stopped state, sending Reset Application by NMT, sending Reset communication by NMT or CAN peripheral gone into Error Passive State. The behaviour of drive after abort communication is set in the object 0x6007

6007h: Abort Connection Option Code

This object shall indicate what action shall be performed when one of the following events occurred: bus-off, heartbeat, life guarding, NMT stopped state entered, reset application, and reset communication.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6007h	Abort Connection Option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00h	Abort Connection Option Code	rw	no	See table	-1

The following value definition is valid:

Value	Meaning	Lafert Servo Drives
-1	Manufacturer-specific: • Bus-Off: FAULT SIGNAL, the drive goes in FAULT state • HeartBeat: NO ACTION, the master controller must define the strategy when the heartbeat message is lost • Life guarding: FAULT SIGNAL, after number of lost message descripted by "Life Time Factor" object (100Dh) the drive goes in FAULT state • NMT Stopped state entered: NO ACTION • Reset application: NO ACTION • Reset communication: NO ACTION	**Available**
0	No action	**Available**
1	Fault signal	**Available**
2	Disable voltage command	**Available**
3	Quick stop command	**-**

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)
- 0x06040030 = the value is out of range (see table Entry Description)

It is possible to change the value in run time.

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6007h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

There are other objects to have more information about abort communication:

- Counter Abort Communication (0x4101:4) → This counter defines the number of times the drive detects an interruption in communication.
- Clear Abort Communication Flag (0x4102:3) → This value clears the abort communication bit set in the statusword when a communication interruption occurs (configuration statusword must be set value 3 in object 0x3100:0)
- Flags Abort Communication (0x4103:3) → The bits selected describe the abort code communication occurred
- First Abort Code Occurred (0x4103:4) → The value describes the first abort code communication that it is occurred

information

Referring to "FUNCTIONS" chapter to know the "ABORT CONNECTION" function.

Object 605A$_h$: Quick Stop Option Code

This object shall indicate what action is performed when the quick stop function is executed.
It indicates the action to be performer upon a transition from “Operation Enabled” to “Quick Stop Active”.
The slow down ramp is the deceleration value of the used mode of operations.

This object is implemented for ONLY use READ.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
605A$_h$	Quick Stop Option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00$_h$	Quick Stop Option Code	rw	no	See table	-1

The following value definition is valid:

Value	Meaning	Lafert Servo Drives
-1	Manufacturer-specific: Immediate Stop	**Available**
0	Disable drive function	-
1	Slow down on slow down ramp and transit into switch on disabled	-
2	Slow down on quick stop ramp and transit into switch on disabled	-
3	Slow down on current limit and transit into switch on disabled	**-**
4	Slow down on voltage limit and transit into switch on disabled	-
5	Slow down on slow down ramp and stay in quick stop active	-
6	Slow down on quick stop ramp and stay in quick stop active	**-**
7	Slow down on current limit and stay in quick stop active	-
8	Slow down on voltage limit and stay in quick stop active	-

information

Referring to “TRANSITION TO EXIT FROM “OPERATION ENABLED” STATE (RUN)” chapter to have additional information.

Object 605B$_h$: Shutdown Option Code

This object shall indicate what action is performed if there is a transition from "Operation Enabled" state to "Ready To Switch On State".
The slowdown ramp is the deceleration value of the used mode of operations defined by object 0x3007:4.

This object is READ/WRITE.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
605B$_h$	Shutdown Option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00$_h$	Shutdown Option Code	rw	no	See table	1

The following value definition is valid:

Value	Description	Lafert
1	**P402: Slow down with slow down ramp; disable of the drive function** Electronic Dynamic Stop is ENABLED. It slows down with ramp- defined by object 0x3007:4. When the velocity is on the range ±5 RPM it disables the drive. Stop with ramp programmed (RAMP TRAPEZOIDAL).	**Available**
0	**P402: Disable drive function (switch-off the drive power stage)** Electronic Dynamic Stop is DISABLED. Disable drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.	**Available**
-5 (1)	**P402: Manufacturer-specific** Electronic Dynamic Stop is ENABLED. It slows down with down S-ramp defined by object 0x3007:8 Stop with ramp programmed (S-RAMP).	**Available**

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x060B0002 = the written is not possible because the drive his configured with other fieldbus (not CANopen)
- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06040047 = internal incompatibility: it is set '-5' but the motion profile type is not SRAMP

It is possible to change the value in run time.

To use new configuration this object must be changed and saved in the memory of e2prom.

E^{2}prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 605B$_h$

- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "TRANSITION TO EXIT FROM "OPERATION ENABLED" STATE (RUN)" chapter to have additional information.

Object $605C_h$: Disable Operation Option Code

This object shall indicate what action is performed if there is a transition from "Operation Enabled" state to "Switched On State".
The slow down ramp is the deceleration value of the used mode of operations defined by object 0x3007:4.

This object is READ/WRITE.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
$605C_h$	Disable Operation Option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00_h	Disable Operation Option Code	rw	no	See table	1

The following value definition is valid:

Value	Description	Lafert
1	**P402: Slow down with slow down ramp; disable of the drive function** Electronic Dynamic Stop is ENABLED. It slows down with ramp- defined by object 0x3007:4. When the velocity is on the range ±5 RPM it disables the drive. Stop with ramp programmed (RAMP TRAPEZOIDAL).	**Available**
0	**P402: Disable drive function (switch-off the drive power stage)** Electronic Dynamic Stop is DISABLED. Disable drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.	**Available**
-1	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It stops immediately and set the Velocity to 0. When the velocity is on the range ±5 RPM it actives the electronic brake motor.	**Available**
-2	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive function until the time delay defined by object 0x3007:2 is expired.	**Available**
-3	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It stops immediately and It locks immediately the brake motor, and set the Velocity to 0	**Available**
-4	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by	**Available**

	inertia. and lock immediately the brake motor.	
-5 (1)	**P402: Manufacturer-specific** Electronic Dynamic Stop is ENABLED: It slows down with down S-ramp defined by object 0x3007:8 Stop with ramp programmed (S-RAMP).	**Available**

information

Referring to "TRANSITION TO EXIT FROM "OPERATION ENABLED" STATE (RUN)" chapter to have additional information.

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $605C_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object $605D_h$: Halt Option Code

This object shall indicate what action is performed when the halt function is executed.
The slow down ramp is the deceleration value of the used mode of operations defined by object 0x3007:4.

This object is implemented for ONLY use READ.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
$605C_h$	Halt Option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00_h	Halt Option Code	rw	no	See table	1

The following value definition is valid:

Value	Meaning	Lafert Servo Drives
-1	Manufacturer-specific: Immediate Stop	**Available**
1	Slow down on slow down ramp and stay in operation enabled	-
2	Slow down on quick stop ramp and stay in operation enabled	-
3	Slow down on current limit and stay in operation enabled	-
4	Slow down on voltage limit and stay in operation enabled	-

information

Referring to "TRANSITION TO EXIT FROM "OPERATION ENABLED" STATE (RUN)" chapter to have additional information.

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $605D_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object $605E_h$: Fault Reaction Option Code

This object shall indicate what action is performed when fault is detected.
The slow down ramp is the deceleration value of the used mode of operations defined by object 0x3007:4.

This object is READ/WRITE.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
$605E_h$	Fault Reaction Option Code	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00_h	Fault Reaction Option Code	rw	no	See table	1

The following value definition is valid:

Value	Description	Lafert
1	**P402: Slow down on slow down ramp** Electronic Dynamic Stop is ENABLED. It slows down with ramp- defined by object 0x3007:4. When the velocity is on the range ±5 RPM it disables the drive. Stop with ramp programmed (RAMP TRAPEZOIDAL).	**Available**
0	**P402: Disable drive function, motor is free to rotate** Electronic Dynamic Stop is DISABLED. Disable drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.	**Available**
-1	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It stops immediately and set the Velocity to 0. When the velocity is on the range ±5 RPM it actives the electronic brake motor.	**Available**
-2	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive function until the time delay defined by object 0x3007:2 is expired.	**Available**
-3	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It stops immediately and It locks immediately the brake motor, and set the Velocity to 0	**Available**

-4	**P402: Manufacturer-specific** Electronic Dynamic Stop is DISABLED. It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. and lock immediately the brake motor.	**Available**
-5 (1)	**P402: Manufacturer-specific** Electronic Dynamic Stop is ENABLED: It slows down with down S-ramp defined by object 0x3007:8 Stop with ramp programmed (S-RAMP).	**Available**

information

Referring to "TRANSITION TO EXIT FROM "OPERATION ENABLED" STATE (RUN)" chapter to have additional information.

To use new configuration this object must be changed and saved in the memory of e2prom.

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $605E_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 6060_h: Modes of Operation

The operational mode is selectable by this object.
This object shows only the value of the requested operation mode, the actual operation mode of the PDS is reflected in the object [Mode of Operation Display: 6061_h]

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6060_h	Modes of Operation	VAR	INTEGER8	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00_h	Mode of operation	rw	YES (default)	See table	-

The following value definition is valid:

Value	Meaning	Lafert Servo Drives
0	no mode change / no mode assigned	
1	profile position mode	
2	velocity mode	
3	profile velocity mode	**Available**
4	profile torque mode	**Available**
5	Reserved	
6	homing mode	
7	interpolated position mode	
8	cyclic synchronous position mode	
9	cyclic synchronous velocity mode	

10	cyclic synchronous torque mode	
-1	manufacturer-specific (analog or hardware control mode)	**Available**
-2	manufacturer-specific (reserved for test)	**Available**
-3	manufacturer-specific (modbus control mode)	**Available**

The Manufacturer-specific **-1** is reserved for the mode in analog or hardware control. To set the drive in analog mode contact the manufacture supply.

The Manufacturer-specific **-2** is reserved for manufacture (CAN automatic Velocity Test).

The Manufacturer-specific **-3** is reserved for the mode MODBUS protocol. To set the drive in this mode contact the manufacture supply

The drive will sent the follow abort codes:

- 0x060B0002 = the written is not possible because the drive hah the torque applied (state is "operation enabled" or "Quick stop Active")
- 0x05040001 = command is invalid because the mode is not supported

Caution

It is not possible to set an other mode of the manufacturer-specific profile (analog mode and test mode). To set these profiles contact Manufacturer

Object 6061h: Modes of Operation Display

This object provides the actual operation mode.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6061h	Modes of Operation Display	VAR	INTEGER8	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00h	Mode of operation display	ro	YES (default)	See table	-

The following value definition is valid:

BIT	Meaning
0	no mode change / no mode assigned
1	profile position mode
2	velocity mode
3	profile velocity mode
4	profile torque mode
5	Reserved
6	homing mode
7	interpolated position mode
8	cyclic synchronous position mode
9	cyclic synchronous velocity mode
10	cyclic synchronous torque mode
-1	manufacturer-specific (analog or hardware control mode)
-2	manufacturer-specific (reserved for test)
-3	manufacturer-specific (modbus control mode)

Object 607Eh: Polarity

This object influences the sign of: [Position Demand Value: 6062h] and/or [Velocity Demand Value: 606Bh]

Object Description:

Index	Name EDS	Object Code	Data Type	Category
607E$_h$	Polarity	VAR	UNSIGNED8	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00$_h$	Polarity	rw	no	See table	-

Bits:

BIT	Meaning
0 .. 5	reserved
6	Velocity/Torque Polarity
7	Position polarity

The following value definition is valid:

- bit value = 0: multiply the demand value by 1
- bit value = 1: multiply the demand value by -1

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E^{2}prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active"
- Write the new value in SDO object 607E$_h$
- Write signature "SAVE" in Store Parameters 1010$_h$ object (pay attention on the processing time)
- NMT Reset Node

Object 60FD$_h$: Digital inputs

This object shall provide digital inputs. The low word contains the states of the digital inputs as defined by the CANopen 402 profile. The high word displays the states of all digital inputs.
The status of digital inputs is output by object 60FD$_h$:

- Limit or reference switch for Homing Profile (not implemented)
- Digital Input 1, 2, 3, 4 programmable or defined by application
- Safe Torque Off (STO)

Object Description:

Index	Name EDS	Object Code	Data Type	Category
60FD$_h$	Digital Inputs	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Name	Access	PDO mapping	Value Range	Default
00$_h$	Digital Inputs	ro	no	See table	-

Bits Structure:

Bit MSB							**Bit LSB**
31	16	15	4	3	2	1	0
Digital Input Status Manufacturer Specific		reserved		Interlock	Home switch	Pos limit switch	Neg limit switch

Data Description:

BIT	Configuration	Value	Definition	Note
0	Negative limit switch	0$_b$ 1$_b$	Negative limit switch not reached Negative limit switch reached	If the function "DCW" is configured then this bit is the state of digital input
1	Positive limit switch	0$_b$ 1$_b$	Positive limit switch not reached Positive limit switch reached	If the function "DCCW" is configured then this bit is the state of digital input
2	Home switch	0$_b$ 1$_b$	Home switch not reached Home switch reached	If the function "HOME" is configured then this bit is the state of digital input
3	Interlock	0$_b$ 1$_b$	Interlock not activated Interlock activated	Not Used
4 ... 15	reserved	-	-	
16	Digital Input - DigIn1	0$_b$ 1$_b$	Read Status: Low Level Read Status: High Level	It depends by Function configured
17	Digital Input – DigIn2	0$_b$ 1$_b$	Read Status: Low Level Read Status: High Level	It depends by Function configured
18	Digital Input – DigIn3	0$_b$ 1$_b$	Read Status: Low Level Read Status: High Level	It depends by Function configured
19	Digital Input – DigIn4	0$_b$ 1$_b$	Read Status: Low Level Read Status: High Level	It depends by Function configured
20	Digital Input – STO1	0$_b$ 1$_b$	Read Status: Low Level Read Status: High Level	Digital Input connected to STO circuit
21	Digital Input – STO2 (*)	0$_b$ 1$_b$	Read Status: Low Level Read Status: High Level	(*) Digital Input NOT connected. It is always High Level.
20 ... 31	Digital Input	-	-	Not Available

Object 60FE$_h$: Digital outputs

This object shall command the digital outputs. This object shall represent the logical output levels.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
60FE$_h$	Digital Outputs	ARRAY	U32	Optional

Entry Description:

Sub-Index	Name	Access	PDO mapping	Value Range	Default Value
00$_h$	Highest sub-index supported	c	no	[1, 2]	2
01$_h$	Physical outputs	rw	possible	See table	00000000$_h$
02$_h$	Bit Mask	rw	no	See table	00000000$_h$

Bits Structure of sub-index 01$_h$:

Bit MSB				Bit LSB
31	16	15	1	0
Digital Output Command Manufacturer - specific		reserved		Motor Brake Command

Value Definition for sub-index 01$_h$:

BIT	Configuration	Value	Definition	Note
0	Motor Brake Command	0$_b$ 1$_b$	Brake Activated → Motor Locked Brake Released → Motor Free	It is available if the Brake is in "Manual Mode"
1 ... 15	reserved (each bit)	-	Reserved	-
16	Digital Output1	0$_b$ 1$_b$	Switched off Switched on	Available
17	Digital Output 2	0$_b$ 1$_b$	Switched off Switched on	Available
18	Digital Output 3	0$_b$ 1$_b$	Switched off Switched on	Available
19	Digital Output 4	0$_b$ 1$_b$	Switched off Switched on	Available

Bits Structure of sub-index 02$_h$:

Bit MSB				Bit LSB
31	16	15	1	0
Digital Output Enable/Disable Manufacturer - specific		reserved		-

Value Definition for sub-index 02$_h$:

BIT	Configuration	Value	Definition	Note
0	Motor Brake Management	0$_b$ 1$_b$	Disable output Enable output	It is ever enabled
1 ... 15	reserved	-	Reserved	-
16	Enable Digital Output1	0$_b$ 1$_b$	Disable output Enable output	It is ever enabled
17	Enable Digital Output 2	0$_b$ 1$_b$	Disable output Enable output	It is ever enabled
18	Enable Digital Output 3	0$_b$ 1$_b$	Disable output Enable output	It is ever enabled
19	Enable Digital Output 4	0$_b$ 1$_b$	Disable output Enable output	It is ever enabled

The sub-index 2 is only READ. The outputs are ever enabled.

6. | CANOPEN OPERATION MODES

MODES OF OPERATIONS

Caution

The operating mode is selected with the object 0x6060 whose change is implemented only if the drive is not in "Operation Enabled" state.

The Drive has the modes of operation below:

- **PROFILE POSITION (not available)**

The Drive in this mode is able to make movements in relation to a defined target position. Set Value number 1 of "Mode of Operation" object (6060_h)

- **PROFILE VELOCITY**

The Drive, in this mode, is able to follow a velocity set point without requiring the definition of a target position. Set Value number 3 of "Mode of Operation" object (6060_h)

- **PROFILE TORQUE**

The Drive, in this mode, is able to follow a Current set point without requiring the definition of a target position. Set Value number 4 of "Mode of Operation" object (6060_h)

- **PROFILE HOMING (not available)**

Use this mode to define a homing position. Set Value number 6 of "Mode of Operation" object (6060_h)

- **ANALOG MODE**

In this mode the commands are determined by analog reference input or digital input. The number -1 indicates the Analog Mode profile in "Mode of Operation" object (6060_h). The CAN protocol can be used to monitor variables.

Caution

To set Analog Mode contact the Manufacturer.

- **CAN TEST VELOCITY**

This mode is reserved. The number -2 indicates the CAN is mode operation reserved .

PROFILE POSITION MODE (1) (not available)

Object 6064h: Position actual value

This object shall provide the actual value of the position measurement device.
This object is 0 on the power-on.

Object Description:

Index	Object Code	Data Type	Category
6064h	VAR	INTEGER32	mandatory if pp

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00h	ro	YES	-	no	[inc]

information

Referring to "APPENDIX" chapter to know the description of "POSITION MONITORING"

PROFILE VELOCITY MODE (3)

DESCRPITION

In the Profile Velocity Mode (PV) the speed of the drive is controlled by a PID controller. This ensures that the drive is operated without deviation from the specified values, provided it is not overloaded.

Prerequisites for the drive to be operated in Profile Velocity Mode:

- The Profile Velocity Mode must be set in the "Mode of Operation"(6060_h) parameter (value "3").
- The drive must be in *"Operation Enabled"* state of state machine of DSP402, verify it with the object "Statusword" (6041_h). To move the state machine it uses the object "controlword" (6040_h)
- Target Velocity and parameters of Profile Velocity Mode must be set correctly.

The target velocity is set via the "Target Velocity" ($60FF_h$) object in the object dictionary.

In Profile Velocity Mode the drive directly follows each new transferred set-point value.
At the same time, the set maximum values for acceleration, deceleration ramp and speed are also taken into account.

Controller structure in Profile Velocity Mode:

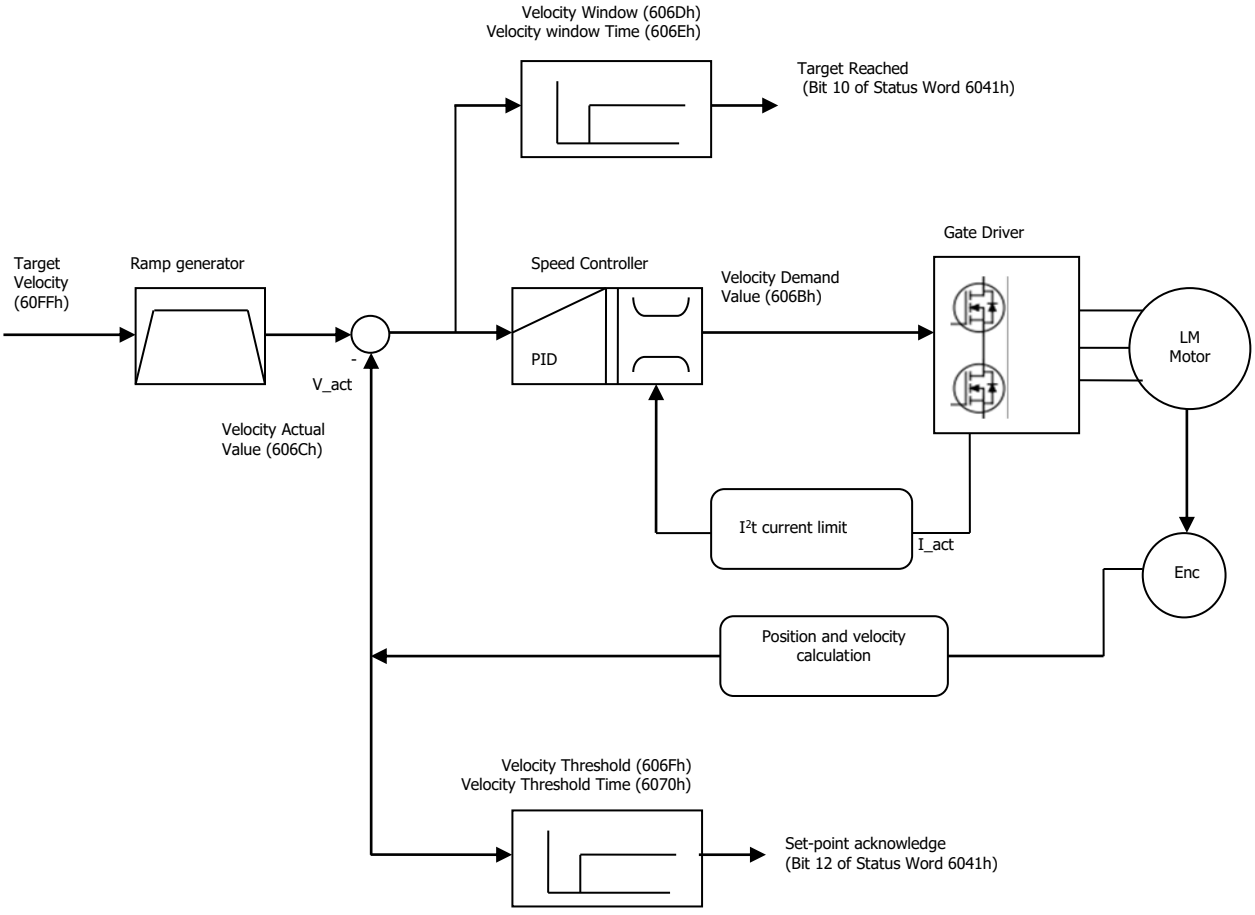


Figure 31 - Controller structure for Profile Velocity

OBJETCS LIST:

The associated objects to control the drive in Profile Velocity Mode are the following:

Index	Sub Index	Name	READ / WRITE	M/O	Data Type	PDO	Available
0x603F	0	Error Code	RO	O	U16	-	x
0x6040	0	Control Word	R/W	M	U16	RPDO	x
0x6041	0	Status Word	RO	M	U16	TPDO	x
0x6060	0	Modes of Operation	R/W	M	I8	RPDO	x
0x6061	0	Modes of Operation Display	RO	M	I8	TPDO	x
0x60FF	0	Target Velocity	R/W	M	I32	RPDO	x
0x607F	0	Max Profile Velocity	R/W	O	U32	-	x
0x6080	0	Max Motor Speed	R/W	O	U32	-	
0x6083	0	Profile Acceleration	R/W	O	U32	TPDO	x
0x6084	0	Profile Deceleration	R/W	O	U32	TPDO	x
0x60C5	0	Max Acceleration	R/W	O	U32	-	x
0x60C6	0	Max Deceleration	R/W	O	U32	-	x
0x607E	0	Polarity	R/W	O	U8	-	x
0x606B	0	Velocity Demand Value	RO	O	I16	TPDO	x
0x606C	0	Velocity Actual Value	RO	M	I32	TPDO	x
0x606D	0	Velocity Window	R/W	O	U16	-	x
0x606E	0	Velocity Window Time	R/W	O	U16	-	x
0x606F	0	Velocity Threshold	R/W	O	U16	-	x
0x6070	0	Velocity Threshold Time	R/W	O	U16	-	x
0x3300	0	Velocity Full Scale	R/W	O	U16	-	x
0x6086	0	Motion Profile Type	R/W	O	I16	-	x
0x60E0	0	Positive torque limit value	R/W	O	U16	-	x
0x60E1	0	Negative torque limit value	R/W	O	U16	-	x
0x6085	0	Quick-Stop deceleration	R/W	O	I16	-	
0x2053	0	Velocity filtered	R/W	O	I16	TPDO	x
0x60A4	1	Profile Jerk 1	R/W	O	U32	-	x
0x60A4	2	Profile Jerk 2	R/W	O	U32	-	x

Table 42 - Velocity Profile's objects

In the Profile Velocity operation mode, the movement profile is defined by velocity and acceleration/deceleration commands.

To initiate a velocity-controlled profile:

- Switch the operation mode to Profile Velocity mode by writing '3' to object "Mode of Operation" (6060_h).
- Use "Controlword" (6040_h) to move in the "Operation Enable" state of Finite State Machine DSP402.
- Set acceleration in object "Profile Acceleration" (6083_h) and the deceleration in object "Profile Deceleration"(6084_h) respectively.
- Start motion by setting the target velocity in object "Target velocity" ($60FF_h$).

If needed, clear Bit 8 in object "Controlword"(6040_h) to start motion.

In this mode the Drive is able to follow a set point of speed "Target velocity" (60FF$_h$).Target velocity can be changed on-the-fly during motion. The set point is reached with the accelerations defined 6083$_h$ and 0x6084$_h$.

The following picture shows the objects of Velocity Profile block diagram:

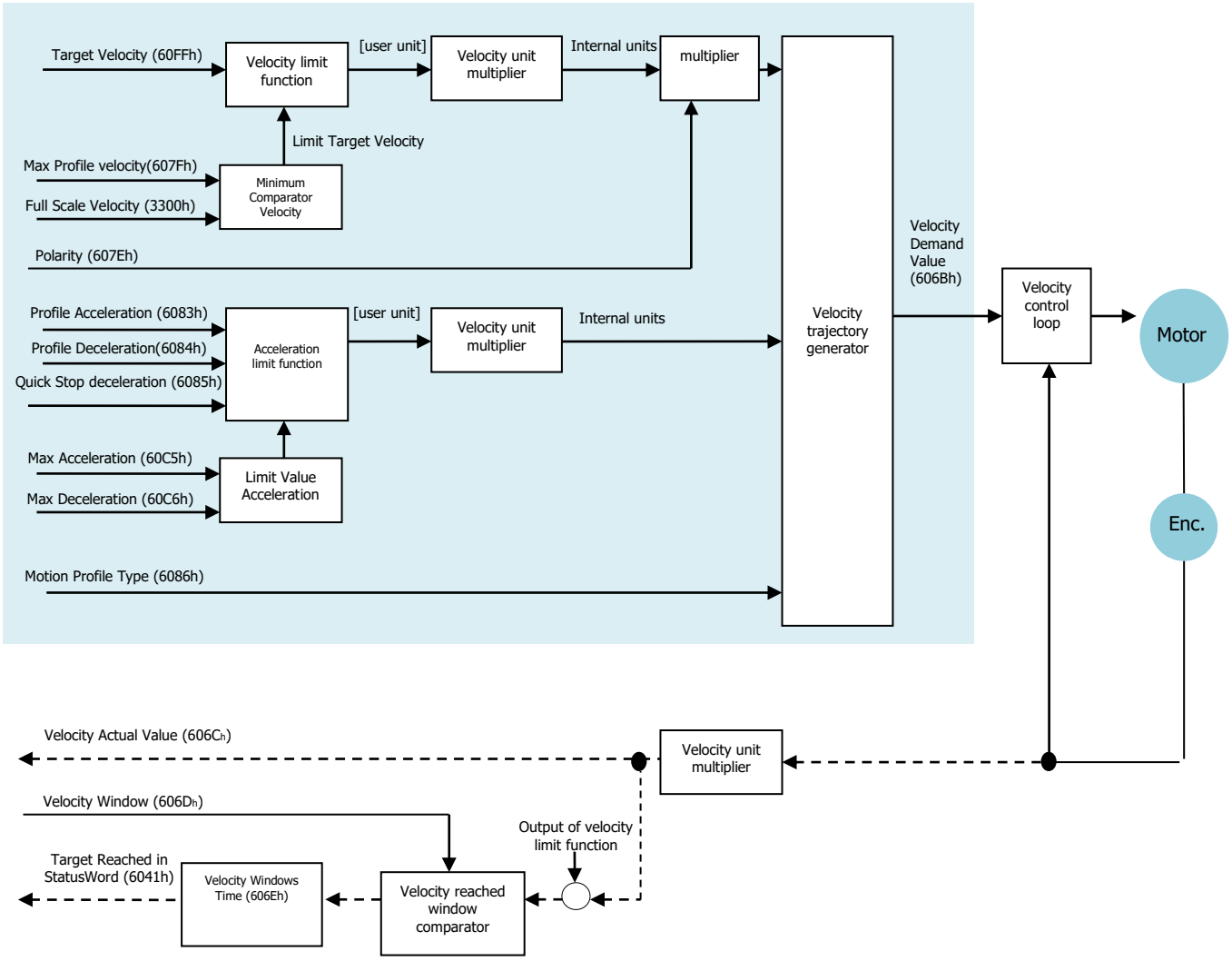


Figure 32 - Profile Velocity Block Diagram

The motion ends when one of the following conditions is met:

- "Target velocity"(60FF$_h$) is set to 0 (in this condition the motor is in torque)
- Stop caused by Halt Bit (8) of "Controlword" (6040$_h$).
- Stop caused by an error (the drive will move in Fault State)
- Stop to exit Operation Enabled State of DSP402 using command "Disable Operation" or "Disable Voltage" or "Quick Stop" in "Controlword"(6040$_h$).
- Stop caused by Safety Condition (STO input)

The result of profile Velocity is in the following bits:

- Object "Velocity actual value" ($606C_h$)
- Object "Velocity Windows " ($606D_h$)→ Target Reached Bit 10 of "Statusword" (6041_h)
- Object "Velocity Threshold"($606F_h$)→ Speed Bit 12 of "Statusword" (6041_h)

Controlword Bits:

The following bits in object controlword (6040_h) have a special function; they are set by Master Controller:

Bit	Value	Definition
Bit 8 = Halt	0_b 1_b	The motion shall be executed or continued Axis shall be stopped according to the halt option code ($605D_h$)

Table 43 - controlword Bit for Velocity Profile

Statusword Bits:

The following bits in object statusword (6041_h) have a special function; they are set by Drive:

Bit	Value	Definition
Bit 10 = Target Reached	0_b	If Halt (bit 8 in controlword) = 0: Target not reached If Halt (bit 8 in controlword) = 1: Axis decelerates
	1_b	If Halt (bit 8 in controlword) = 0: Target reached If Halt (bit 8 in controlword) = 1: Velocity of axis is 0
Bit 12 = Speed	0_b 1_b	Speed is not greater than Velocity threshold Speed is greater than Velocity threshold
Bit 13 = Max Slippage error(*)	0_b 1_b	Maximum slippage not reached Maximum slippage reached

Table 44 - statusword Bit for Velocity Profile

(*) Not managed, It is used only for motor asynchronous

Caution

The torque can be limited with "Positive torque limit value" object $60E0_h$ and "Negative torque limit value" $60E1_h$

OPERATING MODE DESCRIPTION:
In the operating mode Profile Velocity, a movement is made with a desired target velocity.

Procedure:

- Set "Mode of operation" (6060_h) to operating mode Profile Velocity (value 3).
- Set "Profile acceleration" (6083_h) and "Profile deceleration" (6084_h) to the value for the acceleration ramp (user units)
- Set "Target velocity" ($60FF_h$) to the target velocity (user units)
- Set "Controlword" (6040_h) to start the operating mode.

If the power stage is enabled, the new target velocity will become active immediately and the movement will start or set in operating mode with bit halt = 0

Type of Profile:

The type of profile can be set by "Motion Profile Type" (6086_h):

(p402) Linear Ramp: Value "**0**" (DEFAULT)

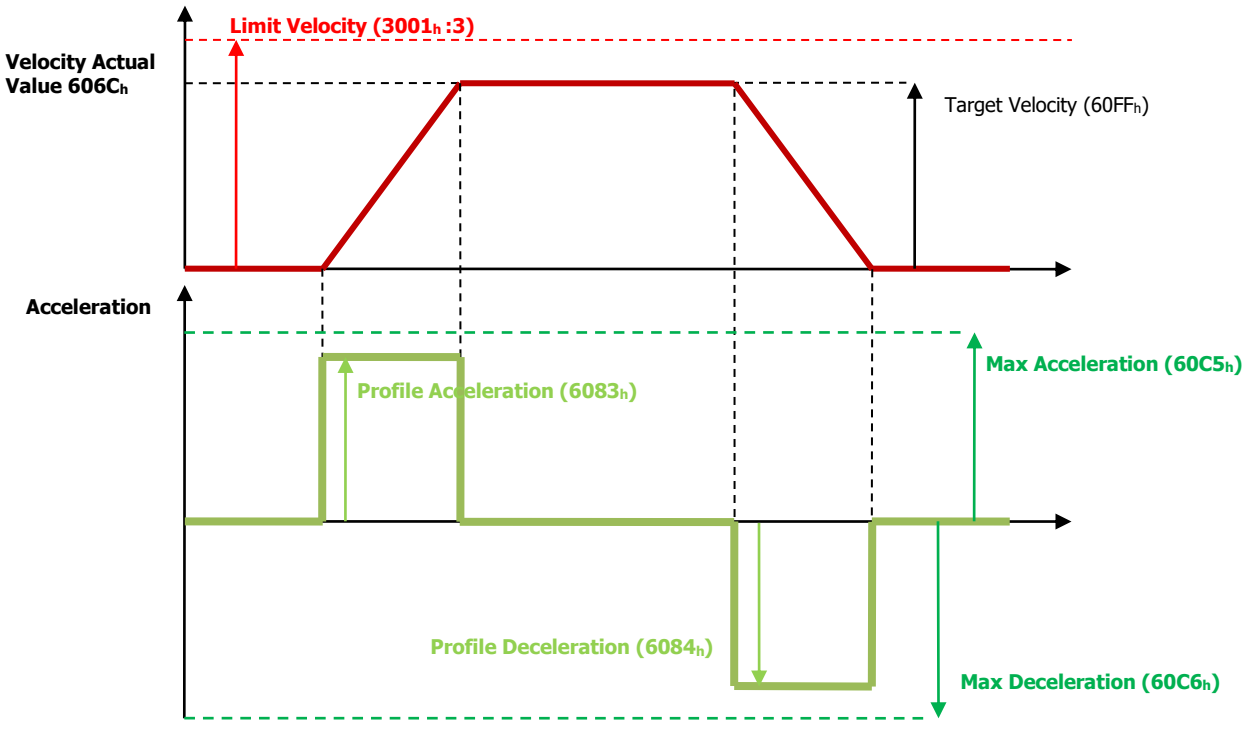


Figure 33 – Type Profile (0) - Trapezoiadal

Manuf. Spec. Immediately: Value "**-1**" (ramps disabled)

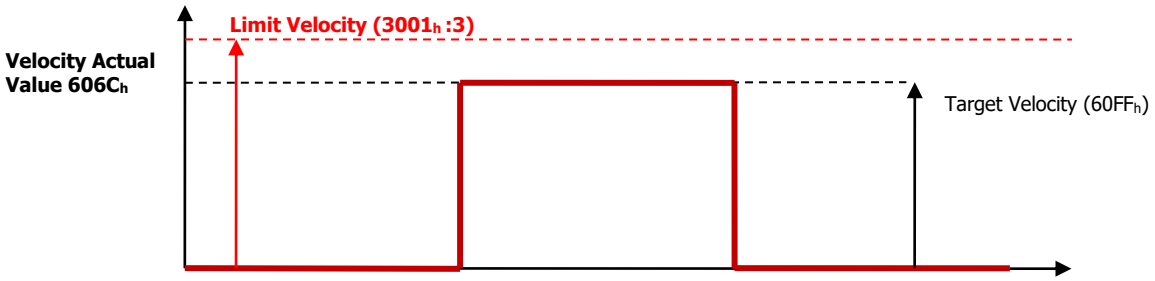


Figure 34 –Type Profile (-1) - Immediately

Manuf. Spec. S-Ramp Jerk-Limited : Value "**-2**"

It means that the S-Ramp is limited by Jerk value calculates in the firmware, if the target velocity (0x60FF) is out of range by jerk calculation the drive answers with "abort code" and it does not accept the new target velocity (to have more information read appendix "SRAMP PROFILE").

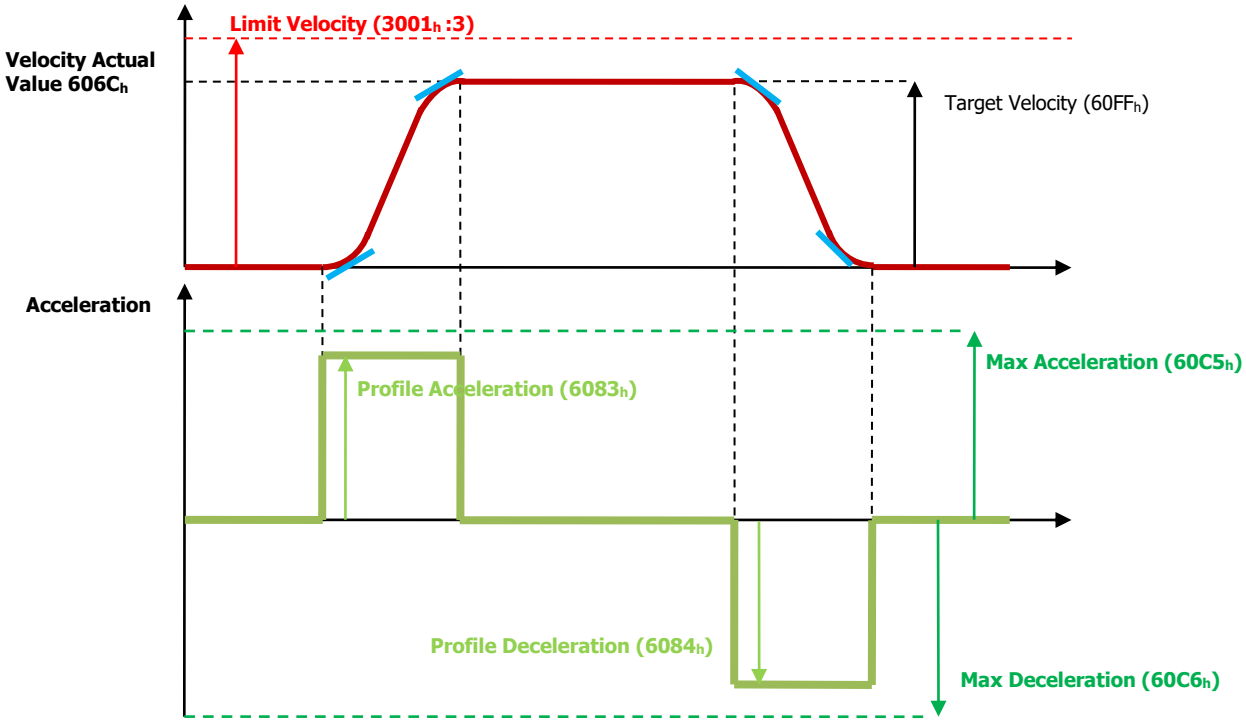


Figure 35 – Type Profile (-2) - S-Ramp Jerk-Limited

Manuf. Spec. S-Ramp Jerk-NoLimited : Value "**-3**"

It means that the S-Ramp is not limited by Jerk (to have more information read appendix "SRAMP PROFILE").

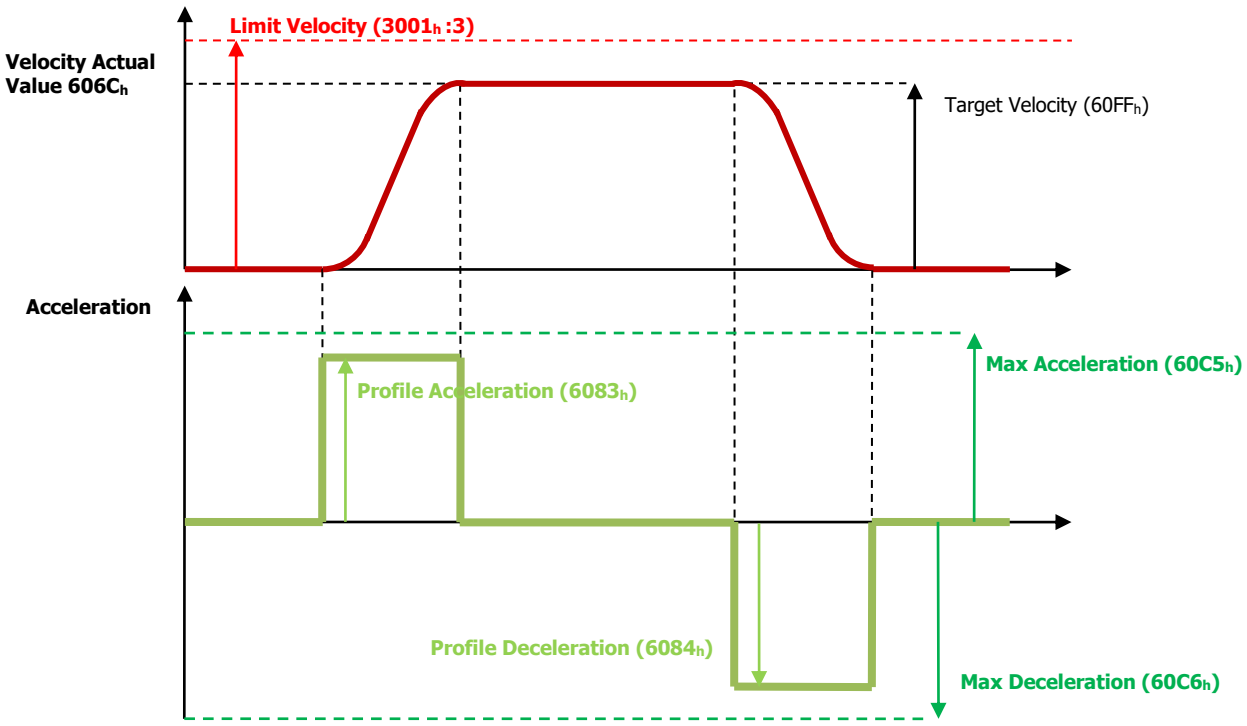


Figure 36 – Type Profile (-3) - S-Ramp Jerk-NoLimited

(p402) Sin² Ramp: Value "**1**".

NOT AVAILABLE

(p402) Jerk-free ramp: Value "**2**".

NOT AVAILABLE

(p402) Jerk-limited ramp: Value "**3**".

NOT AVAILABLE

Output Query:

- Query "Statusword" (6041_h) to get the device status. The value is reset to zero if the operating mode is changed, the power stage is disabled or a Quick Stop is triggered.
- Query "Velocity demand" value ($606B_h$) to get the reference velocity (user units)
- Query "Velocity actual" value ($606C_h$) to get the actual velocity (user units)

Standstill window:

- Query "Velocity window" ($606D_h$) to the value of the velocity window (customer units). It is the step to add ad Target Velocity.

  With the object "Velocity window" ($606D_h$) a tolerance window for the velocity actual value will be defined for comparing the "Velocity Actual" Value ($606C_h$) with the target velocity "Target velocity" ($60FF_h$).

  If the difference is smaller than the "Velocity window" ($606D_h$) for a longer time than specified by the object "Velocity window Time" ($606F_h$) bit 10 "Target Reached" will be set in the object "Statusword" (6041_h).

1. Stop Velocity without Halt Bit:

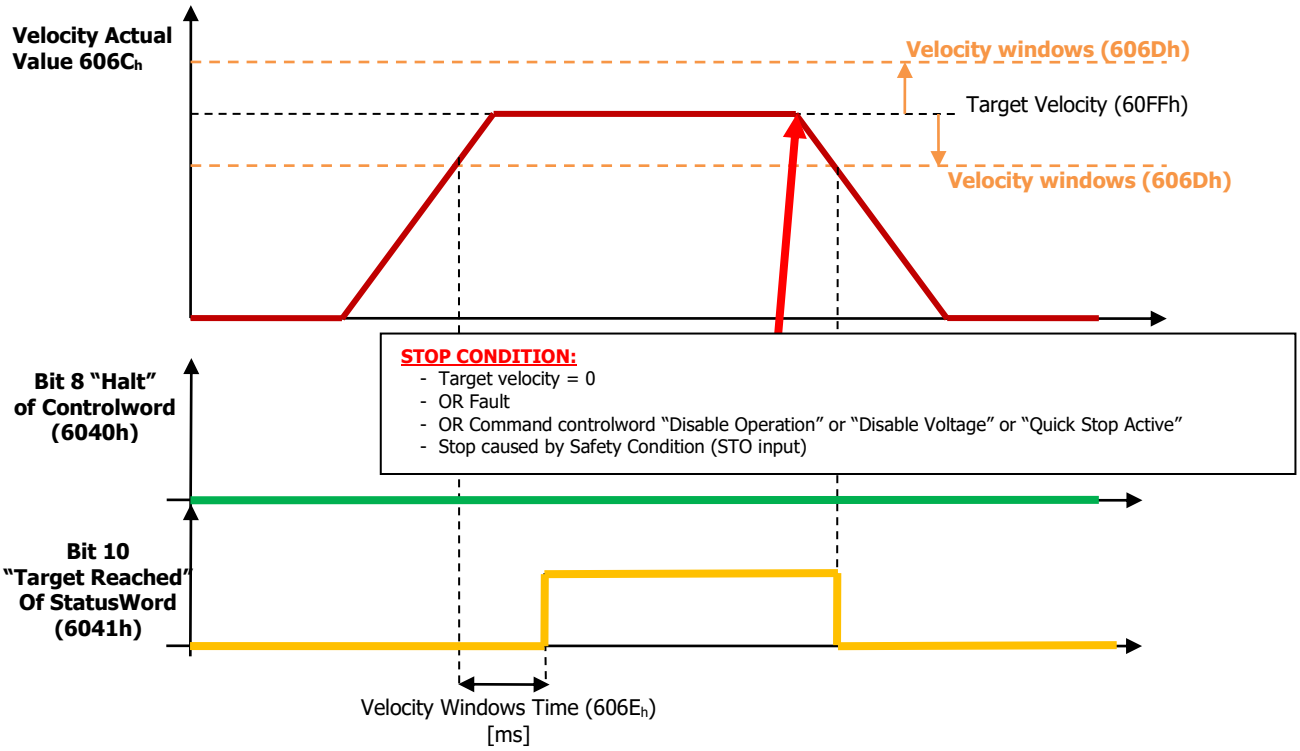


Figure 37 - Velocity Windows without Halt Bit

2. Stop Velocity with Halt Bit = 1

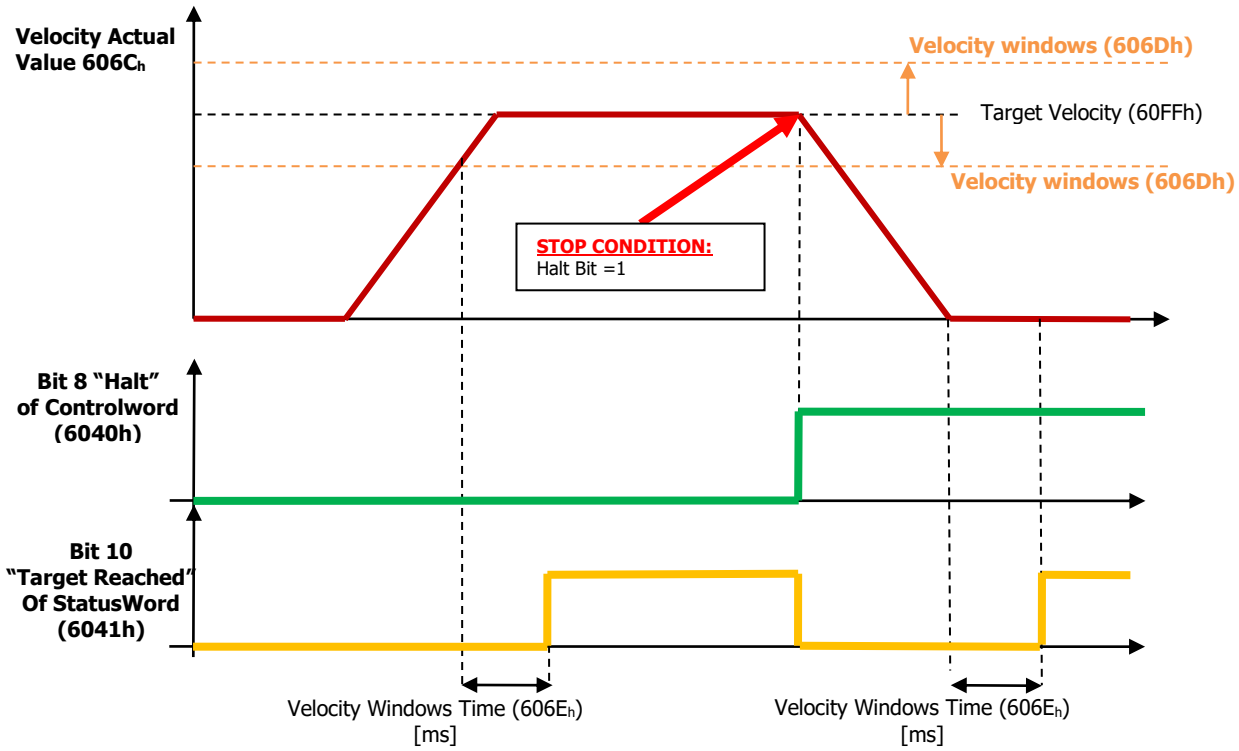


Figure 38 - Velocity Windows with Halt Bit = 1

- Query "Velocity threshold" (606F$_h$) to set the standstill window.

  The object "Velocity threshold" (606F$_h$) determines the velocity underneath the axis is regarded as stationary.

  As soon as the "Velocity Actual" Value (606C$_h$) exceeds the "Velocity threshold" (606F$_h$) longer than "Velocity threshold Time" (6070$_h$) the bit 12 "Speed"vis cleared in the "Statusword" (6041$_h$).

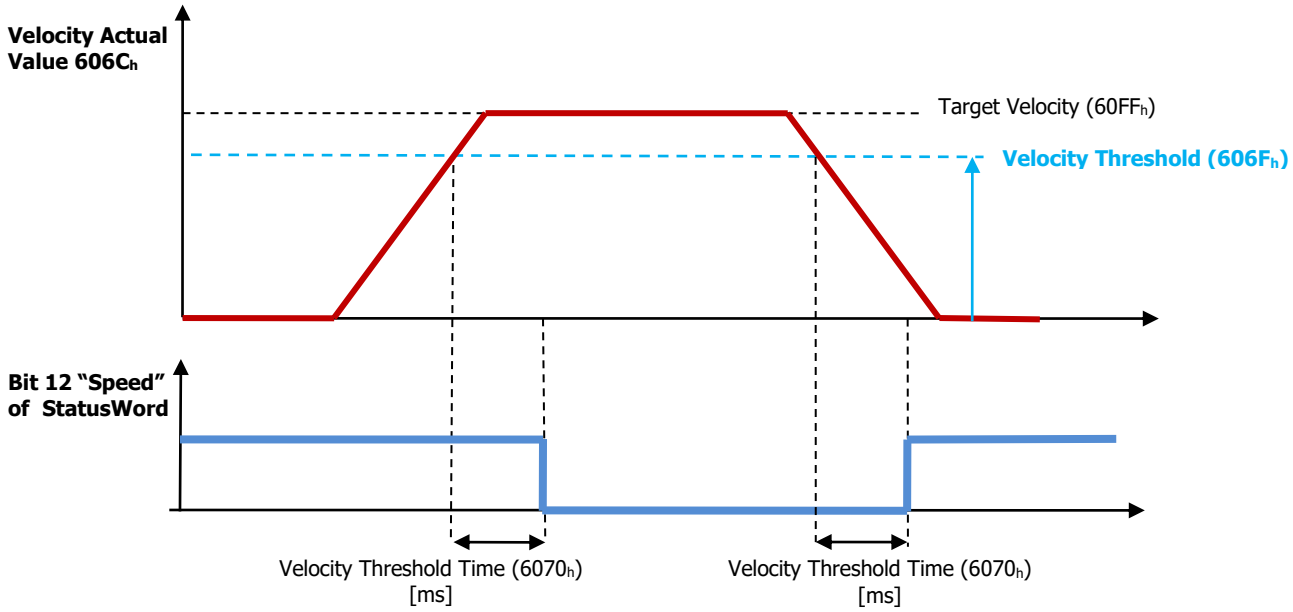


Figure 39 -Velocity threshold

OBJECTS DESCRIPTION FOR VELOCITY PROFILE

Object 60FF$_h$: Target Velocity

This object indicates the configured target velocity and is used as input for the trajectory generator.
Object 60FF$_h$ sets the target velocity when using profile velocity mode.
The drive then accelerates or decelerates to that velocity using the acceleration and deceleration set by objects 6083$_h$ and 6084$_h$.

Object Description:

Index	Object Code	Data Type	Category
60FF$_h$	VAR	INTEGER32	Mandatory

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00$_h$	rw	YES	[-2147483647...2147483647]	Manufacturer Specific	[u.u.]

The drive will sent the follow abort codes:

- 0x06090031 = Value of parameter written too high, the value must be smaller than "Limit Velocity" (3001$_h$:3)
- 0x06040030 = the value is out of range

Caution

Minimum Target Velocity to move the motor is 0,25 rpm

Object 607Fh: Max Profile Velocity

This object indicates the maximal allowed velocity in either direction during a profiled motion.

Object Description:

Index	Object Code	Data Type	Category
607Fh	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00h	rw	no	[1... 2147483647]	Manufacturer Specific	[RPM]

The units are in RPM.

You should program the "Max Profile Velocity" to be smaller or equalthan "Velocity Absolute Maximum Rating" (3001h: 1).

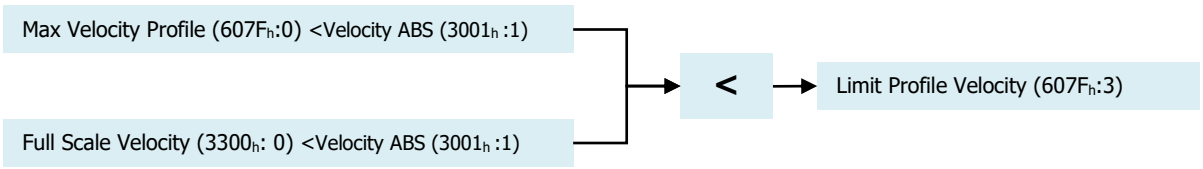
Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B17

The "Max Profile Velocity" (607Fh:0), together "Full Scale Velocity (3300h:0)", defines the limit of Speed. The scheme to set the limit is the follow:



Value The drive will sent the follow abort codes:

- 0x06090031 = Value of parameter written too high, the value must be smaller than "Velocity ABS" (3001h :1)
- 0x06040030 = the value is out of range

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 607Fh
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

This object is used also Profile Position

Object 6086h: Motion profile type

This object shall indicate the configured type of motion profile used to perform a profiled motion.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
6086h	Motion profile type	VAR	INTEGER16	optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00h	Motion profile type	rw	YES	See table	0

Value Definition

Value	Definition	Lafert Servo Drive
-3	Man. Spec: S-Ramp without Jerk	**YES**
-2	Man. Spec: S-Ramp with Jerk Limited	**YES**
-1	Man. Spec: Immediately (ramps disabled)	**YES**
0	p402: Linear Ramp (trapeziodal profile)	**YES (default)**
1	p402: Sin2 ramp	***Not implemented***
2	p402: Jerk-free ramp	***Not implemented***
3	p402: Jerk-limited ramp	***Not implemented***
4 to 32767	reserved	-

The drive will sent the follow abort codes:

- 0x08000024 = No data available (if the value is different value avaliable the drive will send an abord code)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B26

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6086h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

This object is used also Profile Position (future implementation)

Object 6083h: Profile Acceleration

This object indicates the commanded acceleration.

- The range value admissible is [10 ... 319000] rpm/s.
- The profile deceleration must be smaller or equalthan "Max Acceleration" (60C5h :0)

Object Description:

Index	Object Code	Data Type	Category
6083h	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00h	rw	yes	[1...2147483647]	Manufacturer Specific	[u.u.]

It can be converted using Acceleration Factory Group (6097h):

$$Profile\ Acc\ \ [inc/sec^2] = \left(\frac{Factory\ Acc\ Num\ (6097h: 1)}{Factory\ Acc\ Div\ (6097h: 2)}\right) Profile\ Acc\ (6083h)[u.u.]$$

With "numerator" and "Divisor" equal 1, the "Profile Acceleration" (6083h) is in [inc/sec^2] units.

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B13

The drive will sent the follow abort codes:

- 0x06090031 = Value of parameter written too high, the value must be smaller than "Max Acceleration" (60C5h:0)
- 0x06040030 = the value is out of range [10 ... 319000] rpm/s
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

It is possible to change the Acceleration Profile in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E^{2}prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6083h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

This object is used also Profile Position

Object 6084h: Profile Deceleration

This object indicates the commanded deceleration.

- The range value admissible is [10 ... 319000] rpm/s.
- The profile deceleration must be smaller or equalthan "Max Deceleration" (60C6h :0)

Object Description:

Index	Object Code	Data Type	Category
6084h	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00h	rw	yes	[1...2147483647]	Manufacturer Specific	[u.u.]

It can be converted using Acceleration Factory Group (6097h):

$$Profile\ Dec\ \ [inc/sec^2] = \left(\frac{Factory\ Acc\ Num\ (6097h: 1)}{Factory\ Acc\ Div\ (6097h: 2)}\right) Profile\ Dec\ (6084h)[u.u.]$$

With "numerator" and "Divisor" equal 1, the "Profile Deceleration" (6084h) is in [inc/sec^2] units.

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B14

The drive will sent the follow abort codes:

- 0x06090031 = Value of parameter written too high, the value must be smaller than "Max Deceleration" (60C6h:0)
- 0x06040030 = the value is out of range [10 ... 319000] rpm/s
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

It is possible to change the Deceleration Profile in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E^{2}prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6084h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

This object is used also Profile Position

Object 60A3h: Profile Jerk Use

This object shall indicate the configured number of sub-indices used in the profile jerk object (60A4h) for the jerk profile movement.

This object is ONLY read, and the value is 2.

Object Description:

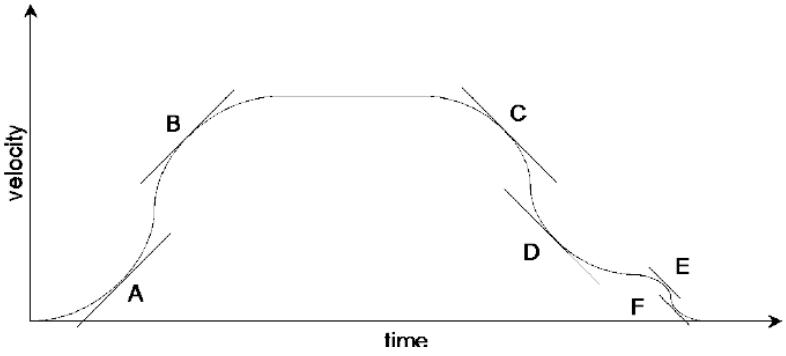
Index	Name EDS	Object Code	Data Type	Category
60A3h	Profile jerk use	VAR	UNSIGNED8	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00h	Profile jerk use	ro	NO	2	2

Object 60A4h: Profile Jerk

This object indicates the configured set of jerk parameters, used during the profile movement.
Figure illustrates the defined jerks (A, B, C, D, E, and F) for the case, the parameters E and F are unequal 0.
The values shall be given in user-defined jerk units.
Table specifies the value assignment to jerks depending of the value of profile jerk use object (60A3h).
The objects 60A4h applies only for the jerk limited ramp.



Value in 60A3h or sub-index 00h of 60A4h if 60A3h is not implemented	Value assignment to jerks					
	A	B	C	D	E	F
01h	01h	01h	01h	01h	–	–
02h	01h	01h	02h	02h	–	–
04h	01h	03h	02h	04h	–	–
06h	01h	03h	02h	04h	05h	06h

The object 0x60A3 has default value number 2, it means that the S-Ramp use the assignments to Jerk:

- For the curve at point A and B the Jerk limitation is the value of the object 60A4h subindex 1
- For the curve at point C and D the Jerk limitation is the value of the object 60A4h subindex 2

Object Description:

Index	Name EDS	Object Code	Data Type	Category
60A4h	Profile Jerk	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default
00h	Profile jerk use	ro	NO	2	2
01h	Profile Jerk 1	rw	NO	[0 ... 2147483647]	20000000
02h	Profile Jerk 2	rw	NO	[0 ... 2147483647]	20000000

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

It is possible to change in run time.

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 60A4h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

information

Referring to "APPENDIX" chapter to know the "SRAMP Profile"

Object 60C5h: Max Acceleration

This object indicates the maximal acceleration. It is used to limit the acceleration to an acceptable value in order to prevent the motor and the moved mechanics from being destroyed.

- The range value admissible is [10 ... 319000] rpm/s.
- The Max Acceleration must be smaller or equal than than "Acceleration ABS" (3001h :2)

Object Description:

Index	Object Code	Data Type	Category
60C5h	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00h	rw	no	[1...2147483647]	Manufacturer Specific	[u.u.]

It can be converted using Acceleration Factory Group (6097h):

$$Max\ Acceleration\ [inc/sec^2] = \left(\frac{Factory\ Acc\ Num\ (6097h{:}\ 1)}{Factory\ Acc\ Div\ (6097h{:}\ 2)}\right) Max\ Acceleration(60C5h)[u.u.]$$

With "numerator" and "Divisor" equal 1, the "Max Acceleration" (60C5h) is in [inc/sec²] units.

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B15

The drive will sent the follow abort codes:

- 0x06090031 = Value of parameter written too high, the value must be smaller than "Acceleration ABS" (3001_h:2)
- 0x06040030 = the value is out of range [10 ... 319000] rpm/s

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $60C5_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

information

This object is used also Profile Position

Object $60C6_h$: Max Deceleration

This object indicates the maximal deceleration. It is used to limit the deceleration to an acceptable value in order to prevent the motor and the moved mechanics from being destroyed.

- The range value admissible is [10 ... 319000] rpm/s.
- The Max Deceleration must be smaller or equal than "Acceleration ABS" (3001_h:2)

Object Description:

Index	Object Code	Data Type	Category
$60C6_h$	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00_h	rw	no	[1...2147483647]	Manufacturer Specific	[u.u.]

It can be converted using Acceleration Factory Group (6097_h):

$$Max\ Deceleration\ [inc/sec^2] = \left(\frac{Factory\ Acc\ Num\ (6097h:1)}{Factory\ Acc\ Div\ (6097h:2)}\right) Max\ Deceleration(60C6h)[u.u.]$$

With "numerator" and "Divisor" equal 1, the "Max Deceleration" ($60C6_h$) is in [inc/sec²] units.

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B16

The drive will sent the follow abort codes:

- 0x06090031 = Value of parameter written too high, the value must be smaller than "Acceleration ABS" (3001_h:2)

- 0x06040030 = the value is out of range [10 ... 319000] rpm/s

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $60C6_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

information

This object is used also Profile Position

Object $606B_h$: Velocity Demand Value

This object provides the output value of the trajectory generator.

Object Description:

Index	Object Code	Data Type	Category
$606B_h$	VAR	INTEGER32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00_h	ro	YES	[-2147483648 ...2147483648]	Manufacturer Specific	[u.u.]

Object $606C_h$: Velocity Actual Value

This object provides the actual velocity value derived either from the velocity sensor or the position sensor.

Object Description:

Index	Object Code	Data Type	Category
$606C_h$	VAR	INTEGER32	Conditional: mandatory if pv or csv is supported

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00_h	ro	YES (default)	[-2147483647 ...2147483647]	Manufacturer Specific	[u.u.]

It can be filtered modifying the object Filter Parameters (3005_h: 6), it is a one pole (default is 50Hz).

Object $606D_h$: Velocity Window

This object indicates the velocity window.

Object Description:

Index	Object Code	Data Type	Category
$606D_h$	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00_h	rw	YES (default)	[1...65535]	Manufacturer Specific	[u.u.]

It can be converted using Acceleration Factory Group (6096_h):

$$Velocity\ Window\ [inc/sec] = \left(\frac{Factory\ Vel\ Num\ (6096h:1)}{Factory\ Vel\ Div\ (6096h:2)}\right) Velocity\ Window\ \ (606Dh)[u.u.]$$

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B1B

The drive will sent the follow abort codes:

- 0x05040001 = command is invalid because the value is 0
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $606D_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object $606E_h$: Velocity Window Time

This object indicates the velocity window time.

Object Description:

Index	Object Code	Data Type	Category
$606E_h$	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00_h	rw	no	[1...65535]	Manufacturer Specific	[ms]

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B1C.

The drive will sent the follow abort codes:

- 0x06090032 = Value of parameter written too low
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $606E_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 606Fh: Velocity Threshold

This object indicates the velocity threshold.

Object Description:

Index	Object Code	Data Type	Category
$606F_h$	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00_h	rw	no	[0 ...65535]	Man. Specific	[u.u.]

It can be converted using Acceleration Factory Group (6096_h):

$$Velocity\ Threshold\ [inc/sec] = \left(\frac{Factory\ Vel\ Num\ (6096h:1)}{Factory\ Vel\ Div\ (6096h:2)}\right) Velocity\ Threshold\ (606Fh)[u.u.]$$

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B1D

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object $606F_h$
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 6070h: Velocity Threshold Time

This object indicates the velocity threshold time.

Object Description:

Index	Object Code	Data Type	Category
6070h	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default Value	Unit
00h	rw	no	[1...65535]	Manufacturer Specific	[ms]

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B1E.

The drive will sent the follow abort codes:

- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)
- 0x06090032 = Value of parameter written too low

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- the drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6070h
- Write signature "SAVE" in Store Parameters 1010h object (pay attention on the processing time)
- NMT Reset Node

PROFILE TORQUE MODE (4)

In the profile Torque operating mode (PT), the motor executes a movement according to a target torque or current sent by the master controller. The current regulator (torque control) is specified a current proportional to the target torque.

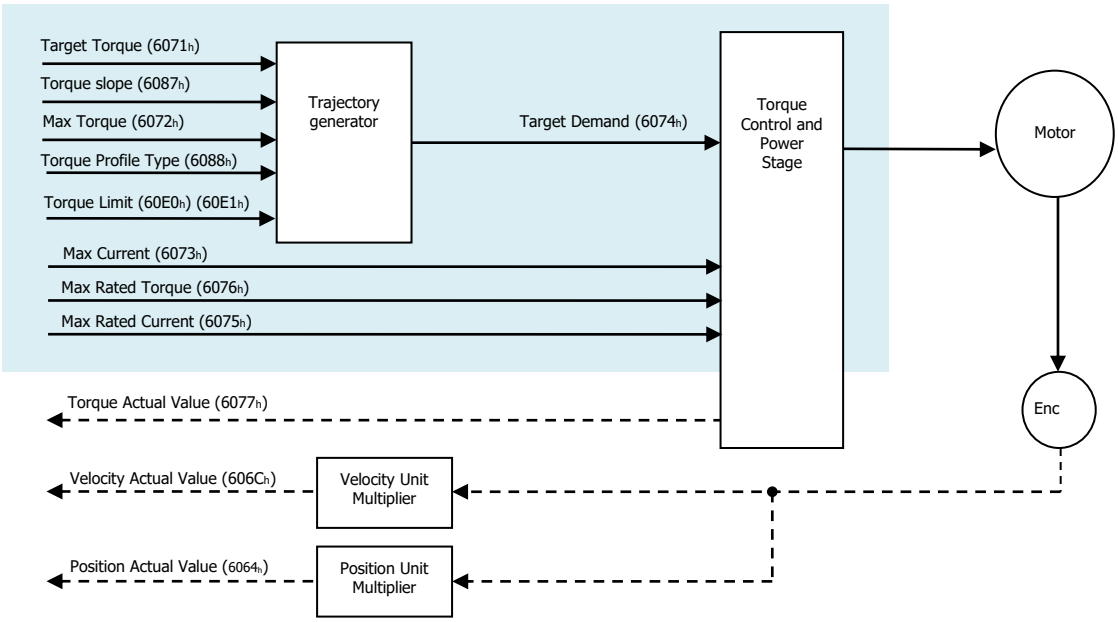


Figure 40 – Torque Profile Block Diagram

Prerequisites for the drive to be operated in Torque Profile Mode:

- The Torque Profile Mode must be set in the "Mode of Operation" (6060_h) parameter (value "4"). The mode operation can be verified using "Mode of Operation Display" (6061_h) which is updated when the current operation is accepted.
- The drive must be in *"Operation Enabled"* state of state machine of DSP402, verify it with the object "Statusword" (6041_h). To move the state machine it uses the object "controlword" (6040_h)
- Target Torque and parameters of torque must be set correctly.

The target Torque is set with object "Target Torque" (6071_h) of the object dictionary.

OBJETCS LIST:

The objects associated to move the drive in Torque Profile Mode are the following:

Index	Sub Index	Name	READ / WRITE	M/O	Data Type	PDO	Available
0x603F	0	Error Code	RO	O	U16	-	x
0x6040	0	Control Word	R/W	M	U16	RPDO	x
0x6041	0	Status Word	RO	M	U16	TPDO	x
0x6060	0	Modes of Operation	R/W	M	I8	RPDO	x
0x6061	0	Modes of Operation Display	RO	M	I8	TPDO	x
0x6071	0	Target torque	R/W	M	I16	RPDO	x
0x6087	0	Torque slope	R/W	O	U32	-	x
0x6072	0	Max torque	R/W	O	U16	-	
0x6073	0	Max current	R/W	O	U16	-	x
0x6076	0	Motor rated torque	R/W	O	U32	-	
0x6075	0	Motor rated current	R/W	O	U32	-	x
0x6077	0	Torque actual value	RO	O	I16	TPDO	x
0x6078	0	Current actual value	RO	O	I16	TPDO	x
0x6079	0	DC link circuit voltage	RO	O	U32	TPDO	x
0x60E0	0	Positive torque limit value	R/W	O	U16	-	x
0x60E1	0	Negative torque limit value	R/W	O	U16	-	x
0x6074	0	Torque demand	RO	O	I16	-	
0x6088	0	Torque profile type	R/W	O	I16	-	x

Table 45 - Torque Profile's objects

The motion ends when one of the following conditions is met:

- "Target Torque"(6071_h) is set to 0 (in this condition the motor is in torque equal 0)
- Stop caused by Halt Bit (8) of "Controlword" (6040_h).
- Stop caused by an error (the drive will move in Fault State)
- Stop to exit Operation Enabled State of DSP402 using command bit "Disable Operation" or "Disable Voltage" or "Quick Stop" in "Controlword" (6040_h).
- Stop caused by Safety Condition (STO input)

The result of profile torque is in the following bits:

- Object "Torque actual value" (6077_h)
- Object "Current actual value" (6078_h)
- Target Reached Bit 10 of "Statusword" (6041_h)

Controlword Bits:

The following bits in object controlword (6040_h) have a special function; they are set by Master Controller:

Bit	Value	Definition
Bit 8 = Halt	0_b 1_b	The motion shall be executed or continued Axis shall be stopped according to the halt option code ($605D_h$) (*)

Table 46 - controlword Bit for Torque Profile

(*) option code $605D_h$ is not implemented

Statusword Bits:

The following bits in object **6041_h** (statusword) have a special function; they are set by Drive:

Bit	Value	Definition
Bit 10 = Target Reached	0_b	If Halt (bit 8 in controlword) = 0: Target not reached If Halt (bit 8 in controlword) = 1: Axis decelerates
	1_b	If Halt (bit 8 in controlword) = 0: Target reached If Halt (bit 8 in controlword) = 1: Velocity of axis is 0

Table 47 - statusword Bit for Torque Profile

Caution

The "Dynamic Stop controlled" is not available in Torque profile.

If the Dynamic Stop feature is set (see object 3007:1) then the drive is controlled by torque (or current) without the Dynamic Stop controlled.

OPERATING MODE DESCRIPTION:

In the operating Torque Profile Mode a movement is made with a desired target torque.

Procedure:

- Set "Mode of operation" (6060_h) to operating mode Profile Velocity (value 4).
- Set "Motor Rated Current" (6075_h) to a value according to motor specifications (unit mArms) (this value is saved in e^2prom, follow the procedure to save the new value in e^2prom)
- Set "Torque profile Type" (6078h) to select the type of torque profile
- If the "Torque profile Type" is a Linear Ramp (Trapezoidal profile) set the rate of change of torque object "Torque slope" (6087_h)
- Set "Controlword" (6040_h) to activate the operating mode and enable movement. When the operating mode is started, the target torque is set to zero.
- Set "Target Torque" (6071_h) to the set point torque

If the power stage is enabled, the new target velocity will become active immediately and the movement will start or set in operating mode with bit halt = 0.

information

The torque can be limited in percent value with $60E0_h$ and $60E1_h$ object.

Type of Profile:

The type of profile can be set by "Torque Profile Type" (6088_h):

(p402) Linear Ramp: Value "**0**".

It means that if "Motor Rated Current" (6075_h) = 12500 mArms and "Torque Slope" (6087_h) = 1000 then the motor will go until 12,5Arms in 1seconds; to go to 1,25Arms it will run for 100 ms.

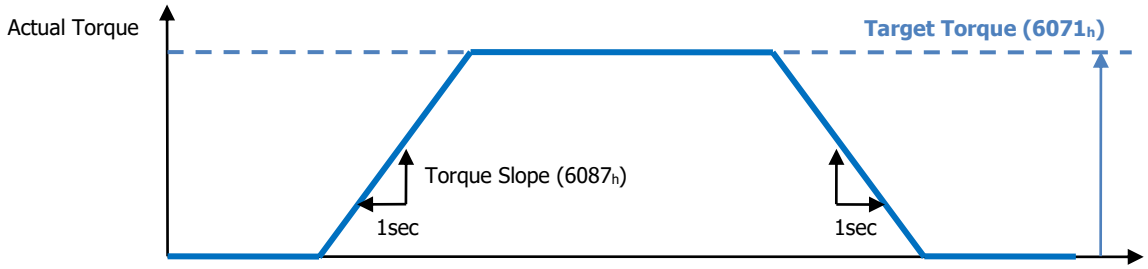


Figure 41 – Diagram Torque Trapezoidal Type

(Manuf.Spec.) Immediately: Value "**-1**".

It means that the drive will go to the target torque instantly. The new target is processed immediately.

(p402) Sin² Ramp: Value "**1**".

NOT AVAILABLE

Output Query:

- Query "Statusword" (6041_h) to get the device status. The value is reset to zero if the operating mode is changed, the power stage is disabled or a Quick Stop is triggered.
- Query "Torque Actual Value" value (6077_h) to get the reference instantaneous current in the drive motor.
- Query "Current Actual Value" value (6078_h) to get the reference filtered current in the drive motor.
- Query "Target Reached" value (bit 10) of object "Statusword" (6041_h).

1. Torque Profile Mode without Halt Bit:

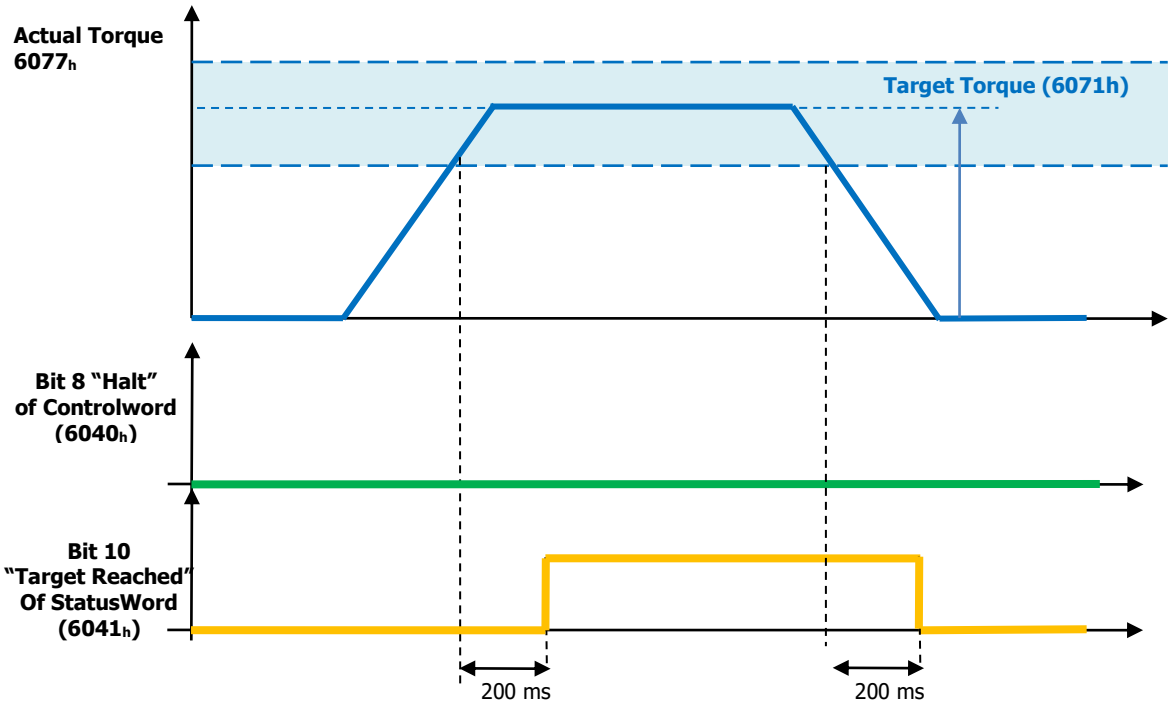


Figure 42 – Torque Reached Bit without Halt Bit

2. Torque Profile Mode with Halt Bit = 1

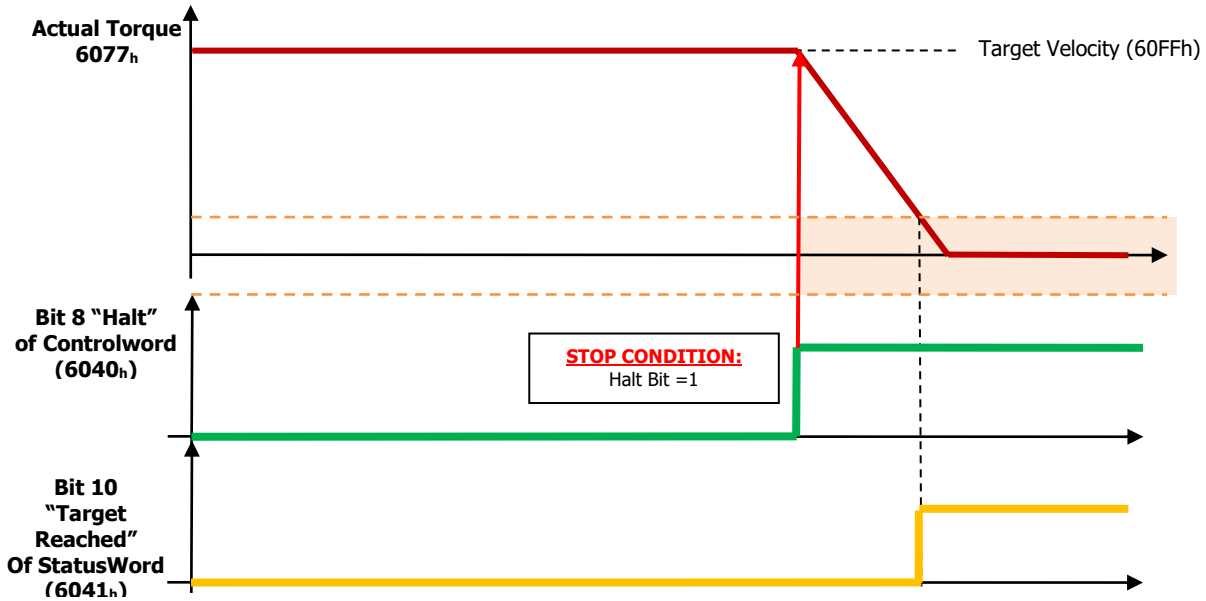


Figure 43 - Velocity Windows with Halt Bit = 1

OBJECTSDESCRIPTIONFOR TORQUE PROFILE

Object 6071h - Target Torque

This object shall indicate the configured input value for the torque controller in profile torque mode. The value shall be given per thousand of rated current.
Target Torque is the torque set-point, which is given here as the torque producing current I_q.

Object Description:

Index	Object Code	Data Type	Category
6071h	VAR	INTEGER16	Mandatory

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	rw	yes	[1 ...65535]	ManufSpecific	[Rated Current/1000]

Example:

If a torque that is relative to current of 2Arms is needed and the object "Motor Rated Current" (6075h) is 12500 mArms, then:

$$TargetTorque[6071h] = \frac{2000\,mArms\ *\ 1000}{12500\,mArms} = 160$$

This number means 16.0% of Motor Rated Current

The drive will send the follow abort code:

- 0x06090031 = Value of parameter written too high

Object 6075h–Motor Rated Current

This object shall indicate the configured motor rated current. It is taken from the motor's name-plate. Depending on the motor and drive technology, this current is DC, peak or r.m.s. (root-mean-square) current. All relative current data refers to this value. The value shall be given in mArms.

Object Description:

Index	Object Code	Data Type	Category
6075h	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	rw	no	[1 ... 2147483647]	Manuf. Specific	[mArms]

The "Motor Rated Current" (6075h: 0) must be lower or equal than Rated Current of Motor, it is defined in the object "Current Parameters" (3003h) sub-index 3 (named "Rated Current Motor") or by Motor Parameters Datasheet.

Motor RatedCurrent (6075h: 0) [mArms] ≤ Nominal RatedCurrent (3003h: 3) (DatasheetParameter) [Arms/100]

Example:

If Nominal Rated Current (3003h: 3) is 125 (it means that is 12,5 Arms), then the "Motor Rated Current" (6075h: 0) must be ≤ 12500 mArms.

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06090031 = Value of parameter written too high (because it must be smaller than 0x3003: 3)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B1F

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6075_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 6073_h–Max Current

This object shall indicate the configured maximum permissible torque creating current in the motor. The value shall be given per thousand of rated current.

Object Description:

Index	Object Code	Data Type	Category
6073_h	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00_h	rw	no	[1 ... 32767]	Manuf.Specific	[Rated Current/1000]

The value "Max Current" (6073_h:0) converted in mArms must be lower or equal than Peak Current [mArms]. The Peak current is defined in the object "Current Parameters" (3003_h) sub-index 2 (named "Peak Current Motor") or by Motor Parameters Datasheet.

MaxCurrent (6073_h: 0) → [mArms] ≤ Peak Current (3003_h: 2) - DatasheetParameter [Arms/100]

Example:

If PeakCurrent (3003_h: 2) is 410 (it means that is 41 Arms), and the "Motor Rated Current" is 12500 mArms then the "Max Current" (6073_h: 0) must be:

$$Max\ Current[6073h] = \frac{(410 * 100)_{(Arms/100)} * 1000}{12500\ (mArms)} = 3280$$

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x06090031 = Value of parameter written too high (the value converted must be smaller than 0x3003:2)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B22

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6073_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 6087_h - Torque slope

This object shall indicate the configured rate of change of torque. The value shall be given inunits of per thousand of rated torque per second.

Object Description:

Index	Object Code	Data Type	Category
6087_h	VAR	UNSIGNED32	Mandatory IF

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00_h	rw	no	[1 ... 2147483647]	Manuf. Specific	[(Rated Current/1000)/s]

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B25

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6087_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 6088_h - Torque profile type

This object shall indicate the configured type of profile used to perform a torque change.

Object Description:

Index	Object Code	Data Type	Category
6088_h	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00_h	rw	no	[-32767 ...32767]	Manuf. Specific	-

Value definition

Value	Definition	Lafert Servo Drive
-1	Immediately	YES
0	Linear ramp (trapezoidal profile) (default)	YES Default
1	sin2 ramp	*(not available)*

The drive will sent the follow abort codes:

- 0x08000024 = No data available (if the value is the type not available)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B18

This object can be changed, to use the new value it must be saved in the memory of e2prom

E²prom Store

- The drive must be supplied with VDC-Link
- The drive mustn't be in "Operational enabled" or "Quick Stop Active" state
- Write the new value in SDO object 6088_h
- Write signature "SAVE" in Store Parameters 1010_h object (pay attention on the processing time)
- NMT Reset Node

Object 6074h - Torque demand

This object shall provide the output value of the trajectory generator. The value shall be givenper thousand of rated current.

Object Description:

Index	Object Code	Data Type	Category
6074h	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	ro	no	[-32767 ...32767]	Manuf. Specific	[Rated Current/1000]

Object 6077h - Torque Actual Value

This object shall provide the actual value of the torque. It shall correspond to the instantaneous torque in the motor. The value shall be given per thousand of rated current.

Object Description:

Index	Object Code	Data Type	Category
6077h	VAR	INTEGER16	Mandatory

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	ro	si	[-32767 ...32767]	Manuf. Specific	[Rated Current/1000]

Object 6078h - Torque Actual Current

This object shall provide the actual value of the current. It shall correspond to the current in the motor. The value shall be given per thousand of rated current. This value is filtered.

Object Description:

Index	Object Code	Data Type	Category
6078h	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	ro	yes	[-32767 ...32767]	Manuf. Specific	[Rated Current/1000]

Object 6079h–DC Link circuit Voltage

This object shall provide the instantaneous DC link current voltage at the drive device. The value shall be given in mV.

Object Description:

Index	Object Code	Data Type	Category
6079h	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	ro	yes	[0 ... 4294967296]	Manuf. Specific	[mV]

information

This object can be read for all profiles

Object 60E0h– Positive torque limit value

This object shall indicate the configured maximum positive torque in the motor. The value shall be given percent of maximum peak current. Positive torque takes effect in the case of motive operation is positive velocity or regenerative operation is negative velocity.

Object Description:

Index	Object Code	Data Type	Category
60E0h	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	rw	yes	[0 ... 100]	100	[%]

information

This object is used also Profile Velocity

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B23

This object can't be saved in e²prom memory but it can be changed in RAM

Object 60E1h– Negative torque limit value

This object shall indicate the configured maximum negative torque in the motor. The value shall be given percent of maximum peak current. Negative torque takes effect in the case of motive operation is negative velocity or regenerative operation is positive velocity.

Object Description:

Index	Object Code	Data Type	Category
60E1h	VAR	UNSIGNED16	Optional

Entry Description:

Sub-Index	Access	PDO mapping	Value Range	Default	Unit
00h	rw	yes	[0 ... 100]	100	[%]

information

This object is used also Profile Velocity

The drive will sent the follow abort codes:

- 0x06040030 = the value is out of range (see table Entry Description)
- 0x08000002 = the written is disabled by manufacturer (it is a option defined by application)

Caution

This object is controlled by some limits of drive. If the value is not correct the drive sends an Abort Code.

If the value is not correct during initialization the drive sends an error messages with Emergency Protocol.

See Error Code 0x8B24

This object can't be saved in e2prom memory but it can be changed in RAM

PROFILE HOMING MODE (6) (not available)

ANALOG MODE

In this operation mode the Drive can be piloted with an analog reference. To use this configuration contact the manufacturer.

The analog reference can be:

- -10V ... +10V (default)
- 0V ... +10V
- -5V ... +5V
- 0V ... +5V
- Threshold ... 10V
- Threshold ... 5V

Giving voltage on +VREF and –VREF it is possible to supply to the drive the speed set point.

The motor speed will depend from the reference voltage given on +VREF and –VREF and from the maximum speed available on the drive.
The Speed Set Point is proportional to the voltage supplied on the concerned input.
Giving a +VREF voltage the motor will set to the maximum speed in clockwise rotation, motor front, while giving a -VREF the motor will set to the maximum speed in counter clockwise rotation, motor front.

Example:

Maximum Speed Configured =Max Speed r.p.m. :

- Input Voltage = +10V →Rotation speed (clockwise) = + Max Speed r.p.m.
- Input Voltage = -10V →Rotation speed (counter clockwise) = - Max Speed r.p.m
- Input Voltage = +5V →Rotation speed (clockwise) = + 1/2 Max Speed r.p.m
- Input Voltage = -5V →Rotation speed (counter clockwise) = - 1/2 Max Speed r.p.m

To move the motor is necessary that the digital Inputs configured as "RUN" and "STOP". See Digital I/O chapter what is the digital Input configuration.

Variable Monitoring

In analog Mode it's possible to monitor a list of variables on drive by CANOpen, connecting a CAN Interface and using Lafert Drive SW:

- Object 2002h: Drive Status Mode
- Object 2003h: Warning Flag
- Object 2004h: State Lafert Servo Drive Machine
- Object 2030h: Temperature Drive
- Object 2031h: Temperature Motor
- Object 2032h: Temperature Heat Sink

- Object 2041h: Voltage Bus
- Object 2050h: Torque Current
- Object 2053h: Velocity Filtered
- Object 2054h: Energy i^2t
- Object 2060h: Impulse
- Object 2070h: Current Torque Windowed
- Object 2080h: STO Statusword
- Object 3020h: Function Digital Input
- Object 3021h: Digital Input 1
- Object 3022h: Digital Input 2
- Object 3023h: Digital Input 3
- Object 3024h: Digital Input 4
- Object 30320h: Function Digital Output
- Object 3031h: Digital Output 1
- Object 3032h: Digital Output 2
- Object 3033h: Digital Output 3
- Object 3034h: Digital Output 4
- Object 4000h: Safety State
- Object 4100h: State Communication CAN
- Object 4101h: Counter Communication CAN
- Object 4102h: Settings Communication CAN
- Object 4103h: Error Communication CAN
- Object 6041h: statusword
- Object 603Fh: Error Code
- Object 6064h: Position Actual Value
- Object 6077h: torque Actual Value
- Object 606Ch: Velocity Actual

OTHER PROTOCOL

The drive can be controlled by other protocols. To have more informations contact the manufacturer.

7. | CANOPEN OBJECT LIST

INDEX	SUB.	DESCRIPTION	CODE	TYPE	O/M	ATTR.	ARGUMENT
		STANDARD OBJECTS DS301					
1000 h	0	Device Type	COST	UINT32	M	RO	Settings
1001 h	0	Error Register	VAR	UINT32	O	RO	Settings
1002 h	0	Manufacturer Status Register	VAR	UINT32	O	RO	Settings
1003 h	0	Pre-Defined Error Field	ARRAY	UINT32	M	RO	Diagnostic
	1	History Error Field		UINT32	M	RO	Diagnostic
	2	History Error Field		UINT32	O	RO	Diagnostic
	3	History Error Field		UINT32	O	RO	Diagnostic
	4	History Error Field		UINT32	O	RO	Diagnostic
	5	History Error Field		UINT32	O	RO	Diagnostic
	6	History Error Field		UINT32	O	RO	Diagnostic
	7	History Error Field		UINT32	O	RO	Diagnostic
	8	History Error Field		UINT32	O	RO	Diagnostic
	9	History Error Field		UINT32	O	RO	Diagnostic
	10	History Error Field		UINT32	O	RO	Diagnostic
	11	History Error Field		UINT32	O	RO	Diagnostic
	12	History Error Field		UINT32	O	RO	Diagnostic
	13	History Error Field		UINT32	O	RO	Diagnostic
	14	History Error Field		UINT32	O	RO	Diagnostic
	15	History Error Field		UINT32	O	RO	Diagnostic
1005 h	0	Cob-ID Sync	VAR	UINT32		R/W	Settings
1008 h	0	Manufacturer Device Name	VAR	STRING	M	RO	Communication
1009 h	0	Manufacturer Hardware Version	VAR	STRING	M	RO	Communication
100A h	0	Manufacturer Software Version	VAR	STRING	M	RO	Communication
100C h	0	Guard Time	VAR	UINT16	O	R/W	Settings
100D h	0	LifeTime Factor	VAR	UINT8	O	R/W	Settings
1010 h	0	Store Parameter Fields	ARRAY	UINT32	O	R/W	Memory Parameters
	1	Save all Parameters			M	R/W	Memory Parameters
	2	Save Communication Parameters			O	R/W	Memory Parameters
	3	Save Application Parameters			O	R/W	Memory Parameters
	4	Save Manufacturer Parameters			O	R/W	Memory Parameters
	5	reserved			O	R/W	reserved
1011 h	0	Restore Default Parameter	ARRAY	UINT32	O	R/W	Memory Parameters
	1	Restore all Default Parameters			O	R/W	Memory Parameters
	2	Restore Communication Default Parameters			O	R/W	not available
	3	Restore Application Default Parameters			O	R/W	not available
	4	Restore Manufacturer Default Parameters			O	R/W	not available
	5	reserved			O	R/W	reserved

1014 h	0	Cob-ID Emergency Message	VAR	UINT32	O	RO	not available
1017 h	0	Producer HeartBeat Time	VAR	UINT16	M	R/W	Settings
1018 h	0	Identity Object	RECORD	UINT32	M	RO	-
	1	Vendor Id			M	RO	Settings
	2	Product Code			O	RO	not available
	3	Revision number			O	RO	not available
	4	Serial number			O	RO	not available
1029 h	0	Error Behaviour	ARRAY	UINT8	O	RO	not available
	1	Communication Error			O	R/W	not available
1200 h	0	Server SDO Parameter 1			O	R/W	Settings
1280 h	0	Client SDO Parameter 1			O	R/W	Settings
1400 h	0	Receive PDO Communication Parameter 1	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1401 h	0	Receive PDO Communication Parameter 2	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1402 h	0	Receive PDO Communication Parameter 3	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1403 h	0	Receive PDO Communication Parameter 4	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1600 h	0	Receive PDO Mapping Parameter 1	RECORD	UINT8	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings

1601 h	0	Receive PDO Mapping Parameter 2	RECORD	UINT32	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
1602 h	0	Receive PDO Mapping Parameter 3	RECORD	UINT32	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
1603 h	0	Receive PDO Mapping Parameter 4	RECORD	UINT8	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
1800 h	0	Transmit PDO Communication Parameter 1	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1801 h	0	Transmit PDO Communication Parameter 2	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1802 h	0	Transmit PDO Communication Parameter 3	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings

	5	Event Timer		UINT16	O	R/W	Settings
1803 h	0	Transmit PDO Communication Parameter 4	RECORD	UINT8	M	R/W	Settings
	1	COB-ID		UINT32	M	R/W	Settings
	2	Transmission Type		UINT8	M	R/W	Settings
	3	Inhibit Time		UINT16	O	R/W	Settings
	4	Compatibility Entry		UINT8	O	R/W	Settings
	5	Event Timer		UINT16	O	R/W	Settings
1A00 h	0	Transmit PDO Mapping Parameter 1	RECORD	UINT8	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
1A01 h	0	Transmit PDO Mapping Parameter 2	RECORD	UINT8	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
1A02 h	0	Transmit PDO Mapping Parameter 3	RECORD	UINT8	M	R/W	Settings
	1			UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
1A03 h	0	Transmit PDO Mapping Parameter 4		UINT32	M	R/W	Settings
	1	Mapping Entry 1		UINT32	M	R/W	Settings
	2	Mapping Entry 2		UINT32	M	R/W	Settings
	3	Mapping Entry 3		UINT32	M	R/W	Settings
	4	Mapping Entry 4		UINT32	M	R/W	Settings
	5	Mapping Entry 5		UINT32	M	R/W	Settings
	6	Mapping Entry 6		UINT32	M	R/W	Settings
	7	Mapping Entry 7		UINT32	M	R/W	Settings
	8	Mapping Entry 8		UINT32	M	R/W	Settings
MANUFACTURER OBJECT							

2000 h	0	ID Node	VAR	UINT8	M	R/W	Settings
2001 h	0	CAN Baud Rate	VAR	UINT16	M	R/W	Settings
2002 h	0	Drive Status	VAR	INT16	O	RO	Monitor
2003 h	0	Warning Flag	VAR	UINT32	O	RO	Monitor
2004 h	0	State Lafert Servo Drive Machine	VAR	INT16	O	RO	Monitor
2005 h	0	**Protocol Settings**	ARRAY	INT16	O	RO	Monitor
	1	Protocol Type		INT16	O	RO	Monitor
	2	RS232 - Mode Of Operation		INT16	O	RO	Monitor
	3	CANopen - Mode Of Operation		INT16	O	RO	Monitor
	4	ModBus - Mode Of Operation		INT16	O	RO	Monitor
	5	Ethercat - Mode Of Operation		INT16	O	RO	Monitor
2030 h	0	Drive Temperature	VAR	INT16	O	RO	Monitor
2031 h	0	Motor Temperature	VAR	INT16	O	RO	Monitor
2032 h	0	Heat Sink Temperature	VAR	INT16	O	RO	Monitor
2040 h	0	Voltage Brake	VAR	INT16	O	RO	reserved
2041 h	0	Voltage Bus	VAR	INT16	O	RO	Monitor
2042 h	0	Voltage Logic Board	VAR	INT16	O	RO	reserved
2043 h	0	Voltage Reference	VAR	INT16	O	RO	Monitor
2050 h	0	Torque Current	VAR	INT16	O	RO	Monitor
2051 h	0	Drive Power	VAR	INT16	O	RO	reserved
2052 h	0	Motor Power	VAR	INT16	O	RO	reserved
2053 h	0	Velocity Filtered	VAR	INT16	O	RO	Monitor
2054 h	0	Energy i^2t	VAR	UINT16	O	RO	Monitor
2060 h	0	Impulse	VAR	INT16	O	RO	Monitor
2061 h	0	ThetaE	VAR	UINT16	O	RO	Monitor
2062 h	0	ThetaM	VAR	INT16	O	RO	Monitor
2070 h	0	Torque Current Windowed	VAR	INT16	O	RO	Monitor
2080 h	0	STO Statusword	VAR	INT16	O	RO	Monitor
2081 h	0	ParGenTell 2	VAR	INT16	O	RO	reserved
2082 h	0	ParGenTell 3	VAR	INT16	O	RO	reserved
2083 h	0	ParGenTell 4	VAR	INT16	O	RO	reserved
2090 h	0	ParGenLongTell 1	VAR	INT16	O	RO	reserved
2091 h	0	ParGenLongTell 2	VAR	INT16	O	RO	reserved
2092 h	0	ParGenLongTell 3	VAR	INT16	O	RO	reserved
2093 h	0	ParGenLongTell 4	VAR	INT16	O	RO	reserved
3001 h	0	**Limits Parameter**	ARRAY	UINT32	O	**RO**	-
	1	Velocity ABS		UINT32	O	RO	Monitor
	2	Acceleration ABS		UINT32	O	RO	Monitor
	3	Limit Velocity Profile		UINT32	O	RO	Monitor
	4	-		UINT32	O	RO	-
	5	-		UINT32	O	RO	-
3002 h	0	**Motor Brake Parameters**	ARRAY	INT16	M IF	RO	Settings
	1	Motor Brake Option		INT16	M IF	R/W	Settings
	2	Motor Brake Delay		INT16	M IF	R/W	Settings

	3	Motor Brake Unlock time		INT16	M IF	R/W	Settings
	4	Motor Brake Timeout		INT16	M IF	R/W	Settings
	5	Mode Manual Automatic		INT16	M IF	R/W	Settings
	6	Motor Brake Status		INT16	M IF	RO	Monitor
	7	Motor Brake Type		INT16	M IF	RO	Settings
	8	Motor Brake Int or Ext		INT16	M IF	RW	Settings
	9	-		INT16	M IF	RO	-
	10	-		INT16	M IF	RO	-
3003 h	0	**Drive Size Parameters**	ARRAY	UINT8	O	RO	-
	1	Maximum Current		INT16	O	RO	Monitor
	2	Peak Current		INT16	O	RO	Monitor
	3	Rated Current		INT16	O	RO	Monitor
	4	reserved		INT16	O	RO	reserved
	5	reserved		INT16	O	RO	reserved
	6	reserved		INT16	O	RO	reserved
	7	reserved		INT16	O	RO	reserved
3004 h	**0**	reserved	ARRAY	INT16	O	**RO**	reserved
	1	reserved		INT16	O	RO	reserved
	2	reserved		INT16	O	RO	reserved
	3	reserved		INT16	O	RO	reserved
	4	reserved		INT16	O	RO	reserved
	5	reserved		INT16	O	RO	reserved
	6	Absolute position H (actual) - 16 bit MSB		INT16	O	RO	Monitor
	7	Encoder Type		INT16	O	RO	Monitor
	8	Reserved		INT16	O	RO	Reserved
	9	Reserved		INT16	O	RO	Reserved
	10	Encoder Command		INT16	O	RW	Monitor
	11	Initial gear number		INT16	O	RO	Monitor
	12	Relative position (initial)		INT16	O	RO	Monitor
	13	Reserved		INT16	O	RO	Reserved
	14	Reserved		INT16	O	RO	Reserved
	15	Reserved		INT16	O	RO	Reserved
3005 h	**0**	**Filter Parameters**	ARRAY	INT16	O	RO	Settings
	1	Reserved		INT16	O	RO	Monitor
	2	Reserved		INT16	O	RO	Reserved
	3	Reserved		INT16	O	RO	Reserved
	4	Reserved		INT16	O	RO	Reserved
	5	Filter Polo 1 Velocity Filtered		INT16	O	RW	Settings
	6	Filter Polo 1Velocity Actual		INT16	O	RW	Settings
	7	Reserved		INT16	O	RO	Reserved
	8	Reserved		INT16	O	RO	Reserved
	9	Reserved		INT16	O	RO	Reserved
	10	Reserved		INT16	O	RO	Reserved
3006 h	0	**Motor Specific Settings**	ARRAY	INT16	O	RO	Monitor

	1	Motor Part Number		INT16	O	RO	Monitor
	2	Max Motor Speed		INT16	O	RO	Monitor
	3	N Poli		INT16	O	RO	Monitor
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3007 h	0	**Dynamic Stop Parameter**	ARRAY	INT16	M IF	RO	Settings
	1	Dynamic Stop Enabled		INT16	M IF	R/W	Settings
	2	Holding Torque Time		INT16	M IF	R/W	Settings
	3	Dynamic Stop Status		INT16	M IF	RO	Monitor
	4	Decrement step ramp		INT16	M IF	R/W	Settings
	5	Max Timeout Dynamic Brake		INT16	M IF	R/W	Settings
	6	reserved		INT16	M IF	RO	reserved
	7	Option Code Braking Mode		INT16	M IF	R/W	Settings
	8	Decrement step S-ramp		INT16	M IF	R/W	Settings
3008 h	0	**Emergency Parameter**	ARRAY	INT16	M IF	RO	Settings
	1	Emergency Enable		INT16	M IF	R/W	Settings
	2	Emergency Level Neg		INT16	M IF	R/W	Settings
	3	Emergency Status		INT16	M IF	RO	Monitor
	4	Option Code		INT16	M IF	R/W	Settings
	5	-		INT16	M IF	R/W	-
	6	-		INT16	M IF	R/W	-
	7	-		INT16	M IF	R/W	-
	0	Alarm Option	ARRAY	INT16	O	RO	Monitor
	0	Disable Alarm 1		INT16	O	RO	Monitor
	0	Disable Alarm 2		INT16	O	RO	Monitor
	0	Disable Alarm 3		INT16	O	RO	Monitor
	0	Disable Alarm 4		INT16	O	RO	Monitor
3020 h	0	**Function Digital Input**	ARRAY	INT16	O	RO	Monitor
	1	Enable Function		INT16	O	RO	Monitor
	2	State Function		INT16	O	RO	Monitor
	3	Level Function		INT16	O	RO	Monitor
	4	Digital Raw Level		INT16	O	RO	Monitor
	5	Configuration Set Point		INT16	O	RO	Monitor
	6	Direction		INT16	O	RO	Monitor
3021 h	0	**Digital Input 1**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3022 h	0	**Digital Input 2**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings

	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3023 h	0	**Digital Input 2**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3024 h	0	**Digital Input 2**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3030 h	0	**Function Digital Output**		INT16	O	RO	Monitor
	1	Enable Function		INT16	O	RO	Monitor
	2	State Function		INT16	O	RO	Monitor
	3	Level Function		INT16	O	RO	Monitor
	4	Digital Raw Level		INT16	O	RO	Monitor
	5	-		INT16	O	RO	-
	6	-		INT16	O	RO	-
3031 h	0	**Digital Output 1**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3032 h	0	**Digital Output 2**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3033 h	0	**Digital Output 3**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-
	5	-		INT16	O	RO	-
3034 h	0	**Digital Output4**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	RW	Monitor/Settings
	2	State		INT16	O	RO	Monitor
	3	Level		INT16	O	RW	Monitor/Settings
	4	-		INT16	O	RO	-

	5	-		INT16	O	RO	-
3040 h	0	**Analog Input**	ARRAY	INT16	O	RO	Monitor
	1	Configuration ADC Ref		INT16	O	RO	Monitor
	2	Value ADC [12 BIT]		INT16	O	RO	Monitor
	3	Value ADC [16 BIT]		INT16	O	RO	Monitor
	4	Threshold Configuration Run with Thrs		INT16	O	RO	Monitor
	5	-		INT16	O	RO	-
3050 h	0	**Analog Output 1**	ARRAY	INT16	O	RO	Monitor
	1	Configuration		INT16	O	R/W	Settings
	2	Value		INT16	O	R/W	Settings
	3	Output		INT16	O	RO	Monitor
	4	-		INT16	O	RO	-
	4	-		INT16	O	RO	-
3051 h	0	**Analog Output 2**	ARRAY	INT16	O	RO	not available
	1	-		INT16	O	RO	-
3100 h	0	Configuration 1 - Statusword	VAR	INT16	O	R/W	Settings
3101 h	0	Configuration 2 – Reserved	VAR	INT16	O	R/W	reserved
3102 h	0	Configuration 3 – STO	VAR	INT16	O	R/W	Settings
3103 h	0	Configuration 4 – Safety FSM	VAR	INT16	O	R/W	Settings
3104 h	0	Configuration 5 – Warning Cfg	VAR	INT16	O	R/W	reserved
3105 h	0	Configuration 6 - Free	VAR	INT16	O	R/W	reserved
3106 h	0	Configuration 7 - Free	VAR	INT16	O	R/W	reserved
3107 h	0	Configuration 8 - Free	VAR	INT16	O	R/W	reserved
3200 h	0	**Current PID**	ARRAY	INT16	M	RO	reserved
	1	PidCur Kp		INT16	M	R/W	reserved
	2	PidCur Ki		INT16	M	R/W	reserved
	3	PidCur Kv		INT16	M	R/W	reserved
	4	PidCur Kd		INT16	M	R/W	reserved
	5	PidCur N		INT16	M	R/W	reserved
	6	PidCur FF		INT16	M	R/W	reserved
3201 h	0	**Speed PID**	ARRAY	INT16	M	RO	Settings
	1	PidVel Kp		INT16	M	R/W	Settings
	2	PidVel Ki		INT16	M	R/W	Settings
	3	PidVel Kv		INT16	M	R/W	Settings
	4	PidVel Kd		INT16	M	R/W	Settings
	5	PidVel N		INT16	M	R/W	Settings
	6	PidVel FF		INT16	M	R/W	Settings
3202 h	0	**Position PID**	ARRAY	INT16	M	RO	reserved
	1	PidPos Kp		INT16	M	R/W	reserved
	2	PidPos Ki		INT16	M	R/W	reserved
	3	PidPos Kv		INT16	M	R/W	reserved
	4	PidPos FF Ra V		INT16	M	R/W	reserved
	5	PidPos FF Ra A		INT16	M	R/W	reserved
	6	PidPos FF Vr V		INT16	M	R/W	reserved

	7	PidPos FF Rd A		INT16	M	R/W	reserved
	8	PidPos FF Rd V		INT16	M	R/W	reserved
	9	PidPos Tc		INT16	M	R/W	reserved
	10	PidPos FF Kpdff		INT16	M	R/W	reserved
	11	PidPos FF Kfriction		INT16	M	R/W	reserved
3300 h	0	Velocity Full Scale	VAR	INT16	O	R/W	Settings
3400 h	0	ParGen 1	VAR	INT16	O	R/W	reserved
3401 h	0	Warning emergency protocol	VAR	INT16	O	R/W	reserved
3402 h	0	ParGen 3	VAR	INT16	O	R/W	reserved
3403 h	0	ParGen 4	VAR	INT16	O	R/W	reserved
3500 h	0	**ParGen Array1**	ARRAY	INT16	O	R/W	reserved
	1	ParGen Array1 - 1		INT16	O	R/W	reserved
	2	ParGen Array1 - 2		INT16	O	R/W	reserved
	3	ParGen Array1 - 3		INT16	O	R/W	reserved
	4	ParGen Array1 - 4		INT16	O	R/W	reserved
	5	ParGen Array1 - 5		INT16	O	R/W	reserved
3501 h	0	**ParGen Array2**	ARRAY	INT16	O	R/W	reserved
	1	ParGen Array2 - 1		INT16	O	R/W	reserved
	2	ParGen Array2 - 2		INT16	O	R/W	reserved
	3	ParGen Array2 - 3		INT16	O	R/W	reserved
	4	ParGen Array2 - 4		INT16	O	R/W	reserved
	5	ParGen Array2 - 5		INT16	O	R/W	reserved
3800 h	0	ParGenLong 1	VAR	INT32	O	R/W	reserved
3801 h	0	ParGenLong 2	VAR	INT32	O	R/W	reserved
3802 h	0	ParGenLong 3	VAR	INT32	O	R/W	reserved
3803 h	0	ParGenLong 4	VAR	INT32	O	R/W	reserved
3900 h	0	**Warning Settings**	ARRAY	INT32	O	R/W	reserved
	1	Warning Enable User		INT32	O	R/W	reserved
	2	Warning Sel Output		INT32	O	R/W	reserved
	3	ParGenLong Array1 - 3		INT32	O	R/W	reserved
	4	ParGenLong Array1 - 4		INT32	O	R/W	reserved
	5	ParGenLong Array1 - 5		INT32	O	R/W	reserved
3901 h	0	**ParGenLong Array2**	ARRAY	INT32	O	R/W	reserved
	1	ParGenLong Array2 - 1		INT32	O	R/W	reserved
	2	ParGenLong Array2 - 2		INT32	O	R/W	reserved
	3	ParGenLong Array2 - 3		INT32	O	R/W	reserved
	4	ParGenLong Array2 - 4		INT32	O	R/W	reserved
	5	ParGenLong Array2 - 5		INT32	O	R/W	reserved
4100 h	0	**Status Communication CAN**	ARRAY	INT16	O	RO	reserved
	1	Safety State		INT16	O	R/W	reserved
	2	STO		INT16	O	RO	reserved
4100 h	0	**Status Communication CAN**	ARRAY	INT16	O	RO	reserved
	1	TEC Register		INT16	O	RO	reserved
	2	REC Register		INT16	O	RO	reserved

	3	Actual Flags Error		INT16	O	RO	reserved
	4	Actual State Machine Communication		INT16	O	RO	reserved
	5	Actual State CAN Controller		INT16	O	RO	reserved
4101 h	0	**Counter Communication CAN**	ARRAY	INT16	O	RO	reserved
	1	Counter BusOff		INT16	O	RO	reserved
	2	Counter Recovery BusOff		INT16	O	RO	reserved
	3	Count Reset CAN		INT16	O	RO	reserved
	4	Counter Abort		INT16	O	RO	reserved
4102 h	0	**Settings Communication CAN**	ARRAY	INT16	O	RO	Settings
	1	Sensibility CAN error		INT16	O	R/W	Settings
	2	Timeout Error Passive		INT16	O	R/W	Settings
	3	Clear Abort Flag		INT16	O	R/W	Settings
4103 h	0	**Error Communication CAN**	ARRAY	INT16	O	RO	Monitor
	1	Last Error Flag		INT16	O	RO	Monitor
	2	All Error Flags		INT16	O	RO	Monitor
	3	Flag Abort code		INT16	O	RO	Monitor
	4	First Abort Code Occurred		INT16	O	RO	Monitor
4200 h	0	**Alarm Monitoring**	ARRAY	UINT16	O	RO	Monitor
	1	Last Error Code		UINT16	O	RO	Monitor
	2	Manufacturer Code		UINT16	O	RO	Monitor
	3	Manufacturer Code		UINT16	O	RO	Monitor
4400 h	0	reserved	VAR	UINT8	O	RW	reserved
	1	reserved		UINT8	O	RW	reserved
4401 h	0	reserved	ARRAY	UINT16	O	RO	reserved
	1	reserved		UINT16	O	RW	reserved
	2	reserved		UINT16	O	RW	reserved
	3	reserved		UINT16	O	RW	reserved
	4	reserved		UINT16	O	RW	reserved
	5	reserved		UINT16	O	RW	reserved
	6	reserved		UINT16	O	RW	reserved
	7	reserved		UINT16	O	RW	reserved
	8	reserved		UINT16	O	RW	reserved
	9	reserved		UINT16	O	RW	reserved
	10	reserved		UINT16	O	RW	reserved
4402 h	0	reserved	ARRAY	UINT16	O	RO	reserved
	1	reserved		UINT16	O	RW	reserved
	2	reserved		UINT16	O	RW	reserved
	3	reserved		UINT16	O	RW	reserved
	4	reserved		UINT16	O	RW	reserved
	5	reserved		UINT16	O	RW	reserved
	6	reserved		UINT16	O	RW	reserved
	7	reserved		UINT16	O	RW	reserved
	8	reserved		UINT16	O	RW	reserved
	9	reserved		UINT16	O	RW	reserved

	10	reserved		UINT16	O	RW	reserved
5000 h	0	**Logger Alarm Data**		INT16	O	RO	Diagnostic
	1	reserved		INT16	O	RO	Diagnostic
	2	reserved		INT16	O	RO	Diagnostic
5001 h	0	**Logger Alarm 1**		INT16	O	RO	Diagnostic
	1	Data 1		INT16	O	RO	Diagnostic
	...	...		INT16	O	RO	Diagnostic
	32	Data 32		INT16	O	RO	Diagnostic
5002 h	0	**Logger Alarm 2**		INT16	O	RO	Diagnostic
	1	Data 1		INT16	O	RO	Diagnostic
	...	...		INT16	O	RO	Diagnostic
	32	Data 32		INT16	O	RO	Diagnostic
5003 h	0	**Logger Alarm 3**		INT16	O	RO	Diagnostic
	1	Data 1		INT16	O	RO	Diagnostic
	...	...		INT16	O	RO	Diagnostic
	32	Data 32		INT16	O	RO	Diagnostic
5004 h	0	**Logger Alarm 3**		INT16	O	RO	Diagnostic
	1	Data 1		INT16	O	RO	Diagnostic
	...	...		INT16	O	RO	Diagnostic
	32	Data 32		INT16	O	RO	Diagnostic
5010 h	0	**Logger Alarm Reserved**		INT32	O	RO	Diagnostic
	1	reserved		INT32	O	RO	Diagnostic
	2	reserved		INT32	O	RO	Diagnostic
	3	reserved		INT132	O	RO	Diagnostic
	4	reserved		INT132	O	RO	Diagnostic
STANDARD OBJECTS DSP402							
6007 h	0	Abort Connection Option Code	VAR	UINT16	O	R/W	State Machine DS402
603F h	0	Error Code	VAR	UINT16	O	RO	Diagnostic
6040 h	0	Control Word	VAR	UINT16	M	R/W	State Machine DS402
6041 h	0	Status Word	VAR	UINT16	M	RO	State Machine DS402
605A h	0	Quick Stop Option Code	VAR	INT16	O	R/W	State Machine DS402
605B h	0	Shutdown Option Code	VAR	INT16	O	R/W	State Machine DS402
605C h	0	Disable Option Code	VAR	INT16	O	R/W	State Machine DS402
605D h	0	Halt Option Code	VAR	INT16	O	R/W	State Machine DS402
605E h	0	Fault Reaction Code	VAR	INT16	O	R/W	State Machine DS402
6060 h	0	Modes of Operation	VAR	INT8	M	R/W	State Machine DS402
6061 h	0	Modes of Operation Display	VAR	INT8	M	RO	State Machine DS402
6062 h	0	Position Demand Value	VAR	INT32	O	RO	not available
6063 h	0	Position Actual internal Value	VAR	INT32	O	RO	not available
6064 h	0	Position Actual Value	VAR	INT32	M	RO	Monitor
6065 h	0	Following Error Windows	VAR	UINT32	O	R/W	not available
6066 h	0	Following Error Timeout	VAR	UINT16	O	R/W	not available
6067 h	0	Position Windows	VAR	UNIT32	O	R/W	not available
6068 h	0	Position Window Time	VAR	UINT16	O	R/W	not available

606B h	0	Velocity Demand Value	VAR	INT32	O	RO	Profile Velocity
606C h	0	Velocity Actual Value	VAR	INT32	M	RO	Profile Velocity
606D h	0	Velocity Window	VAR	UINT16	O	R/W	Profile Velocity
606E h	0	Velocity Window Time	VAR	UINT16	O	R/W	Profile Velocity
606F h	0	Velocity Threshold	VAR	UINT16	O	R/W	Profile Velocity
6070 h	0	Velocity Threshold Time	VAR	UINT16	O	R/W	Profile Velocity
6071 h	0	Target Torque	VAR	INT16	M	R/W	Torque Profile
6072 h	0	Max Torque	VAR	UINT16	O	R/W	reserved
6073 h	0	Max Current	VAR	UINT16	O	R/W	Torque Profile
6074 h	0	Torque Demand	VAR	INT16	O	RO	Torque Profile
6075 h	0	Motor Rated Current	VAR	UINT32	O	R/W	Torque Profile
6076 h	0	Motor Rated Torque	VAR	UINT32	O	R/W	reserved
6077 h	0	Torque Actual Value	VAR	INT16	O	RO	Torque Profile
6078 h	0	Current Actual Value	VAR	INT16	O	RO	Torque Profile
6079 h	0	DC Link Circuit Voltage	VAR	UINT32	O	RO	Monitor
607A h	0	Target Position	VAR	INT32	M	R/W	not available
607B h	0	Position Range Limit	VAR	INT32	O	R/W	not available
607C h	0	Home Offset	VAR	INT32	O	R/W	not available
607D h	0	Software Position Limit	VAR	INT32	O	R/W	not available
607E h	0	Polarity	VAR	UINT8	O	R/W	Profile Velocity (and Profile Position)
607F h	0	Max Profile Velocity	VAR	UINT32	O	R/W	Profile Velocity (and Profile Position)
6080 h	0	Max Motor Speed	VAR	UINT32	O	R/W	reserved
6081 h	0	Profile Velocity	VAR	UINT32	M	R/W	not available
6082 h	0	End Velocity	VAR	UINT32	O	R/W	not available
6083 h	0	Profile Acceleration	VAR	UINT32	O	R/W	Profile Velocity (and Profile Position)
6084 h	0	Profile Deceleration	VAR	UINT32	O	R/W	Profile Velocity (and Profile Position)
6085 h	0	Quick Stop Deceleration	VAR	UINT32	O	R/W	reserved
6086 h	0	Motion Profile Type	VAR	INT16	O	R/W	Profile Velocity
6087 h	0	Torque Slope	VAR	UINT32	M	R/W	Torque Profile
6088 h	0	Torque Profile Type	VAR	INT16	O	R/W	Torque Profile
6096 h	0	Velocity Factor Group	ARRAY	UINT32	O	R/W	Settings
	1	Num Velocity Factor		UINT32	O	R/W	Settings
	2	Div Velocity Factor		UINT32	O	R/W	Settings
6097 h	0	Acceleration Factor Group	ARRAY	UINT32	O	R/W	Settings
	1	Num Acceleration Factor		UINT32	O	R/W	Settings
	2	Div Acceleration Factor		UINT32	O	R/W	Settings
6098 h	0	Homing Method	VAR	INT8	M	R/W	not available
6099 h	0	Homing Speeds	VAR	UINT32	M	R/W	not available
609A h	0	Homing Acceleration	VAR	UINT32	O	R/W	not available
60A2 h	0	Jerk Factor Group	ARRAY	UINT32	O	R/W	Settings
	1	Num Jerk Factor		UINT32	O	R/W	Settings
	2	Div Jerk Factor		UINT32	O	R/W	Settings

60A3 h	0	Profile Jerk Use	VAR	UINT8	O	RO	Settings
60A4 h	0	Profile Jerk	ARRAY	UINT32	O	R/W	Settings
	1	Profile Jerk 1		UINT32	O	R/W	Settings
	2	Profile Jerk 2		UINT32	O	R/W	Settings
	3	Profile Jerk 3		UINT32	O	R/W	not available
	...			UINT32	O	R/W	not available
	6	Profile Jerk 6		UINT32	O	R/W	not available
60C5 h	0	Max Acceleration	VAR	UINT32	O	R/W	Profile Velocity (and Profile Position)
60C6 h	0	Max Deceleration	VAR	UINT32	O	R/W	Profile Velocity (and Profile Position)
60E0 h	0	Positive Torque Limit Value	VAR	UINT16	O	R/W	Torque Profile, Profile Velocity and Profile Position
60E1 h	0	Negative Torque Limit Value	VAR	UINT16	O	R/W	Torque Profile, Profile Velocity and Profile Position
60E3 h	0	Supported homing methods	ARRAY	INT8	O	RO	not available
	1	1st supported homing method		INT8	O	RO	not available
	2	2nd supported homing method		INT8	O	RO	not available
	3	3rd supported homing method		INT8	O	RO	not available
	4	4th supported homing method		INT8	O	RO	not available
	5	5th supported homing method		INT8	O	RO	not available
	6	6th supported homing method		INT8	O	RO	not available
	7	7th supported homing method		INT8	O	RO	not available
	8	8th supported homing method		INT8	O	RO	not available
	9	9th supported homing method		INT8	O	RO	not available
	10	10th supported homing method		INT8	O	RO	not available
60F2 h	0	Position Option Code	VAR	UINT16	O	R/W	not available
60F4 h	0	Following Error Actual Value	VAR	INT32	O	RO	not available
60FA h	0	Control Effort	VAR	INT32	O	RO	not available
60FC h	0	Position Demand Internal Value	VAR	INT32	O	RO	not available
60FD h	0	Digital Inputs	VAR	UINT32	O	RO	Monitor
60FE h	0	Digital Outputs	VAR	UINT32	O	RO	Settings
60FF h	0	Target Velocity	VAR	INT32	**M**	R/W	Profile Velocity
6402 h	0	Motor Type	VAR	UINT16	O	R/W	Monitor
6403 h	0	Motor Catalogue Number	VAR	STRING	O	R/W	Monitor
6404 h	0	Motor Manufacturer	VAR	STRING	O	R/W	Monitor
6502 h	0	Supported Drive Modes	VAR	UINT32	**M**	RO	Monitor

8. | FUNCTIONS

OVER SPEED

The drive has the alarm Over Speed set as default 10% of Maximum Limit Velocity defined.
To change the percent contact the Manufacturer.

DIGITAL I/O

The drive has:

- 4 DIGITAL Input
- 4 DIGITAL Output
- 2 SAFETY Digital Input

Digital Input

The digital Inputs can be configured with different functions (see object 3020_h).
To change digital input configuration, it needs write 0x3021, 0x3022, 0x3023, 0x3024 subindex 1.
The drive has the digital inputs configured by manufacturer as default.
The standard of level to activate the function in digital input is edge positive.
To change digital input level it needs write 0x3021, 0x3022, 0x3023, 0x3024 subindex 3.

The DEFAULT configuration is

- DIG-IN1 = function RUN used by Analog Mode Control
- DIG-IN2 = function STOP used by Analog Mode Control
- DIG-IN3 = function EMERGENCY used to go from RUN State to STANDBY in emergency condition with dynamic brake.
- DIG-IN4 = function RESET HARDWARE (with timeout)

If the drive is controlled by CAN via SDO/PDO commands, the "controlword" (6060h) set the states of DSP402 state machine and Digital Input (RUN and STOP) are not used.

Digital Output

The digital Outputs can be configured with different functions (see object 3030_h).
To change digital input configuration It needs write 0x3031, 0x3032, 0x3033, 0x3034 subindex 1.
The drive has the digital outputs configured by manufacturer as default.
The standard of level to activate the function in digital output is edge positive.
To change digital input level It needs write 0x3031, 0x3032, 0x3033, 0x3034 subindex 3.

The DEFAULT configuration is

- DIG-OUT1 = Function Drive Status
- DIG-OUT2 = Function Warning Status
- DIG-OUT3 = Function Ready
- DIG-OUT4 = Function Brake Status

Some outputs functions can be read with bit of statusword, for example bit 7 "Warning" and bit 8 "Emergency".

Digital Safety Input

There are two digital Input connected a STO circuit.
If it is available the "STO Safety" the STO is active the drive goes in SAFETY status independently to other selection. In case of FAULT the drive goes in FAULT State.

SAFETY

The follow picture shows the STO state machine:

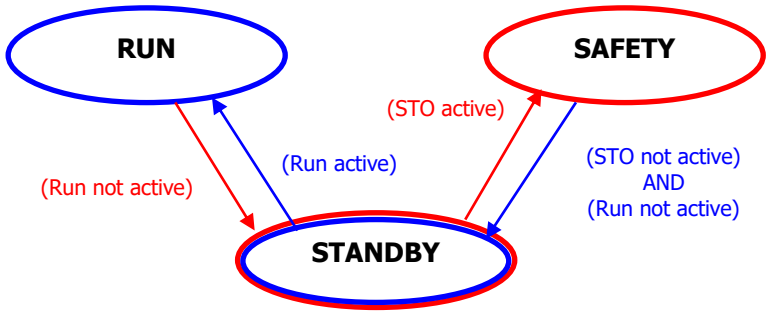


Figure 44 – STO transition State Machine

Caution

To reactive the standby status is mandatory that STO and RUN are not active and do the procedure

- in Analog Mode: SAFETY → STANDBY → RUN
- in CANopen Mode: SAFETY → "SWITCH ON DISABLED" → "READY TO SWITCH ON" → "SWITCH ON" → "OPERATION ENABLED"

If an application requires controlled braking, before using the STO function, the drive must be braked first and the STO function must be activate with a delay:

- Controlled braking of drive
- Once standstill is reached, disable the drive.
- In the case of a suspended load, mechanically lock the drive as well
- Activate STO function.

Every single output relay is read.
Safety controller can read every single command corresponding to every relay output: fully monitoring of safety functions.

Caution

The Drive cannot hold the load with the STO function activated because the motor no longer supplies any torque.

- If the STO function is activated during operation, the drive will stop in an uncontrolled manner.
- If the drive has the Safety Torque OFF (STO), verify that this circuit is correctly supplied before all operation functions.

This procedure is necessary to read the drive Safety Mode.

Master Controller must active the STO Hardware Input to move the drive in safety state (SAFETY).

information

To enable Safety Mode you don't have to connect pin STO1 or STO2 at +24V.

See chapter SAFETY to have additional information about safety object.

It is possible change the transition to go to "SAFETY" state using object $3102_h:0_h$. If the value is 0 is a standard mode, if the value is 1 then the brake is locked immediately

EMERGENCY DIGITAL INPUT

The Emergency Input is the digital input function (it is not safety certificated) to exit the drive from power-on state. The input can be configured as enabling signal hardware to move:

- from "Operation Enable" state [RUN] to "Switched On" state [STANDBY]
- from "Quick Stop Active" state [STOP] to "Switched On" state [STANDBY]

Caution

It can be considered as emergency signal but it isn't safety certificated.
For disabling the power in safety certificated condition referred to STO chapter of Drive User Guide.

information

To configure the Emergency Input it needs to configure Digital Input as "EMERGENCY" and configure object 0x3008.

If the function "Emergency Input" is programmed in one digital input, then the digital is used to move in the state machine of DSP402:

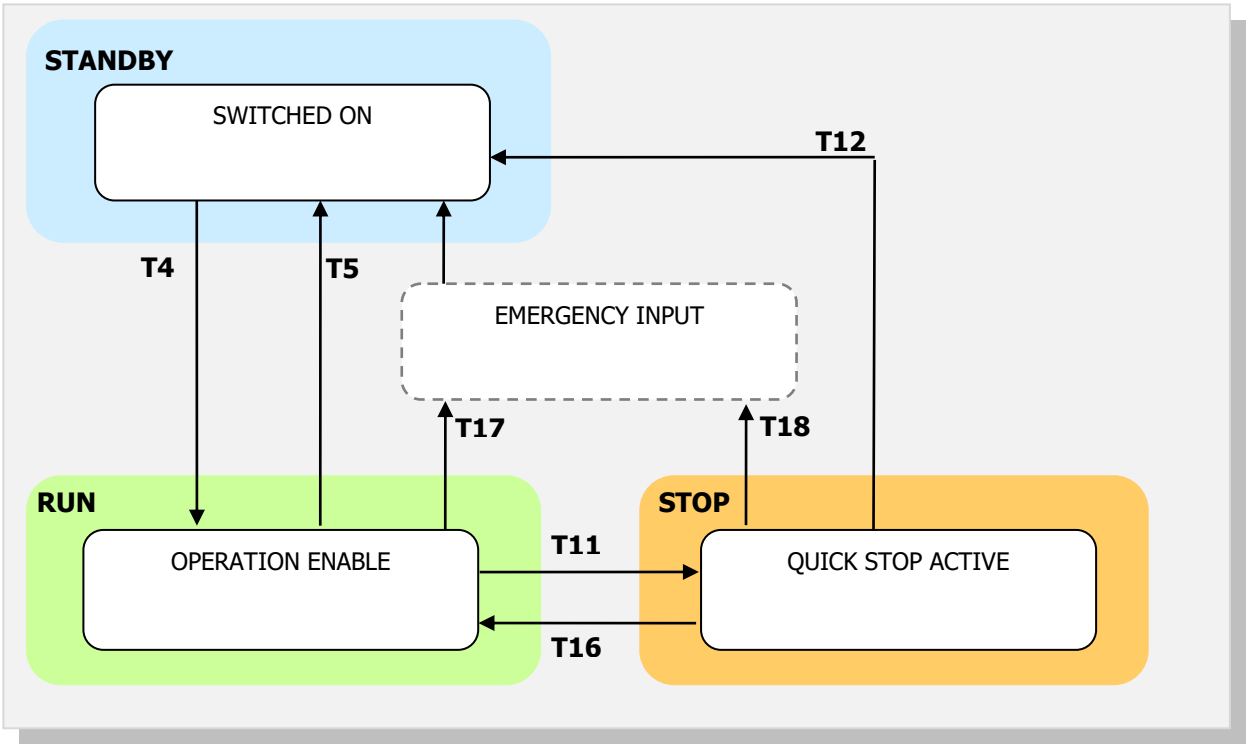


Figure 45 - Emergency enable configuration

Transition	Commands	Description
T4	Controlword command= "enable operation" **AND** Status Emergency Input **LOW** level	Drive will go in "Operation Enable", it will wait the target of profile to run it.
T5	Controlword command= "disable operation" **AND** Status Emergency Input **LOW** level	Drive will go in "switched On". If Dynamic Stop is activated then the motor will decrement speed with a programmed ramp.
T11	Controlword command= "Quick Stop"	Drive will go in "quick Stop Active", it will stop with torque equal zero.
T16	Controlword command= "enable operation"	Drive will go in "Operation Enable", it will wait the target of profile to run it.
T12	Controlword command= "disable operation"	Drive will go in "switched On" and the power will turn off. The drive will be in standby.
T17	Status Emergency Input **HIGH** level	The input emergency is ON. If it is enabled the drive will go in "switched On". If Dynamic Stop is activated then the motor will decrement speed with a programmed ramp.
T18	Status Emergency Input **HIGH** level	The input emergency is ON. If it is enabled the drive will go in "switched On".

Table 48 - Transition Description with emergency Input Enabled

CANopen State Transition:

Transition 4: SWITCHED ON → OPERATION ENABLE
Controlword (6040_h) with "Enable Operation" value AND digital Emergency Input Enabled in low level hardware:

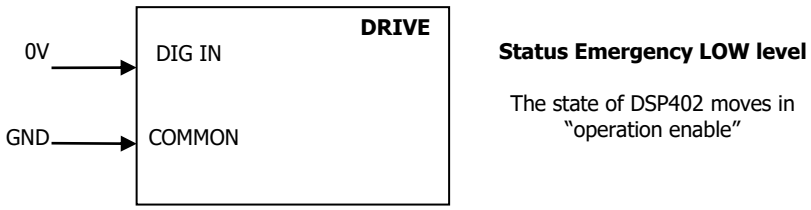


Figure 46 - Emergency Enable Status Low LevelTo "operation enable" state

Transition 5 and 12: OPERATION ENABLE → SWITCHED ON or QUICK STOP ACTIVE →SWITCHED ON
Controlword (6040_h) with "Disable Operation" value AND digital Emergency Input Enabled in low level hardware:

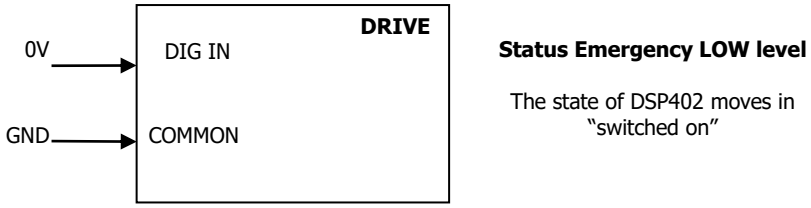


Figure 47 - Emergency Enable Status Low Level To "switched-on" state

Transition 17 and 18: OPERATION ENABLE → SWITCHED ON
Digital Emergency Input Enabled High Level hardware:

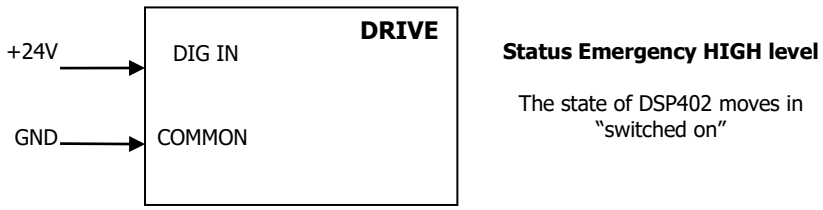


Figure 48 - Emergency Enable Status High Level

It can be changed the configuration level (default Positive Edge, to change the level contact manufacturer)

The mode to exit from "operation enabled" state (RUN) to "switched on" (STANDBY) is defined by object 0x3008:4 called "option code". It has available the following modes:

Option Code Transition	Description
1	It slows down with down ramp defined by object 0x3007:4 then when the velocity is on the range ±5 RPM it disables the drive.
0	Disable drive function (switch-off the drive power stage) and the servo motor will stop with inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.
-1	It stops immediately and set the Velocity to 0.
-2	It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive with torque applied until the time delay defined by object 0x3007:2 is expired.
-3	It stops immediately, set the Velocity to 0 and It looks immediately brake motor.
-4	It disables drive function (switch-off the drive power stage) and It locks immediately brake motor.
-5	It slows down with down S-ramp programmable defined by object 0x3007:8.

TRANSITION STOPPING MODE

It is possible to define the different mode to exit from RUN state ("operation enabled") using the object

- **$605A_h:0_h$ - Quick stop option code:** from "Operation Enable" (RUN) State to go to "Quick Stop" (STOP) State with CONTROLWORD command or Limit Switch clockwise and counter-clockwise
- **$605B_h:0_h$ - Shutdown option code:** from "Operation Enable" (RUN) State to go to "Ready To Switch On" or "Switch on Disabled" (INIT) State with CONTROLWORD command
- **$605C_h:0_h$ - Disable operation option code:** from "Operation Enable" (RUN) State to go to "Switched On" (STANDBY) State with CONTROLWORD command
- **$605D_h:0_h$ - Halt option code:** from "Operation Enable" (RUN) State to stay to "Operation Enable" (RUN) State with CONTROLWORD command
- **$605E_h:0_h$ - Fault reaction option code:** from all States to go to "Fault Reaction Active" (FAULT) State
- **$3008_h:4_h$ - Emergency option code:** from "Operation Enable" (RUN) State to go to "Switched On" (STANDBY) State with EMERGENCY command

Option Code Transition Mode	Description
1	It slows down with down ramp programmable defined by object 0x3007:4. When the velocity is on the range ±5 RPM it disables the drive.
0	Disable drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor.
-1	It stops immediately and set the Velocity to 0.
-2	It disables drive function (switch-off the drive power stage) and the servo motor will stop by inertia. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive with torque applied until the time delay defined by object 0x3007:2 is expired.
-3	It stops immediately, set the Velocity to 0 and It looks immediately brake motor.
-4	It disables drive function (switch-off the drive power stage) and It locks immediately brake motor.
-5	It slows down with down S-ramp programmable defined by object 0x3007:8. This stopping mode is available only with motion profile configured as SRAMP (object 0x6086:0 = -2 or -3)

TRANSITION STOPPING MODE '1':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is applied.

It slows down with down **LINEAR-RAMP** programmable defined by object 0x3007:8. When the velocity is on the range ±5 RPM it disables the drive.

This condition will be present when the drive state move from RUN state (operation enabled in DSP402) to STANDBY state (Switched ON in DSP402).

- **STANDARD CASE "Operation Enabled" (RUN) → "Switched-On" (STANDBY) :**

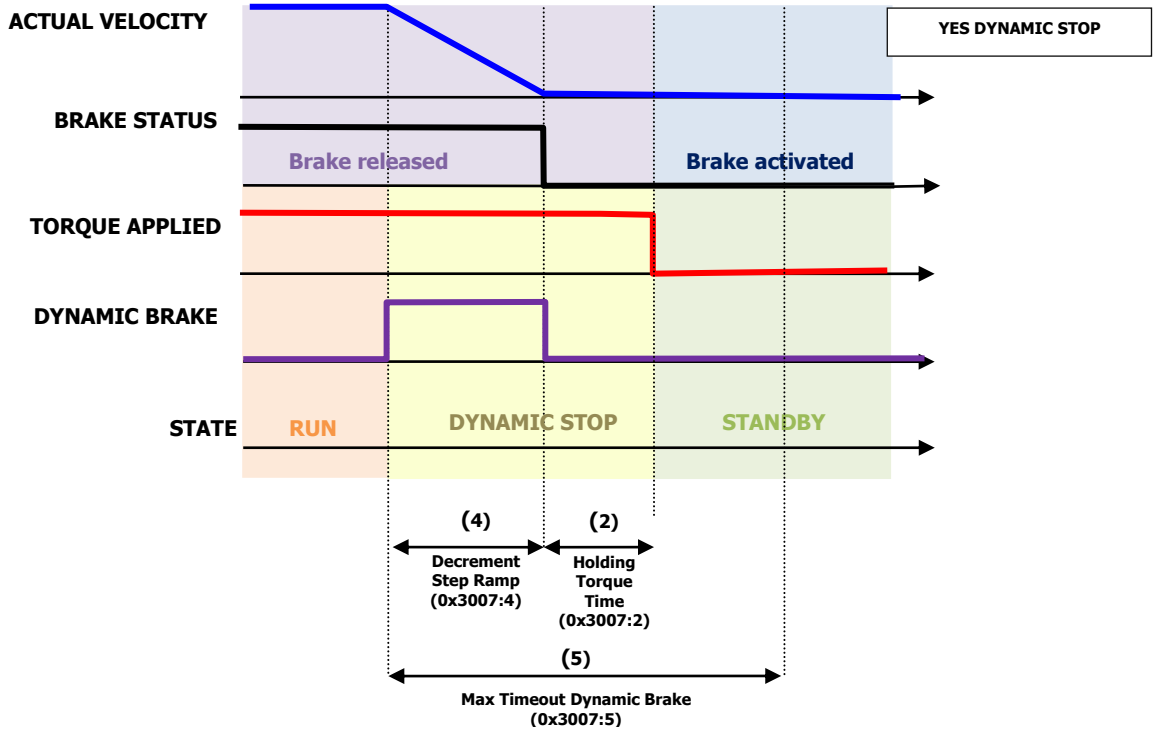


Figure 49 - Dynamic Stop timeframe: to Standby with Dynamic Stop (1)

SPECIAL CASE "Operation Enabled" (RUN) → "Switched-On" (STANDBY) : the actual velocity fails to follow the controlled braking ramp.

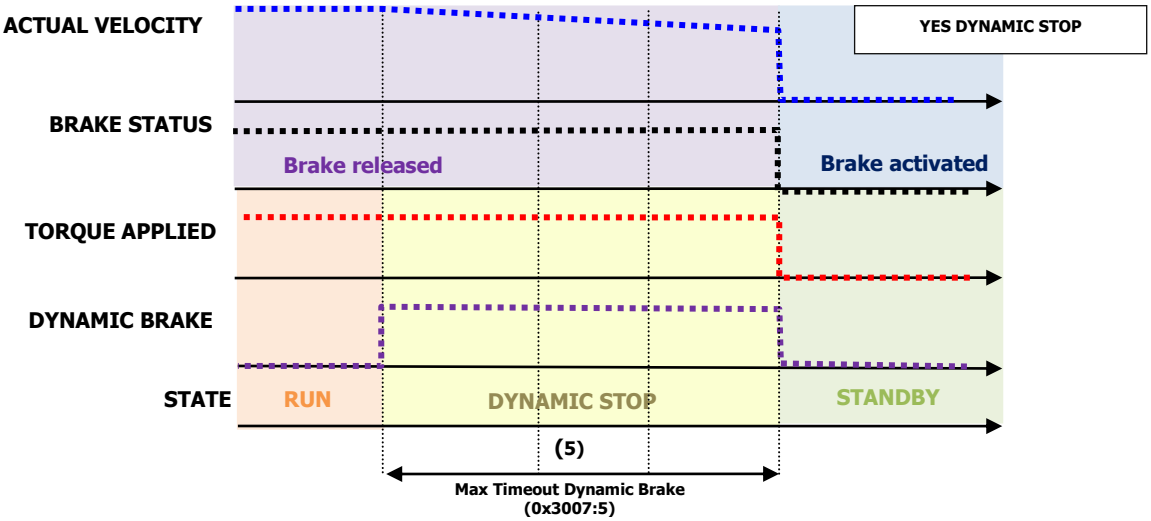


Figure 50 - Dynamic Stop timeframe: to Standby with Dynamic Stop (special case)

TRANSITION STOPPING MODE '0':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is not applied.

The motor will stop because it is switch-off the drive power stage end it locks the motor brake.

If the motor fails to stop and the time "Brake timeout" (object 3002_h: 4) expired, then the torque will be released immediately (see dashed line).

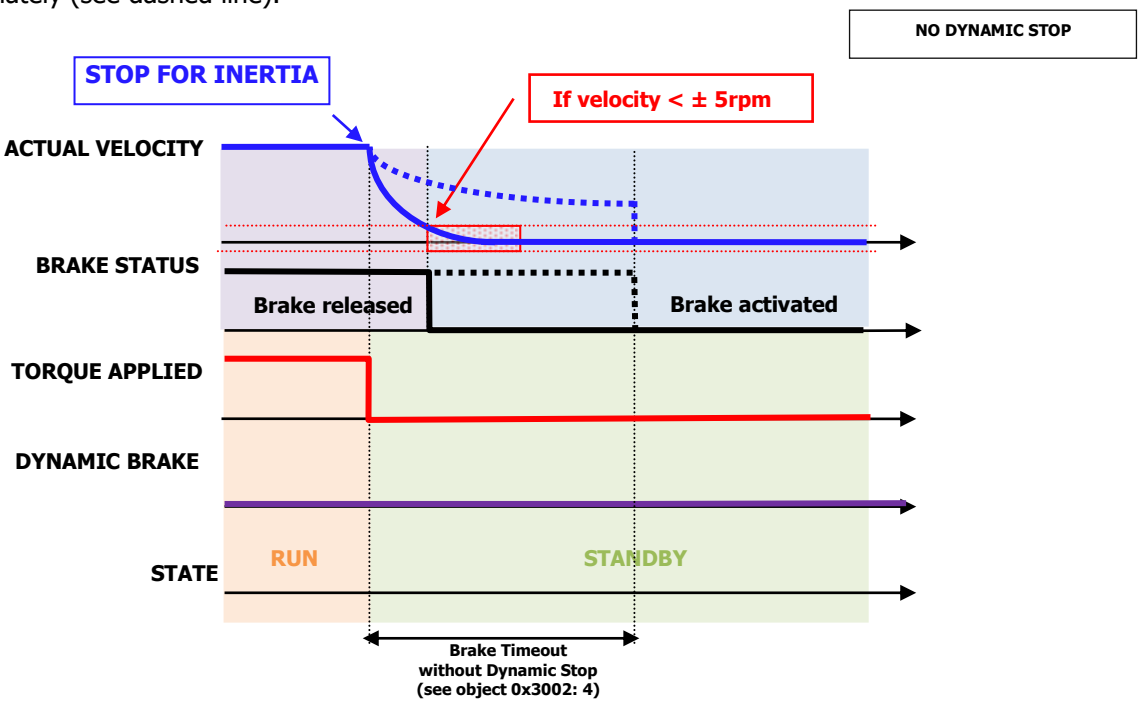


Figure 51 - Dynamic Stop timeframe: to Standby without Dynamic Stop (0)

TRANSITION STOPPING MODE '-1':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is not applied.

The motor will stop with set point = 0, when the velocity is lower ±5 rpm the drive locks motor brake and the torque is applied for "Holding torque time" (object 3007_h: 2).

If the motor fails to stop and the time "Max Brake timeout" (object 3002_h: 4) is expired, then the torque will be released immediately and the brake is locked (see dashed line).

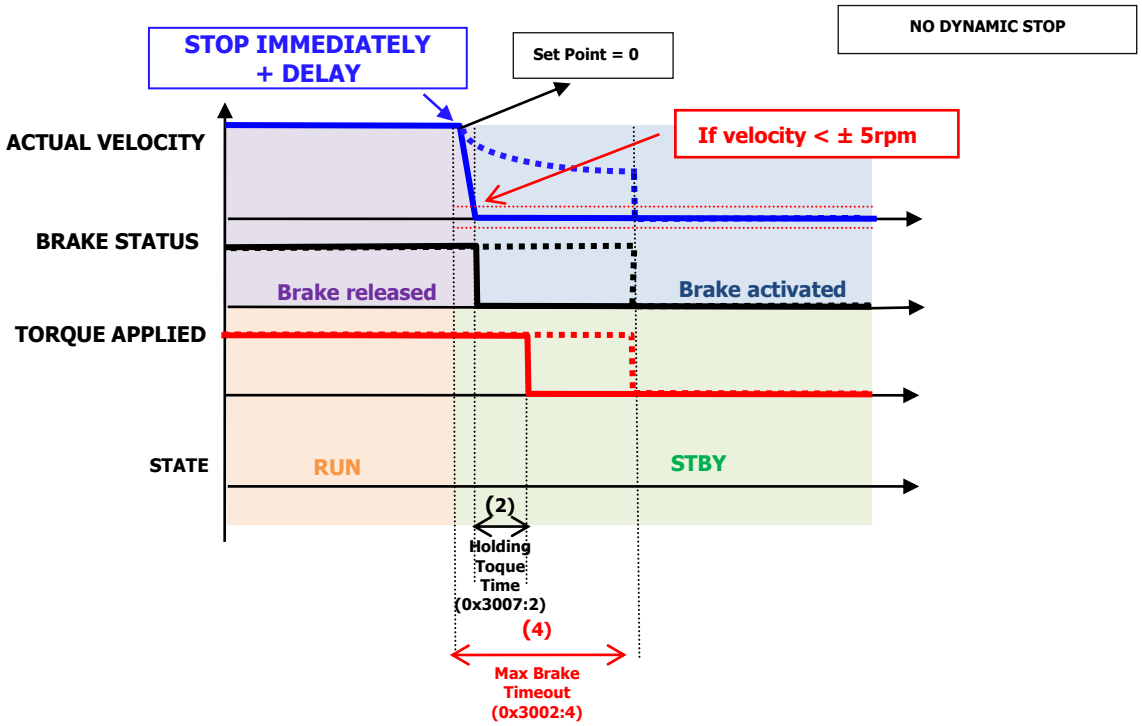


Figure 52 – Dynamic Stop timeframe: To Standby without Dynamic Stop (-1)

TRANSITION STOPPING MODE '-2':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is not applied.

The motor will stop because it is switch-off the drive power stage. When the velocity is on the range ±5 RPM it actives the electronic brake motor and It re-enables the drive function until the time delay defined by object 0x3007:2 is expired.

If the motor fails to stop and the time "Max Brake timeout" (object 3002_h: 4) is expired, then the torque will be released immediately and the brake is locked (see dashed line).

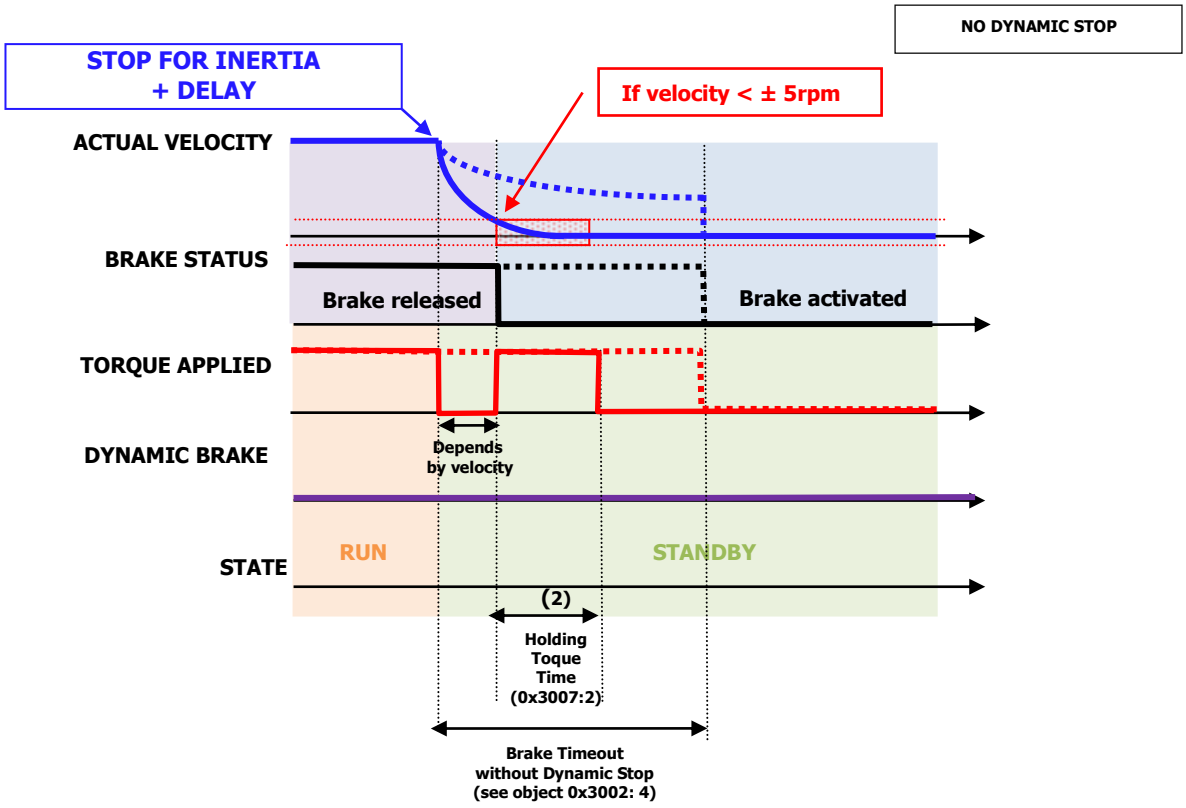


Figure 53 - Dynamic Stop timeframe: to Standby without Dynamic Stop (-2)

TRANSITION STOPPING MODE '-3':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is not applied.

The motor will stop immediately and the brake motor is locked immediately.

In this case the set point is also set to 0.

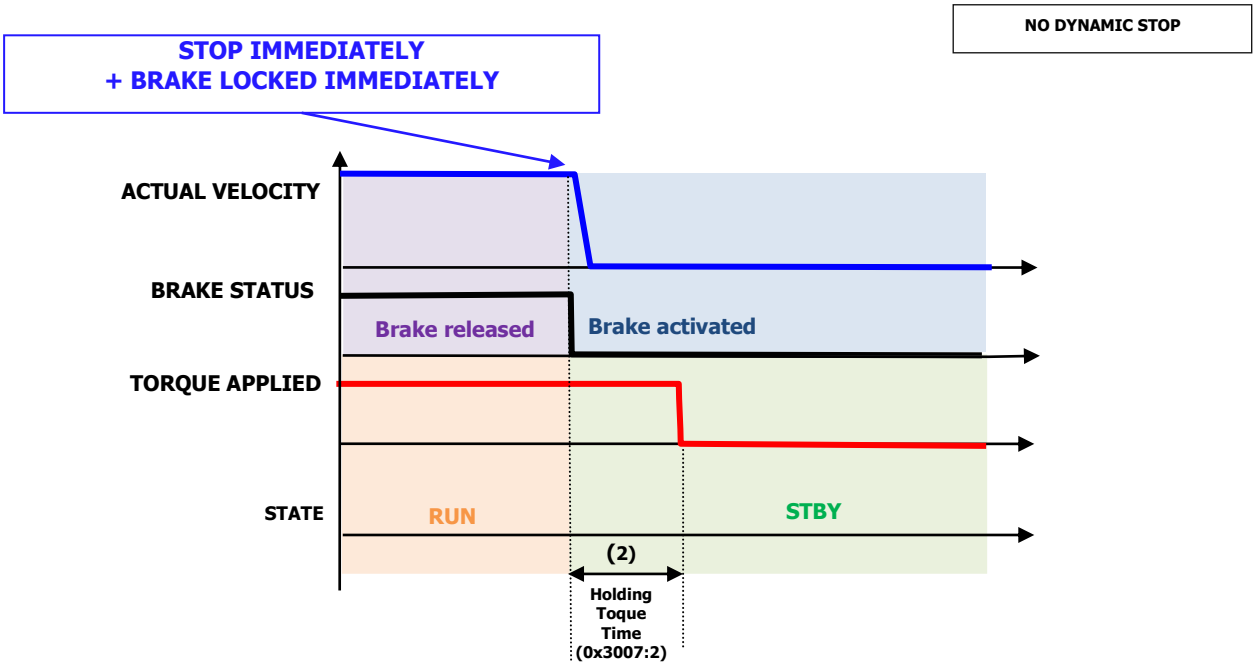


Figure 54 – Dynamic Stop timeframe: To Standby without Dynamic Stop (-3)

TRANSITION STOPPING MODE '-4':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is not applied.

The motor will stop immediately and the brake motor is locked immediately.

In this case the motor is switch off and it should be stop by inertia, but the brake stops the motor.

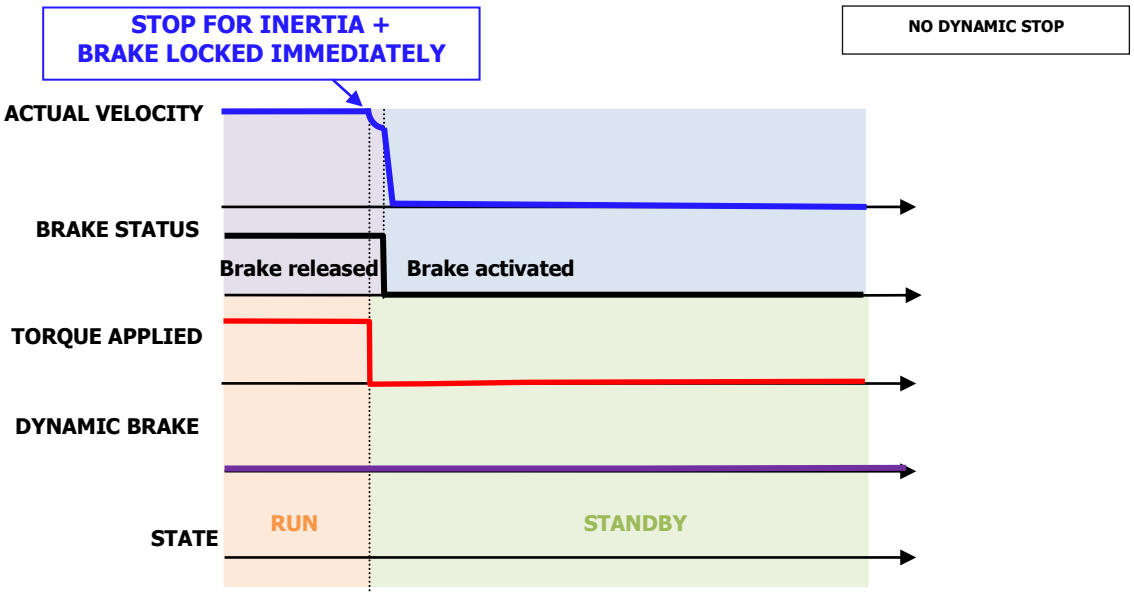


Figure 55 - Dynamic Stop timeframe: to Standby without Dynamic Stop (-4)

TRANSITION STOPPING MODE '-5':

The follow graph describes the timing stop of drive when the electronic Dynamic Stop is applied.

It slows down with down **S-RAMP** programmable defined by object 0x3007:8. When the velocity is on the range ±5 RPM it disables the drive.

This stopping mode is available only with motion profile configured as SRAMP (object 0x6086:0 = -2 or -3)

This condition will be present when the drive state move from RUN state (operation enabled in DSP402) to STANDBY state (Switched ON in DSP402).

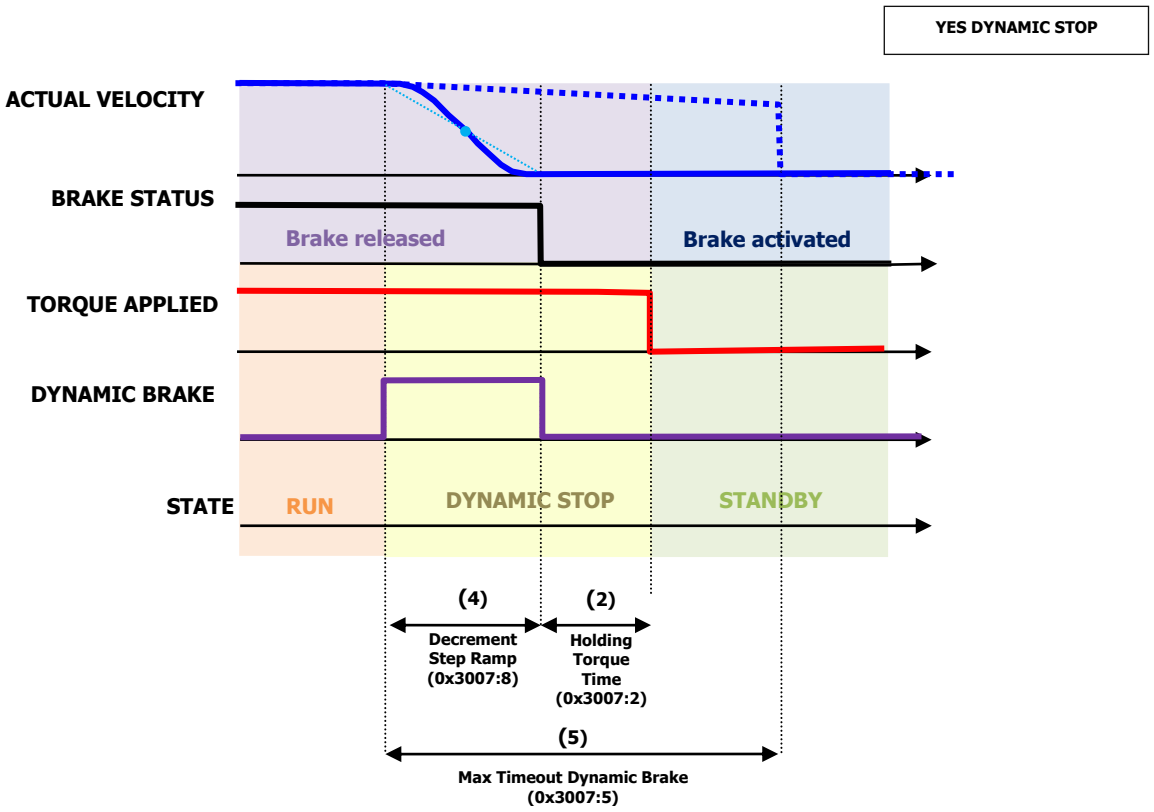


Figure 56 - Dynamic Stop timeframe: to Standby with Dynamic Stop (-5)

ABORT CONNECTION

From firmware 242 is possible to set the different behaviour in response to a failure/interruption in CAN bus communication defined "abort connection".

The failure/interruption can be occurred for bus-off, heartbeat (only if device is master), life guarding, NMT stopped state entered, reset application, and reset communication.

The object that define this feature is 0x6007 "Abort Connection Option Code".

Before firmware 242 this object was ONLY read and it was set with value -1- From firmware 2432 can be changed by editing in RAM or saving in e^2prom.

Value	Name	Meaning
-1	Manufacturer-specific	No reaction, the unit continues to run with the last set point except for • Bus-Off: FAULT SIGNAL, the drive goes in FAULT state • Life guarding: FAULT SIGNAL, after number of lost message descripted by "Life Time Factor" object (100Dh) the drive goes in FAULT state This mode is default and it is the same behaviour for old firmware
0	No action	No reaction, the unit continues to run with the last set point.
1	Fault signal	The drive will enter into Fault status . Execute specific stopping routine with Object 605Eh "Fault reaction option code" the drive will go in "FAULT" state.
2	Disable voltage command	If the failure/interruption in CAN bus Command the drive will move like command "Disable voltage". Execute specific stopping routine with Object 605Bh "Shutdown option code", the drive will go in "SWITCH ON DISABLED" state.

The configuration of statusword bits can be set to know when connection interruption occurs. By writing 0bject 0x3100 "Configuration 1 - Statusword" with value 3, bit 15 will take on the meaning of when connection interruption occurs.

To clear this bit the master must be set '1' the object 0x4102 "Settings Communication CAN" subindex 3.

Pay attention that if the interruption is for "NMT stopped state entered" the drive will be in STOPPED STATE for NMT protocol, this means that the PDO protocol is not enable and the master must move the NMT state in "OPERATIONAL" state.

The following picture shows the State Machine with "abort Connection Option Code" is set 1 or 2:

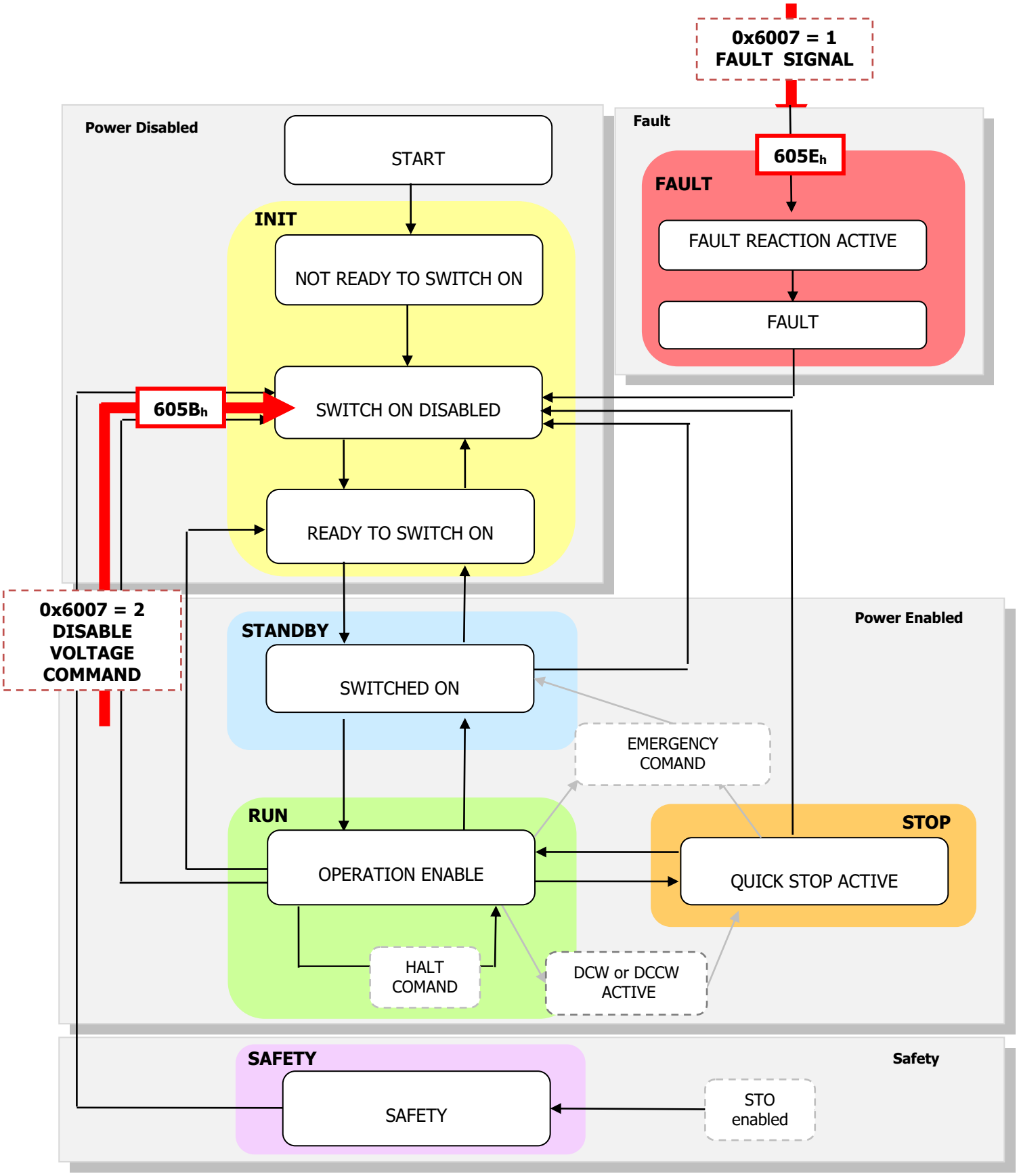


Figure 57 – Finite State Machine with object 0x6007

9. | DIAGNOSTIC

LED CODES

In the Smartris Standard there are 2 leds on right of the top that describe the status of drive

LED	2	3
Meaning	Status1 (green)	Status2 (yellow)
Led View	Led Status 1	Led Status 2

Table 49 - - Led Color Smartris Standard

In the Smartris Compact there are 5 leds near Power Connector that describe the status of drive

LED	1	2	3	4	5
Meaning	Power (red)	Status1 (green)	Status2 (yellow)	Led Run (blue)	Pulse (white)
Led View	Fixed The Drive is ON	Led Status 1	Led Status 2	Fixed when motor is energized	Rapid Blink (drive alive)

Table 50 - - Led color Smartris Compact

The combination of leds describe the status of Finite State Machine of drive:

- The **GREEN Led** (Status 1) is the status 1 of the drive
- The **YELLOW Led** (Status 2) is the status 2 of the drive
- The **LIGHT BLUE Led** (Run) is the RUN status [only Smartris "Compact"]

MACRO DRIVE STATE	CANOpen STATE	STATUS 1 LED GREEN	STATUS 2 LED YELLOW	LED VIEW SMARTRIS STANDARD	LED VIEW SMARTRIS COMPACT
INIT	Not Ready To Switch On	"BLINK" simultaneously	"BLINK" simultaneously	1 simultaneously 2 simultaneously	2 simultaneously 3 simultaneously 4 OFF
	Switch On Disabled Ready to Switch On	"BLINK" alternately	"BLINK" alternately	1 alternately 2 alternately	2 alternately 3 alternately 4 OFF
STANDBY	Switched On	"BLINK"	OFF	1 BLINK 50% 2 OFF	2 BLINK 50% 3 OFF 4 OFF
FAULT	Fault Fault reaction fault	"BLINK" [x]	"BLINK" [y]	1 see fault 2 chapter	2 see fault 3 chapter 4 OFF
RUN (RUNV or RUNC)	Operation Enabled	ON	OFF	1 ON 2 OFF	2 ON 3 OFF 4 ON
STOP	Quick Stop Active	ON	ON	1 ON 2 ON	2 ON 3 ON 4 ON
SAFETY	-	OFF	"BLINK"	1 OFF 2 BLINK	2 OFF 3 BLINK 4 OFF
COMMUN ERROR	-	OFF	ON	1 OFF 2 ON	2 OFF 3 ON 4 OFF

Table 51 - Led Status Smartris

ALARM

When fault alarm is occurred the Green Led and Yellow Led blink to defined an Erro Code:

Alarm	STATUS 1 CODE LED GREEN 1st Code	STATUS 2 CODE LED YELLOW 2nd Code	Alarm Description
A Group: (Temperature)			
Motor Over Temperature	**1**	**10**	Motor Temperature over threshold. The motor has reached a too high temperature for correct operation.
Heat Sink Over Temperature	**1**	**1**	Heat Sink Temperature over threshold. The Heat Sink has reached a too high temperature for correct operation.
Heat Sink Temperature Out Of Range	**1**	**3**	Heat Sink Temperature Sensor is out of range. Potential malfunction of the temperature sensor. (Contact the supplier).
Board Over Temperature	**1**	**4**	Internal Board Temperature over threshold. Too high a temperature for correct operation inside the drive.
Board Temperature Out Of Range	**1**	**5**	Internal Temperature Sensor out of range. Potential malfunction of the temperature sensor. (Contact the supplier).
Motor Temperature Out Of Range	**1**	**6**	Motor Temperature Sensor is out of range. • Potential malfunction of the temperature sensor. • (Contact the supplier).
B Group: (Feedback)			
Resolver	**2**	**10**	Check resolver connections, connectors and wiring of both sides.
Resolver Initialization	**2**	**4**	Initialization Fault for Resolver Device. (Contact the supplier).
Encoder	**2**	**5**	Incremental Encoder Fault
SinCosFault	**2**	**6**	SinCos Encoder Fault
Hall	**2**	**7**	Hall Sensors Fault
Distance Hall	**2**	**8**	Hall Sensors Fault
C Group: (Current and safety)			
Offset Current Sensor	**3**	**10**	Offset current sensor is out of range. [1]
Over Current	**3**	**1**	The current absorbed by the motor is beyond the set limit. Check Phase Motor connection and wire. Look for any short circuits.
STO Circuit	**3**	**4**	Reserved (contact Manufacturer)
Safety	**3**	**5**	STO occurred (If alarm activated, Contact Supplier)
D Group: (Voltage)			
Under Voltage	**4**	**1**	DC Bus voltage value lower than the limit threshold. Check mains voltage at terminals +, -.
Over Voltage	**4**	**2**	DC Bus voltage value higher than the limit threshold. Check mains voltage at terminals +, -
E Group: (Functionality)			
Velocity Fault	**5**	**10**	Following Error: The actual speed different from the target Speed.
I^2T Overload Protection	**5**	**2**	I2T overload motor protection reached.
Hardware	**5**	**3**	Error Hardware (Contact Supplier)
External HW	**5**	**4**	Error CAN Interface (Contact Supplier)
Over Speed	**5**	**8**	Over Speed
F Group: (Communication and Configuration)			

Alarm	STATUS 1 CODE LED GREEN 1st Code	STATUS 2 CODE LED YELLOW 2nd Code	Alarm Description
E²prom	**6**	**1**	Parameter Fault stored in E2prom.
CanOpen	**6**	**2**	Communication Faultwith CANOpen
Sincos Fault	**6**	**3**	Internal Communication Fault(Contact Supplier).
Configuration Parameters	**6**	**4**	Configuration Parameters Fault(Contact Supplier).
Profile Generic	**6**	**5**	Error Configuration Profile: Mode Of Operation
Torque Profile	**6**	**6**	Error Torque Profile
Velocity Profile	**6**	**7**	Error Velocity Profile
Homing Profile	**6**	**8**	Error Homing Profile
G Group: (Programming)			
Program Fault	**7**	**1**	Code Programming Fault (Contact Supplier).
H Group: (Programming)			
Factory Param	**8**	**1**	Error Programming Factory Parameters (Contact Supplier).
CAN Param	**8**	**2**	Error Programming CAN Parameters (Contact Supplier).
E2prom Function	**8**	**3**	Function Code for EEPROM R/W (Contact Supplier).
I Group: (protoocol)			
ModBus	**9**	**1**	Communication Lost ModBus
L Group: (Positioner)			
Positioner	**10**	**1**	Error Positioner Profile

Table 52 - Diagnostic

To have more information about diagnostic It is possible:

- Know the Error Code of alarm occurred read object 0x1001 and 0x603F: it describes the last error before to clear with a reset alarm command (with controlword or via NMT protocol)
- Have other information about the error, some alarms have also a subcode to know it this it needs to read the Object 0x4200 (Alarm Monitoring)
- Read the History List alarm with object 0x1003: it contains an error history list of 15 alarms saved in the FIFO queue.
- Read the photograph of sensitive variables during the alarm occurred. Photographs of the last 5 errors are saved
    - Log Error 1: 0x5001 (Last occurred)
    - Log Error 2: 0x5002
    - Log Error 3: 0x5003
    - Log Error 4: 0x5004
    - Log Error 5: 0x5005 (Old occurred)

  The object 0x5000 describe the state of error logger, the object 0x5010 are reserved for Manufacturer.
- Read alarm settings with objects 0x3010 and 0x3011, but are reserved by Manufacturer.
- From firmware 236 is possible to clean the alarm with bit 7 of controlword, the drive will move from "fault" state to "switch-On disable" state.

Object 1001h: Error Register

This object is of standard CiA DS301. The error register is a field of 8 bits, each for a certain error type. If an error occurs the bit has to be set.

Object Description:

Index	Name	Object Code	Data Type	Category
1001h	Error Register	VAR	U8	O

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Error Register	ro	no	-

Value Definition:

BIT	Meaning
0	Generic Error
1	Current
2	Voltage
3	Temperature
4	Communication Error (overrun, error state)
5	Device Profile Specific
6	Reserved
7	Manufacturer Specific

Object 603Fh: Error Code

This object is of standard CiA DS402. This object shall provide the error code of the last error which occurred in the drive device.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
603Fh	Error code	VAR	U16	Optional

Entry Description:

Sub-Index	Name	Access	PDO mapping	Value Range	Default Value
00h	Error code	ro	yes	See table Emergency	-

The 603Fh object is the error code of (last) alarm occurred. The meaning is described in the Table 22 - Emergency Description of Emergency Chapter. The column "Error Code" is the corresponding value.

Object 4200h: Alarm Monitoring

This object describes the Alarm Monitor.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
4200h	Alarm Monitoring	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	5	5
01h	Error Code	rw	no	[0, 32767]	0
02h	Manufacturer Code LSB	rw	no	[0, 32767]	0
03h	Manufacturer Code MSB	ro	no	[0, 32767]	0
04h	free	ro	no	[0, 32767]	-
05h	free	ro	no	[0, 32767]	-

Example:

Read Error Code:

Rx: Id 0x601 – 40 00 42 01 00 00 00 00
Tx: Id 0x581 – 4B 00 42 01 01 64 00 00

Rx: [cmd: 0x40] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00000000] – Read Error Code
Tx: [resp: 0x4B] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00006401] – Error Code 0x6401

Read Manufacturer Code LSB:

Rx: Id 0x601 – 40 00 42 02 00 00 00 00
Tx: Id 0x581 – 4B 00 42 02 0A 00 00 00

Rx: [cmd: 0x40] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00000000] – Read Manufacturer Code LSB
Tx: [resp: 0x4B] [Index: 0x4200] [sub-index: 0x01] [Data: 0x0000000A] - Man Code 0xA

Read Manufacturer Code MSB:

Rx: Id 0x601 – 40 00 42 03 00 00 00 00
Tx: Id 0x581 – 4B 00 42 03 00 00 00 00

Rx: [cmd: 0x40] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00000000] – Read Manufacturer Code MSB
Tx: [resp: 0x4B] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00000000] – Man Code 0x0

Error Code = Error Code 0x6401
Manufacturer Code = 0x000000000000000000A

Example:

Read Error Code:

Rx: Id 0x601 – 40 00 42 01 00 00 00 00
Tx: Id 0x581 – 4B 00 42 01 04 8C 00 00

Rx: [cmd: 0x40] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00000000] – Read Error Code
Tx: [resp: 0x4B] [Index: 0x4200] [sub-index: 0x01] [Data: 0x00008C04] – Error Code 0x8C04

Error Code = Error Code 0x8C04
Manufacturer Code = 0x0000000000000000000

Object 1003h: Pre-defined Error Field

This object contains an error stack with up to eight entries. It holds errors that have occurred on the device and have been signalled via Emergency Object. It is an error history.
Writing to sub index 0 deletes the entire error history.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
1003h	Pre-defined Error Field	VARIABLE	U32	Mandatory

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Number of Errors	rw	no	-

01_h	Error Code last alarm occurred	ro	no	-
02_h	Error Code before last alarm	ro	no	-
03_h... FF_h	Error Code Older Alarm	ro	no	-

If a new error occurs, it is entered in sub-index 1. The already existing entries in sub-indices 1 to 15 are moved back one position. The error in sub-index 15 is thereby removed.

The number of errors that have already occurred can be read from the object with sub-index 0.

If no error is currently entered in the error stack, it is not possible to read one of the 15 sub-indices 1–15 and an error is sent in response. The drive responses with an SDO abort message (abort code: 0800 0024_h).

The pre-defined error field has the following structure

Bit MSB **31** **24**	**23** **16**	**Bit LSB** **150**
Manufacturer-specific error code	Error register	Error code

Writing 00h to sub-index 00h shall delete the entire error history (empties the array).
Other values than 00h are not allowed and shall lead to an abort message (error code: 0609 0030_h).

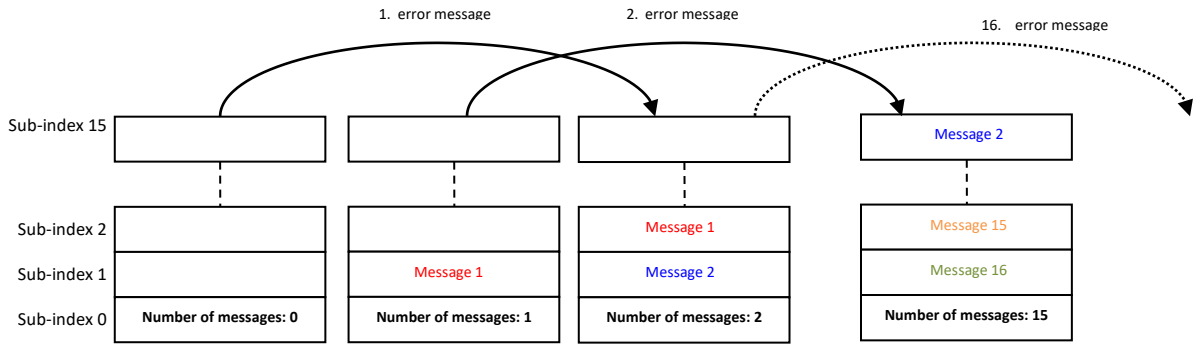


Figure 58 - History Message List

Example:

1- This procedure is to read the emergency history.

Rx: Id 0x601 – 40 03 10 00 00 00 00 00 - Read number of errors (sub-Index 0)
Tx: Id 0x581 – 4F 03 10 00 02 00 00 00 - Response from CANopen. Byte 5: 02h means there are 2 error messages recorded

Rx: [cmd: 0x40] [Index: 0x1003] [sub-index: 0x00] [Data: 0x00000000] – Command Read
Tx: [resp: 0x4F] [Index: 0x1003] [sub-index: 0x00] [Data: 0x00000002] – Successful

2- To delete the emergency messages by writing 0 to sub-index 0:

Rx: Id 0x601 – 22 03 10 00 00 00 00 00 - Delete the emergency messages
Tx: Id 0x581 – 60 03 10 00 00 00 00 00 - Response from CANopen

Rx: [cmd: 0x22] [Index: 0x1003] [sub-index: 0x00] [Data: 0x00000000] – Command delete
Tx: [resp: 0x60] [Index: 0x1003] [sub-index: 0x00] [Data: 0x00000000] – Successful

3- To Read error message (sub-index 1 ...15). The error message code description is in section

"Error Code" on the Emergency chapter.

Rx: Id 0x601 – 40 03 10 01 00 00 00 00 - Read error message
Tx: Id 0x581 – 43 03 10 01 00 FF 81 00 - Response from CANopen

Rx: [cmd: 0x40] [Index: 0x1003] [sub-index: 0x01] [Data: 0x00000000] – Command read
Tx: [resp: 0x43] [Index: 0x1003] [sub-index: 0x00] [Data: 0x00008100] – Successful

4- To Read error message (sub-index 1...15) without alarm. Request a sub-index without occurred error the following error message will be received

Rx: Id 0x601 – 40 03 10 01 00 00 00 00 - Read error message
Tx: Id 0x581 – 80 03 10 01 11 00 09 06 - Response from CANopen → ABORT CODE

Rx: [cmd: 0x40] [Index: 0x1003] [sub-index: 0x01] [Data: 0x00000000] – Command read
Tx: [resp: 0x80] [Index: 0x1003] [sub-index: 0x00] [Data: 0x06090011] – ABORT CODE

Object 5000_h: Logger Alarm Data

This object describe the state of Logger Error.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5000_h	Logger Alarm Data	ARRAY	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Index Logger	ro	no	-
02_h	Alarm Counter Total	ro	no	-

Object 5001_h: Logger Error 1

This object describes the photography of some important variables during the occurrence of the fault. There are 32 data available. Contact the manufacturer for more information. This is Error is Last occurred.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5001_h	Logger Error 1	ARRAY	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Data 1	ro	no	-
02_h	Data 2	ro	no	-
....	....	...	...	...
20_h	Data 32	ro	no	-

Object 5002_h: Logger Error 2

This object describes the photography of some important variables during the occurrence of the fault. There are 32 data available. Contact the manufacturer for more information

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5002_h	Logger Error 2	ARRAY	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Data 1	ro	no	-
02_h	Data 2	ro	no	-
....	....	...	...	...
20_h	Data 32	ro	no	-

Object 5003_h: Logger Error 3

This object describes the photography of some important variables during the occurrence of the fault. There are 32 data available. Contact the manufacturer for more information

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5003_h	Logger Error 3	ARRAY	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Data 1	ro	no	-
02_h	Data 2	ro	no	-
....	....	...	...	...
20_h	Data 32	ro	no	-

Object 5004_h: Logger Error 4

This object describes the photography of some important variables during the occurrence of the fault. There are 32 data available. Contact the manufacturer for more information

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5004_h	Logger Error 4	ARRAY	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Data 1	ro	no	-
02_h	Data 2	ro	no	-
....	....	...	...	...
20_h	Data 32	ro	no	-

Object 5005_h: Logger Error 5

This object describes the photography of some important variables during the occurrence of the fault. There are 32 data available. Contact the manufacturer for more information

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5005_h	Logger Error 5	ARRAY	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Number of Errors	rw	no	-
01_h	Data 1	ro	no	-
02_h	Data 2	ro	no	-
....	....	...	...	...
20_h	Data 32	ro	no	-

Object 5010h: Logger Alarm (RESERVED)

This object is reserved. Contact the manufacturer for more information.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
5010h	Logger Alarm Data	ARRAY	INT32	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Number of Errors	rw	no	-
01h	Key Log Alarm	ro	no	-
02h	Last Saved Seconds	ro	no	
03h	Power On Seconds	ro	no	-
04h	Run seconds	…	…	…

Object 3010h: Alarm Option (RESERVED)

Contact Manufacturer for additional information

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3010h	Alarm Option	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	4	3
01h	Alarm Option 1	ro	no	[0 … 32767]	defined by application
02h	Alarm Option 2	ro	no	[0 … 32767]	defined by application
03h	Alarm Option 3	ro	no	[0 … 32767]	defined by application
04h	Alarm Option 4	ro	no	[0 … 32767]	defined by application

Object 3011h: Alarm Mask (RESERVED)

Contact Manufacturer for additional information.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3011h	Alarm Mask	ARRAY	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Number Of Entries	ro	no	3	4
01h	Mask Alarm 1	ro	no	[0 … 32767]	defined by application
02h	Mask Alarm 2	ro	no	[0 … 32767]	defined by application
03h	Mask Alarm 3	ro	no	[0 … 32767]	defined by application
03h	Mask Alarm 4	ro	no	[0 … 32767]	defined by application

WARNING

The warnings defined are the following:

- Warning communications CAN
- Warning i^2t Limit
- Warning E^2prom
- Warning Update Parameters
- Warning Factory Parameters
- Warning CANopen parameters
- Warning DAC configuration
- Warning Temperature
- Warning Limitation Torque Function
- Warning Analog Mode
- Warning Protocol

From firmware 242 the monitoring of warning is programmable.

There is object 0x3900 Warning Settings to program the monitor warning (subindex 1 and 2).

- Warning Enable User (subindex 1): it is possible enable the warning FLAG by setting the corresponding bit to '1'. If the warning occurs, the bit corresponding to the warning in the 'Warning' object 0x2003 will be 1.
- Warning Select Output (subindex 2): it is possible enable the warning OUTPUT of bit of statusword and/or the digital output setting the bit at '1'. If the at least warning selected occurs, the output will be 1.

Every warning has your emergency code, see table in the chapter "EMERGENCY MESSAGE (EMCY)" .
It is possible to clear all Warning Code from emergency protocol (see column "Emergency Error Code" in the "Warning Table") writing '1' the object 0x3401:0.

When the bit of warning flag is set, the bit stays set to 1 until it is cleared with controlword (bit 11). This is the standard configuration for Lafert. If it wants that the warning flag clears automatically it needs to set the configuration, it has to write the object 0x3104:0 at 1.

WARNING TABLE:

Bit	Warning	EMRGY ERROR CODE	Description	Autom Validation Enable	Default Enabled
0	Warning Node Guarding	0x8133	Warning to communicate that the master lost 1 Node Guarding Message. This warning is enable only if the Node Guarding is active	NO	YES
1	Warning I2T Limit	0x2351	The warning will be 1 if the drive is in i2T limitation. If the overload alarm is active, the warning will become 1 along with the fault alarm.	YES	YES
2	Warning command CAN E^2prom	0x8B01	Warning needs to communicate that command store/restore/load was not accepted because the drive isn't in "ready to switch-on" or "disabled" state	NO	YES
3	Warning Update Param Manufacturer	0x6001	Reserved (manufacturer)	YES	YES
4	Warning Golden Image Writing	0x5A01	Reserved (manufacturer)	NO	YES

5	CANopen Disabled	**No Code**	Reserved (manufacturer)	NO	YES
6	Warning error Init object CanOpen	**No Code**	Reserved (Analog Mode)	NO	YES
7	Warning DAC configuration	0x6002	Configuration DAC is not valid	NO	YES
8	warning Temperature Heat Sink	0x4301	Warning Heat Sink Temperature. This bit will be '1' before the alarm fault.	YES	YES
9	warning Temperature Board	0x4501	Warning Logic Board Temperature. This bit will be '1' before the alarm fault.	YES	YES
10	warning Temperature Motor	0x4A01	Warning Motor Temperature. This bit will be '1' before the alarm fault.	YES	YES
11	Warning Error Bus CAN (Passive or Busoff)	0x8001	Warning Error Bus CAN (Passive or Bus-Off). This bit will be '1' before the alarm fault. - If the drive is configured with the 'high sensibility' ("settings communication CAN" object 0x4102:1 = 0) the drive will go to fault when the CAN peripheral is in error passive, the warning will become 1 along with the fault alarm. - If the drive is not configured with the 'high sensibility' ("settings communication CAN" object 0x4102:1 = 1) the drive will go to fault when the CAN peripheral is in error passive, the warning will set '1' without fault alarm.	YES	YES
12	warning Function Tq Limit Activated	0x8301	Warning to communicate that that the Torque Limitation is activated (object 0x60E0 and 0x60E1 value < 100%)	YES	YES
13	warning Tq Limitation	0x8322	Warning enable only if object 0x60E0 or 0x60E1 have a value <100%, in this case if the drive is in limitation this warning will be 1	YES	YES
14	warning Analog Parameters	0x8D06	Reserved (Analog Mode)	NO	NO
15	warning force 232	**No Code**	Reserved (manufacturer)	NO	NO
16	warning Limit Pos disabled	**No Code**	Reserved (future implementation)	NO	NO
17	warning Profile aborted Target Pos Limited	**No Code**	reserved (future implementation)	NO	NO
18	warning Profile aborted "Change Set Immediately"	**No Code**	reserved (future implementation)	NO	NO
19	warning Following Error	**No Code**	reserved (future implementation)	NO	NO
20	Warning i2t not rearmed	0x2353	Warning to protection the motor. During power-on if the drive reaches the maximum number of consecutive i^2t fault, this bit is set until the time expires. If the drive will go in RUN with warning actives then the alarm 0x2352 is occurred.	YES	YES
21	Warning Speed PID Saturation	**No Code**	Reserved (manufacturer)	NO	NO

22	Warning other protocol type set	0x8C06	Warning to communicate that the protocol used is different with protocol set	NO	NO
23	warning Calculator jerk	**No Code**	Reserved (manufacturer)	NO	NO

Table 53 - Warning Table

Object 3900h: Warning Settings

This object configure the warning.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3900h	Warning Settings	ARRAY	INT32	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00h	Number of Errors	rw	no	-
01h	Warning Enable User	rw	no	1081343
02h	Warning Select Output	rw	no	1081343
03h	free	rw	no	-
04h	free	rw	no	-

Example:

I want to enable only warning Temperature: bit 8, 9, 10.

I have to write object Warning Enable User (0x3900 subindex 1) the value:
➔ 1792 = 0111 0000 0000b = 0x700.
I have to write object Warning Select Output (0x3900 subindex 2) the value:
➔ 1792 = 0111 0000 0000b = 0x700.

If the Warning Temperature Board occurs then:
- the object Warning flag (0x2003) will write the value: 512 = 0010 0000 0000b = 0x200.
- the output warning bit of statusword (0x6041) and digital output will be 1

If other warnings (other than temperature warnings) occur, they will not be recorded in them 0x2003 object.

With the same configuration I want to enable the warning I2T Limit: bit 1.
- I have to write object Warning Enable User (0x3900 subindex 1) the value:
➔ 1794 = 0111 0000 0010b = 0x702.

I don't change object Warning Select Output (0x3900 subindex 2) the value:
➔ 1792 = 0111 0000 0000b = 0x700.

If the Warning i2t occurs then:
- the object Warning (0x2003) will write the value: 2 = 0000 0000 0010b = 0x2.
- the output warning bit of statusword (0x6041) and digital output will be 0

If also the Warning Temperature Board occurs then:
- the object Warning (0x2003) will write the value: 258 = 0001 0000 0010b = 0x102.
- the output (warning bit of statusword and digital output) will be 1

Object 2003h: Warning Flag

This object logs the drive's warning. Every bit is a flag of warning occurs, they are defined in warning table. To enable this word you have to write the object 0x3900:1.
This is only read and it is a monitor.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
2003h	Warning Flag	VAR	UNSIGNED32	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value
00h	Warning Flag	ro	YES	See table "Warning Table"

Object 3104h: Configuration 5 – Warning Configuration

The object defines the different configuration of warning.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3104h	Config 5 – Warning	VAR	INTEGER16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value Range	Default Value
00h	Config 5 - Warning	rw	no	[0 … 32767]	0

Configuration:

Value	Name	Description
0	**STANDARD LAFERT**	When the bit of warning flag is set, the bit stays set to 1 until it is cleared with controlword.
1	**AUTOMATIC VALIDATION**	The bit stays set to 1 until the warning is present, when the warning is not present, the flag is cleared automatically. (Attention not all warnings have this function active see column "Automatic Validation Enable" in the "Warning Table")

Object 3401h: Warning Emergency Protocol

This object clear all error code of warning from emergency protocol.

Object Description:

Index	Name EDS	Object Code	Data Type	Category
3401h	Warning Emergency Protocol	VAR	INT16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Value
00h	Warning Emergency Protocol	rw	NO	0

Configuration:

Value	Name	Description
0	STANDARD LAFERT	The error code of warning is displayed in emergency protocol (CAN BUS the error code can be read with COBID 0x80+IdNode
1	CLEAR WARNING CODE IN EMERGENCY PROTOCOL	All error code of warning are cleaned in emergency protocol

10. | APPENDIX - FIRST CONFIGURATION

POWER-ON

On the Power-On the, if the CAN communication is OK, the drive sends these message:

ID	Name	Node	Transfer data	Error	Data
701	HB_NG_001	Node1	Boot-up		00
81	EMCY_001	Node1	00 00 00 00 00 00 00 00	E	00 00 00 00 00 00 00 00

The drive has default values.

It is possible to change the default value writing via SDO protocol the corresponding index and sub-index object and store in e^2prom.

After that you do not need to write at every power on, and values are updated by memory e^2prom.

Caution

During STORE procedure, drive must not be in "Operation Enabled" state or in "Quick Stop Active" state.

To Store parameter in e^2prom (permanently) it need to send the object 1010_h and reset

ID	Name	Node	Transfer data	Error	Data
601	CSDO_001	Node1	73 61 76 65		23 10 10 01 73 61 76 65
581	SSDO_001	Node1			60 10 10 01 00 00 00 00

and Reset all Nodes (or switch-off/switch-on)

ID	Name	Node	Transfer data	Error	Data
0	NMTZeroMsg				81 00
701	HB_NG_001	Node1	Boot-up		00
81	EMCY_001	Node1	00 00 00 00 00 00 00 00	E	00 00 00 00 00 00 00 00

HOW TO CHANGE ID-NODE

Id-Node has default Value = 1. The following steps describe how to change the Id-Node.

Caution

To Change Id-Node it is mandatory connect one drive on the time with Master Controller

Procedure Set New Id-Node Value (Write SDO)

The Master Control sends SDO message ID = 0x601 (defined 0x600 + Id node)

- Data Value "command" = 0x2F
- Data Value "Index" = 0x2000
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = new Id-Node value (for Example 3)

The drive answers SDO message ID = 0x581 (defined 0x580 + Id node)

- Data Value "command" = 0x60
- Data Value "Index" = 0x2000
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

The following picture shows the SDO message:

ID	Name	Node	Transfer data	Interpretation	Data
601	CSDO_001	Node1	03	[2000,00] Initiate Download Rq. expedited	2F 00 20 00 03 00 00 00
581	SSDO_001	Node1		[2000,00] Initiate Download Rsp	60 00 20 00 00 00 00 00

Procedure Save New Value in e²prom (Write SDO)

Caution

During STORE procedure, drive must not be in "Operation Enabled" state or in "Quick Stop Active" state.

The Master Control sends SDO message ID = 0x601 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x1010 (Store)
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 0x73617665 (means "save" in ASCII code)

The drive answers SDO message ID = 0x581 (defined 0x580 + Id node)

- Data Value "command" = 0x60
- Data Value "Index" = 0x1010
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 0

The following picture shows the SDO message:

ID	Name	Node	Transfer data	Interpretation	Data
601	CSDO_001	Node1	73 61 76 65	[1010,01] Initiate Download Rq. expedited "save"	23 10 10 01 73 61 76 65
581	SSDO_001	Node1		[1010,01] Initiate Download Rsp	60 10 10 01 00 00 00 00

The procedure continues …

after stored parameters continue with reset

Reset All Nodes (NMT Protocol)

The sends message ID = 0x00 (NMT protocol)

- Data Value "command" = 0x81
- Data Value "Index" = 0x00

The following picture shows the SDO message:

Dir	ID	Name	Node	Transfer data	Interpretation	Error	Data
Rx	0	NMTZeroMsg			Reset all nodes		81 00

After Reset (NMT Protocol)

The drive answers message BOOT-UP message ID = 0x703 (defined 0x700 + Id node)

- Data Value "Index" = 0x00

ID	Name	Node	Transfer data	Interpretation	Error	Data
703			Boot-up			00
703		_	Boot-up			00

The procedure continues …

Then the drive sends emergency messages (emergency protocol)

The drive sends message ID = 0x83 (defined 0x80 + Id node)

- Data Value "Error Code" = 0x0
- Data Value "Reg" = 0x0
- "Data" = 0

It means "ERROR RESET or NO ERROR"

ID	Name	Node	Transfer data	Interpretation	Error	Data
83	EMCY_003		00 00 00 00 00 00 00 00	Error reset or no error	E	00 00 00 00 00 00 00 00

The procedure continues …

Then, to be sure, the drive accepted previous id-node changed.

Procedure Verify New Id-Node (Read SDO)

The Master Control sends SDO message ID = 0x603 (defined 0x600 + New Id node)

- Data Value "command" = 0x40
- Data Value "Index" = 0x2000
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0x00

The drive answers SDO message ID = 0x583 (defined 0x580 + New Id node)

- Data Value "command" = 0x4F
- Data Value "Index" = 0x2000
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0x3

The following picture shows the SDO messages:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003			[2000,00] Initiate Upload Rq.		40 00 20 00 00 00 00 00
583	SSDO_003		03	[2000,00] Initiate Upload Rsp. expedited		4F 00 20 00 03 00 00 00

HOW TO CHANGE BAUDRATE

BaudRate Default is 1000Kbit. The following steps describe how to change the BaudRate.

Procedure Set New Baudrate Value (Write SDO)

The Master Control sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x2B
- Data Value "Index" = 0x2001
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = new BaudRate (for Example 500K = 0x01F4)

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x60
- Data Value "Index" = 0x2001
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

The following picture shows the SDO messages:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003		f4 01	[2001,00] Initiate Download Rq. expedited		2B 01 20 00 F4 01 00 00
583	SSDO_003			[2001,00] Initiate Download Rsp		60 01 20 00 00 00 00 00

Procedure Save New Value In e²prom (Write SDO)

Caution

During STORE procedure, drive must not be in "Operation Enabled" state or in "Quick Stop Active" state.

The Master Control sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x1010 (Store)
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = *0x65766173* (means "save" in ASCII code)

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x60
- Data Value "Index" = 0x1010
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 0

The following picture shows the SDO message:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003		73 61 76 65	[1010,01] Initiate Download Rq. expedite...		23 10 10 01 73 61 76 65
583	SSDO_003			[1010,01] Initiate Download Rsp		60 10 10 01 00 00 00 00

The procedure continues ...

After stored parameters proceed with Reset

Reset All Nodes (NMT Protocol)

The Master Control sends message ID = 0x00 (NMT protocol)

- Data Value "command" = 0x81
- Data Value "Index" = 0x00

The following picture shows the SDO message:

Dir	ID	Name	Node	Transfer data	Interpretation	Error	Data
Rx	0	NMTZeroMsg			Reset all nodes		81 00

After Reset (NMT Protocol)

The drive answers message BOOT-UP message ID = 0x703 (defined 0x700 + Id node)

- Data Value "Index" = 0x00

ID	Name	Node	Transfer data	Interpretation	Error	Data
703			Boot-up			00
703	_		Boot-up			00

The procedure continues ...

then the drive sends emergency messages (emergency protocol)

The drive sends message ID = 0x83 (defined 0x80 + Id node)

- Data Value "Error Code" = 0x0
- Data Value "Reg" = 0x0
- "Data" = 0

It means "ERROR RESET or NO ERROR"

ID	Name	Node	Transfer data	Interpretation	Error	Data
83	EMCY_003		00 00 00 00 00 00 00 00	Error reset or no error	E	00 00 00 00 00 00 00 00

The procedure continues ...

then, to be sure, the drive accepted previous id-node changed

Procedure Verify New BaudRate (Read SDO)

The Master Control sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x40
- Data Value "Index" = 0x2001
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0x00

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x4B
- Data Value "Index" = 0x2000
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0x01F4

The following picture shows the SDO messages:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003			[2001,00] Initiate Upload Rq.		40 01 20 00 00 00 00 00
583	SSDO_003		f4 01	[2001,00] Initiate Upload Rsp. expedited		4B 01 20 00 F4 01 00 00

HOW TO CHANGE THE USER UNITS

Lafert Servo Drive has a default unit [inc/s] for velocity objects and [inc/s^2] for acceleration objects. If it is necessary to change the user unit (for example in [rpm] for velocity objects and [rpm/s] for acceleration objects) it has to change the factory group object.

The velocity factory group is

$$Velocity\ Factor = \frac{Numerator}{Divisor}$$

Numerator and divisor of the Velocity Factor has to be entered separately.

The default value is [inc/s]. The numerator and the divisor are set "1" in e^{2}prom.

To change the default unit it has to write the numerator and divisor in the object index 0x6096 and save in e^{2}prom the new value:

$$Velocity[internalunit] = Velocity[user\ unit] \times (\frac{Numerator}{Divisor})$$

Example:

The speed-set point provision is to be made in revolutions per minute (rpm).

$$Velocity[inc/sec] = Velocity[rpm] \times \left(\frac{Numerator}{Divisor}\right)$$

If the resolution of encoder is 2^{13} =16384 then the Numerator is 16384 and the Divisor is 60

The Acceleration Factory Group has the same consideration (object index 0x6097)

Procedure Set New Factory Group Values (Write SDO)

Write **NUMERATOR** Velocity Factory Group (value = 16384):

The Master Control sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x6096
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 16384 = 0x4000

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x6096
- Data Value "Index" = 0x01
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

Write **DIVISOR** Velocity Factory Group (value = 60):

The Master Control sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x6096
- Data Value "Sub-Index" = 0x02

- Data Value "Data" = 60 = 0x3C

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x6096
- Data Value "Index" = 0x02
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

Write **NUMERATOR** Acceleration Factory Group (value = 16384):

The Master Controller sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x6097
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 16384 = 0x4000

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x6097
- Data Value "Index" = 0x01
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

Write **DIVISOR** Acceleration Factory Group (value = 60):

The Master Controller sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x6097
- Data Value "Sub-Index" = 0x02
- Data Value "Data" = 60 = 0x3C

The drive answers SDO message ID = 0x583(defined 0x580 + Id node)

- Data Value "command" = 0x6097
- Data Value "Index" = 0x02
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

The following picture shows the SDO messages:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003		00 40 00 00	[6096,01] Initiate Download Rq. expedited ".@.."		23 96 60 01 00 40 00 00
583	SSDO_003			[6096,01] Initiate Download Rsp		60 96 60 01 00 00 00 00
603	CSDO_003		3c 00 00 00	[6096,02] Initiate Download Rq. expedited "<..."		23 96 60 02 3C 00 00 00
583	SSDO_003			[6096,02] Initiate Download Rsp		60 96 60 02 00 00 00 00
603	CSDO_003		00 40 00 00	[6097,01] Initiate Download Rq. expedited ".@.."		23 97 60 01 00 40 00 00
583	SSDO_003			[6097,01] Initiate Download Rsp		60 97 60 01 00 00 00 00
603	CSDO_003		3c 00 00 00	[6097,02] Initiate Download Rq. expedited "<..."		23 97 60 02 3C 00 00 00
583	SSDO_003			[6097,02] Initiate Download Rsp		60 97 60 02 00 00 00 00

The procedure continues …

Now the user units are ready to save in E²prom, but they aren't available. **It MUST STORE and RESET**.

Procedure Save New Value in e²prom (Write SDO)

Caution

During STORE procedure, drive must not be in "Operation Enabled" state or in "Quick Stop Active" state.

The Master Controller sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x1010 (Store)
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = *0x65766173* (means "save" in ASCII code)

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x60
- Data Value "Index" = 0x1010
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 0

The following picture shows the SDO message:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003		73 61 76 65	[1010,01] Initiate Download Rq. expedite...		23 10 10 01 73 61 76 65
583	SSDO_003			[1010,01] Initiate Download Rsp		60 10 10 01 00 00 00 00

The procedure continues …

After stored parameters proceed with drive reset

Reset All Nodes (NMT Protocol)

The Master Controller sends message ID = 0x00 (NMT protocol)

- Data Value "command" = 0x81
- Data Value "Index" = 0x00

The following picture shows the SDO message:

Dir	ID	Name	Node	Transfer data	Interpretation	Error	Data
Rx	0	NMTZeroMsg			Reset all nodes		81 00

After Reset (NMT Protocol)

The drive answers message BOOT-UP message ID = 0x703 (defined 0x700 + Id node)

- Data Value "Index" = 0x00

ID	Name	Node	Transfer data	Interpretation	Error	Data
703			Boot-up			00
703		_	Boot-up			00

The procedure continues ...

Then the drive sends Emergency Messages (EMERGENCY PROTOCOL)

The drive sends message ID = 0x83 (defined 0x80 + Id node)

- Data Value "Error Code" = 0x0
- Data Value "Reg" = 0x0
- "Data" = 0

It means "ERROR RESET or NO ERROR"

ID	Name	Node	Transfer data	Interpretation	Error	Data
83	EMCY_003		00 00 00 00 00 00 00 00	Error reset or no error	E	00 00 00 00 00 00 00 00

OBJECT WITH DIFFERENT DEFAULT

If the factory group changed than it is mandatory to change in [user unit] all values related to velocity and acceleration/deceleration of velocity profile mode and save in e^2prom.

After reset (or power-on) the drive initializes the object from e^2prom.

Caution

It is important to change before the maximum value.

If after power-on the value is not correct the drive send an emergency message.

Error Code	Object Error in Initialization
0x8B11	Init Object CANopen 0x6081
0x8B12	Init Object CANopen 0x6082
0x8B13	Init Object CANopen 0x6083
0x8B14	Init Object CANopen 0x6084
0x8B15	Init Object CANopen 0x60C5
0x8B16	Init Object CANopen 0x60C6
0x8B17	Init Object CANopen 0x607F
0x8B18	Init Object CANopen 0x6088
0x8B19	Init Object CANopen 0x6096
0x8B1A	Init Object CANopen 0x6097
0x8B1B	Init Object CANopen 0x606D
0x8B1C	Init Object CANopen 0x606E
0x8B1D	Init Object CANopen 0x606F
0x8B1E	Init Object CANopen 0x6070
0x8B1F	Init Object CANopen 0x6075
0X8B20	Init Object CANopen 0x6076

0X8B21	Init Object CANopen 0x6072
0X8B22	Init Object CANopen 0x6073
0X8B23	Init Object CANopen 0x60E0
0X8B24	Init Object CANopen 0x60E1
0X8B25	Init Object CANopen 0x6087
0X8B26	Init Object CANopen 0x6086
0X8B27	Init Object CANopen 0x607B
0X8B28	Init Object CANopen 0x607D
0X8B29	Init Object CANopen 0x6099
0X8B2A	Init Object CANopen 0x609A
0X8B2B	Init Object CANopen 0x607C
0X8B2C	Init Object CANopen 0x6065
0X8B2D	Init Object CANopen 0x6066
0X8B2E	Init Object CANopen 0x6067
0X8B2F	Init Object CANopen 0x6068
0X8B30	Init Object CANopen 0x60F2

It has to change the following objects (example new value):

Index	Sub Index	Name Object	Default Value [internal unit]	New Value [user unit]
0x606D	0	Velocity Window	13653 inc/sec	50 rpm
0x606F	0	Velocity Threshold	1365 inc/s	5 rpm
0x6083	0	Profile Acceleration	273066 inc/s^2	1000 rpm/s
0x6084	0	Profile Deceleration	273066 inc/s^2	1000 rpm/s
0x60C5	0	Max Acceleration	608393 inc/s^2	2228 rpm/s
0x60C6	0	Max Deceleration	608393 inc/s^2	2228 rpm/s

Caution

Pay attention at the order to write the new values.

It is important to define before the max value before the other objects.

Procedure Set New Values in User Unit (Write SDO)

Save Max Acceleration: new value 2228 [rpm/s]

The Master Controller sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x60C5
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 2228 = 0x08B4

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x60C5

- Data Value "Index" = 0x00
- Data Value "Sub-Index" = 0x00
- Data Value "Data" = 0

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003		b4 08 00 00	[60c5,00] Initiate Download Rq. expedited		23 C5 60 00 B4 08 00 00
583	SSDO_003			[60c5,00] Initiate Download Rsp		60 C5 60 00 00 00 00 00

The procedure continues ...

It has to write all objects with new values in [user unit]. It has to use the same procedure.

Procedure Save New Value in E²prom (Write SDO)

Caution

During STORE procedure, drive must not be in "Operation Enabled" state or in "Quick Stop Active" state.

The Master Controller sends SDO message ID = 0x603 (defined 0x600 + Id node)

- Data Value "command" = 0x23
- Data Value "Index" = 0x1010 (Store)
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = *0x65766173* (means "save" in ASCII code)

The drive answers SDO message ID = 0x583 (defined 0x580 + Id node)

- Data Value "command" = 0x60
- Data Value "Index" = 0x1010
- Data Value "Sub-Index" = 0x01
- Data Value "Data" = 0

The following picture shows the SDO message:

ID	Name	Node	Transfer data	Interpretation	Error	Data
603	CSDO_003		73 61 76 65	[1010,01] Initiate Download Rq. expedite...		23 10 10 01 73 61 76 65
583	SSDO_003			[1010,01] Initiate Download Rsp		60 10 10 01 00 00 00 00

The procedure continues ...

After Stored Parameters Proceed With reset drive

Reset All Nodes (NMT Protocol)

The Master Controllersends message ID = 0x00 (NMT protocol)

- Data Value "command" = 0x81
- Data Value "Index" = 0x00

The following picture shows the SDO message:

Dir	ID	Name	Node	Transfer data	Interpretation	Error	Data
Rx	0	NMTZeroMsg			Reset all nodes		81 00

After Reset (NMTProtocol)

The drive answers message BOOT-UP message ID = 0x703 (defined 0x700 + Id node)

- Data Value "Index" = 0x00

ID	Name	Node	Transfer data	Interpretation	Error	Data
703			Boot-up			00
703	_		Boot-up			00

The procedure continues ...

Then The Drive sends emergency messages (EMERGENCY PROTOCOL)

The drive sends message ID = 0x83 (defined 0x80 + Id node)

- Data Value "Error Code" = 0x0
- Data Value "Reg" = 0x0
- "Data" = 0

It means "ERROR RESET or NO ERROR"

ID	Name	Node	Transfer data	Interpretation	Error	Data
83	EMCY_003		00 00 00 00 00 00 00 00	Error reset or no error	E	00 00 00 00 00 00 00 00

11. | APPENDIX - EXAMPLE PROGRAMS

In this chapter the typical course of action is shown to launch a CANopen-drive.

In these examples the drive's Id Node is 1.

The value of message is composed

	Cmd	**index**		**subi**	**value**			
BYTES	0	1	2	3	4 (LSB)	5	6	7 (MSB)
Value	2F	60	60	00	03	00	00	00

For example the value **2F 60 60 00 03 00 00 00** is the SDO download (write):

- Byte 0: "$2F_h$" is the command "write request" of 1 byte
- Byte 1,2: "6060_h" is the index 0x6060 mode of operation
- Byte 3: "00_h" is the subindex 0x00
- Byte 4,5,6,7: 0x00000003 it is the number '3' profile velocity mode

SET PROFILE VELOCITY PROCEDURE with SDO

Set Mode of Operation

Caution

To configure the Profile Velocity Mode the drive isn't in "Operation enabled" state.

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	2f 60 60 00 03 00 00 00	Request: Set Mode of Operation num. Profile velocity
Tx	0x581	60 60 60 00 00 00 00 00	Answer: successfull

Go to the State "Switched-On"

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	2b 40 60 00 06 00 00 00	Request: Change state in "ReadyToSwitchOn"
Tx	0x581	60 40 60 00 00 00 00 00	Answer: request successfull
Rx	0x601	2b 40 60 00 07 00 00 00	Request: Change state in "SwitchedOn"
Tx	0x581	60 40 60 00 00 00 00 00	Answer: successfull
Rx	0x601	40 41 60 00 00 00 00 00	Request: Read SDO Status Word
Tx	0x581	4b 41 60 00 23 00 00 00	Answer: the status word is 0x0023. The "SwitchedOn" is xxxxxxxxx01x0011b

Set Acceleration e Deceleration

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	23 83 60 00 e8 03 00 00	Request: set Acceleration 1000 rpm/s
Tx	0x581	60 83 60 00 00 00 00 00	Answer: successfull
Rx	0x601	23 84 60 00 e8 03 00 00	Request: set Deceleration 1000 rpm/s
Tx	0x581	60 84 60 00 00 00 00 00	Answer: successfull

Go to the State "Operation Enabled".

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	2b 40 60 00 0F 00 00 00	Request: Change state in "OperationEnabled"
Tx	0x581	60 40 60 00 00 00 00 00	Answer: successfull
Rx	0x601	40 41 60 00 00 00 00 00	Request: Read SDO Status Word
Tx	0x581	4b 41 60 00 27 00 00 00	Answer: the status word is 0x0027. The "OperationEnabled" is xxxxxxxxx01x0011b

Set Target Velocity

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	23 ff 60 00 e8 03 00 00	Request: Set Point Velocity i.e. 1000 RPM
Tx	0x581	60 ff 60 00 00 00 00 00	Answer: successfull
Rx	0x601	40 6c 60 00 00 00 00 00	Request: Read Actual Velocity
Tx	0x581	43 6c 60 00 e8 03 00 00	Answer: the act velocity is 0x03E8 = 1000 rpm
Rx	0x601	23 ff 60 00 00 00 00 00	Request: Set the value of target velocity at 0 to stop
Tx	0x581	60 ff 60 00 00 00 00 00	Answer: successfull

information

In this case drive will stop with proper ramp and will stay in Powered (with torque applied)

Stop Velocity

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	2b 40 60 00 07 00 00 00	Request: Set "SwitchedOn".No torque applied
Tx	0x581	60 40 60 00 00 00 00 00	Answer: successfull
Rx	0x601	40 41 60 00 00 00 00 00	Request: Read SDO Status Word
Tx	0x581	4b 41 60 00 23 00 00 00	Answer: the status word is 0x0023. The "SwitchedOn" is xxxxxxxxx01x0011b
Rx	0x601	2b 40 60 00 00 00 00 00	Request: Set "SwitchedOnDisabled".No torque applied
Tx	0x581	60 40 60 00 00 00 00 00	Answer: successfull
Rx	0x601	40 41 60 00 00 00 00 00	Request: Read Status Word
Tx	0x581	4b 41 60 00 40 00 00 00	Answer: the status word is 0x0040. The "SwitchedOnDisabled" is xxxxxxxxx1xx0000b
Rx	0x601	2b 40 60 00 02 00 00 00	Request: Set State "QuickStopActive"
Tx	0x581	60 40 60 00 00 00 00 00	Answer: successfull
Rx	0x601	40 41 60 00 00 00 00 00	Request: Read SDO Status Word
Tx	0x581	4b 41 60 00 07 00 00 00	Answer: the status word is 0x0040. The "QuickStopActive" is xxxxxxxxx00x0111b

Trace Log Drive with SDO protocol (Target Velocity 1000 rpm)

ID	Name	Node	Transfer data	Interpretation	Error	Data	Counte
701	HBGuard_001	Node1	Boot-up			00	1
81	EMCY_001	Node1	00 00 00 00 00 00 00 00	Error reset or no error	E	00 00 00 00 00 00 00 00	1
81	EMCY_001	Node1	70 81 11 00 00 00 00 00	Communication - generic	E	70 81 11 00 00 00 00 00	2
601	CSDO_001	Node1	03	[6060,00] Initiate Download Rq. expedited		2F 60 60 00 03 00 00 00	1
581	SSDO_001	Node1		[6060,00] Initiate Download Rsp		60 60 60 00 00 00 00 00	1
601	CSDO_001	Node1	06 00	[6040,00] Initiate Download Rq. expedited		2B 40 60 00 06 00 00 00	2
581	SSDO_001	Node1		[6040,00] Initiate Download Rsp		60 40 60 00 00 00 00 00	2
601	CSDO_001	Node1		[6041,00] Initiate Upload Rq.		40 41 60 00 00 00 00 00	3
581	SSDO_001	Node1	31 10	[6041,00] Initiate Upload Rsp. expedited "1."		4B 41 60 00 31 10 00 00	3
601	CSDO_001	Node1	07 00	[6040,00] Initiate Download Rq. expedited		2B 40 60 00 07 00 00 00	4
581	SSDO_001	Node1		[6040,00] Initiate Download Rsp		60 40 60 00 00 00 00 00	4
601	CSDO_001	Node1		[6041,00] Initiate Upload Rq.		40 41 60 00 00 00 00 00	5
581	SSDO_001	Node1	33 10	[6041,00] Initiate Upload Rsp. expedited "3."		4B 41 60 00 33 10 00 00	5
601	CSDO_001	Node1	0f 00	[6040,00] Initiate Download Rq. expedited		2B 40 60 00 0F 00 00 00	6
581	SSDO_001	Node1		[6040,00] Initiate Download Rsp		60 40 60 00 00 00 00 00	6
601	CSDO_001	Node1	e8 03 00 00	[60ff,00] Initiate Download Rq. expedited		23 FF 60 00 E8 03 00 00	7
581	SSDO_001	Node1		[60ff,00] Initiate Download Rsp		60 FF 60 00 00 00 00 00	7
601	CSDO_001	Node1		[606c,00] Initiate Upload Rq. !UNUSED FIELDS USED	P	4B 6C 60 00 00 00 00 00	8
581	SSDO_001	Node1	e7 03 00 00	[606c,00] Initiate Upload Rsp. expedited		43 6C 60 00 E7 03 00 00	8
601	CSDO_001	Node1	07 00	[6040,00] Initiate Download Rq. expedited		2B 40 60 00 07 00 00 00	9
581	SSDO_001	Node1		[6040,00] Initiate Download Rsp		60 40 60 00 00 00 00 00	9
601	CSDO_001	Node1		[6041,00] Initiate Upload Rq.		40 41 60 00 00 00 00 00	10
581	SSDO_001	Node1	33 14	[6041,00] Initiate Upload Rsp. expedited "3."		4B 41 60 00 33 14 00 00	10

READ VERSION RELEASE

Command to read the version release $100A_h$ object:

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	40 0A 10 00 00 00 00 00	Request: Read Firmware released
Tx	0x581	43 0A 10 00 31 30 38 00	Answer: in ASCII code is 0x31, 0x30, 0x38, 0x00 = "108" version firmware released

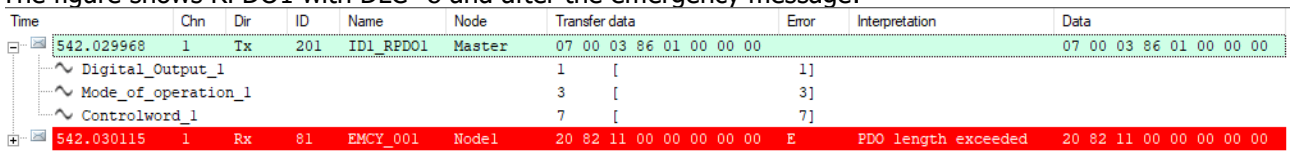
CONTROL VIA PDO

RPDOs:

Example set RPDOs default by master controller:

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x201	06 00 03 00 00 00 00	RPDO1 total size 7 bytes: controlword = 0x0006 = set "ReadyToSwitchOn" mode of operation = 0x03 digital output = 0x00000000
Rx	0x301	07 00 A0 86 01 00	RPDO2 total size 6 bytes: controlword = 0x0007 = set "SwitchedOn" target position = 0x186A0 = 100000 increments
Rx	0x401	0F 00 E8 03 00 00	RPDO3 total size 6 bytes: controlword = 0x000F = set "OperationEnabled" target velocity = 0x000003E8 = 1000 rpm
Rx	0x501	00 00 E8 03	RPDO3 total size 4 bytes: controlword = 0x0000 = set "DisableVoltage" target position = 0x003E8 =1000 motor rated current

If the length is wrong then the drive sends an emergency message.
The figure shows RPDO1 with DLC=8 and after the emergency message:



Time	Chn	Dir	ID	Name	Node	Transfer data	Error	Interpretation	Data
542.029968	1	Tx	201	ID1_RPDO1	Master	07 00 03 86 01 00 00 00			07 00 03 86 01 00 00 00
Digital_Output_1						1 [1]			
Mode_of_operation_1						3 [3]			
Controlword_1						7 [7]			
542.030115	1	Rx	81	EMCY_001	Node1	20 82 11 00 00 00 00 00	E	PDO length exceeded	20 82 11 00 00 00 00 00

RPDOs and TPDOs with SYNC:

If the RPDO1 and TPDO1 are defined SYNC type communication the master has to send the RPDOs and after the SYNC message. After SYNC message the drive transmits the TPDOs.

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x201	0F 00 03 00 00 00 00	RPDO1: controlword = 0x000F = set "OperationEnabled" mode of operation = 0x03, digital output = 0x00000000
Rx	0x401	0F 00 E8 03 00 00	RPDO3: controlword = 0x000F = set "OperationEnabled" target velocity = 0x000003E8 = 1000 rpm
Rx	0x80		**SYNC message**
Tx	0x181	C0 02 03 00 00 00 00	TPDO1: statusword = 0x02C0, mode of operation=0x03, digital input =0x00000000
Tx	0x381	C0 02 00 00	TPDO3: statusword = 0x02C0, actual velocity= 0x000003E8 = 1000 rpm

RPDOs RTR type:

If the RPDO set RTR the master has to

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x201	0F 00 03 00 00 00 00	RPDO1: controlword = 0x000F = set "OperationEnabled" mode of operation = 0x03, digital output = 0x00000000

Rx	0x401	0F 00 E8 03 00 00	RPDO3: controlword = 0x000F = set "OperationEnabled" target velocity = 0x000003E8 = 1000 rpm
Rx	0x181	RTR	
Tx	0x181	C0 02 03 00 00 00 00	TPDO1: statusword = 0x02C0, mode of operation=0x03, digital input =0x00000000
Rx	0x381	RTR	
Tx	0x381	C0 02 00 00	TPDO3: statusword = 0x02C0, actual velocity= 0x000003E8 = 1000 rpm

REMAPPING TPDO (RPDO) PROCEDURE

Remap the TPDO4 (1803_h) with the following maps:

- statusword (6041_h) in position 1
- actual velocity ($606C_h$) in position 2

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	23 03 18 01 0000 00 80	Request: Disable TPDO4 Communication
Tx	0x581	60 03 18 01 00 00 00 00	Answer: successfull
Rx	0x601	2F 03 1A 00 00 00 00 00	Request: Disable TPDO4 Mapping Parameters
Tx	0x581	60 03 1A 00 00 00 00 00	Answer: successfull
Rx	0x601	2F 03 18 02 01 00 00 00	Request: Set transmission type n.1 SYNC
Tx	0x581	60 03 18 02 00 00 00 00	Answer: successfull
Rx	0x601	23 03 1A 01 10 00 41 60	Request: Mapin pos.1 object 0x6041 statusword
Tx	0x581	60 03 1A 01 00 00 00 00	Answer: successfull
Rx	0x601	23 03 1A 02 20 00 6C 60	Request: Map in pos.2 object 0x606C actvelocity
Tx	0x581	60 03 1A 02 00 00 00 00	Answer: successfull
Rx	0x601	2F 03 1A 00 02 00 00 00	Request: Write number objects mapped (num 2)
Tx	0x581	60 03 1A 00 00 00 00 00	Answer: successfull
Rx	0x601	23 03 18 01 81 04 00 00	Request: Enable TPDO4 and Set Cob-Id
Tx	0x581	60 03 18 01 00 00 00 00	Answer: successfull

Read object Communication object (1803_h) and TPDO Mapping Parameters ($1A03_h$)

Tx/Rx	ID	VALUE	DESCRIPTION
Rx	0x601	40 03 18 00 00 00 00 00	Request: read 1803:0
Tx	0x581	4F 03 18 00 05 00 00 00	Answer: number subindex present, the value is 5
Rx	0x601	40 03 18 01 00 00 00 00	Request: read 1803:1
Tx	0x581	43 03 18 01 81 04 00 00	Answer:COB-ID is 0x00000481
Rx	0x601	40 03 18 02 00 00 00 00	Request: read 1803:2
Tx	0x581	4F 03 18 02 01 00 00 00	Answer:Tx Type: Sync 1
Rx	0x601	40 03 18 03 00 00 00 00	Request: read 1803:3
Tx	0x581	4B 03 18 03 05 00 00 00	Answer:Inhibit Time value is 5
Rx	0x601	40 03 18 04 00 00 00 00	Request: read 1803:4
Tx	0x581	80 03 18 04 11 00 09 06	Answer: Abort Code this subindex not exixt
Rx	0x601	40 03 18 05 00 00 00 00	Request: read 1803:6
Tx	0x581	4B 03 18 05 00 00 00 00	Answer:Event Time value is 0
Rx	0x601	40 03 1A 00 00 00 00 00	Request: read 1A03:0
Tx	0x581	4F 03 1A 00 02 00 00 00	Answer: number objects mapped, the value is 2
Rx	0x601	40 03 1A 01 00 00 00 00	Request: read 1A03:1
Tx	0x581	43 03 1A 01 10 00 41 60	Answer: 1° obj mapped:statusword 0x6041
Rx	0x601	40 03 1A 02 00 00 00 00	Request: read 1A03:2
Tx	0x581	43 03 1A 02 20 00 6C 60	Answer: 2° obj mapped:Act Velocity 0x6041
Rx	0x601	40 03 1A 03 00 00 00 00	Request: read 1A03:3
Tx	0x581	43 03 1A 03 00 00 00 00	Answer: 3° obj mapped:none
Rx	0x601	40 03 1A 04 00 00 00 00	Request: read 1A03:4

```
Tx      0x581     43 03 1A 04 00 00 00 00       Answer: 4° obj mapped:none

Rx      0x601     40 03 1A 05 00 00 00 00       Request: read 1A03:5
Tx      0x581     43 03 1A 05 00 00 00 00       Answer: 5° obj mapped:none

Rx      0x601     40 03 1A 06 00 00 00 00       Request: read 1A03:6
Tx      0x581     43 03 1A 06 00 00 00 00       Answer: 6° obj mapped:none

Rx      0x601     40 03 1A 07 00 00 00 00       Request: read 1A03:7
Tx      0x581     43 03 1A 07 00 00 00 00       Answer: 7° obj mapped:none

Rx      0x601     40 03 1A 08 00 00 00 00       Request: read 1A03:8
Tx      0x581     43 03 1A 08 00 00 00 00       Answer: 8° obj mapped:none
```

PROFILE VELOCITY WITH PDO MAPPING DYNAMIC

First initialization with TPDO1 and RPDO1, transmission type SYNC and following maps:

- RPDO1 : controlword and Target velocity
- TPDO : statusword and Actual Velocity

1. Power cycle (boot-up and emergency "NO ERROR")

Rx	0	NMTZeroMsg				81 00
Rx	701	HB_NG_001	Node1	Boot-up		00
Rx	81	EMCY_001	Node1	00 00 00 00 00 00 00 00	E	00 00 00 00 00 00 00 00

2. Configuration RPDO1 (Tx-Type SYNC and maps: controlword and target velocity)

Rx	601	CSDO_001	Node1	00 00 00 80	23 00 14 01 00 00 00 80
Rx	581	SSDO_001	Node1		60 00 14 01 00 00 00 00
Rx	601	CSDO_001	Node1	00	2F 00 16 00 00 00 00 00
Rx	581	SSDO_001	Node1		60 00 16 00 00 00 00 00
Rx	601	CSDO_001	Node1	01	2F 00 14 02 01 00 00 00
Rx	581	SSDO_001	Node1		60 00 14 02 00 00 00 00
Rx	601	CSDO_001	Node1	10 00 40 60 [6040,00]	23 00 16 01 10 00 40 60
Rx	581	SSDO_001	Node1		60 00 16 01 00 00 00 00
Rx	601	CSDO_001	Node1	20 00 ff 60 [60ff,00]	23 00 16 02 20 00 FF 60
Rx	581	SSDO_001	Node1		60 00 16 02 00 00 00 00
Rx	601	CSDO_001	Node1	02	2F 00 16 00 02 00 00 00
Rx	581	SSDO_001	Node1		60 00 16 00 00 00 00 00
Rx	601	CSDO_001	Node1	01 02 00 00	23 00 14 01 01 02 00 00
Rx	581	SSDO_001	Node1		60 00 14 01 00 00 00 00

3. Configuration TPDO1 (Tx-Type SYNC and maps: statusword and actual velocity)

Rx	601	CSDO_001	Node1	00 00 00 80	23 00 18 01 00 00 00 80
Rx	581	SSDO_001	Node1		60 00 18 01 00 00 00 00
Rx	601	CSDO_001	Node1	00	2F 00 1A 00 00 00 00 00
Rx	581	SSDO_001	Node1		60 00 1A 00 00 00 00 00
Rx	601	CSDO_001	Node1	01	2F 00 18 02 01 00 00 00
Rx	581	SSDO_001	Node1		60 00 18 02 00 00 00 00
Rx	601	CSDO_001	Node1	10 00 41 60 [6041,00]	23 00 1A 01 10 00 41 60
Rx	581	SSDO_001	Node1		60 00 1A 01 00 00 00 00
Rx	601	CSDO_001	Node1	20 00 6c 60 [606c,00]	23 00 1A 02 20 00 6C 60
Rx	581	SSDO_001	Node1		60 00 1A 02 00 00 00 00
Rx	601	CSDO_001	Node1	02	2F 00 1A 00 02 00 00 00
Rx	581	SSDO_001	Node1		60 00 1A 00 00 00 00 00
Rx	601	CSDO_001	Node1	81 01 00 00	23 00 18 01 81 01 00 00
Rx	581	SSDO_001	Node1		60 00 18 01 00 00 00 00

4. Send a SDO mode of operation 0x6060 to set the "Profile Velocity"

Rx	601	CSDO_001	Node1	03	2F 60 60 00 03 00 00 00
Rx	581	SSDO_001	Node1		60 60 60 00 00 00 00 00

5. Move the drive in "operational" state

```
    Rx      0      NMTZeroMsg                                          01 00
```

or

```
    Rx      0      NMTZeroMsg  Node1                                   01 01
```

6. The TPDO1 and RPDO1 are not working because they wait a SYNC message.

   Master must send a SYNC message periodically timer and RPDO1. The drive accepts the RPDO1 after SYNC message, and it sends a TPDO1.

```
1884.673491   1   Rx   80    SYNC
1884.673605   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1885.464957   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1885.673863   1   Rx   80    SYNC
1885.673979   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1886.464980   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1886.674130   1   Rx   80    SYNC
1886.674245   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1887.465025   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1887.674461   1   Rx   80    SYNC
1887.674597   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1888.463951   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1888.674822   1   Rx   80    SYNC
1888.674960   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1889.464949   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1889.675120   1   Rx   80    SYNC
1889.675234   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1890.464948   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1890.675429   1   Rx   80    SYNC
1890.675543   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1891.464958   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1891.675753   1   Rx   80    SYNC
1891.675868   1   Rx   181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
1892.464956   1   Tx   201   ID1_RPDO1   Master   00 00 00 00 00 00    00 00 00 00 00 00
1892.676166   1   Rx   80    SYNC
```

7. the master moves the drive using control word (command 0x06)

   The statusword changes the state: 0x9250 = "Switch On disabled" -> 0x9231 = "Ready To Switch On"

```
   Rx    181   ID1_TPDO1   Node1    50 92 00 00 00 00    50 92 00 00 00 00
   Tx    201   ID1_RPDO1   Master   06 00 00 00 00 00    06 00 00 00 00 00
   Rx    80    SYNC
   Rx    181   ID1 TPDO1   Node1    31 92 00 00 00 00    31 92 00 00 00 00
```

8. the master moves the drive using control word (command 0x07)

   The statusword changes the state: 0x9231 = "Ready To Switch On" -> 0x9033 "Switched On"

```
  Rx    80    SYNC
  Rx    181   ID1_TPDO1   Node1    31 92 00 00 00 00    31 92 00 00 00 00
  Tx    201   ID1_RPDO1   Master   07 00 00 00 00 00    07 00 00 00 00 00
  Rx    80    SYNC
  Rx    181   ID1_TPDO1   Node1    33 90 00 00 00 00    33 90 00 00 00 00
```

9. Now, the master moves the drive using control word (command 0x0F)

   The statusword changes the state: 0x9033 "Switched On" -> 0x9237 "Operation Enabled"

```
1    Rx    80    SYNC
1    Rx    181   ID1_TPDO1   Node1    33 90 00 00 00 00    33 90 00 00 00 00
1    Tx    201   ID1_RPDO1   Master   0f 00 00 00 00 00    0F 00 00 00 00 00
1    Rx    80    SYNC
1    Rx    181   ID1 TPDO1   Node1    37 92 00 00 00 00    37 92 00 00 00 00
```

10. Now, the drive is in RUN, it waits the target Velocity:

    This case the master sends a target velocity in RPDO1 = 1000rpm = 0x03E8

1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 dc 03 00 00	37 86 DC 03 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00
1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 dc 03 00 00	37 86 DC 03 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00
1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 dc 03 00 00	37 86 DC 03 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00
1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 01 04 00 00	37 86 01 04 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00
1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 dc 03 00 00	37 86 DC 03 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00
1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 01 04 00 00	37 86 01 04 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00
1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 dc 03 00 00	37 86 DC 03 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00

11. You can see the Actual Velocity in TPDO1, here it is 0x3dc = 988 RPM

1	Rx	80	SYNC			
1	Rx	181	ID1_TPDO1	Node1	37 86 dc 03 00 00	37 86 DC 03 00 00
1	Tx	201	ID1_RPDO1	Master	0f 00 e8 03 00 00	0F 00 E8 03 00 00

12. | APPENDIX – HEARTBEAT MECHANISM

One of the protective mechanisms available in CANopen is the heartbeat mechanism.

This mechanism allows the network master to detect a loss of communication from the network slaves, and it also allows the network slaves to react to a loss of communication from the master.

Lafert servo drives are compliant with the DS-301 and DS-402 versions of the CANopen protocol which define functions related to the heartbeat mechanism.

If the heartbeat is activated, when the drive detects a communication loss, it goes in "Fault" state automatically and the alarm is sent.

Heartbeat Sources and Message Structures

The standard DS301 describes that the CANopen nodes can be configured to transmit heartbeat messages and they can also be configured to monitor heartbeats from the host.
Nodes that generate heartbeats are called "producers", and nodes that monitor heartbeats are called "consumers".

The following picture is of DS301 standard document CiA301:

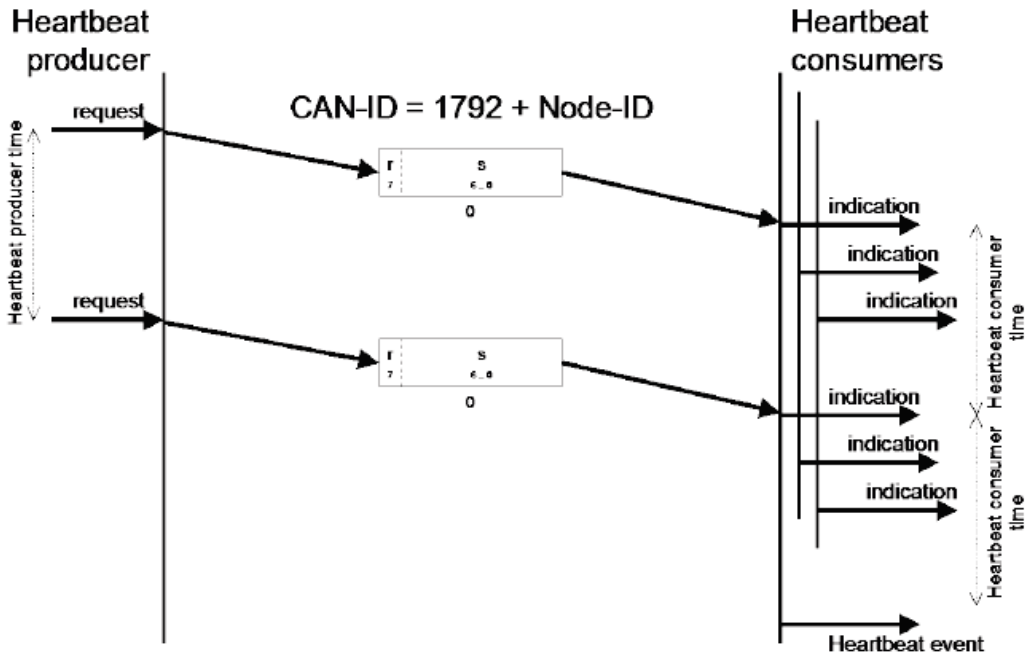


Figure 59 - Heartbeat Mechanism by DS301

Master Heartbeat:

The master heartbeat has the following characteristics:

- Produced by the CANopen master

- Consumed by CANopen slave nodes
- COB ID is 0x700
- The data frame can be empty

Master heartbeat message:

COB-ID	Rx/Tx	DLC	Byte							
			0	1	2	3	4	5	6	7
0x700	Rx	0								

Slave Heartbeat:

The slave heartbeat has the following characteristics:

- Produced by slave nodes on the network
- Consumed by the CANopen master
- The COB ID range is in the range 0x701 – 0x77F
- The data frame is 1 byte in length and contains a description of the slave node's communication state according to the table below:

Heartbeat Value	Description
0x0	Boot-up
0x1	Off bus
0x4	Stopped
0x5	Operational
0x7F	Pre-operational

Slave heartbeat message:

COB-ID	Rx/Tx	DLC	Byte							
			0	1	2	3	4	5	6	7
0x700 + IdNode	Tx	1	NMT State						-	

If the network is composed by 1 PLC and two drive than the drives must be configured as producer and the PLC as consumer.

Drive Configuration:

Lafert Servo Drive can be ONLY as a Heartbeat Producer by settings object 0x1017.

If the heartbeat producer time defined by 1017_hobject is configured on the Lafert Servo Drive, the producer heartbeat protocol begins immediately and transmits producer heartbeat messages periodically.

Heartbeat monitoring starts as soon as the time interval of the producer is greater than zero. If the Heartbeat protocol is already active during the NMT state transition to "Pre-Operational", Heartbeat protocol starts with sending of the boot-up message. The boot-up message is Heartbeat message with one data byte 00_h.

To configure the 1017_h object (Producer Heartbeat Time) see the relative paragraph.

The time intervals are set in increments of 1ms steps; the values for the consumer must not be less than the values for the producer. Whenever the "Heartbeat" message is received, the time interval of the producer is restarted.

Example:

Tx/Rx	ID	VALUE Define: Cmd + index + subi + value Bytes: (1) + (2) + (1) + (4)	DESCRIPTION
Rx	0x601	2b 17 10 00 64 00 00 00	Request: Drive Set Time HeratBeat 100ms (0x64)
Tx	0x581	60 17 10 00 00 00 00 00	Answer:The drive send periodically the heartbeat message with 0x5 value (State Drive is Operational)
Tx	0x701	05	
Tx	0x701	05	
Tx	0x701	05	
...	...	...	

Master Configuration:

The master controller must be configured as consumer. It has to define the 1016h object.

This object defines the period of time (ms) where a heartbeat from the master node is expected at the beginning of the period.

Monitoring begins on reception of the first heartbeat. A value of 0 disables heartbeat monitoring.

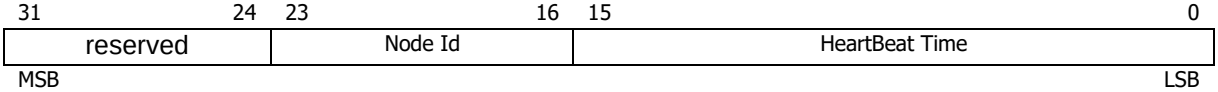
Object Description:

Index	Name	Object Code	Data Type	Category
1016_h	Consumer Heartbeat Time	ARRAY	U16	Optional

Entry Description:

Sub-Index	Description	Access	PDO mapping	Default Value
00_h	Highest sub-index supported	ro	no	0
01_h	Consumer Heartbeat Time 1st Node	rw	no	0
02_h	Consumer Heartbeat Time 2nd Node	rw	no	0

Value Definition:



Example:

Tx/Rx	ID	VALUE Define: Cmd + index + subi + value Bytes: (1) + (2) + (1) + (4)	DESCRIPTION
Rx	0x601	23 16 10 01 64 00 00 00	Request: Set the consumer heartbeat time on node 1 to 100 ms (0x64)
Tx	0x581	60 16 10 01 00 00 00 00	Answer: successfull

13. | APPENDIX – POSITION MONITORING

Between the motor shaft and the AGV wheel there is a gear box with a gear ratio. This value means that one revolution of AGV wheel corresponds to 21 motor shaft revolutions.

For example: diameter 21 mm.

When motor shaft makes one revolution the value of "Position Actual Value" 6064_h object increment or decrement (depending by direction of rotation) by resolution units.

Caution

The object 6064_h is initialized a 0 every time you reset or power up the drive, it is independent of position wheel.

Wheel Rotation:

Supposing the drive moves the motor with a clockwise rotation of motor shaft.

The "Position Actual Value" (6064_h) object was increased from 0.

If the value is 386007 increments then the motor shaft is rotated of

$$Motor\ Rotation = \frac{Actual\ Position\ (6064h: 0)}{Feedback\ Resolution\ (3004h: 2)} = \frac{386007}{16384} = 23{,}56\ rounds$$

The wheel will rotate of

$$\frac{Motor\ Rotation}{Gear\ Ratio\ \emptyset} = \frac{23{,}56}{21} = 1{,}122\ turns$$

The result of operation is equals 1 with a remainder of 0,122.

Angle Calculation:

The turn of the wheel is mapped in 360 degrees. When the "Position Actual Value" (6064_h) object is 386007 counts then the motor shaft will rotate of 23,56 rounds. The wheel will rotate 1,122 turns.

The angle of the motor wheel will be

$$Angle = \left(\left[\frac{Motor\ Rotation}{Feedback\ Resolution\ (3004h: 2)}\right] div\ (Gear\ Ratio\ \emptyset)\right) x\ 360 = 0{,}122\ x\ 360 = 43{,}89\ degrees$$

Where 0,122 is the remainder of division $\left(\left(\frac{386007}{16384}\right)/21\right)$

14. | APPENDIX – SRAMP PROFILE

Introduction:

Theory

In a point-to-point move, the rate of change in acceleration is known as jerk.
In some applications, high rates of jerk can cause excessive mechanical wear or material damage.
To support varying levels of jerk tolerance, it needs to introduce new motion profiles defined S-Curve.
An S-Curve is a trajectory which is constrained to jerk, acceleration and velocity. By smoothing out the edges, S-curve motion profiles reduce abrupt acceleration changes and thereby smooth out the motion.

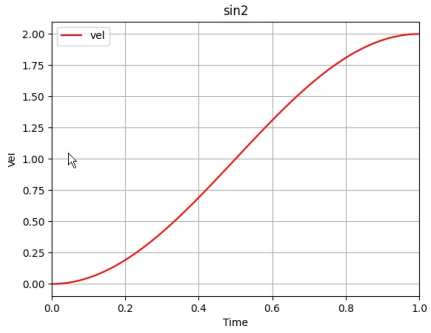
An S-Curve sin^2 based consists of 5 phases:

- Smooth acceleration from 0 to maximum acceleration value (at T/2 of acceleration phase)
- Smooth decrease of acceleration from max to 0 (target velocity reached)
- constant velocity phase (if required)
- Smooth negative acceleration from 0 to maximum deceleration value (at T/2 of deceleration phase)
- Smooth decrease of deceleration from max to 0 (0 velocity reached)

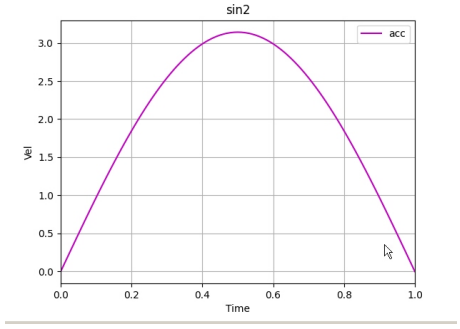
Description

S-curve profile the built-in reference generator computes a profile with an S-curve shape of the speed using sin^2 calc. S-Ramp defines a trajectory for the velocity set point to move from a V0 to Vr (Velocity Target) in each time.

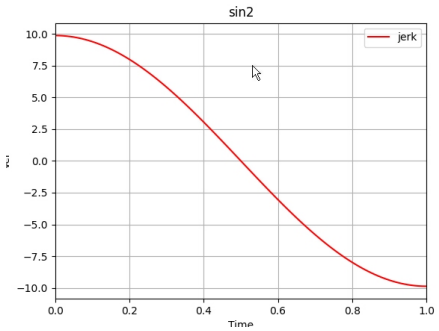
Example of a velocity trajectory from V0 = 0 to Vr = 2 in 1 sec



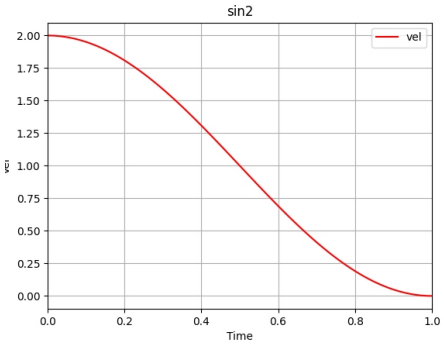
Example of an acceleration trajectory from V0 = 0 to Vr = 2 in 1 sec



Example of a jerk trajectory from V0 = 10 to Vr = 2 in 1 sec



Example of a velocity trajectory from V0 = 2 to Vr = 0 in 1 sec



Programmability

The motion S-Curve is set by a parameter using programming in serial interface (file parameters *file.ibd*) or it is possible write parameter with object via CANopen (DS402) or with register using Modbus.

Object used in CANopen:

S-ramp or trapezoidal ramp will be programmed specifying value

- Object 6086h = Motion profile type

Parameters to define trajectory:

- Object 60FFh = Target Velocity
- Object 6083h = Profile Acceleration
- Object 6084h = Profile Deceleration

Other relevant Parameters:

- object 607Fh = Max Velocity
- object 60C5h = Max Acceleration
- object 60C6h = Max Deceleration

Max Jerk for acceleration and max jerk for deceleration:

- object 60A3h = 2
- object 60A4h sub 00 = 2
- object 60A4h sub 01 – Begin and End jerk during acceleration
- object 60A4h sub 02 - Begin and End jerk during deceleration

Register used in Modbus:

The same values will be written in registers for Modbus configuration (see Modbus Manual).

Functionality

The S ramps can be run with Jerk control or without control. This behaviour will be programmable with parameter "Motion Profile Type".

Upon receipt of a new speed set point or a new acceleration value, the parameters for the ramp generation will be calculated and verified its correspondence with the set limits of maximum acceleration and jerk (in case of Jerk controlled).

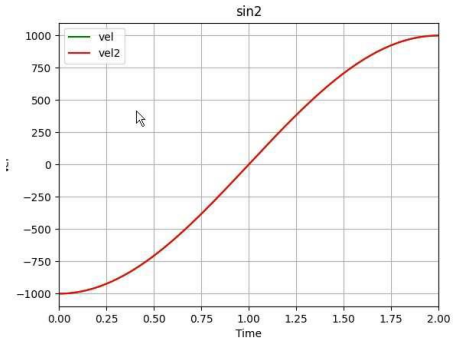
In case the above limits are not respected the ramp will not be executed and an error message will be provided with different actions.

The different actions can be:

- discard the received value of speed set point that doesn't meet Jerk limit (only in case of jerk controlled)
- Recalculation in real time of new lower acceleration for meet jerk limit and execute trajectory to new speed setpoint (at the moment this behaviour is not implemented and will be verified implementation feasibility)

If you work with different values for Profile Acceleration and Profile Deceleration you need to know when acceleration and deceleration are applied. Acceleration is applied when you perform a ramp where absolute value of final velocity is greater than absolute value of initial velocity. Deceleration instead is applied when absolute value of final velocity is lower than absolute value of initial velocity, regardless the sign of velocity. A special case is when you perform a "cross S ramp".
A cross S ramp is a motion where the sign of initial velocity is opposite to the sign of final velocity.

Example of an S ramp from -1000 Rpm to 1000 Rpm



In the case above is applied the Profile acceleration parameter. If initial velocity is positive and final velocity is negative instead profile deceleration is applied.

It will be possible to perform emergency braking with ramp S. This feature will be programmable and the parameters for ramp S braking will be separated from the others.

Parameters for quick stop and Emergency are to be determined.

- Object 605Ch = Disable operation option code
- Object 605Bh = Shut down option code
- Object 605Eh = Fault reaction option code
- Object 3008h Sub 4 = Emergency option code

For perform emergency breaking with S ramp you need to program value = -5 in relevant parameter.

Each type of emergency breaking, S ramp or linear ramp, will cause a discontinuity in speed and acceleration, with jerk not limited.

Limitations

Does not support modification of current move, once the ramp has started it cannot be interrupted. This means that any new speed set point will only be considered after the ramp has been completed. A series of moves requires a series of discrete profiles

Measure Unit

- Target Velocity = RPM or Inc/s
- Acceleration or Deceleration = RPM/s or Inc/s^2
- Jerk = Velocity/s^2

Functional Requirements

The mode of operations is "profile Velocity" with digital set point (not analog).
The fieldbus are CANopen or Modbus.

Configuration:

Before to move the motor with S-Ramp the drive must be configured.
It needs to have the firmware version ≥ 240, if the drive has old release, it must be upload.

Procedure to configure parameters:

Write object Motion profile type" (0x6086):

- value '-2'→ "*Man*. Spec S-Ramp with Jerk Limited": It means that the S-Ramp is limited by Jerk value calculates in the firmware, if the target velocity (0x60FF) is out of range by jerk calculation the drive answers with "abort code".
- value '-3'→ "*Man*. Spec S-Ramp without Jerk": It means that the S-Ramp is not limited by Jerk.

Write Factor Group of Jerk like Velocity and acceleration Factor Group. If the user units of drive is

- INC: write "Jerk Factor" num and div (0x60A2:1 and 0x60A2:2) with value '1'
- RPM: write "Jerk Factor" num (0x60A2:1) with value '16384' (equal resolution) and write Jerk Factor div (0x60A2:2) with value '60'

Time	Chn	Dir	ID	Name	Node	Error	Data	Counter
7.462324	1	Rx	601	M_SDO_001	Master		23 A2 60 01 00 40 00 00	1
0.000189	1	Rx	581	S_SDO_001	Smartris_DX		60 A2 60 01 00 00 00 00	1
4.426085	1	Rx	601	M_SDO_001	Master		23 A2 60 02 3C 00 00 00	2
0.000174	1	Rx	581	S_SDO_001	Smartris_DX		60 A2 60 02 00 00 00 00	2
2.976684	1	Rx	601	M_SDO_001	Master		23 10 10 01 73 61 76 65	3
0.000194	1	Rx	581	S_SDO_001	Smartris_DX		60 10 10 01 00 00 00 00	3
1.417012	1	Rx	0	NMT_Zero			81 00	1
2.390108	1	Rx	701	HB_NG_001	Smartris_DX		00	1
0.000135	1	Rx	81	EMCY_001	Smartris_DX	E	00 00 00 00 00 00 00 00	1
4.145211	1	Rx	601	M_SDO_001	Master		40 A2 60 01 00 00 00 00	4
0.000162	1	Rx	581	S_SDO_001	Smartris_DX		43 A2 60 01 00 40 00 00	4
1.183293	1	Rx	601	M_SDO_001	Master		40 A2 60 02 00 00 00 00	5
0.000188	1	Rx	581	S_SDO_001	Smartris_DX		43 A2 60 02 3C 00 00 00	5

The "Jerc Factor" object must be saved in eeprom before to use.

Write object "Jerk Profile 1" (0x60A4:1) and "Jerk Profile 2" (0x60A4:2) according by application and user units.

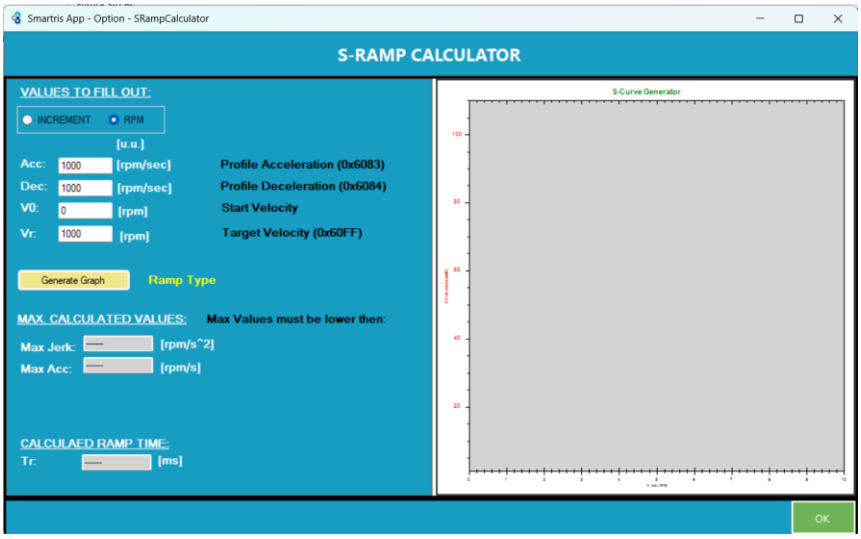
Programmability:

To choose the parameters of the S CURVES, there is an application integrated with the SmartrisApp that generates the curve.

Open SmartrisApp and click button "Utilities" 

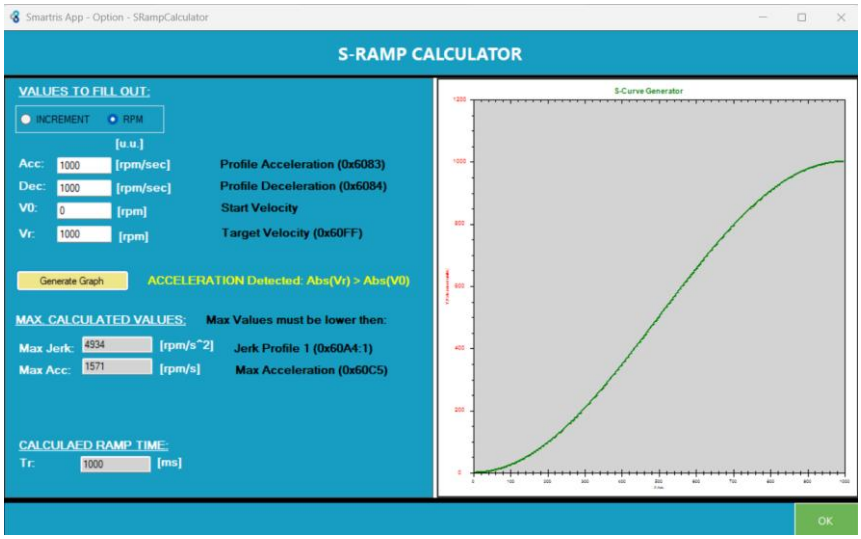
and "SRamp Calculator".

This window is open:



According by application write camps Acc, Dec, V0 and Vr.
Fill in Acceleration and Deceleration Parameters likes object 0x6083 and 0x6084.
Write "Start velocity" (V0) and "Final Velocity" (Vr) and generate the graph.

Graph generated:



The "Max Jerk", "Max Acc" and "Tr" are value calculated after graph generation.

The user must verify that the values are correct and compatible with the application.

- The "Max Jerk" calculated must lower than "Jerk Profile 1" object (0x60A4:1) or "Jerk Profile 2" object (0x60A4:2), depending by slope of curve.
- The "Max Acc" calculated must be lower than "Max acceleration" (0x60C5) or Max Deceleration (0x60C6), depending by slope of curve.

The "Tr" value is the time it takes to make the whole curve.

Note:

- It is not possible to change on the fly the set point of S-ramp. If the new target is sent during S-ramp, the new value will be used at the end of the curve with new calculation

- It is possible to move the drive from "operation enabled" state (RUN) to "Switched on disabled" by controlword or emergency using S-Ramp programmable. It needs to set object 0x3007 "Electronic Dynamic Stop Parameters" subindex 8 the value "Decrement step ramp – SRAMP" (unit in rpm/s) and the value '-5' relative at movement (object 0x605C Disable Operation Option Code)

- The behaviour of "Quick Stop" and "halt" bit is the same, are not affected by S ramp

Lafert S.p.A.
J.F. Kennedy, 43 – I-30027 San Donà di Piave (Venice), Italy | Phone +39 / 0421 229 611 | Fax +39 / 0421 222 908 | lafert.info@shi-g.com | **www.lafert.com**

REVISION HISTORY

Rel.	Date	Description
0.0	**23/10/2019**	• Draft of CANOpen User Guide for AGV
0.1	**06/11/2019**	• 2nd Draft of CANOpen User Guide for AGV
0.2	**03/01/2020**	• 3th Draft of CANOpen User Guide for AGV
0.3	**07/01/2020**	• Changes in table - I/O SIGNAL AGV
0.4	**16/01/2020**	• Added Object - 0x1008: Manufacturer Device Name - 0x1009: Manufacturer Hardware Version - 0x100A: Manufacturer Software Version - 0x3002: Brake Parameters - 0x3007: Dynamic Stop Parameters • Modified Object - 0x3020: Drive Digital Input - 0x60FD: Digital Inputs • Added Table Identifier • Added Example Programs in Appendix • Added Map Object Dictionary Memory • Added Mapping Default PDO • Added Store and Restore
0.5	**17/01/2020**	• Added Object - 0x3008: Emergency Enable Input Parameters
0.6	**06/02/2020**	• Added Object - 0x100C: Guard Time - 0x100D: Life Time Factor - 0x1017: Producer Heartbeat Time - 0x2003: Warning • Modified Object: - 0x6040: Control word – Bit Warning - 0x6041: Status Word – Bit Warning • Modified Node Guarding Protocol • Modified Heart-Beat Protocol • Update Error List CANopen - Add Communication Canopen: Error Code 0x8100
0.7	**04/03/2020**	• Added Object - 0x2041: Voltage Bus - 0x2050: Torque Current - 0x2053: Velocity Filtered - 0x1001: Error Register - 0x1003: Pre-defined Error Field • Object 0x3024 erased because It is in 0x3008 (Emergency Enable Parameters) • Modified Error List CANopen (Chapter Emergency messages) • Added Example Programs: • Emergency History
0.8	**06/04/2020**	• Added Object: - 0x1002:Manufacturer Status Register - 0x3050: Analog Output 1 - 0x3200: Current Pid

		- 0x3201: Speed Pid - 0x3202: Position Pid • Update Object: - 0x1010: Store parameters - 0x1011: Restore default parameters (page - 0x2003: Warning • Added SDO Abort Protocol • Update Error List CANopen - Data Set Param: add error code 0x6309, 0x630A, 0x630B - Resolver Fault Error: add error code 0x7370, 0x7373, 0x7374, 0x7375, 0x7376, 0x7377 - Warning: add error code 0x6001, 0x8B01, 0x8B06 • Update Store and Restore Chapter because changed e2prom
0.9	**19/06/2020**	• Update Object - 0x2003: Warning • Update Error List CANopen: - Golden Data Image:add error code 0x5A01, 0x5A02 - Can Protocol Communication: add error code 0x7530, 0x7531,0x7532 - Incremental Encoder Error: add error code 0x7390, 0x7391, 0x7392, 0x7393, 0x7394 - None Error Profile: add error code 0x8C04 - Hardware Error : 0x5501 • Added Graphic Velocity Profile Mode • Add Object 0x3001: Limits Parameters
1.0	**13/07/2020**	• Update Error List CANopen: - Hardware Error : 0x5501 - Error Parameters:0x6321 • Update Object - 0x2002: Drive Mode change in Drive Control State - 0x3020: Drive Digital Input • Add Object - 0x3006: Max Motor Speed - 0x6402: Motor Type • Update Diagnostic Led • ERROR STATE MACHINE transition T12 • Add SAFETY Chapter
1.1	**03/09/2020**	• Update Error List CANopen: - Init Object CANopen from E^2prom: 0x8B06 - Data record no. 14: 0x630E - Data record no. 15: 0x630F • Update Object - 3002_h: Motor Brake Parameters - 3007_h: Dynamic Stop Parameters • Add Object - 3050_h : Analog Output 1 - $603F_h$:Error Code - 1018_h: Identity object • Add Abort Code for the following object: - 0X2000, 0X2001, 0X3001, 0x3002, 0x3008, 0x3050, 0x3200, 0x3201, 0x3202, 0x3202, 0x3300, 0x3004, 0x6060, 0x607E, 0x60FF, 0x607F, 0x6080, 0x6083, 0x6084, 0x60C5, 0x60C6, 0x606D, 0x606E, 0x6070 • Modify unit measure of objects: 0x2030, 0x2031, 0x2032, 0x2041 • Added Appendix :"FIRST CONFIGURATION" • Modified the appendix "Example Program"

		• Modified Chapter :"FUNCTIONS"
1.2	**21/10/2020**	• Modify unit measure of objects: 0x2041, 0x2050 • Modify Value Range Factory Group • Modify access of Object 6060h: rw (not ro) • Add digital Input 4 management • Add Appendix Heartbeat mechanism • Modify Network Management (NMT) Chapter • Add Graph Time frame Brake when the drive moves from "operation enabled" to "switched-on"
1.3	**27/11/2020**	**Manual for firmware 200:** • Add Chapter Torque Profile • SYNC Protocol • Update Error List CANopen: - Torque Profile: 0x9341, 0x8351 - Error EEprom Programming :0x5563 … 0x556D • Modify PDO Protocol Chapter (add dynamic mapping features)
1.4	**05/02/2021**	**Manual for firmware 201:** • Update Error List CANopen: - Data Set Programming: 0x6401, 0x6402,0x6403 - Load Level (i2t): 0x2352 - Warning Temperature : 0x4301, 0x4501, 0x4A01 - Init Object From EEprom : 0x8B10, 0x8B11, 0x8B12, 0x8B13, 0x8B14, 0x8B15, 0x8B16, 0x8B17, 0x8B18, 0x8B19, 0x8B1A, 0x8B1B, 0x8B1C, 0x8B1D, 0x8B1E, 0x8B1F, 0x8B20, 0x8B21, 0x8B22, 0x8B23, 0x8B24, 0x8B25, 0x8B26 • Modify object 0x3050 "Analog Output 1" : add function "Current Monitoring" • Add subindex 4 e 5 in "Motor Specific Settings" object 0x3006 • Modify object "Warning" 0x2003 - Add bit 8,9,10 warning temperature - Add bit 11 warning communication busoff/passive - Add bit 12, 13 Limitation Torque Limit • Modify Object 0x3020 "Digital Input Function" • Add Object for read the digital Input configuration: - Digital Input 1: 0x3021 - Digital Input 2: 0x3022 - Digital Input 3: 0x3023 - Digital Input 4: 0x3024 - Positive torque limit value: 0x60E0 - Negative torque limit value: 0x60E0 • Modify chapter digital Input in Function • Add Chapter "APPENDIX – POSITION MONITORING" • Add Led Code Status in "Diagnostic" : "Communication Error" • Modify Chapter "CAN Error Communication"
1.5	**15/06/2021**	**Manual for firmware 201:** • Errata corrige: - "Positive torque limit value" (0x60E0) description - "the value shall be given percent of rated current maximum peak current" (not motor rated current) - "Negative torque limit value" (0x60E1) description - "the value shall be given percent of rated current maximum peak current" (not motor rated current) - in "CANOPEN OBJECT LIST" chapter n.7 the object 0x60E0 and 0x60E1 are available. - "Motor RatedCurrent" (0x6075) description - the units of "Nominal Rated Current" (3003:3) is not mArms but in Arms/100. The example was incorrect. - "Max Current" (0x6073) description - the units of "Nominal Rated Current" (3003:3) is not mArms but in Arms/100. The example was incorrect. - In Re-Mapping Procedure point (1) – "The 31 bit must be set to 1" (not 0).

		- In SYNC chapter the DLC of message SYNC is 0 and not 8. • Add Chapter "APPENDIX EXAMPLE PROGRAMS" the example for "Control via PDO" and "REMAPPING TPDO/RPDO PROCEDURE" • Add Drive Size Parameters object (object 0x3003) only read • Update Error List CANopen in EMERGENCY chapter: - Protocol Error: 0x8200, 0x8210, 0x8220, 0x8230, 0x8240, 0x8250 • Modify Velocity filtered 0x2053: - it can be mapped - the filter pole cab be be modified using object 0x3005:5 • Add object Filter Parameters 0x3005 • Add object Energy i^2t (0x2054)
1.6	**29/06/2021**	**Manual for Firmware 225:** • Errata corrige: - Dynamic Stop option (0x3007:1) is wrong: 0 is "not enabled" and 1 is "activated" • Add Graph "Brake Timeframe" in FAULT condition in object 0x3002 • Modify Graph Dynamic Stop timeframe in object 0x3007 • Update Error List CANopen: - Parameter Error:0x6322 - Data Set Programming: 0x6404, 0x6405,0x6406, 0x6407, 0x6408 - Following Error: 0x8410, 0x8411 - Over Speed: 0x8412 • Add in "Functions" chapter the new digital Input functions: SETVEL1, SETVEL2, SETVEL3, SETVEL4, VEL/CUR, DIRECTION • Add in object Digital Input Function (0x3020) the subindex 4, 5, 6 • Add object Communication CAN: - 0x4100 State Communication CAN - 0x4101 Counter Communication CAN - 0x4102 Settings Communication CAN - 0x4103 Error Communication CAN • Add Table Diagnostic Led Code - Over Speed (5,8) - Following Error (5,10) • Add in "Functions" chapter the Over Speed • Add Alarm Monitoring object 0x4200 • Modify object description 0x3008 Emergency Input Enable
1.7	**06/09/2021**	**Update Manual for firmware 225:** • Errata corrige: - In default mapping the transmission Type of TPDO1 is 0xFF and not 0xFD - Misprint Error Code: 0x8411 - Object 0x1000 Data Type is U32 (not U8) • Rewritten chapter PDO • Add chapter "TRANSITION TO EXIT FROM "OPERATION ENABLED" STATE" • Add object (ONLY read) 0x605A, 0x605B, 0x605C, 0x605D, 0x605E, 0x6007 • Modify object 0x3005:6 (filter for Velocity Actual Speed 0x606C)
1.8	**10/11/2021**	**Manual for firmware 226:** • Errata corrige: - Misprint Error for Factory Group equation (all document) - Modified Graphics Dynamic Stop - Object 3002:4 Brake timeout eliminate the phrase: "Only without Dynamic Stop (see object 0x3007)" - Object 0x3007: in graph "Dynamic Stop Disabled" changed the phrase "then the torque will be apply immediately" with "then the torque will be released immediately"

		and the red line continuous and dashed (Torque) must fall before. • Modified example of Factory Group Velocity and Acceleration (Chapter "Measuring Unit conversion" • Re-written paragraph of controlword object 0x6040 and statusword 0x6041 • Modified Object 0x6086 (add type profile) • Add PDO mapping for object 0x606B, 0x6078, 0x6079 • Update Error List CANopen: - 0x2353 Warning Load Level (i^2t) not rearmed - 0x6409, 0x640A, 0x640B, 0x640C, 0x640D Programming Error Data for ModBus Protocol - 0x7520 Serial Interface Error n°2 – ModBus - 0x7521 Communication Error Modbus – Lost Communication - 0x8121 Communication Error CANbus - BusOff • Add "Warning i2t not rearmed" bit 20 of Warning Object 0x2003 • Update CANopen Object List • Add Analog Input object 0x3040 • Errata corrige Dynamic Stop changed in Dynamic Stop
1.9	**7/11/2022**	**<u>Manual for firmware 230:</u>** • Errata corrige: - In Transition to "Ready To Switch On" or "Switch On Disabled" 605Bh (chapter TRANSITION TO EXIT FROM "OPERATION ENABLED" STATE) the exit from this state is not for inertia but depending by the dynamic brake configuration - Typing Error example re-mapping PDO : RX: Id 0x601 – 23 00 14 01 01 ~~20~~ 00 00 → RX: Id 0x601 – 23 00 14 01 01 02 00 00 - Object Warning - **Typing Error**: "To clear the warnings, set fault reset bit (#7) in Controlword (6040h)" is wrong. **Correct**: "To clear the warnings, set warning bit (#11) in Controlword (6040h)" - Transition machine state table chapter controlword there were 2 transitions wrong. - Chapter "STORE and RESTORE", after "the following Objects can be changed and stored in E2prom by writing in object 1010h: 2h (Communication Parameters)" → modified the object list • Update CANopen Object List: - Add object 0x2043 - Add object 0x2070 - Add object 0x2080, 0x2081, 0x2081, 0x2082, 0x2083 - Add object 0x2090, 0x2091, 0x2092, 0x2093 - Add object 0x3030, 0x3031, 0x3032, 0x3033 - Add object 0x3100, 0x3101, 0x3102, 0x3103, 0x3104, 0x3105, 0x3106, 0x3107 - Add object 0x3400, 0x3401, 0x3402, 0x3403, 0x3500, 0x3501 - Add object 0x3800, 0x3801, 0x3802, 0x3803, 0x3900, 0x3901 • Update Error List CANopen: - Add error code 0x2353, 0x3211, 0x556E, 0x556F, 0x6410, 0x6411, 0x8121 • Added Digital Output Object 0x3030, 0x3031, 0x3032, 0x3033, 0x3034 • Modify object digital Input 0x3020: add EXAMPLE • Added description digital ouput in chapter FUNCTIONS paragraph "DIGITAL I/O" • Re-written SYNC protocol (add bit definition COBID message) • Added object 0x3100 Configuration 1 – statusword • Re-written object status word because bits manufacturer can be modified by object 0x3100 • Added "PROFILE VELOCITY WITH PDO MAPPING DYNAMIC" • Modify object 0x3004 "Feedback Parameters", add subindex for encoder multiturn • Added example in Analog Output object 0x3050 • Added example in STORE and RESTORE object and correct list parameters

1.9a	**10/01/2023**	**Manual for firmware 231:** **ERRATA CORRIGE:** • Object 0x2050 and 0x2070 modified units: from A/100 → Arms/100
1.9b	**20/01/2023**	**Manual for firmware 232:** **ERRATA CORRIGE:** • Misprint unit measure object 0x6071, 0x6073, 0x6087, 0x6074, 0x6077, 0x6078 → **Typing Error**: [1000~~/Rated Current~~] **Correct**: [Rated Current/1000]
1.10	**27/11/2023**	**Manual for firmware 235:** **ERRATA CORRIGE:** • none **Chapter 1 – PORPUSE OF MANUAL** • none **Chapter 2 – CANOPEN OPERATION** • Update Error List CANopen: - Added Error Code 0x6412, 0x6413, 0x6414, 0x6415, 0x6416 - Added Error Code 0x8C06 - Added Error Code 0xFF11 - Added Error Code 0xFF21, 0xFF22, 0xFF23, 0xFF24 **Chapter 3 – MEASURING UNIT CONVERSION** • none **Chapter 4 – SAFETY** • none **Chapter 5 – CANOPEN OBJECT DICTIONARY** • Added object description 0x2080:0 STO statusword • Modify object 0x6083 and 0x6084 → these objects became PDO mapping. • Modify object description Digital Input 0x3020, 0x3021, 0x3022, 0x3023, 0x3024. • Modify object description Digital Output 0x3030, 0x3031, 0x3032, 0x3033, 0x3034. **Chapter 6 – CANOPEN OPERATION MODES** • none **Chapter 7 – CANOPEN OBJECT LIST** • Added object 0x4400, 0x4401, 0x4402. • Added object 0x3101. • Modify object 0x3021, 0x3022, 0x3023, 0x3024. • Modify object 0x3031, 0x3032, 0x3033, 0x3034. **Chapter 8 – FUNCTIONS** • Added "Emergency" paragraph that it describes the emergency input behavior. Move the diagram of emergency from object 0x3008 • Move example of FUNCTIONS chapter in the relative object **Chapter 9 – DIAGNOSTIC** • Added Error Led code for "STO Circuit" group

1.11	**27/05/2024**	**Manual for firmware 236, 237 and 238:** **ERRATA CORRIGE:** • Chapter Safety – correct scheme of Finite State Machine: the exit from state SAFETY goes to "Switched On Disabled" and NOT to "Not ready to Switch ON" • Chapter RPDO Mapping Default and Table RPDO4 Mapping it was written ~~Target Velocity~~ → correct is Target Torque • In "APPENDIX - FIRST CONFIGURATION" when it describes save in eeprom procedure Data the phrase has the value wrong: *Value "Data" = ~~0x73617665~~ (means "save" in ASCII code) → Value "Data" = 0x65766173 (means "save" in ASCII code)* **Chapter 1 – PORPUSE OF MANUAL** • none **Chapter 2 – CANOPEN OPERATION** • Update Error List CANopen: - Added Error Code: 0x5B00 … 0x5B18. - Added Error Code: 0x6323 … 0x6325. - Added Error Code: 0x6417 … 0x641B. - Added Error Code: 0xFF01. - Regroup sequential Error Codes • Modify chapter "TRANSITION TO EXIT FROM OPERATION ENABLE" state. Added new mode of Braking , added different configurations. • In "Store and Restore" chapter updated object that they can saved in eeprom **Chapter 3 – MEASURING UNIT CONVERSION** • none **Chapter 4 – SAFETY** • Modify STO management: add different mode for transition into FSM. • Added STO smartris Compact. • Move object description in the chapter 5. **Chapter 5 – CANOPEN OBJECT DICTIONARY** • Modify object 0x1005:0 (add abort code and description) • Modify description 0x3007: different mode for fw 236 and add subindex 7 • Modify description 0x3008: add subindex 4 that define transition mode to exit from RUN state with emergency command • Add object 0x3102:0, 0x3103:0 • Modify object 0x605C:0. • Modify object 0x605E:0. • Modify object 0x605B:0. • Modify object controlword 0x6040 → add emergency command 12 bit. **Chapter 6 – CANOPEN OPERATION MODES** • none **Chapter 7 – CANOPEN OBJECT LIST** • Update CANopen Object List: - Added object 0x3102:0, 0x3008:4

		- Modify 0x605C:0, 0x605E:0, 0x605B:0 **Chapter 8 – FUNCTIONS** • Added chapter braking Mode. • Move example of FUNCTIONS chapter in the relative object • Modified description and graph object Electronic Dynamic Stop 0x3007 **Chapter 9 – DIAGNOSTIC** • Added Led Code for alarm "E2prom function" group • Added Led Code for alarm "Safety" group
1.12	**03/07/2024**	**Manual for firmware 240, 241 and 242:** **ERRATA CORRIGE:** • Misalignment with firmware: object 0x2060, 0x2070 can be configured PDO mapping • Error written: Object 60E0: the access was ~~ro~~ , it must be rw • Error written: Object 60E1: the access was ~~ro~~ , it must be rw • Missing information: Mode "Safety FAULT" is not available, to enable please contact manufacturer (see the chapter SAFETY) **Chapter 1 – PORPUSE OF MANUAL** • none **Chapter 2 – CANOPEN OPERATION** • Update Error List CANopen "Emergency Protocol": - Added Error Code: 0x5B19, 0x6326, 0x6420, 0x6421, 0x6422, 0x6423, 0x8135, 0x8136, 0x8137, 0x8B41, 0x8B42 - Changed name of macro group (Group of error codes from 0x8B10 to 0x8C00) : ~~*"Init Object CANopen from Eeprom"*~~ → Init values P402 from Eeprom • Chapter "STORE AND RESTORE" added new objects that they can stored **Chapter 3 – MEASURING UNIT CONVERSION** • Added Object 60A2h: Jerk Factor **Chapter 4 – SAFETY** • none **Chapter 5 – CANOPEN OBJECT DICTIONARY** • Added Object 0x60A3 (for S-Ramp Profile) • Added Object 0x60A4 (for S-Ramp Profile) • Modified object 0x6086, added values -2 and -3 • Modified object 0x3007, added subindex 8 • Modified object 0x3008, 0x605C, 0x605B, 0x605E added value '-5' (for S-Ramp Profile) • Added Object 0x3104 (for warning Programmable) • Added Object 0x3401 (for warning Programmable) • Added Object 0x3900 (for warning Programmable) • Modified Object 0x6007, added values 0, 1, 2 • Modified object 0x3100 , added values 2, 3 • Modified object 0x4101, added subindex 4 • Modified object 0x4102, added subindex 3

		• Modified object 0x4103, added subindex 3 and 4 • Moved objects 0x1001, 0x1003, 0x4200, 0x2003, 0x603F, 0x3010 have been added in the DIAGNOSTIC chapter. **Chapter 6 – CANOPEN OPERATION MODES** • none **Chapter 7 – CANOPEN OBJECT LIST** • Updated CANopen Object List: - Replaced in column "Argument" the word ~~Alarm~~ with Diagnostic - Added object 0x5000, 0x5001, 0x5002, 0x5003, 0x5004, 0x5005, 0x5010 - Added object 0x60A2, 0x60A3, 0x60A4, 0x3007:8 - Modified 0x6007 ~~RO~~ with R/W - Added subindex in object 0x4101:3, 0x4101:4, 0x4102:3, 0x4103:3, 0x4103:4 - Modified 0x3900:1 and 0x3900:2 - Modified 0x2003 → name is began "Warning Flag" **Chapter 8 – FUNCTIONS** • Changed title TRANSITION → TRANSITION STOPPING MODE • Added chapter "ABORT CONNECTION" **Chapter 9 – DIAGNOSTIC** • Re-Written the table of Leds code, add led code combination of Smartris Compact. It divided the Chapter in 3 argument : LED CODE, ALARM and WARNING • Copied object 0x1001, 0x603F, 0x4200, 0x3010, 0x1003 (they are removed from 'CANOPEN OBJECT DICTIONARY') • Added description Logger error table saved in eeprom (last 5 errors occurred), added object 0x5000, 0x5001, 0x5002, 0x5003, 0x5004, 0x5005, 0x5010 • Copied objects 0x2003 (it is removed from 'CANOPEN OBJECT DICTIONARY'), and change the name "warning" → "Warning Flag" • Added Warning object 0x3104, 0x3401, 0x3900 **APPENDIX** • Added SRAMP profile