



PRELIMINARY

**Technical Customer Information**

**Actuator SENSE 42**

Sonceboz SA  
8565R1000

## Table of contents

Revision History .....	3
1. Introduction .....	4
2. Power supply .....	4
2.1. Harness Installation Consideration.....	5
3. Technical Data.....	7
4. Mechanic .....	9
4.1. Overall dimensions of Actuator .....	9
4.2. Plug definition .....	10
4.3. Gear rack interface .....	10
5. Detailed Functionalities.....	11
5.1. Motor State Diagram.....	11
5.2. Positions Configuration and Parameters .....	12
5.3. Motor Speed Configuration and Limitations .....	13
5.4. Integrated Encoder .....	14
5.5. Sensor calibration with the valve.....	15
5.6. Stall Detection.....	16
5.7. Dead Zone Detection.....	18
5.7.1. Full Precision Mode .....	18
5.7.2. Low Precision Mode .....	18
5.8. Linearization .....	19
5.9. Boost Operation.....	20
5.10. Errors .....	22
5.10.1. Under- and overvoltage state Diagram.....	23
5.10.2. High Temperature function state Diagram.....	23
6. Communication.....	25
6.1. CAN J1939 Communication.....	25
6.1.1. System operation.....	25
6.1.2. Special Parameter groups (data link layer) .....	27
6.1.3. Transport protocol (Data link layer) .....	31
6.1.4. Network management.....	34
6.1.5. Diagnostics .....	39
6.1.6. Identification messages .....	47
6.1.7. Memory access.....	50
6.1.8. Actuator control.....	69
6.2. CAN J1939 Bootloader Protocol .....	82
6.2.1. State diagram power up sequence .....	82



6.2.2.	Protocol Overview.....	83
6.2.3.	Protocol details .....	86
6.2.4.	Address claim procedure .....	103
6.2.5.	DM1 Active diagnostic trouble codes .....	105
6.2.6.	Not supported J1939 functionality.....	107
7.	Product validation .....	108

## Revision History

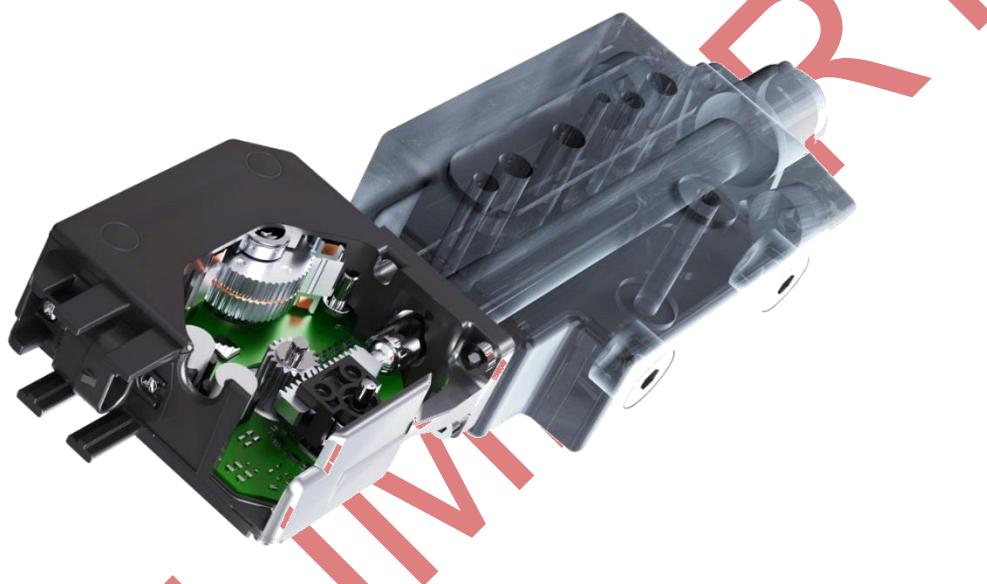
Version	Description	By	Date
0.1	New Document – DRAFT (not released)	MCH	01.06.2016
0.2	DRAFT: Modification of plan to integrate “12/24V”, and clarification of hysteresis on voltages	MCH	24.06.2016
0.3	Integration of CAN communication specification Add of Harness Consideration	MCH	29.08.2016
0.4	New revision after review: Precision on resolution Change connector pinning to include 12V and 24V Various language corrections	MCH	08.09.2016
1.0	First Preliminary release	MCH	08.09.2016
1.1	Integration of Boost function Clarification of Errors and addition of some New Linearization	MCH	30.09.2016
2.0	New revision after review: Update of Memory data and Memory Access (NAU) Addition of some information on speed management (saturation and configuration) §0 Update of Boost Function (current values)	MCH	19.01.2017
3.0	Update of Memory data and Memory Access (NAU) Update of Sensor state value (measured position) (NAU) Update boost Saturation value (NAU) Update of Stall Detection with encoder description (MCH) Addition of a § on Positions Configuration (MCH)	NAU / MCH	28.02.2017

## 1. Introduction

The SENSE 42 is a linear actuator with integrated position encoder which makes part of the S42 actuator series.

The S42 actuator series has been designed for electromechanical actuation of proportional hydraulic valves. Depending of the valve design, the actuator provides an accurate direct motion of the spool up to 180 l/min of flow. The movement is realized by a specific electric brushless DC motor called hybrid stepper motor, a gear and a toothed rack. The electronics features on board diagnostics and a CAN bus communication.

The S42 actuator series is designed for mobile hydraulics applications like agricultural or construction machines.



The SENSE 42 version has been specially designed for applications where the position information is mandatory, like for valves with combined manual override function, or to replace a separate stand-alone position sensor on the valve. The SENSE 42 actuator integrates a contactless and robust Hall Effect absolute position encoder.

## 2. Power supply

The SENSE 42 actuator is compatible with both 12 V and 24 V power supply but maximum performances will be limited depending on Power Supply (See §):

Temp	Power Supply		
	Low_Voltage_Mode	12V_Mode	24V_Mode
Normal_Temp_Mode	40 mm/s	80 mm/s	100 mm/s
Low_Temp_Mode	40 mm/s	60 mm/s	60 mm/s
VeryLow_Temp_Mode	10 mm/s	10 mm/s	10 mm/s

## 2.1. Harness Installation Consideration

Please take note that the harness must be designed with attention according to the total length of the cable (from battery to the actuators), the contact resistance of plugs along the harness, the number of actuators supplied by the harness and so on.

### **IMPORTANT:**

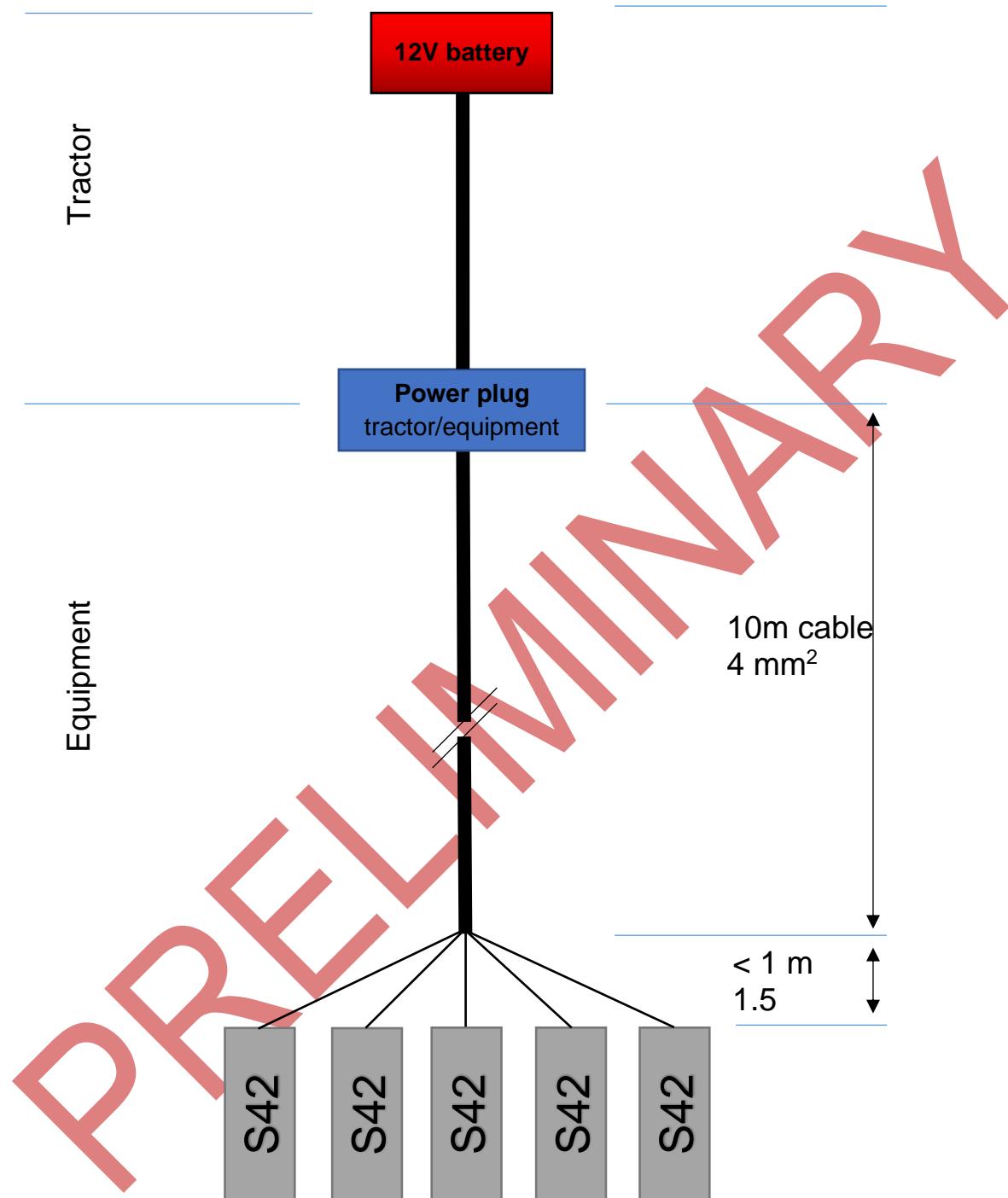
If several actuators are powered by a common cable, the harness must be designed for all actuators even if only one actuator of all is used at the same time because all actuators will start simultaneously at power on.

According to the data from product specification (under-voltage, low speed mode voltage), the section of the cable must be chosen to guarantee the required behaviour on actuator considering the worst case (with a margin) of 5A peak per actuator.

Example of harness design:

	Example 1	Example 2
Cable length	3m harness (6m: VBAT line + GND line)	10m harness (20m: VBAT line +GND line)
Max peak current	3 actuators = $3 \times 5 \text{ A}$	5 actuators = $5 \times 5 \text{ A}$
Average current	$3 \text{ actuators} \times 2.5 \text{ A} = 7.5 \text{ A}$	$5 \text{ actuators} \times 2.5 \text{ A} = 12.5 \text{ A}$
Nominal battery voltage	13.5 V	13.5 V
Min allowed Supply voltage on actuator	9 V never declare undervoltage but enter in low speed mode	10.8 V never enter in low speed mode
Max allowed voltage drop	$13.5 \text{ V} - 9 \text{ V} = 4.5 \text{ V}$	$13.5 \text{ V} - 10.8 \text{ V} = 2.7 \text{ V}$
Max allowed total cable resistance (2 lines)	$4.5 \text{ V} / (3 \times 5 \text{ A}) = 0.3 \Omega$ $6 \text{ m of } 0.35 \text{ mm}^2 = 0.291 \Omega$	$2.7 \text{ V} / (5 \times 5 \text{ A}) = 0.11 \Omega$ $20 \text{ m of } 4 \text{ mm}^2 = 0.085 \Omega$
Cable section	calculation from cable resistance gives a section of $0.35 \text{ mm}^2$ taking into account the average current of 7.5A and a max current density of $6 \text{ A} / \text{mm}^2$ gives $1.5 \text{ mm}^2$ → use $1.5 \text{ mm}^2$	calculation from cable resistance gives a section of $4 \text{ mm}^2$ taking into account the average current of 12.5 A and a max current density of $6 \text{ A} / \text{mm}^2$ gives $2 \text{ mm}^2$ → use $4 \text{ mm}^2$

Example for 5 actuators (example 2 above):

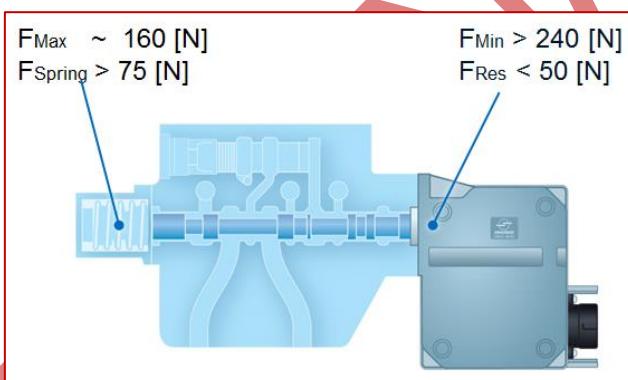


### 3. Technical Data

Feature	Values and units	Remark, comment		
<b>Stroke</b>				
Working stroke: $S_{\text{nom}}$	25mm			
<b>Actuating force</b> (See also recommended actuating forces on the valve)				
Minimum actuating force provided by the actuator $F_{\text{min}}$	240 N	At standard speed		
Minimum actuating "boost" force provided by the actuator $F_{\text{min\_boost}}$	300N	At standard Speed @ 30% Boost		
Maximum recommended load force	160 N	Margin to 240 N		
Holding force $F_h$	250 N	Not in neutral position		
Maximum residual force	50 N	Unpowered motor		
<b>Actuator linear speed</b>				
Standard speed:	100mm/s 80mm/s	For 24V Power Supply Selection For 12V Power Supply Selection		
Restriction	60mm/s	Low Speed 1 Speed to float position		
	10mm/s	Low Speed 2		
	40 mm/s	Low voltage Speed		
Priority	For Details See §5.9			
<b>Resolution</b>				
<small>Rq: Limited by CAN protocol</small>				
Min. linear resolution with 8 Bit	0.028 mm/CAN Value	With $\pm 7\text{mm}$ Stroke		
<b>Linearization of actuation</b>				
Displacement / command	Adjustable on customers request	See §5.8		
<b>Power Supply</b>				
Operating voltage: $U_{\text{min}} \dots U_{\text{max}}$	8.4V ... 32,5 V	Fully functional (with performances depending on value) (See §2)		
Speed Restriction	16 V ... 21 V	Low voltage speed (See §5.9)		
	9V ... 11.2V	For 24V Power Supply Selection For 12V Power Supply Selection		
CAN Available	8V to 35V			
Operating current	Typical values Operation: 2.6 A Safety: 40 mA Inrush current: 10 A @ 11 V for 500 us Neutral: 130 mA			
<b>Operating Temperature</b>				
Ambient Operating temperature range	-40 °C to 85 °C	Restrictions in duty cycle may allow higher temperatures		
Restrictions based on internal temperature sensors are influenced by internal self-heating	For Details see §0			
Max storage temperature	125°C	Max 3 hours with no function		
<b>Communication</b>				

Feature	Values and units	Remark, comment
CAN-BUS <u>Rq:</u> No terminal resistance included	J1939 Compatible	See §6.1
Bootloader	J1939 Bootloader	See §6.2
<b>Internal Protection</b>		
Automatic shut-off	< 8.4V > 32.5V Temperature > 118°C Temperature < -40°C	UnderVoltage (See §5.10) OverVoltage (See §5.10) Internal temperature measurement (See §5.1 and §5.10.2)
<b>Special features</b>		
Mechanical play/wear detection	Automatic play detection between 0.1 to 1 mm	Depend on application environment
<b>Lifetime</b>		
Minimum requirement	10 Million cycles	At force 100 N ; Stroke +/-7mm
<b>Dimensions</b>		
L x B x H	120 x 115 x 42	See drawing for details (§4)

Recommended actuating force of the valve including design margins (information only):

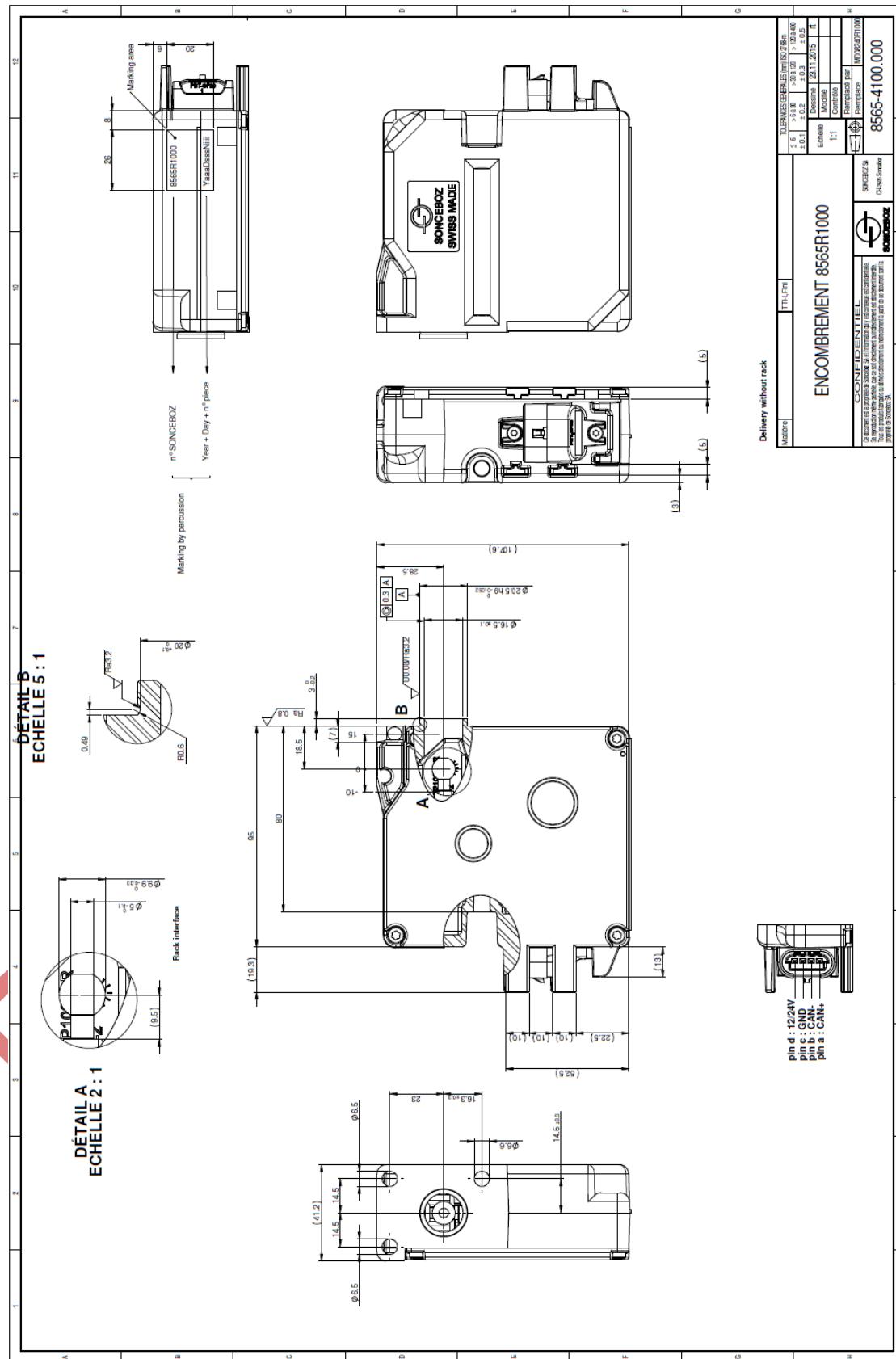


Definition of  $F_{max}$  :  $F_{max} = F_{spring} + F_{oilflow} + F_{friction} (\text{seal})$

PRE

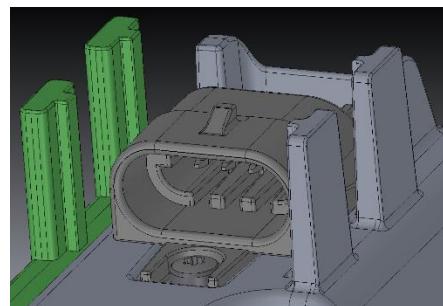
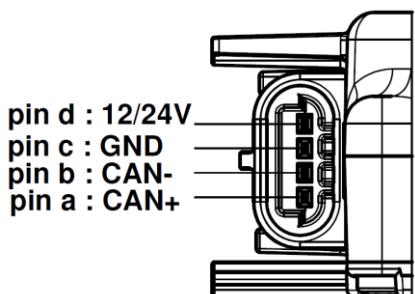
## 4. Mechanic

#### **4.1. Overall dimensions of Actuator**



## 4.2. Plug definition

Actuator connector: Delphi Metri Pack 150.2



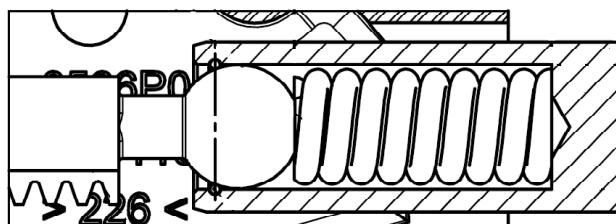
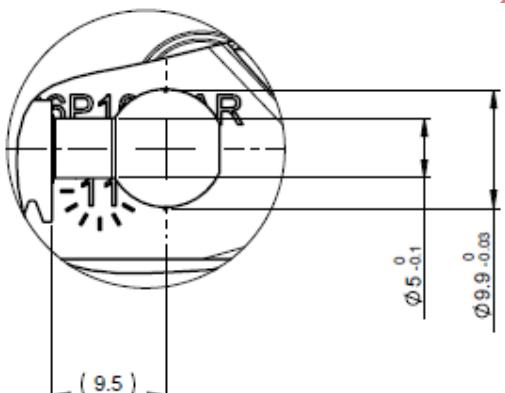
Recommended mating plug:

- Body: 12162188
- Contacts: 12103881



## 4.3. Gear rack interface

Recommended design:

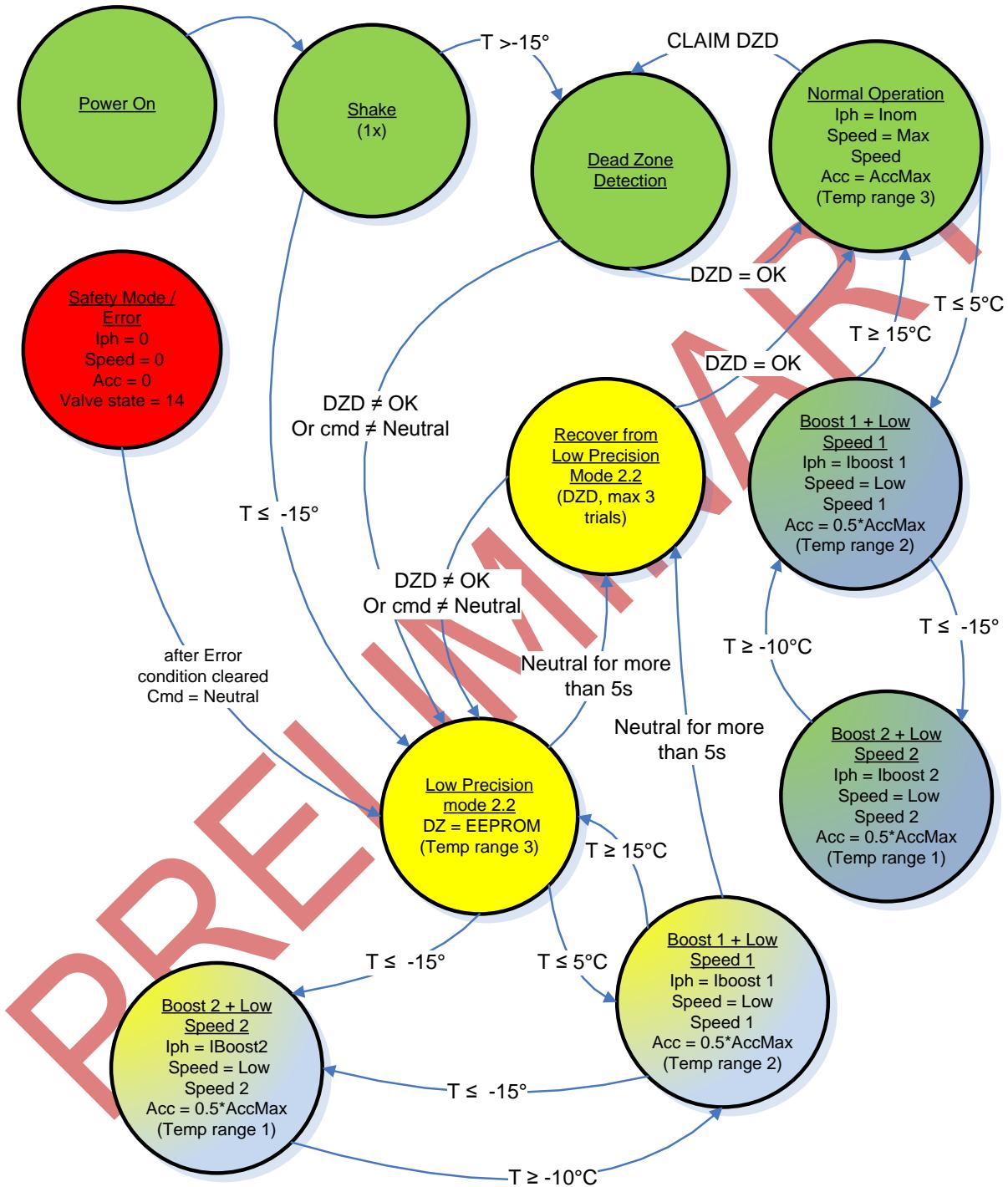


P1

## 5. Detailed Functionalities

### 5.1. Motor State Diagram

The motor will apply the following state diagram depending on external and/or internal conditions.



The shake function is made in order to avoid stick-slip effects of the seal on the valve spool valve.

## 5.2. Positions Configuration and Parameters

The SENSE42 enables the configuration of stroke parameters as described below:

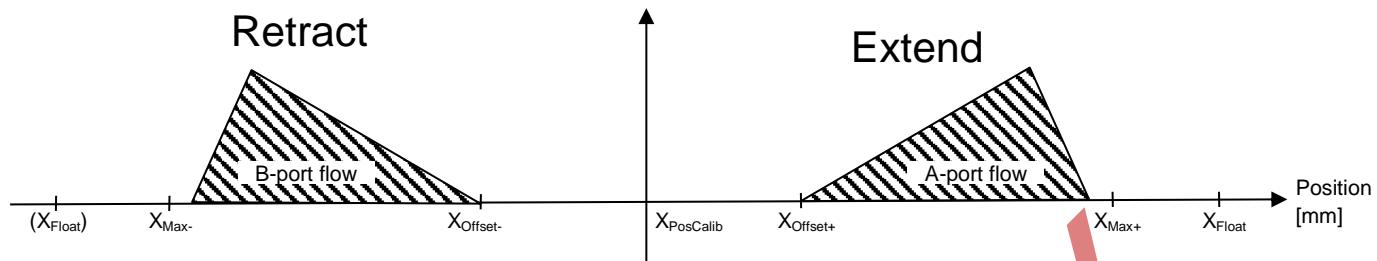


Figure 5-1 Valve positions parameters

The 2  $X_{\text{offset}+/-}$  parameters enable to stay on a “ready to have oil” position in order to suppress the possible delay due to the “No Hydraulic Flow” zone.

The 2  $X_{\text{max}+/-}$  parameters enable to limit the maximum stroke of the valve.

The 2 parameters  $X_{\text{float}}$  and  $X_{\text{float\_dir}}$  (not represented) enable to define a Float position and its direction.

The configuration can be changed through the parameters stored in the non-volatile memory (see §6.1.7)

PREELIMINAR

### 5.3. Motor Speed Configuration and Limitations

The Speed of the motor will depend of its configuration (stored in the non-volatile memory) and of the external conditions:

- §5.1 describes the “Low” Temperature conditions
- §5.10.1 describes Power Supply conditions

The configuration can be changed through the following parameters stored in the non-volatile memory (see §6.1.7), except for Low Speed 2 which is the lowest limitation and cannot be changed:

- Normal Speed 
- Float Speed 
- Low Voltage Speed 
- Low Speed 1 (Low Temp Speed) 
- Low Speed 2 (Very Low Temp Speed) 

The table below explains which parameter sets the speed of the motor, depending on conditions. The speed on the table are also to be considered as maximum/saturation speeds. Any parameter above those values will be limited:

		Power Supply		
		Low_Voltage_Mode	12V_Mode	24V_Mode
Temp	Normal_Temp_Mode	40 mm/s	80 mm/s	100 mm/s
	Low_Temp_Mode	40 mm/s	60 mm/s	60 mm/s
	VeryLow_Temp_Mode	10 mm/s	10 mm/s	10 mm/s

The minimum configurable speed is 10mm/s. Any parameter below this value will be limited:

Example: If the motor is in Low\_VoltageMode (<10.8V) and in Low\_Temp\_Mode (<5°C), the speed of the motor will be controlled by the parameter Low Voltage Speed. This parameter can be configured between 10mm/s and 40mm/s.

## 5.4. Integrated Encoder

The SENSE 42 actuator integrates a patented Hall Effect absolute position encoder:

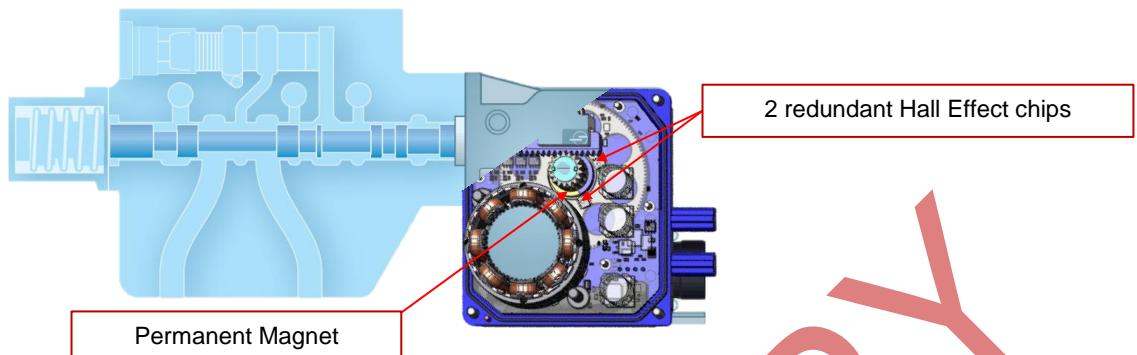


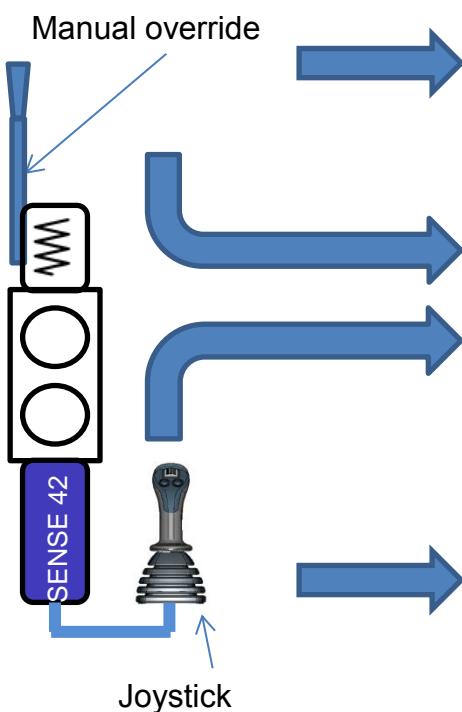
Figure 5-2 Encoder Integration

The position encoder enables to know the current absolute position of the valve spool which is mechanically linked with the sensor through the gear rack. It will particularly allow to check if the valve spool is back in the "No Hydraulic Flow" zone after it has been moved back in that position by the valve spring.

Feature	Values and units	Remark, comment
<b>Encoder</b>		
Resolution	0.025mm	Worst case
Precision	+/- 1.2mm +/- 0.8mm	Over full temperature range Over +/-7mm working stroke Over +/-1mm working stroke

The SENSE 42 actuator is perfectly suited to be used in combination with a manual override lever on the valve, following the principle describe below:

PRE  
MANUAL OVERRIDE

**Manual Command:**

- The SENSE42 is «shut-off» and does not interfere with the movement
- The encoder position is sent through CAN

**Simultaneous Manual and Joystick Command:**

- The SENSE42 declares a «stall» detected by the sensor and shuts-off itself.  
→ Manual override priority
- The encoder position is sent through CAN
- To reset the «stall», Manual and Joystick shall be in Neutral position for a minimum configurable time.  
(The SENSE42 checks that the encoder is in the “No Hydraulic Flow” zone for at least  $T_{Reset}$ )

**Joystick Command:**

- SENSE42 controls the valve according to the CAN commands

**Figure 5-3 Dead Zone Detection function*****5.5. Sensor calibration with the valve***

In order for the actuator to learn the “No Hydraulic Flow” position of the valve, a calibration is needed when coupling the actuator to its valve. This calibration is made automatically with a new actuator. As soon as the actuator succeeded in doing its first Dead Zone Detection (See §5.7), the zero position of the encoder will be defined and saved as the middle of the Dead Zone.

In case of changing the valve, it is possible to reset this calibration by clearing the corresponding parameter in the non-volatile memory through CAN. For more details see §6.1.7.

## 5.6. Stall Detection

To guarantee a correct operation of the actuator in all working conditions (temperature, vibration, etc.) the maximum valve load must be adapted to the maximum force of the actuator. For more information refer to the recommended actuating force of the valve in chapter 3.

In case the external load is higher than what the actuator can provide as maximum force (i.e. a sticking spool), the Stall Detection function will detect this problem, shut-off the power supply of the actuator and send back this information via the CAN bus.

The stall detection function of the SENSE 42 actuator is based on two principles which are working in parallel:

- 1) Back-EMF principle (see Figure 5-4). The back electromotive force is the voltage that occurs in electric motors where there is relative motion between the stator (fixed part with the coil windings) and the magnetic field from the rotor (moving part with the permanent magnet). This principle, which works with a software algorithm based on voltage and current measurements, has some limitations especially at very low speeds. So it cannot be considered as a 100% guaranteed detection.

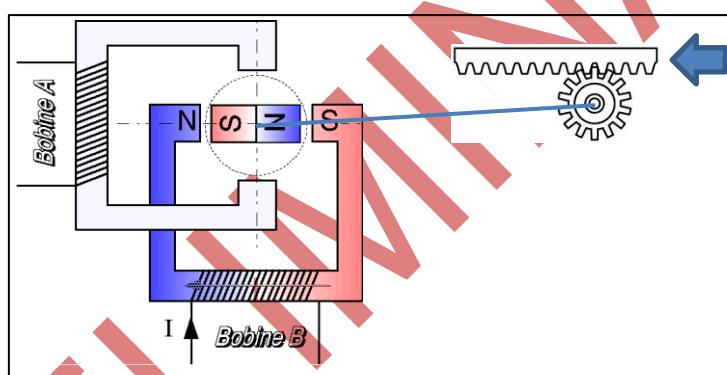


Figure 5-4 Back EMF is measured in the actuator windings

- 2) Absolute position encoder. This second principle uses the absolute encoder position to detect if there is a stall.

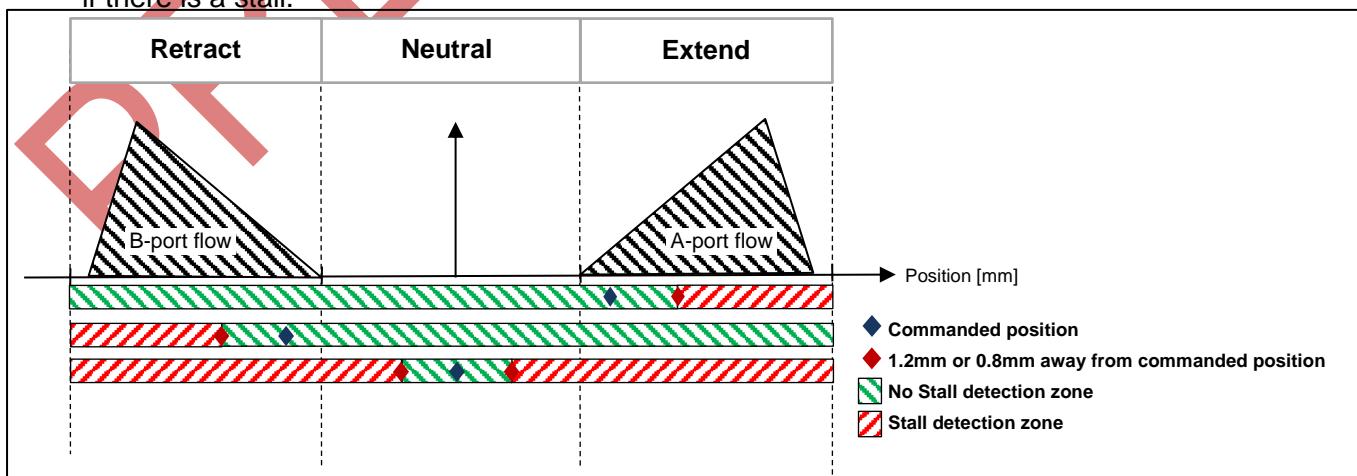


Figure 5-5 Stall detection principle

The detection of the stall is made following the principle describe in the figure above:

- In Extend positioning, the limit is set to "Commanded position + 1.2mm"
- In Retract positioning, the limit is set to "Commanded position - 1.2mm"
- While commanded in Neutral position, the limits are set to "Xoffset- +0.8mm" and "Xoffset+ - 0.8mm" and limited to a zone of +/-0.8mm around the 0 position.

The error will be declared if the measured position is in a "stall detection zone" for more than a confirmation time. The confirmation time is dependent on the current working speed of the actuator. The configuration of those confirmation times can be changed through the parameters stored in the non-volatile memory (see §6.1.7):

Description	Default value
<b>Protection reaction time for 10mm/s</b>	<b>1500 ms</b>
<b>Protection reaction time for 20mm/s</b>	<b>850 ms</b>
<b>Protection reaction time for 40mm/s</b>	<b>450 ms</b>
<b>Protection reaction time for 60mm/s</b>	<b>350 ms</b>
<b>Protection reaction time for 80mm/s</b>	<b>250 ms</b>
<b>Protection reaction time for 100mm/s</b>	<b>250 ms</b>

The combination of both principles will allow a quick and 100% guaranteed stall detection of the SENSE 42 actuator (in the limit of the encoder performances and of the electronic detection).

To reset a Stall Detection the actuator needs the two conditions below for at least the "Neutral fault Recovery -  $T_{Reset}$ ":

- A "Neutral" command received via the CAN bus
- A confirmation of the "No Hydraulic Flow" zone position by the integrated encoder

This "Neutral fault Recovery -  $T_{Reset}$ " is configurable between 0 and 25s (by step of 0,1s) by writing the corresponding value in the non-volatile memory (for more details see §6.1.7).

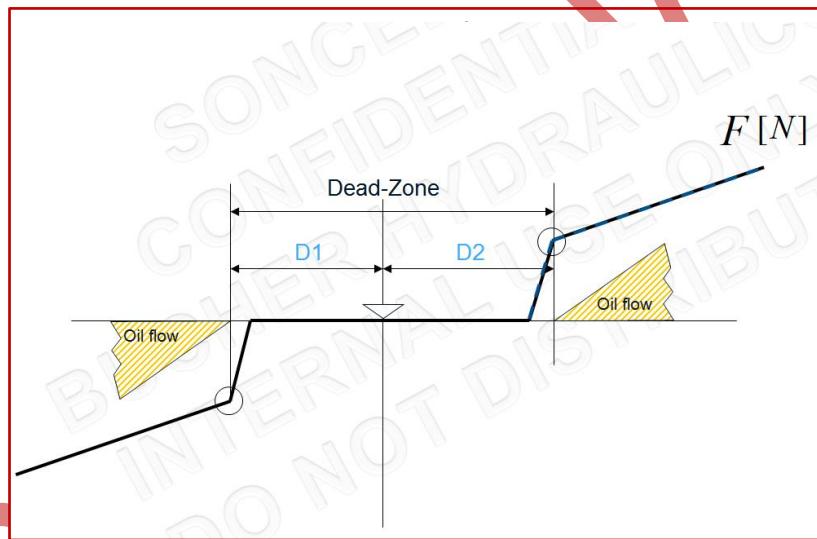
## 5.7. Dead Zone Detection

### 5.7.1. Full Precision Mode

The patented Dead Zone Detection (DZD) function allows to detect a steep increase of force, like the one of the spring of a hydraulic valve spool, in order to measure and compensate the mechanical play. This function increases the positioning precision since the Dead-Zone (mechanical play) will be invisible to the end user. The DZD function will also compensate for possible wear of the internal or external gear of the actuator. The result is a system (actuator and mechanics) that has identical position precision and behaviour when shipped and after several years of service.

The Dead Zone Detection procedure is made at each power on of the actuator (or upon request from CAN message). The procedure takes about 3 sec and after the procedure the actuator is put in a ready to use neutral mode.

The patented DZD function method is based on the motor back-EMF voltage. By making a specific sequence of forward and backwards steps combined with the measure of back-EMF voltage the length and absolute position of a steep increasing force can be determined, see Figure 5-6.



**Figure 5-6 Dead Zone Detection function**

At first calibration with the valve, if the dead zone is not found (for example no play or too much play...), the actuator won't authorize any movement (see §5.5).

After first calibration, the actuator will automatically switch to "Low Precision Mode" (see §5.7.2).

### 5.7.2. Low Precision Mode

If the motor receive a position command while performing its DZD, the motor will change its DZD from "Full Precision Mode" to "Low Precision Mode" (see chapter §5.1).

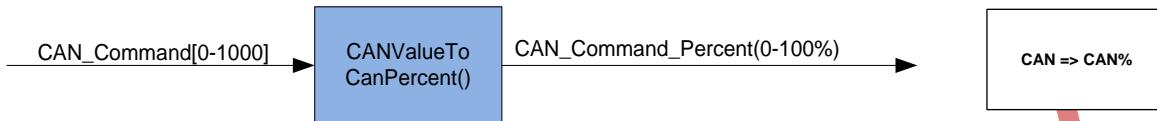
In this case, the Dead Zone will be determined by finding (only) one of the two spring and by applying the last value of Dead Zone detected (saved in the non-volatile memory).

The time to find the Dead Zone will be reduced to 350ms maximum.

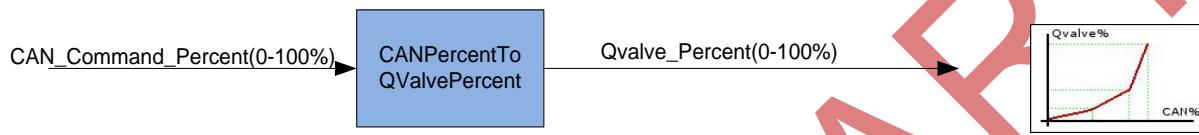
## 5.8. Linearization

In order to map any characteristic valve curve to a programmable (often linear) oil flow at the valve ports A and B the position command received on the CAN bus is reshaped using the following steps:

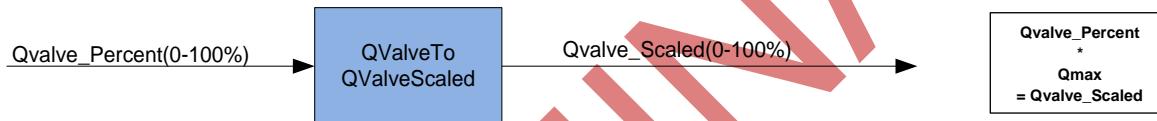
**Step 1:** The CAN command received (8 bits or 10 bits) will be transposed into a 0 – 100% value



**Step 2:** Application of the “Customer Curve” (configurable)



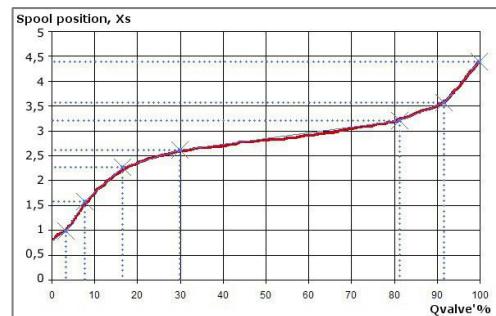
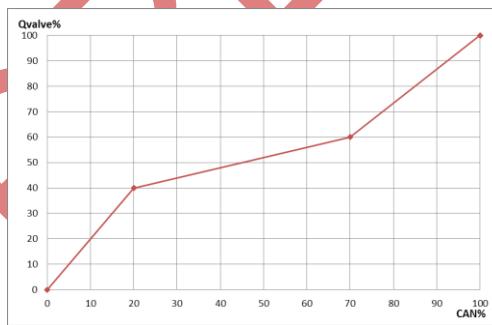
**Step 3:** Adaptation to maximum flow authorized (parameter in non-volatile memory)



**Step 4:** Application of the “Spool Linearization Curve”



Those steps enable, whatever the valve curve is (flow versus position), to change the reaction of the system (actuator + valve) under the same command. For example, to change from a linear control of the flow to an exponential one, just by setting the correct “Customer Curve”.

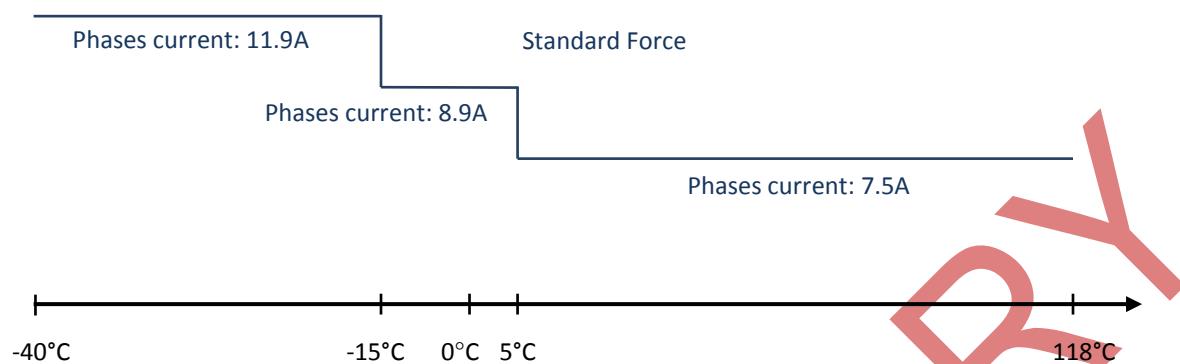


**Note:**

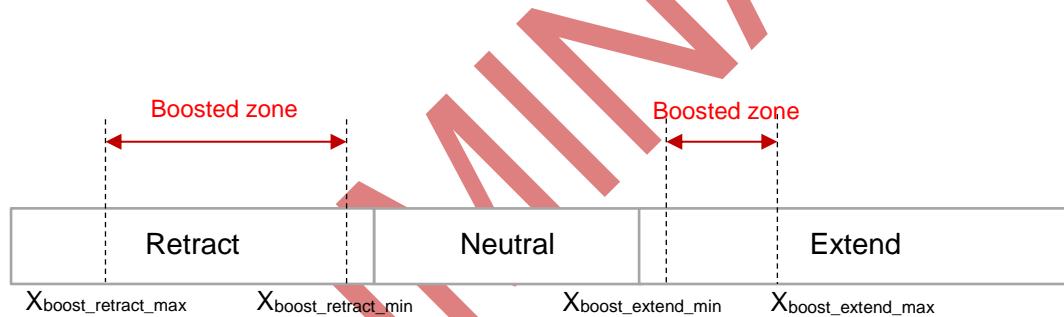
- The Customer curve can be set thanks to 10 configurable points (5 in extend and 5 in retract) saved in the non-volatile memory (see §6.1.7).
- The Spool Linearization curve can be set thanks to 16 configurable points (8 in extend and 8 in retract) saved in the non-volatile memory (see §6.1.7).

## 5.9. Boost Operation

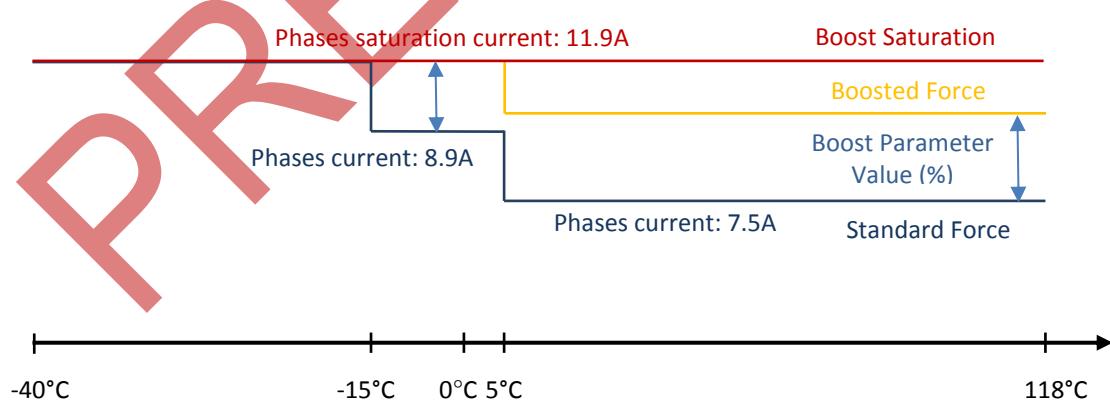
As a standard, the motor boosts its performance (by increasing the current flowing through the phases) when the temperature decreases, in order to compensate the increasing viscosity of the oil (see diagram below):



With the located position Boost It is possible by defining 4 points into the non-volatile memory (see §6.1.7) to parameter 2 zones (one in retract and one in extend) where the force of the actuator will be boosted.



This position located Boost effect can be controlled through the "Boost parameter" (saved in non-volatile memory), by setting a value between 0% and 30%.



Thus will enable the user to avoid some stall detection due to "hard points" in the application. This function is disabled by default (all points set to 0).

**WARNING:**

- A "Boost Saturation" saturation exist. When increasing too much the "located boost", the performance of the motor will be saturated

Saturation value :

	Low Voltage Mode	12V Mode	24V Mode
10 mm/s	ID_NORMAL	ID_BOOST_TWO	ID_BOOST_TWO
40 mm/s	ID_NORMAL	ID_BOOST_TWO	ID_BOOST_TWO
60 mm/s	ID_NORMAL	ID_BOOST_TWO	ID_BOOST_TWO
80 mm/s	ID_NORMAL	ID_BOOST_ONE	ID_BOOST_TWO
100 mm/s	ID_NORMAL	ID_NORMAL	ID_BOOST_TWO

ID\_NORMAL = 7,5A

ID\_BOOST\_ONE = 8,9A

ID\_BOOST\_TWO = 11,9A

- Using this function will make the motor heat-up more than in standard use. Care shall be taken to avoid overheating of the motor.

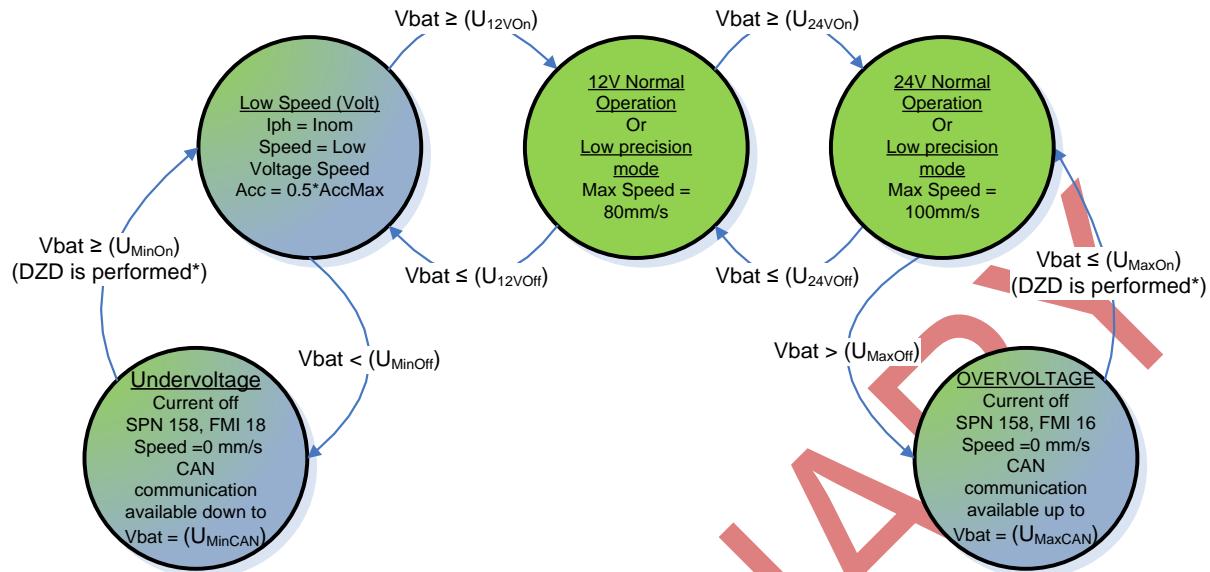
PRELIMINARY

## 5.10. Errors

The following errors will be detected and declared by the motor

SPN	Fault	ACU Action	ACU Restart
CAN message	Inconsistent CAN Control message	ACU drives spool back to neutral position and waits for valid message	New valid CAN Neutral position message AND Motor in Neutral Position
	No CAN Control message received = TIME OUT (configurable parameter in the non-volatile memory)		
FLASH / EEPROM inconsistent	EEPROM parameter write/read error	ACU turns off motor current and the central spring returns the spool to neutral position	No possible restart since Flash/EEPROM data are not reliable anymore. In case of temporary error, it could be cleared after a power up sequence
Output stage / Force (Motor Stall) / Manual Override	Stepper motor stall detected.	ACU turns off motor current and the central spring returns the spool to neutral position	New valid CAN Neutral position message AND Motor in Neutral Position for at least "Neutral fault Recovery - TReset"
Electrical failures	Undervoltage SONCEBOZ value ( $VDC < U_{MINOFF}$ ) during time more than 3 ms	ACU turns off motor current and the central spring returns the spool to neutral position	$VDC > U_{MINON}$ AND New valid CAN Neutral position message AND Motor in Neutral Position
	Oversupply SONCEBOZ value ( $VDC > U_{MAXOFF}$ ) during time more than 3 ms		
Temperature	SCU temperature higher than $T_{MAXOFF}$ during time more than 2 sec	ACU turns off motor current and the central spring returns the spool to neutral position	SCU temp $< T_{MaxON}$ AND New valid CAN Neutral position message AND Motor in Neutral Position
	SCU temperature below $T_{MINOFF}$ during time more than 2 sec		
Current	Current exceeds upper limit $I_{MAX}$ during time more than 1 ms	ACU turns off motor current and the central spring returns the spool to neutral position	New valid CAN Neutral position message AND Motor in Neutral Position
Calculation Position Error	Internal SW Error of calculation position in IDLE Mode	ACU turns off motor current and the central spring returns the spool to neutral position	New valid CAN Neutral position message AND Motor in Neutral Position
Linearization	Range error for Linearization parameter in EEPROM	ACU turns off motor current and the central spring returns the spool to neutral position	New valid CAN Neutral position message AND Motor in Neutral Position

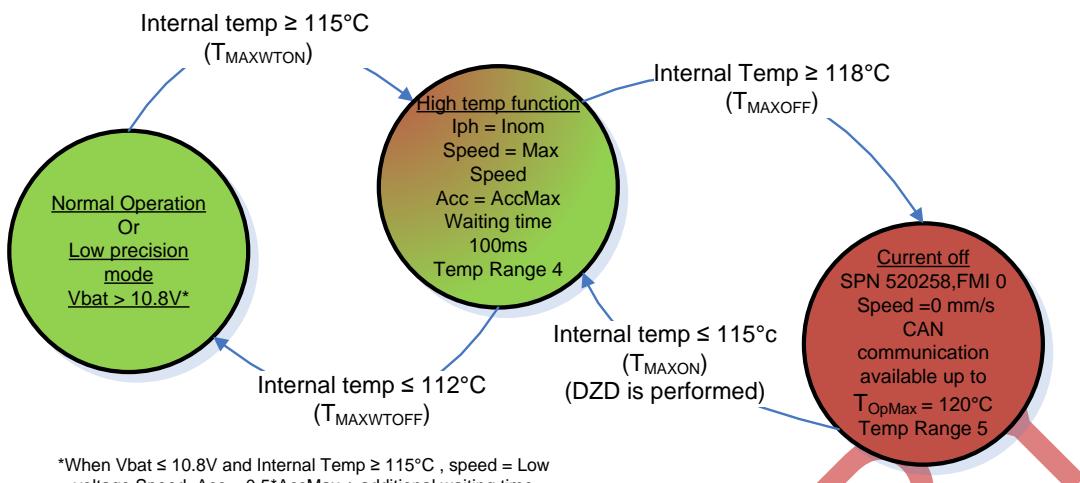
### 5.10.1. Under- and overvoltage state Diagram



Voltages Measured (SW)	
U <sub>MinOn</sub>	9.8V +/- 0.5V
U <sub>MinOff</sub>	8.4V +/- 0.5V
U <sub>12VOFF</sub>	10.8V +/- 0.5V
U <sub>12VOn</sub>	11.2V +/- 0.5V
U <sub>24VOFF</sub>	18.5V +/- 0.5V
U <sub>24VOn</sub>	19V +/- 0.5V
U <sub>MaxOff</sub>	32.5V +/- 0.5V
U <sub>MaxOn</sub>	32V +/- 0.5V
U <sub>minCAN</sub>	8V +/- 0.5V
U <sub>maxCAN</sub>	35V +/- 0.5V

### 5.10.2. High Temperature function state Diagram

PRE



PRELIMINAR

## 6. Communication

### 6.1. CAN J1939 Communication

#### 6.1.1. System operation

##### 6.1.1.1. Start-up sequence

At power up the ACU initialize communication modules and sends firstly, after a maximum delay of 500ms, an address claimed and analyses network traffic for 250ms. As soon as the ACU may begin regular network communication it sends a DM1 message or an estimated flow message in case of no active error.

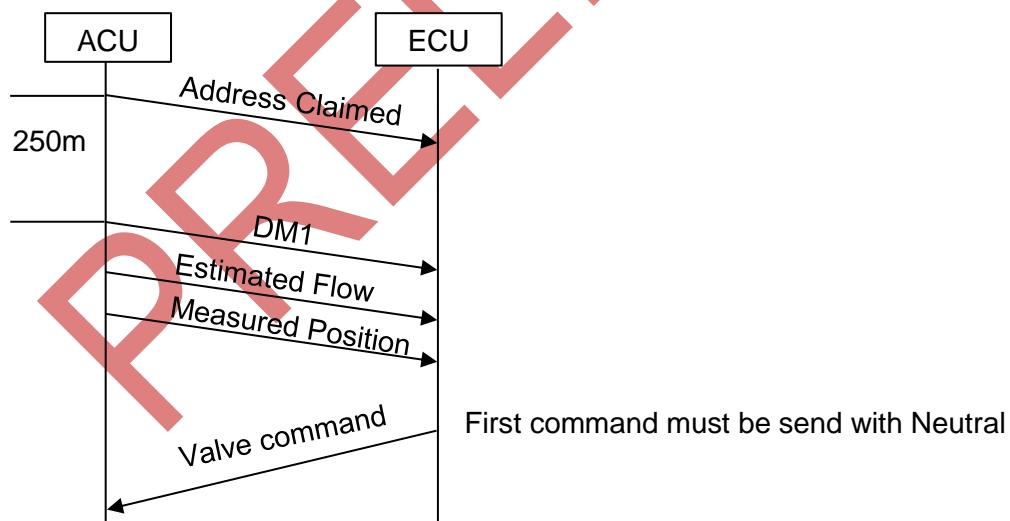
The FLASH checksum will be calculated at power up before sending the address claim. If the checksum is not correct it means that memory is corrupted or that the last bootload operation failed (application not correctly loaded in memory).

The first position command must be Neutral (Valve State = 0) otherwise it will result in sending a DM1 with the fault code 13 "Inconsistent CAN Control message".

The CAN timeout is activated with the first received position command.

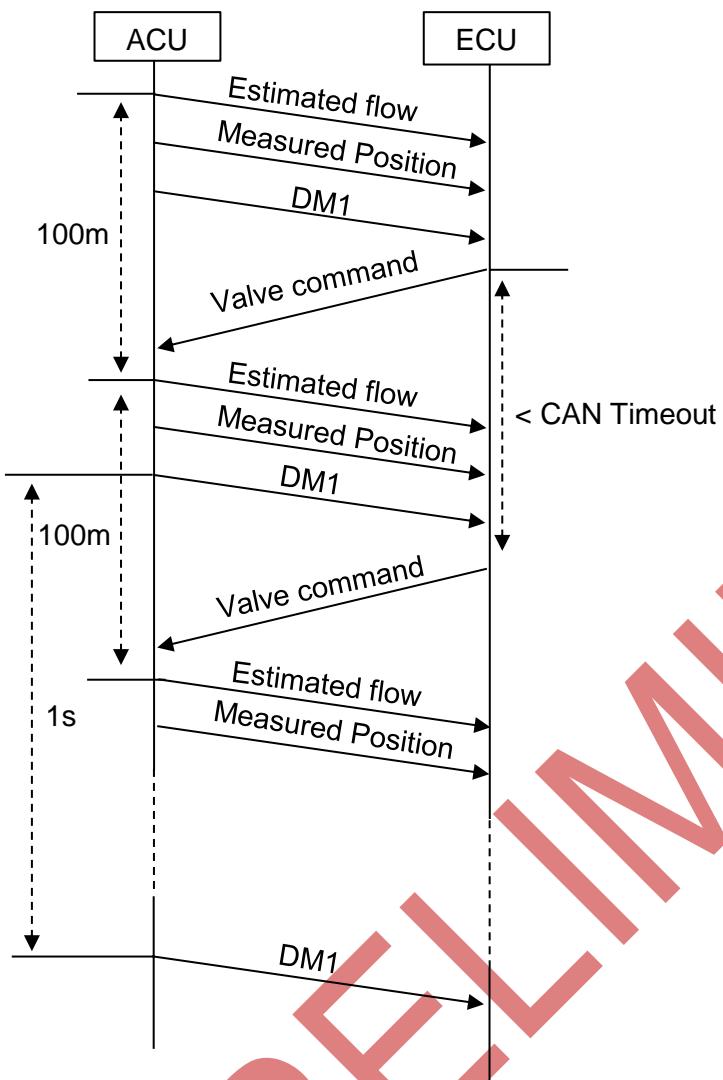
If at start-up of the actuator control algorithm the supply voltage has not reached its nominal level, an under-voltage error will be generated.

##### Typical sequence :



### 6.1.1.2. Normal operation

**Typical sequence :**



See Fault code table when an errors appears.

### 6.1.1.3. Power-down sequence

No particular command must be sent before power-off.

Depending on the voltage ramp, under-voltage error can be send on the last DM1 message (See Fault code table).

## 6.1.2. Special Parameter groups (data link layer)

### 6.1.2.1. Request

This message provides the capability to request information globally or to a specific destination (destination specific request).

The response to such a request can be:

- A CAN frame with maximum 8 bytes or a multi-packet data transfer (depends on the PGN requested)
- A NACK (see acknowledge message) which means "negative acknowledgement" if the requested module is not able to send the requested information (PGN not supported).
- An ACK which means "acknowledgement" if the requested module supports the PGN but doesn't need to send back a specific message.

The default delay from the answer is defined to 100ms.

#### Request message:

Message ID	18 EA "DA" "SA"
Default Priority	6
PGN	00EA00 (59904)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer (could become broadcast if DA = 255)
MP (multipacket)	NO
Message direction	Receive (ECU to ACU).
Message length	3 bytes
Repetition rate	N/A
Timeout	N/A



**Bytes 1, 2, 3:** Parameter group number being requested.

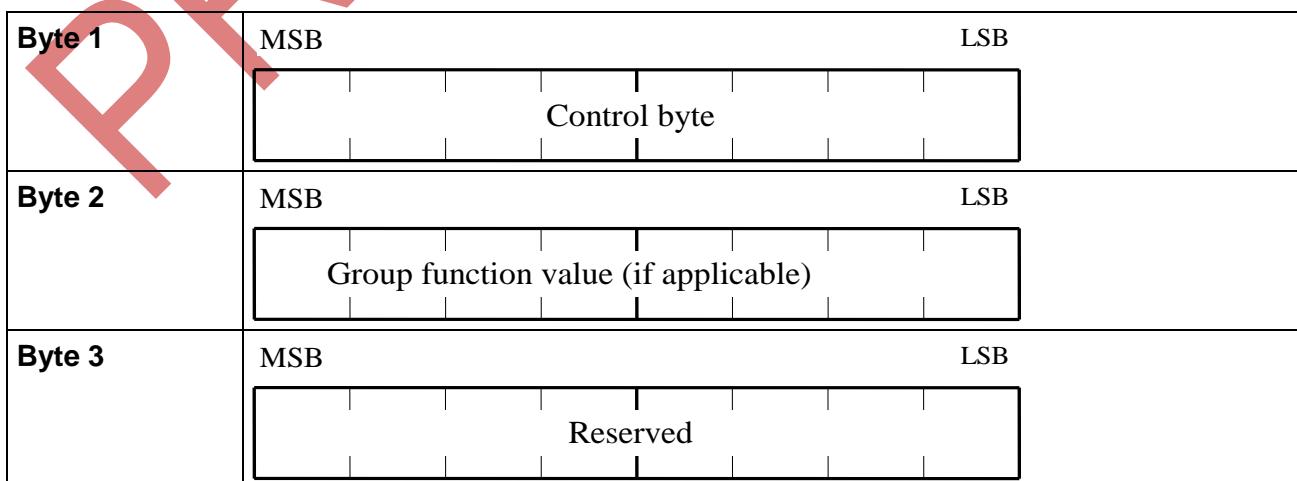
The next table summarizes the suitable response types to a request:

Request Message DA	Data byte 1..3 (PGN requested)	CA Response type	CA Response DA (PDU1 only)	TP response type
Specific DA (0...253)	PGN requested <b>not supported</b> by the CA	NACK	Global DA (DA = 255)	NA
Global DA (255)	PGN requested <b>not supported</b> by the CA	None	NA	NA
Specific DA (0...253)	PGN requested <b>supported</b> by the CA	Message requested	Specific DA	RTS/CTS
		ACK	Global DA (DA = 255)	NA
Global DA (255)	PGN requested <b>supported</b> by the CA	Message requested	Global DA (DA = 255)	BAM

Note that the CA doesn't support the "Request 2" message (PGN 51456) because it doesn't need to report data about more than one device. The "transfer" message (PGN 51712) is not supported too because it is a respond to a request2 message.

### **6.1.2.2. Acknowledgement**

Message ID	18 E8 FF "SA"
Default Priority	6
PGN	00E800 (59392)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer (but becomes broadcast with DA = 255)
MP (multi-packet)	NO
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU).
Data length	8 bytes
Repetition rate	N/A
Timeout	N/A



<b>Byte 4</b>	MSB	LSB
Reserved		
<b>Byte 5</b>	MSB	LSB
Address		
<b>Byte 6</b>	MSB	LSB
PGN of requested info (LSB)		
<b>Byte 7</b>	MSB	LSB
PGN of requested info		
<b>Byte 8</b>	MSB	LSB
PGN of requested info (MSB)		

**Byte 1:**

Control byte	Acknowledge type
0	Positive acknowledgement (ACK). PGN supported
1	Negative acknowledgement (NACK). PGN not supported
2	Access denied (PGN supported, but security access denied)
3	Can't respond. PGN supported but ECU (or ACU) is busy.

**Byte 2:** This field is not coded and is always filled with 0xFF.

**Byte 3-4:** Reserved: filled with 0xFF

**Byte 5:** Source address (ACU AD)

**Byte 6-8:** PGN value not applicable: filled with 0xFF

The CA will send an acknowledgement in the cases of such a response is required (specified in J1939-21). The ACU will transmit a NACK (Control byte = 1) if the PGN of a message is not supported (for example DM13 Stop/Start Broadcast message) or if the PGN specified in a request message is not identified. The ACU will transmit an ACK (control byte = 0) when a PGN is supported and no existing message can't be sent back. In our application the use of Acknowledge message will be restricted as much as possible. The CA will never send an acknowledgement message with the control bit set to 2. If the PGN is supported, the CA will always send back the requested message if available. However, if the CA is already busy by answering with a message using the transport protocol and a new request arrives, it will send an ACK with the Control byte set to 3 informing that the PGN is supported but ACU is busy. The CA behave like a slave in the network and will ignore acknowledgements received.

A NACK will be sent in the next situation:

- Response to a specific request (not global, DA = 0...253) message when the PGN is not supported.
- Response to a specific command (not global, DA = 0...253) when the PGN is not supported (both with and without TP use)

An ACK will be sent in the next situation:

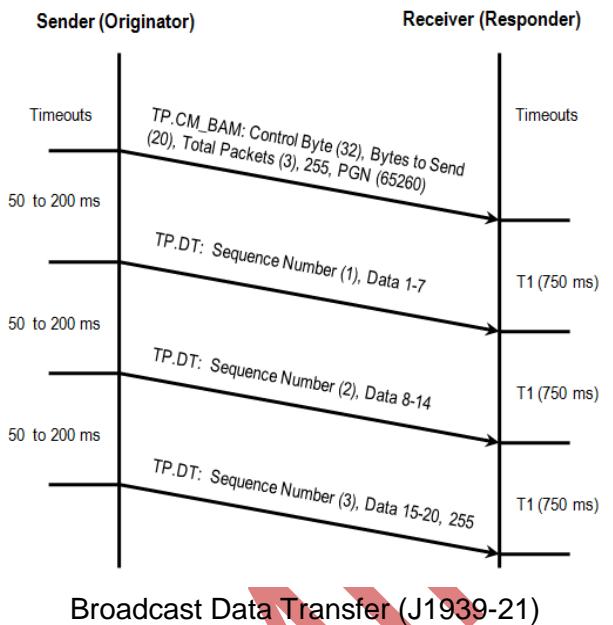
- Acknowledgement of a few proprietary commands

**PRELIMINARY**

### 6.1.3. Transport protocol (Data link layer)

#### 6.1.3.1. Broadcast Transport Protocol

The ACU typical time between 2 transmissions is between 50 and 100ms.



##### 6.1.3.1.1. Connection Initialisation

###### 6.1.3.1.1.1. TP.CM\_BAM – Transport Protocol. Connexion Management Broadcast Announce Message

Message ID	1C EC "DA" "SA"
Default Priority	7
PGN	00EC00 (60416)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast with DA = 255
MP (multipacket)	NO
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU).
Message length	8 bytes
Repetition rate	-
Timeout	As defined into the J1939-21

<b>Byte 1</b>	MSB	LSB
Control byte		
<b>Byte 2</b>	MSB	LSB
Total message size LSB		
<b>Byte 3</b>	MSB	LSB
Total message size MSB		
<b>Byte 4</b>	MSB	LSB
Number of packets		
<b>Byte 5</b>	MSB	LSB
Reserved		
<b>Byte 6</b>	MSB	LSB
PGN of the packeted message		
<b>Byte 7</b>	MSB	LSB
PGN of the packeted message		
<b>Byte 8</b>	MSB	LSB
PGN of the packeted message		

#### Specific bytes content for a BAM message: Only BAM is supported

Byte number	Value	Allowed values
1	Control byte	0x20 (BAM)
2,3	Total message size, number of bytes. Byte 2 = LSB	9 to 1785
4	Number of packets	2 to 255
5	Reserved	0xFF
6,7,8	Parameter group number of the packeted message. Byte 6 = LSB	PGN of supported messages > 8 bytes

### 6.1.3.1.2. Data transfer

#### 6.1.3.1.2.1. TP.DT – Transport Protocol.Data Transfer

Message ID	1C EB "DA" "SA"
Default Priority	7
PGN	00EB00 (60160)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast with DA = 255
MP (multipacket)	NA
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU).
Message length	8 bytes
Repetition rate	-
Timeout	As defined into the J1939-21



Byte 2-8: Byte 2 = LSB

#### 6.1.3.2. Peer to peer transport protocol – Not supported

## 6.1.4. Network management

### 6.1.4.1. Address configuration

The ACU will use the "Command configurable address" procedure in the Single Address capable capabilities group. As the ACU will use the "Commanded address" message, it can be classified as well in the "Service configurable address CA".

The ACU default address is not included in the range 0-127 or 248-253 and must therefore wait for 250ms (even if the address is changed after delivery by the user) before starting to communicate.

### 6.1.4.2. Default addresses

By default the ACU industry group is "agricultural and forestry" (industry group number 2).

The ACU source address is set by default to 128 (0x80).

The ACU will use by default the destination address 34 (0x22). Note that the DA will not be filtered and the ACU will accept commands from any destination address.

### 6.1.4.3. CA NAME

<b>Byte 1</b>	MSB	LSB
Identity Number 8 ... 1		
<b>Byte 2</b>	MSB	LSB
Identity Number 16... 9		
<b>Byte 3</b>	MSB	LSB
Manufacturer Code 3 ... 1      Identify Number 21 ... 17		
<b>Byte 4</b>	MSB	LSB
Manufacturer Code 11...4		
<b>Byte 5</b>	MSB	LSB
Function Instance      ECU Instance		

<b>Byte 6</b>	MSB	LSB
Function		
<b>Byte 7</b>	MSB	LSB
	Vehicle system	Reserved
<b>Byte 8</b>	MSB	LSB
	Arbitrary Address Capable	Industry Group
		Vehicle system Instance

**Identity number:** Assigned by the manufacturer (SBZ), this 21 bits field is unique. It is the serial manufacturing number from 0 to 2'097'152.

**Manufacturer code:** Manufacturer code 327d (0x147) has been assigned by the SAE to SBZ products

**Function instance:** By default, the value is set to 0

**ECU instance:** By default, the value is set to 0

**Function:** The default ACU function is 129d (0x81 - auxiliary valves control).

**Vehicle system:** The vehicle system for tractor auxiliary valves control is 1 (0x1).

**Arbitrary address capable:** The ACU is not "arbitrary address capable". This bit is set to 0.

**Industry group:** By default the ACU group is "agricultural and forestry" (industry group n°2). This bit is set to 2d (0x02).

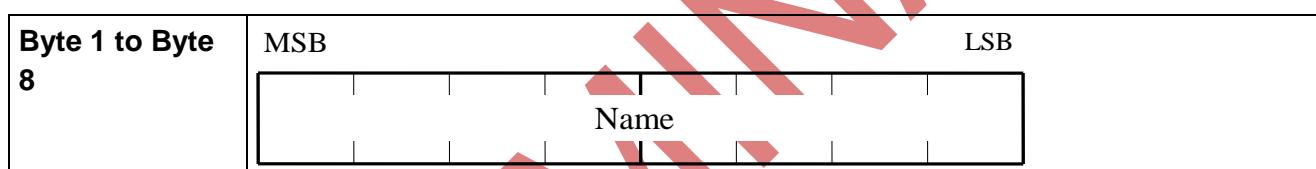
**Vehicle system instance:** By default the vehicle system instance is set to 0.

The default ACU NAME with an identity number set to 0 is: **0x20 02 81 00 28 E0 00 00**

#### 6.1.4.4. Network management procedures

##### 6.1.4.4.1. Address claimed

Message ID	18 EE FF "SA"
Default Priority	6
PGN	00EE00 (60928)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer (but becomes broadcast with DA = 0xFF)
MP (multipacket)	NO
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU)
Message length	8 bytes
Repetition rate	Sent at power on, after a "Request" for Address Claim and after a "Commanded Address"
Timeout	N/A



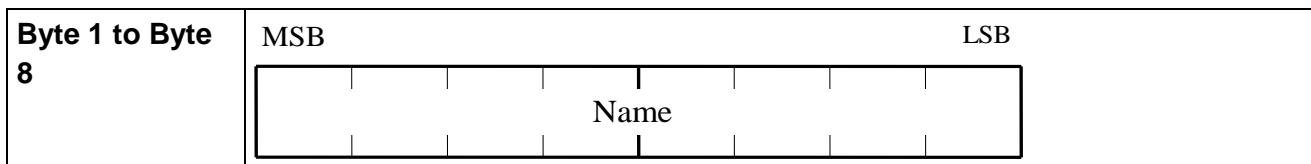
**Byte 1...8: NAME**

Example:

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x18EEFF81	0x00	0x00	0xE0	0x20	0x08	0x81	0x02	0x20	Address Claimed

##### 6.1.4.4.2. Cannot claim address

Message ID	18 EE FF FE
Default Priority	6
PGN	00EE00 (60928)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer (but becomes broadcast with DA = 0xFF)
MP (multipacket)	NO
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU)
Message length	8 bytes.
Repetition rate	N/A. Sent if address was not successfully claimed
Timeout	N/A



### Byte 1...8: NAME

Example:

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x18EEFFFE	0x00	0x00	0xE0	0x20	0x08	0x81	0x02	0x20	Cannot claim address

#### 6.1.4.4.3. Request for address claimed

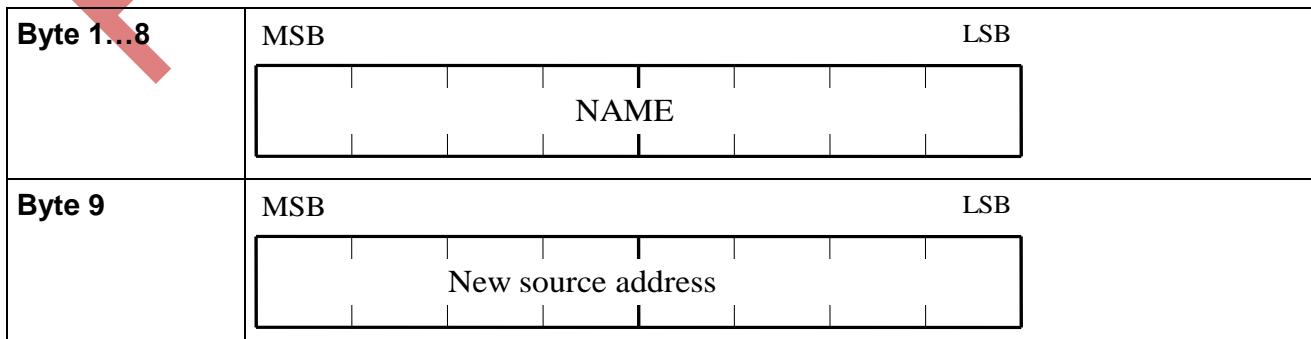
The ACU will only receive "request for address claimed" message and never transmit it.

Request message data: PGN of Address claimed 60928 (3 data). See "Request" message description at chapter 6.1.2.1, "Request".

#### 6.1.4.4.4. Commanded address

The ACU will support this message to change its source address.

PGN	00FED8 (65240)
(Default Priority)	(6)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	YES
Message direction	Receive (ECU to ACU).
Message length	9 bytes. → Uses BAM TP
Repetition rate	As required
Timeout	N/A



**Byte 1...8:** NAME**Byte 9:** Source address.

Example:

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x1CECFF22	0x20	0x09	0x00	0x02	0xFF	0xD8	0xFE	0x00	TP.CM: BAM, 9 bytes, 2 packets, PGN Commanded Address
0x1CEBFF22	0x01	0x00	0x00	0xE0	0x28	0x00	0x81	0x02	TP.DT: packet 1, Name MSB
0x1CEBFF22	0x02	0x20	0x81	0xFF	0xFF	0xFF	0xFF	0xFF	TP.DT: packet 2, Name LSB

PRELIMINARY

## 6.1.5. Diagnostics

### 6.1.5.1. DM1 - Active diagnostic trouble codes

Message ID	18 FE CA "SA"
Default Priority	6
PGN	00FECA (65226)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	NO if only one DTC is active YES when more than one DTC is active (BAM)
Message direction	Transmit (ACU to ECU).
Message length	8 bytes, uses TP when more than one previous DTC is active
Repetition rate	<ul style="list-style-type: none"> <li>- 1 per second</li> <li>- On change (DTC becomes active or inactive)</li> <li>- On request (using request message PGN 59904)</li> <li>- If more than one error is active and the TP is already in use, the DM1 will be sent as soon the TP is available again.</li> <li>- During initialisation, eventual sending of a DM1 message will be delayed until the end of the dead zone detection procedure.</li> </ul>
Timeout	N/A

PRELIM

<b>Byte 1</b>	MSB				LSB					
SAE Lamp Status	Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp						
<b>Byte 2</b>	MSB				LSB					
SAE Flash Lamp	Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp						
<b>Byte 3</b>	MSB				LSB					
SPN	SPN 1 ... 8									
<b>Byte 4</b>	MSB				LSB					
SPN	SPN 9 ... 16									
<b>Byte 5</b>	MSB				LSB					
SPN / FMI	SPN 17 ... 19		FMI							
<b>Byte 6</b>	MSB				LSB					
Reserved	0	Occurrence count								
<b>Byte 7</b>	MSB				LSB					
Reserved	1	1	1	1	1	1	1	1		
<b>Byte 8</b>	MSB				LSB					
Reserved	1	1	1	1	1	1	1	1		

- Byte 1 & 2 :** For every 2 bits lamp : 00 Lamp OFF, 01 Lamp ON, 10 Reserved, 11 Reserved
- Byte 3 & 4:** Suspect Parameter Number 1..16 (LSB)
- Byte 5, Bit 6..8:** Suspect Parameter Number 17..19 (MSB)
- Byte 5, bit 1..5:** Failure Mode Identifier
- Byte 6, bit 8:** SPN conversion method (Always 0)
- Byte 6, bit 1..7:** The occurrence count is not managed in the DM1 message and is set to 127d for all SPN.
- Byte 7 & 8:** Filler bytes (0xFF)

### Message format:

A = Lamp Status

B = SPN

C = FMI

D = Occurrence Counter

#### 1 DTC Active (no TP):

Message form will be as follows: A,B,C,D

#### More than 1 DTC (TP used):

Message form will be as follows: A,B,C,D,B,C,D,B,C,D,... etc

### Examples (With DA → 98h):

#### No DTC active

0x18FECA98      00 FF 00 00 00 00 FF FF      (bytes 7 and 8 → filler bytes)

#### 1 DTC Active:

0x18FECA98      00 FF 9E 00 10 7F FF FF      (bytes 7 and 8 → filler bytes)

#### 2 DTC active (using Transport Protocol)

0x1CECFF98      20 0A 00 02 FF CA FE 00  
0x1CEBFF98      01 00 FF 3F F0 F3 7F 9E  
0x1CEBFF98      02 00 10 7F FF FF FF FF      (on this example, bytes 5 to 8 → filler bytes)

Lamp Status  
DTC 1  
DTC 2



## Fault code table:

SPN	BYTE 1 SAE Lamp Status	BYTE 2 SAE Flash Lamp	FMI	SBZ fault code	Fault	ACU Action	ACU Restart
00h	00h	FFh	00h	00h	Normal operation		
520255d (7 F0 3Fh) CAN message	00h	FFh	19 (13h)	13d	Inconsistent CAN Control message	ACU drives spool back to neutral position and waits for valid message	New valid CAN Control message <sup>1</sup>
			9 (09h)	19d	No CAN Control message received = TIME OUT		
628d (02 74 h) FLASH / EEPROM inconsistent	00h	FFh	2 (02h)	44d	EEPROM parameter write/read error	ACU turns off motor current and the central spring returns the spool to neutral position	No possible restart since Flash/EEPROM data are not reliable anymore. In case of temporary error, it could be cleared after a power up sequence
520256d (7 F0 40h) Output stage / Force	00h	FFh	0 (00h)	47d	Stepper motor stall detected.	ACU turns off motor current and the central spring returns the spool to neutral position <sup>3</sup>	New valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
158d (00 9E h) Electrical failures	00h	FFh	18 (12h)	31d	Undervoltage SONCEBOZ value ( $VDC < U_{MINOFF}$ ) during time more than 3 ms	ACU turns off motor current and the central spring returns the spool to neutral position	$VDC > U_{MINON}$ , new valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
			16 (10h)	32d	Ovvoltage SONCEBOZ value ( $VDC > U_{MAXOFF}$ ) during time more than 3 ms		$VDC < U_{MAXON}$ , new valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
520258d (7 F0 42h) Temperature	00h	FFh	0 (00h)	29d	SCU temperature higher than $T_{MAXOFF}$ during time more than 2 sec	ACU turns off motor current and the central spring returns the spool to neutral position	$SCU \text{ temp} < T_{MaxOn}$ , new valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
			1 (01h)	26d	SCU temperature below $T_{MINOFF}$ during time more than 2 sec		$SCU \text{ temp} > T_{MinOn}$ , new valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
520260d (7 F0 44h) Current	00h	FFh	0 (00h)	43d	Current exceeds upper limit $I_{MAX}$ during time more than 1 ms	ACU turns off motor current and the central spring returns the spool to neutral position	New valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
520259d (7 F0 43h) Error position	00h	FFh	2 (02h)	36d	Error position in IDLE Mode	ACU turns off motor current and the central spring returns the spool to neutral position	New valid CAN Control message <sup>1</sup> , reinitialisation after stabilisation time <sup>2</sup>
630d (02 76 h) Linearization parameter Error	00h	FFh	2 (02h)	45d	Range error for Linearization parameter in EEPROM	ACU turns off motor current and the central spring returns the spool to neutral position	New valid range in EEPROM1, reinitialisation after stabilisation time2



**1** ECU must be commanded to Neutral (Valve State = 0) in order to accept a new position command after error. Once the error is cleared, if the first command is not “Neutral” it will result in sending a DM1 with the fault code 13 “Inconsistent CAN Control message”

**2** Stabilisation time when Temperature range 1 =~ 5000ms, else ~800ms

**3** The stall detection algorithm was successfully validated on a hydraulic valve without oil pressure. It may not detect 100% of the step losses in the final application. In practice, within or close to the deadband, a step loss is not detected in every case.

**4** ACU turns off motor current immediately when he detects the undervoltage but the DM1 message is sent only after DM1 Undervoltage reaction delay if the undervoltage lasted at least DM1 Undervoltage reaction delay.

#### FMI definitions:

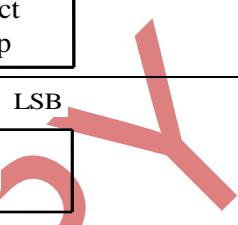
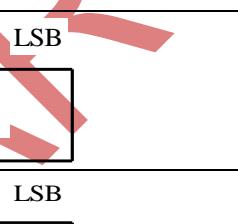
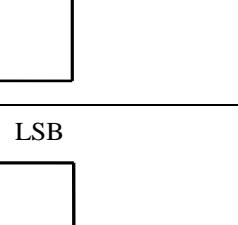
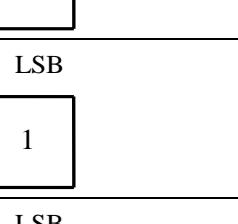
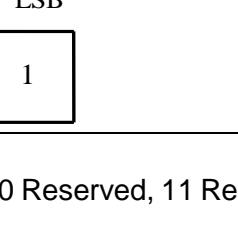
## SAE J1939 FMIs

- **FMI=0:** Data Valid but Above Normal Operation Range - Most Severe Level
- **FMI=1:** Data Valid but Below Normal Operation Range - Most Severe Level
- **FMI=2:** Data Erratic, Intermittent or Incorrect
- **FMI=3:** Voltage Above Normal or Shorted to High Source
- **FMI=4:** Voltage Below Normal or Shorted to Low Source
- **FMI=5:** Current Below Normal or Open Circuit
- **FMI=6:** Current Below Normal or Grounded Circuit
- **FMI=7:** Mechanical System not Responding or Out of Adjustment
- **FMI=8:** Abnormal Frequency or Pulse Width or Period
- **FMI=9:** Abnormal Update Rate
- **FMI=10:** Abnormal Rate of Change
- **FMI=11:** Root Cause Not Known
- **FMI=12:** Bad Intelligent Device or Component
- **FMI=13:** Out of Calibration
- **FMI=14:** Special Instructions
- **FMI=15:** Data Valid but Above Normal Operation Range - Least Severe Level
- **FMI=16:** Data Valid but Above Normal Operation Range - Moderately Severe Level
- **FMI=17:** Data Valid but Below Normal Operation Range - Least Severe Level
- **FMI=18:** Data Valid but Below Normal Operation Range - Moderately Severe Level
- **FMI=19:** Received Network Data In Error
- **FMI=31:** Not Available or Condition reported by the SPN exists

### 6.1.5.2. DM2 - Previously active diagnostic trouble codes

Message ID	18 FE CB "SA"
Default Priority	6
PGN	00FECB (65227)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	NO if only one previous DTC is active YES when more than one previous DTC is active (BAM)
Message direction	Transmit (ACU to ECU).
Message length	8 bytes (uses TP when more than one previous DTC is active)
Repetition rate	On request (using request message PGN 59904)
Timeout	N/A

PRELIMINAR

Byte 1	MSB	LSB								
SAE Flash Lamp	<table border="1"><tr><td>Malfunction Indicator</td><td>Red Stop</td><td>Amber Warning</td><td>Protect Lamp</td></tr></table>	Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp					
Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp							
Byte 2	MSB	LSB								
SAE Flash Lamp	<table border="1"><tr><td>Malfunction Indicator</td><td>Red Stop</td><td>Amber Warning</td><td>Protect Lamp</td></tr></table>	Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp					
Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp							
Byte 3	MSB	LSB								
SPN	<table border="1"><tr><td>SPN 1 ... 8</td></tr></table>	SPN 1 ... 8								
SPN 1 ... 8										
Byte 4	MSB	LSB								
SPN	<table border="1"><tr><td>SPN 9 ... 16</td></tr></table>	SPN 9 ... 16								
SPN 9 ... 16										
Byte 5	MSB	LSB								
SPN / FMI	<table border="1"><tr><td>SPN 17 ... 19</td><td>FMI</td></tr></table>	SPN 17 ... 19	FMI							
SPN 17 ... 19	FMI									
Byte 6	MSB	LSB								
Reserved	<table border="1"><tr><td>0</td><td>Occurrence count</td></tr></table>	0	Occurrence count							
0	Occurrence count									
Byte 7	MSB	LSB								
Reserved	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1			
Byte 8	MSB	LSB								
Reserved	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1			

~~PPR~~ Byte 1 & 2 :

For every 2 bits lamp : 00 Lamp OFF, 01 Lamp ON, 10 Reserved, 11 Reserved  
Suspect Parameter Number 1..16 (LSB)

Byte 3 & 4:

Suspect Parameter Number 17..19 (MSB)

Byte 5, Bit 6..8:

Failure Mode Identifier

Byte 5, bit 1..5:

SPN conversion method (Always 0)

Byte 6, bit 8:

Occurrence count

Byte 6, bit 1..7:

Filler bytes (0xFF)

Byte 7 & 8 :

### Message format:

A = Lamp Status  
B = SPN  
C = FMI  
D = Occurrence Counter

#### 1 DTC Active (no TP):

Message form will be as follows: A,B,C,D

#### More than 1 DTC (TP used):

Message form will be as follows: A,B,C,D,B,C,D,B,C,D,B,C,D,... etc

### Examples (With DA → 98h):

#### No DTC active

0x18FECB98      00 FF 00 00 00 00 FF FF      (bytes 7 and 8 → filler bytes)

#### 1 DTC Active:

0x18FECB98      00 FF 9E 00 10 01 FF FF      (bytes 7 and 8 → filler bytes)

#### 2 DTC active (using Transport Protocol)

0x1CECFF98	20 0A 00 02 FF CB FE 00
0x1CEBFF98	01 00 FF 3F F0 F3 01 9E
0x1CEBFF98	02 00 10 02 FF FF FF FF

(on this example, bytes 5 to 8 → filler bytes)

Lamp Status  
DTC 1  
DTC 2

### 6.1.5.3. DM3 – Diagnostic data clear / reset of previously active DTC

Request with PGN:	00FECC (65228)
-------------------	----------------

All the diagnostic information pertaining to the previously active trouble codes should be erased when this PG (65228) is requested (only DM2 for the ACU). The diagnostic tool shall send the request PGN 59904 directed specifically to our motor controller with the PGN 65228 as the requested PGN. The diagnostic data associated with active trouble code will not be affected. Our actuator controller shall respond with the Acknowledgement (PGN 59392) indicating that the action was successfully completed. No positive or negative acknowledgement is sent when the request was sent to the global address.

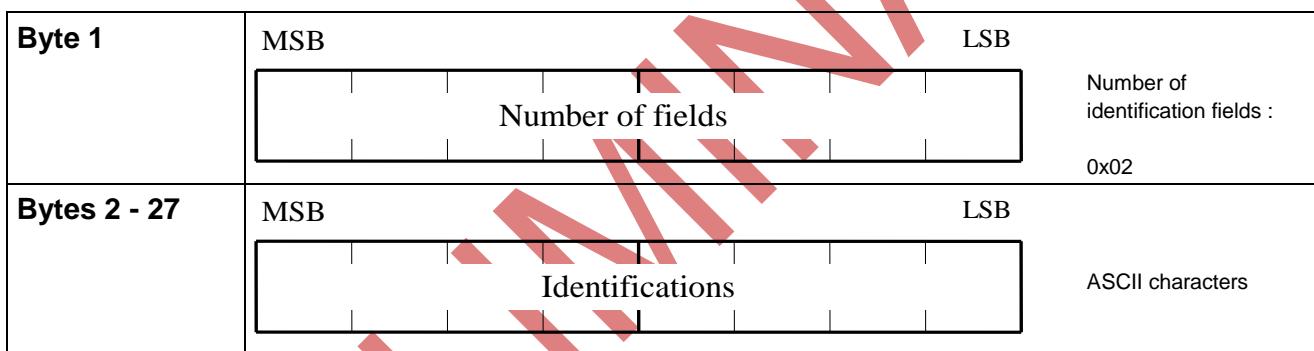
The actuator will clear the DTCs and send a positive acknowledge to this request message only when no error is active.

### 6.1.5.4. DM11 – Diagnostic data Clear/Reset for Active DTCs – Not supported

## 6.1.6. Identification messages

### 6.1.6.1. Software identification

Message ID	18 FE DA "SA"
Default Priority	6
PGN	00FEDA (65242)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	YES (BAM)
Message direction	Transmit (ACU to ECU).
Message length	27 bytes (specific to ACU)
Repetition rate	On request (using request message PGN 59904)
Timeout	N/A



Byte 1: Number of software identification fields  
Byte 2-27: Software identifications

Note: An ASCII "\*" is used as a delimiter to separate software identifications. An ASCII "\*" is required at the end of the last software identification field.

ACU software ID codes :  
Byte 2 – 13: Bios software ID (SBZ definition)  
Byte 14: Separator  
Byte 15- 26: Bootloader software ID (SBZ definition)  
Byte 27: Separator (End of software ID)

Example of SBZ software identification coding fields:

Byte nbr	Char	ASCII	Byte nbr	Char	ASCII
2	8	0x38	15	8	0x38
3	5	0x35	16	5	0x35
4	6	0x36	17	6	0x36
5	5	0x35	18	5	0x35
6	S	0x53	19	S	0x53
7	0	0x30	20	9	0x39
8	0	0x30	21	9	0x39
9	0	0x30	22	9	0x39
10	.	0x2E	23	.	0x2E
11	0	0x30	24	0	0x30
12	0	0x30	25	0	0x30
13	0	0x30	26	0	0x30
14	*	0x2A	27	*	0x2A

PRELIMINARY

### 6.1.6.2. ECU identification

Message ID	18 FD C5 "SA"
Default Priority	6
PGN	00FDC5 (64965)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	YES (BAM)
Message direction	Transmit (ACU to ECU).
Message length	32 bytes (specific to ACU)
Repetition rate	On request (using request message PGN 59904)
Timeout	N/A

Field	SPN	Byte number
ECU part number	2901	1 to 13
ASCII "*" (0x2A)	-	14
ECU serial number	2902	15 to 26
ASCII "*" (0x2A)		27
ECU location	2903	28,29
ASCII "*" (0x2A)		30
ECU type	2904	31
ASCII "*" (0x2A)		32

Note: An ASCII "\*" is used as a delimiter to separate the different ECU identification fields. An ASCII "\*" is required at the end of the last ECU identification field.

#### ECU part number

Composed of SBZ part number (9 bytes) and part number revision (4 bytes)

Refer to the notes 6 (SBZ part number) and 4 (part number revision) of the chapter §6.1.7.1 (System parameters), except that the byte numbers are 1 to 9 and 10 to 13

#### ECU serial number

Refer to the note 7 of the chapter §6.1.7.1 (System parameters), except that the byte numbers are 15 to 26

#### ECU location

Refer to the note 8 of the chapter §6.1.7.1 (System parameters), except that the byte numbers are 28 to 29

#### ECU type

Refer to the note 9 of the chapter §6.1.7.1 (System parameters), except that the byte number is 31

## 6.1.7. Memory access

### 6.1.7.1. System parameters

The parameters can be read and modified by accessing the memory (with DM14 (Pointer Extension = 1), DM15, DM16 messages). The next memory map defines these parameters and their associated address: All these parameters will be stored into the EEPROM. If parameters are set to other values than the permitted ones, this may lead to non-specified behaviour of the Smart actuator.

Address [DEC]	Data length [bytes].	Definition	CUS/SBZ Note 1	Permitted WR/RD data range. Note 2	Default value
4	12	Bios software ID (note 5)	SBZ	ASCII chars	-
25	9	ECU part number (note 6)	SBZ	ASCII chars	-
61	2	Identity number 16 bits LSB	SBZ	note 3	note 3
63	2	Identity number 5 bits MSB + Manufacturer code 3 bits LSB + Manufacturer code 8 bits MSB	SBZ		
65	2	ECU instance (3 bits) + Function instance / ECU location (5 bits) + Function (8 bits)	SBZ		
67	4	ECU Part Number Revision (note 4)	SBZ	ASCII chars	-
71	12	ECU serial number (note 7)	SBZ	ASCII chars	-
83	1	ECU type (note 9)	CUS	ASCII Char "A" or "G"	"A"
84	2	ECU location (Note 8)	SBZ	ASCII Chars "0"0"...1"5"	ASCII :chars"0""0"
105	1	Source address *	CUS	0x80 ...0x8F	0x80
106	1	Destination address	CUS	0x00 ... 0xFD	0x22
107	2	Reserved (1 bit) + Vehicle system (7 bits) + Vehicle system instance (4 bits) + Industry group (3bits) + Arbitrary address capable (1bit)	CUS (excepted that write to arbitrary address capable bit will be ignored)	Note 3	Note 3

\* Source address is not writable through the memory access. This parameter could only be changed with the "Commanded address" command.



Address [DEC]	Data length [bytes].	Definition	CUS/SBZ Note 1	Permitted WR/RD data range. Note 2	Default value
109	1	CAN Timeout (note 10)	CUS	0-100d => (10ms - 1s)	100d (1s)
111	2	Bias value pos : Xoffset + (note 10)	CUS	500d	0
113	2	Bias value neg : Xoffset - (note 10)	CUS	500d	0
115	2	Max position : Xmax + (note 10)	CUS	500...1250d	700d
117	2	Min position : Xmax – (note 10)	CUS	500...1250d	700d
119	2	Float position : Xfloat (note 10)	CUS	0...1250d	1200d
121	1	Float direction : Xfloat DIR (note 10)	CUS	0,1	0
122	4	CAN baudrate : CANBTC (note 10)	CUS	0x000E01FA (100kBaud) 0x000B01FA (125kBaud) 0x000501FA (250kBaud) 0x000501B1 (500kBaud)	0x000501FA (250kBaud)
126	1	Position curve : (note 10) Direct or programmable	CUS	0x00: Direct (position) 0x01: Programmable	0x00 : Direct
127	5*1	User Curve Extend CAN Percent	CUS	0-150	0
132	5*1	User Curve Extend QValve	CUS	0-150	0
137	5*1	User Curve Retract CAN Percent	CUS	0-150	0
142	5*1	User Curve Retract QValve	CUS	0-150	0
147	2	User Curve Extend QMax	CUS	0-1500	0
149	2	User Curve Retract QMax	CUS	0-1500	0
151	8*2	Linear Curve Extend QValve Scaled	CUS	0-1500	0
167	8*1	Linear Curve Extend Spool Pos	CUS	0-255	0
175	8*2	Linear Curve Retract QValve Scaled	CUS	0-1500	0
191	8*1	Linear Curve Retract Spool Pos	CUS	0-255	0
199	2	Boot Seed	CUS	0x02...0xFFFF	0x05
201	2	Boot Key	CUS	0x01...0xFFFF	0x7B5B
642	1	Sensor Enable Functionality	CUS	0-1	1 (Enable)
643	1	Sensor Zero Reference Valid	CUS	0 (Invalid)	0 (Invalid)
644	1	Measured Position Status Transmission	CUS	0-1	1 (Enable)
645	1	Sensor Neutral Fault Recovery Timeout	CUS	0 – 250d => (0s - 25s)	20d (2s)

Address [DEC]	Data length [bytes].	Definition	CUS/SBZ Note 1	Permitted WR/RD data range. Note 2	Default value
646	1	Sensor Stall Neutral Enable	CUS	0-1	1 (Enable)
647	1	Sensor Stall Normal Enable	CUS	0-1	1 (Enable)
648	2	Sensor Threshold Offset	CUS	120 - 1000d	120d
650	1	Sensor Prot React Time 10mm/s	CUS	150 - 250d => (1.5s – 2.5s)	150d (1.5s)
651	1	Sensor Prot React Time 20mm/s	CUS	85 - 250d => (0.850s – 2.5s)	85d (850ms)
652	1	Sensor Prot React Time 40mm/s	CUS	45 - 250d => (0.45s – 2.5s)	45d (450ms)
653	1	Sensor Prot React Time 60mm/s	CUS	35 - 250d => (0.35s – 2.5s)	35d (350ms)
654	1	Sensor Prot React Time 80mm/s	CUS	25 - 250d => (0.25s – 2.5s)	25d (250ms)
655	1	Sensor Prot React Time 100mm/s	CUS	25 - 250d => (0.25s – 2.5s)	25d (250ms)
667	1	ECU Address Filter	CUS	0-1	0 (Disable)
668	1	DM1 Undervoltage reaction delay	CUS	0 – 250d => (0s - 25s)	0
688	1	Boost Mode Enable	CUS	0-1	0 (Disable)
689	2	Xboost Extend Min	CUS	0...2000d	0
691	2	Xboost Extend Max	CUS	0...2000d	700d
693	2	Xboost Retract Min	CUS	0...2000d	0
695	2	Xboost Retract Max	CUS	0...2000d	700d
697	1	Xboost Value	CUS	0-100d	30d
712	1	Normal Speed	CUS	10-100d	100d
716	1	Float Speed	CUS	10-100d	60d
718	1	Low Speed 1	CUS	10-60d	60d
719	1	Low Voltage Speed	CUS	10-40d	40d
736	1	Stall Detection Back-EMF Enable	CUS	0-1	1 (Enable)

For parameters with more than 1 Byte, LSB is the first address.

**Note 1:**

CUS : Customer and SBZ write permitted.

SBZ : SBZ write permitted only

**Note 2:**

Parameters values can be written only if they are in the permitted range.

The parameter “Source Address” is read only. To change this value, use the “Commanded Address” command.

**Note 3:**

NAME parameter	Possible read/write range [dec]	Default Value [dec]
Identity number	0...2'097151 (21 bits)	Number incremented in production
Manufacturer code	327	327
ECU instance	0...7	0
Function instance	0...15	13
Function	0...255	129
Vehicle system	0...127	1
Vehicle system instance	0...15	0
Industry group	0...7	2
Arbitrary address capable	0	0

**Note 4 ECU Part Number Revision (example)**

The ECU Part Number Revision is the complement to the ECU Part Number

Byte number	Definition	Characters	ASCII
67	Revision type	A = Proto A B = Proto B C = Proto C S = Serial	A = 0x41 B = 0x42 C = 0x43 S = 0x53
68, 69	Revision number	00...99	0 = 0x30... 9 = 0x39
70	Manufacture type	P = Production (standard) R = Return product after defect T = Test (R&D use, specific unit not for production)	P = 0x50 R = 0x52 T = 0x54

**Note 5**

Example of SBZ software identification coding fields:

Bios software ID		
Byte nbr	Char	ASCII
4	8	0x38
5	5	0x35
6	6	0x36
7	5	0x35
8	S	0x53
9	0	0x30
10	0	0x30
11	0	0x30
12	.	0x2E
13	0	0x30
14	0	0x30
15	0	0x30

**Note 6:** ECU part number (example):

Byte number	Character	ASCII
25	8	0x38
26	5	0x35
27	6	0x36
28	5	0x35
29	R	0x52
30	1	0x31
31	0	0x30
32	0	0x30
33	0	0x30

**Note 7:** ECU serial number

Format: YxxDyyyNzzzz

Byte number	Definition	ASCII
71	Y : Year	0x59
72, 73	xx: Manufacture year (09...99 , or XX) 0 = 0x30 ... 9 = 0x39 or X = 0x58	0 = 0x30 ... 9 = 0x39 or X = 0x58
74	D: Day	0x44
75...77	yyy: Manufacture Day in the year (1...365 or XXX)	0 = 0x30 ... 9 = 0x39 or X = 0x58
78	N:Number	0x4E
79...82	Manufactured Product number in the day (0...9999)	0 = 0x30 ... 9 = 0x39 or X = 0x58

X = Not Available

For prototypes the fields "year" and "manufacture day" and "product Number" could be coded with the character "X"

#### Note 8: ECU location

The ECU location is defined as the function instance (same data present in the NAME), but this field is coded in ASCII character. By default the function instance is set to 0. This field is programmable from 0 to 15d. The function instance is used to determine the valve number which determines the supported PGN for actuator control

Byte number	Definition	Character	ASCII
84, 85	Function instance (valve number)	00...15 "	2 chars from 0 = 0x30 to 9 = 0x39

#### Note 9: ECU type

The next table illustrates the system behaviour as function of ECU type and ECU location fields.

Type of valve	ECU type Character	ECU type ASCII	ECU location (3 LSB)	Command message PGN	Estimated flow message PGN
Auxiliary valve 0	A	0x41	0	FE30	FE10
Auxiliary valve 1	A	0x41	1	FE31	FE11
Auxiliary valve 2-14	A	0x41	2-14	FE32...FE3E	FE12...FE1E
Auxiliary valve 15	A	0x41	15	FE3F	FE1F
General Purpose valve	G	0x47	0	C400	C600

If the ECU type is "A", the system behaves like an auxiliary valve.

If the ECU type is "G", the system behaves like a general purpose valve.

#### Note 10 Specific parameter definitions

##### Valve offset

Xoffset+: 16-bit extended offset value in [100\*mm] with reference to X<sub>0+/-</sub> (Example 1.81 mm equals 181d → 00B5h).

Xoffset- : 16-bit retracted offset value in [100\*mm] with reference to X<sub>0+/-</sub> (Example 1.81 mm equals 181d → 00B5h).

##### Max position – Xmax+

16-bit extended maximum position value in [100\*mm] with reference to X<sub>0+</sub>.

Values outside the Min/Max interval may lead to activation of a fault code.

Example : 6.21 mm equals 621

Min value : Xoffset+

Max value : 15.00 mm (0x5DC)

##### Min position – Xmax-

16-bit retracted maximum position value in [100\*mm] with reference to X<sub>0-</sub>.

Values outside the Min/Max interval may lead to activation of a fault code.

Example : 6.21 mm equals 621

Min value : Xoffset-

Max value : 10.00 mm (0x3E8)

### Float position – Xfloat POS

16-bit float position value in [100\*mm] with reference to  $X_{0+/-}$ .

Values outside the Min/Max interval may lead to activation of a fault code.

Example : 6.21 mm equals 621

Min value : XfloatDIR = 0x00: Xoffset+; XfloatDIR = 0x01: Xoffset-

Max value : XfloatDIR = 0x00: 15.00 mm (0x5DC); XfloatDIR = 0x01: 10.00 mm (0x3E8)

### Float direction – Xfloat DIR

Float position direction: Extend = 00h, Retract = 01h.

### CAN baudrate – CANBTC

Allows the user to set the baudrate of the CAN protocol.

**IMPORTANT:** Set this parameter wrong, will permanently inhibit the communication of the SCU. The only way to recover the system is downloading a new firmware using the bootloader. (This setting does not influence the baudrate of the bootloader)

32 bits possible CANBTC value:

500kBaud: 0x000501B1

250kBaud: 0x000501FA (default)

125kBaud: 0x000B01FA

100kBaud: 0x000E01FA

Note that the J1939 standard specifies the transmission rate at 250 kbps. Change the baudrate implies loose totally the J1939 network compatibility with the other nodes.

### CAN Timeout

This parameter allows the user to set the CAN Timeout. The unit value is tens ms.

For example value 16 in EEPROM will give a CAN Timeout of 160ms.

### Position curve

This parameter allows the user to set if the position curve has to follow the direct or programmable curve

0x00 : Direct CAN to Position

0x01 : Programmable

### User Curve Extend/Retract CAN Percent

8-bit value to configure the CAN Percent parameter for User Curve in [%]

Values outside the Min/Max interval active the “Linearization Parameter Error”

These values must be sorted in ascending order in EEPROM table to avoid “Linearization Parameter Error”

Example : 98% equals 98 (0x62)

Min value : 0

Max value : 150% (0x96)

### User Curve Extend/Retract QValve

8-bit value to configure the QValve parameter for User Curve in [%]

Values outside the Min/Max interval active the “Linearization Parameter Error”

Example : 98% equals 98 (0x62)

Min value : 0

Max value : 150% (0x96)

### User Curve Extend/Retract QMax

16-bit value to configure the QMax parameter in [0.001\*]

Values outside the Min/Max interval active the “Linearization Parameter Error”

Example : 0.95 equals 950 (0x3B6)

Min value : 0

Max value : 1.5 equals 1500 (0x5DC)

### Linear Curve Extend/Retract QValve Scaled

16-bit value to configure the QValve Scaled for Linear Curve in [10\*%]

Values outside the Min/Max interval active the “Linearization Parameter Error”

These values must be sorted in ascending order in EEPROM table to avoid “Linearization Parameter Error”

Example : 98.6% equals 986 (0x3DA)

Min value : 0

Max value : 150% (0x5DC)

### User Curve Extend/Retract Spool Pos

8-bit value to configure the Spool Position for Linear Curve in [10\*mm]

Values outside the Min/Max interval active the “Linearization Parameter Error”

Example : 6.2 mm equals 62 (0x3E)

Min value : 0

Max value : 25.5 mm (0xFF)

### Sensor Enable Functionality

This parameter allows the user to enable or disable the sensor functionality (stall detection by sensor, measured position frame etc.)

0x00 : Sensor Functionality Disable

0x01 : Sensor Functionality Enable

### Sensor Zero Reference Valid

This parameter allows the user to set the zero position of the sensor. The actuator should be used in neutral mode and the dead-zone procedure must be completed to perform calibration of the sensor.

0x00 : The actuator must perform its dead-zone search and define the zero position (Neutral) of the sensor.

0x01 : The actuator has a zero position value

**After each disassembly of the actuator / valve, it is necessary to write 0 on this parameter to redefine the zero position of the sensor.**



### Measured Position Status Transmission

This parameter allows the user to enable or disable the transmission of measured position frame

0x00 : Measured Position Frame Transmission Disable

0x01 : Measured Position Frame Transmission Enable

### Sensor Neutral Fault Recovery Timeout

This parameter allows the user to set the time to define how long the sensor must be in neutral before resetting the error in [10\*s]

Example : 2s equals 20 (0x14)

Min value : 0

Max value : 25s (0xFA)

### Sensor Stall Neutral Enable

This parameter allows the user to enable or disable the stall detection around neutral position

0x00 : Sensor Stall Neutral Disable

0x01 : Sensor Stall Neutral Enable

### Sensor Stall Normal Enable

This parameter allows the user to enable or disable the stall detection outside neutral position

0x00 : Sensor Stall Normal Disable

0x01 : Sensor Stall Normal Enable

### Sensor Threshold Offset

This parameter allows the user to set the position threshold offset to detect a stall in Normal Mode [100\*mm]

Example : 1.20 mm equals 120

Min value : 120

Max value : 1000

### Sensor Prot React Time 10 mm/s

This parameter allows the user to set the protection reaction time to detect a stall in Normal Mode with Speed = 10mm/s [100\*s]

Example : 1.50 s equals 150

Min value : 150

Max value : 250

### Sensor Prot React Time 20 mm/s

This parameter allows the user to set the protection reaction time to detect a stall in Normal Mode with Speed = 20mm/s [100\*s]

Example : 1.50 s equals 150

Min value : 85

Max value : 250

#### Sensor Prot React Time 40 mm/s

This parameter allows the user to set the protection reaction time to detect a stall in Normal Mode with Speed = 40mm/s [100\*s]

Example : 1.50 s equals 150

Min value : 45

Max value : 250

#### Sensor Prot React Time 60 mm/s

This parameter allows the user to set the protection reaction time to detect a stall in Normal Mode with Speed = 60mm/s [100\*s]

Example : 1.50 s equals 150

Min value : 35

Max value : 250

#### Sensor Prot React Time 80 mm/s

This parameter allows the user to set the protection reaction time to detect a stall in Normal Mode with Speed = 80mm/s [100\*s]

Example : 1.50 s equals 150

Min value : 25

Max value : 250

#### Sensor Prot React Time 100 mm/s

This parameter allows the user to set the protection reaction time to detect a stall in Normal Mode with Speed = 100mm/s [100\*s]

Example : 1.50 s equals 150

Min value : 25

Max value : 250

#### ECU Address Filter

This parameter allows the user to accept the CAN command only if the CAN ID of transmitter corresponds to the "Destination Address" value in EEPROM.

0x00 : Don't filter the CAN ID (accept all ECU ID)

0x01 : Filter the CAN ID (accept only one ECU ID)

#### DM1 Undervoltage reaction delay

This parameter allows the user to set the time from voltage drops below undervoltage limit to the DM1 is transmitted in [10\*s]

Example : 2s equals 20 (0x14)

Min value : 0

Max value : 25s (0xFA)

#### Boost Mode Enable

This parameter allows the user to enable or disable the Boost Mode.

0x00 : Boost Mode Disable

0x01 : Boost Mode Enable

### Xboost Extend Min

16-bit extended minimum position value for Boost Mode in [100\*mm]  
Example : 1.00 mm equals 100 (0x64)  
Min value : 0  
Max value : 20.00 mm (0x7D0)

### Xboost Extend Max

16-bit extended maximum position value for Boost Mode in [100\*mm]  
Example : 7.00 mm equals 700 (0x2BC)  
Min value : 0  
Max value : 20.00 mm (0x7D0)

### Xboost Retract Min

16-bit extended minimum position value for Boost Mode in [100\*mm]  
Example : 1.00 mm equals 100 (0x64)  
Min value : 0  
Max value : 20.00 mm (0x7D0)

### Xboost Retract Max

16-bit extended maximum position value for Boost Mode in [100\*mm]  
Example : 7.00 mm equals 700 (0x2BC)  
Min value : 0  
Max value : 20.00 mm (0x7D0)

### Xboost Value

This parameter allows the user to set the boost value [%]  
Example : 30% equals 30 (0x1E)  
Min value : 10  
Max value : 30% (0x64)

### Normal Speed

This parameter allows the user to set the normal speed [mm/s]  
Example : 80mm/s equals 80 (0x50)  
Min value : 10 (0x0A)  
Max value : 100 (0x64)

### Float Speed

This parameter allows the user to set the float speed [mm/s]  
Example : 80mm/s equals 80 (0x50)  
Min value : 10 (0x0A)  
Max value : 100 (0x64)

### Low Speed 1

This parameter allows the user to set the Low Speed 1[mm/s]  
Example : 60mm/s equals 60 (0x3C)  
Min value : 10 (0x0A)  
Max value : 60 (0x3C)

### Low Voltage Speed

This parameter allows the user to set the Low Voltage Speed [mm/s]

Example : 40mm/s equals 40 (0x28)

Min value : 10 (0x0A)

Max value : 40 (0x28)

### Stall Detection Back-EMF Enable

This parameter allows the user to enable or disable the stall detection based on Back EFM

0x00 : Stall Detection Back EMF Disable

0x01 : Stall Detection Back EMF Enable

PRELIMINARY

### 6.1.7.2. System Variables

The variables can be read by accessing the memory (with DM14 (Pointer Extension = 2), DM15 and DM16 messages). All variables are stored in RAM.

Address [DEC]	Data length [bytes].	Name	RD/WR	Definition
1	7	Claim main info	RD	Byte 1: Actual temperature – Note 1 Byte 2: Dead zone value [um] (MSB) Byte 3: Dead zone value [um] (LSB) Byte 4: Pos Origo value [um] (MSB) Byte 5: Pos Origo value [um] (LSB) Byte 6: VDC [V] (MSB) – Note 2 Byte 7: VDC [V] (LSB) – Note 2

**Note 1:** The value read on the byte 1 corresponds to the actual temperature + 80°C

Example: Byte 1 set to 128 corresponds to 48°C

**Note 2:** The value read on the bytes 6 and 7 corresponds to the VDC \* 1000

Example: Bytes 6 and 7 set to 23456 corresponds to 23.456V

### 6.1.7.3. DM14 – Memory access request

Message ID	18 D9 "DA" "SA"
Default Priority	6
PGN	00D900 (55552)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer
MP (multipacket)	No
Message direction	Receive (ECU to ACU).
Message length	8 bytes
Repetition rate	N/A
Timeout	N/A

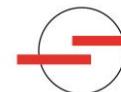
<b>Byte 1</b>	MSB	LSB
Length / Number requested LSB		
<b>Byte 2</b>	MSB	LSB
Length / Number Requested MSB		
Pointer Type		
Command		
1		
<b>Byte 3</b>	MSB	LSB
Pointer		
<b>Byte 4</b>	MSB	LSB
Pointer		
<b>Byte 5</b>	MSB	LSB
Pointer		
<b>Byte 6</b> Reserved	MSB	LSB
Pointer extension		
<b>Byte 7</b>	MSB	LSB
Key / User level		
<b>Byte 8</b>	MSB	LSB
Key / User level		

#### Byte 1, Byte 2 (bit 6..8) : Length / Number

For a read access on the EEPROM, the number of bytes will be limited to 7d.

#### Byte 2, bit 5 : Pointer type

For diagnostic messages, this value should be set to 1



## Byte 2 (bit 2..4) : Command

Commands for memory access request:

Data Value	Command definition	Meaning / Extra info
1	Read	Read memory. When this command is accepted, the device transfers the appropriate memory contents to the tool (use TP if necessary)
2	Write	Write memory. The device may force an erase before write.
4	Operation Completed	Sent by the tool during a close sequence to indicate to the device that it has heard the device's close and that the close sequence is completed. The device should have a time-out (100ms +/- 25ms) function to close the memory access operation if the expected "operation completed" is not received.
5	Operation Failed	Sent by the tool during the close sequence of a read command. It indicates that the expected memory contents were not received
6	Boot Load	Transfer the execution of a device.

## Byte 2 (bit 1) : Always 1

## Byte 3..5 : Pointer, Byte 3 = LSB

## Byte 6 : Pointer extension

The 8-bit parameter is the high order 8 bits of a complete direct memory address, and is used as a memory type selector. It will be set to 0 for a flash access, 1 for an EEPROM access and 2 for variable access.

## Byte 7, 8 : Key / User level, byte 7 = LSB

The ACU doesn't manage User level information. The ACU will only treat this field as a key.

Key value	Meaning
0	Use long key or key from data security message. The actual seed or key is in the data security message (DM18) and this is a simple flag
0x2505	Key value for write memory access
Application specific	Key value for executable code (bootload) a specific key can be defined for each application in EEPROM → KEY BOOT
0xFFFF	No key available

The ACU doesn't use keys longer than 16 bits (possible with DM18 - data security message). So the key field can't be set to 0 in our application.

#### 6.1.7.4. DM15 – Memory access response

Message ID	18 D8 "DA" "SA"
Default Priority	6
PGN	00D800 (55296)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer
MP (multipacket)	No
Message direction	Transmit (ACU to ECU).
Message length	8 bytes
Repetition rate	As needed
Timeout	N/A

Byte 1	MSB	Length / Number allowed LSB	LSB
Byte 2	MSB	Length / Number allowed MSB	R Status R
Byte 3	MSB	Error indicator / EDC parameter	LSB
Byte 4	MSB	Error indicator / EDC parameter	LSB
Byte 5	MSB	Error indicator / EDC parameter	LSB
Byte 6	MSB	EDCP extension	LSB
Byte 7	MSB	Seed	LSB
Byte 8	MSB	Seed	LSB

**Byte 1, byte 2 (bit 6..8) :** Length / Number allowed, byte 1 = LSB

**Byte 2 (bit 2..4) : Status**

Return the device status

Status value	Status definition	Meaning	Note
0	Proceed	The tool can continue with the memory sequence it requested	-
1	Busy	Sent from a device to indicate that there is a condition which prevents the sequence from continuing	The "Length/Number allowed" parameter is set to 0. The tool will not treat the "seed" field.
2	Reserved	-	-
3	Reserved	-	-
4	Operation Completed	Inform that the memory access was successfully completed. Sent as a status during a close sequence (Erase, Read, Write, Boot load or EDCP generation) or in response to a status request. The device should have a timeout 100+/-25ms to close the session if the expected "operation complete" in DM14 from the tool is not received.	-
5	Operation failed	Failure in memory access. Sent during a close sequence or in response to a request status command. Can be sent only if the sequence is not completed.	
6	Reserved	-	-
7	Reserved	-	-

**Byte 3..5 : Error indicator/EDC parameter**

The data bytes "Error Indicator/EDC Parameter" are not used and are filled with 0xFFFFFFF (no error indicator available in the ACU).

**Byte 6 : EDCP extension**

The "EDCP extension field" must be filled with 0xFF (error indicator/EDC parameter not available) because the Error Indicator/EDC parameter is not managed in the ACU.



### Byte 7..8: Seed (byte 7 = MSB)

Seed values:

0 : Seed completed – begin sending key. It indicates that no seed is going to be provided by the device.

1 : Use long seed or key from data security message (DM18)

2-FFFE : Seed values

FFFF : No further key required from tool to begin the operation

Memory operations are only allowed for parameters defined in the Memory map and bootloader procedure. The ACU doesn't use long seed or key from data security messages (Seed = 0x0001). For memory read access or EDCP generation, the ACU doesn't require a seed/key security system (Seed = 0xFFFF).

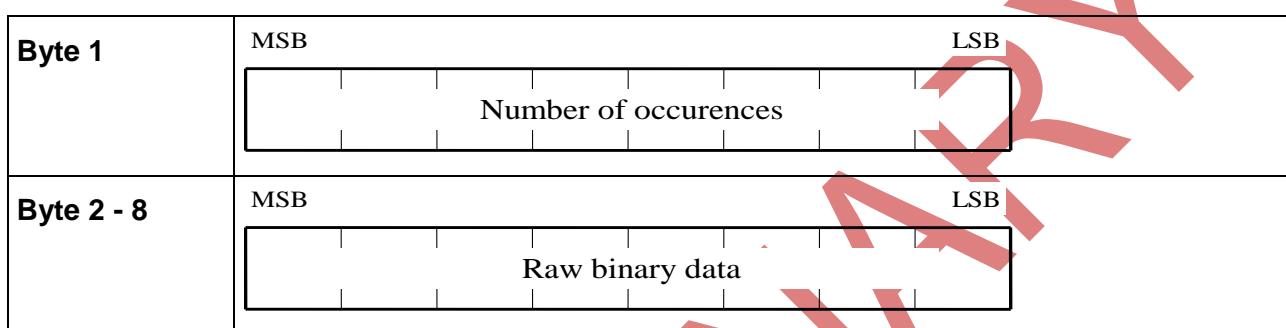
For memory write access, a key is always required from the tool:

Parameter to write	Seed
Executable code (bootload)	A specific Seed can be defined for each application in EEPROM → SEED BOOT
Customer parameter	0xFFFF

PRELIMINARY

### 6.1.7.5. DM16 - Binary data transfer

Message ID	18 D7 "DA" "SA"
Default Priority	6
PGN	00D700 (55040)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer
MP (multipacket)	No
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU)
Message length	8 bytes
Repetition rate	As needed
Timeout	N/A



### 6.1.7.6. Memory access Timeout

If the response from the tool is operation failed or if there is no response from the tool within 100ms (+additional delay for retransmission) the device close itself the memory access sequence (return to initial status). The ACU time-out will be ~125ms.

PRELIMINARY

### **6.1.8. Actuator control**

#### **6.1.8.1. Auxiliary valve command 0...15**

The actuator is able to receive successive CAN messages every 100 $\mu$ s without losing any messages. (Note that it is impossible to write faster than this at 250Kbaud, only the 29 bit identifier takes >100 $\mu$ s to write.)

Message ID	0C FE 30 "SA" (valve 0) 0C FE 31 "SA" (valve 1) ... 0C FE 3E "SA" (valve 14) 0C FE 3F "SA" (valve 15)
Default Priority	3
PGN	00FE30 (65072) (valve 0) ... 00FE3F (65087) (valve 15)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	No
Message direction	Receive (ECU to ACU).
Message length	8 bytes
Repetition rate	10 to 100ms when active
Timeout	Maximum time between two General purpose valve Command message Configurable from 10ms to 1000ms, default 1s The timeout check is activated with the first received position command. Refer to the fault code table when timeout appears.

<b>Byte 1</b>	MSB	LSB
Flow (0...100%)	Standard Flow [0-100%]	
<b>Byte 2</b>	MSB	LSB
Reserved	1   1   1   1   1   1   1   1	
<b>Byte 3</b>	MSB	LSB
State	Fail safe mode: mode: Reserved	1   1 Valve state:
<b>Byte 4</b>	MSB	LSB
Reserved	1   1   1   1   1   1   1   1	
<b>Byte 5</b>	MSB	LSB
Reserved	1   1   1   1   1   1   1   1	

Byte 6	MSB	LSB								
Reserved	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1			
Byte 7	MSB	LSB								
Reserved	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1			
Byte 8	MSB	LSB								
Reserved	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	1	1	
1	1	1	1	1	1	1	1			

### **Byte 1..2: Standard Flow**

The percentage of flow depends on the position curve which can be manufacturer specific. By default the position curve is linear.

Standard resolution : 0 → No flow, neutral, 250 is the highest flow 0.4% resolution, 251...255: Neutral

### **Byte 3 (bit 7..8) : Fail safe mode**

For security reasons, the only safe operation permitted is "return to Neutral" although others possibilities are defined in the J1939 standard. These bits are ignored.

### **Byte 3 (bit 5..6): Reserved**

These bits are not considered for the message consistency check.

### **Byte 3 (bit 1..4) : Valve states**

- 0x0: Block (neutral), the actuator returns to neutral (origin position) with current in the phases. If the actuator receives a "neutral position" command, the current is reduced once the position is reached.
- 0x1: Extend, the actuator extends to the corresponding position defined by the flow parameter
- 0x2: Retract, the actuator retracts to the corresponding position defined by the flow parameter
- 0x3: Float, the actuator moves to the float position defined in the parameters.
- 0xA: Force re-initialisation, ACU is driven back to neutral and starts a new initialisation. Only for Sonceboz test use.
- 0xE: Safety mode, phase current switched off.

All other values will command the valve to neutral (Blocked) state (answer with error 0xE + DM1: inconsistent CAN Control message).

### **Bytes 4 to 8 are ignored by the ACU**

These bytes are reserved. They must be filled with 0xFF

#### **Note :**

If "ECU Address Filter" (@667) is set to disable (default value), this command accepts all addresses independent of "Destination Address" EEPROM value (@106).

Extend and Retract with regard to spool movement direction



### 6.1.8.2. General Purpose valve command

The actuator is able to receive successive CAN messages every 100µs without losing any messages. (Note that it is impossible to write faster than this at 250Kbaud, only the 29 bit identifier takes >100µs to write.)

Message ID	0C C4 "DA" "SA"
Default Priority	3
PGN	00C400 (50176)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer
MP (multipacket)	No
Message direction	Receive (ECU to ACU).
Message length	8 bytes
Repetition rate	10 to 100ms when active
Timeout	Maximum time between two General purpose valve Command message Configurable from 10ms to 1000ms, default 1s The timeout check is activated with the first received position command. Refer to the fault code table when timeout appears.

<b>Byte 1</b> Standard Resolution Flow	MSB	LSB
Standard Flow [0-100%]		
<b>Byte 2</b> Reserved	MSB	LSB
1 1 1 1 1 1 1 1		
<b>Byte 3</b> State	MSB	LSB
Fail safe mode: 1 1 Reserved		
Valve state:		
<b>Byte 4</b> Extended Resolution Flow (0...100%)	MSB	LSB
Extended Flow [0-100%] LSB (D7 to D0)		
<b>Byte 5</b> Extended Resolution Flow (0...100%)	MSB	LSB
Extended Flow [0-100%] MSB (D15 to D8)		
<b>Byte 6</b> Reserved	MSB	LSB
1 1 1 1 1 1 1 1		
<b>Byte 7</b> Reserved	MSB	LSB
1 1 1 1 1 1 1 1		

Byte 8	MSB	LSB
Reserved	1 1 1 1 1 1 1 1	

### **Byte 1: Standard Flow**

The percentage of flow depends on the position curve which can be manufacturer specific. By default the position curve is linear. It is possible to control the flow with 8 bits (standard resolution) or 16 bits resolution (extended resolution) with Byte4 and Byte5.

Standard resolution : 0 → No flow, neutral, 250 is the highest flow 0.4% resolution, 251...255: Neutral

Byte4 and byte5 must be set to 0 or 65535 to use 8bit resolution

### **Byte 2 are ignored by the ACU**

This byte is reserved. It must be filled with 0xFF

### **Byte 3 (bit 7..8) : Fail safe mode**

For security reasons, the only safe operation permitted is "return to Neutral" although others possibilities are defined in the J1939 standard. These bits are ignored.

### **Byte 3 (bit 5..6): Reserved**

These bits are not considered for the message consistency check.

### **Byte 3 (bit 1..4) : Valve states**

- 0x0: Block (neutral), the actuator returns to neutral (origin position) with current in the phases. If the actuator receives a "neutral position" command, the current is reduced once the position is reached.
- 0x1: Extend, the actuator extends to the corresponding position defined by the flow parameter.
- 0x2: Retract, the actuator retracts to the corresponding position defined by the flow parameter.
- 0x3: Float, the actuator moves to the float position defined in the parameters.
- 0xA: Force re-initialisation, ACU is driven back to neutral and starts a new initialisation. Only for Sonceboz test use.
- 0xE: Safety mode, phase current switched off.

All other values will command the valve to neutral (Blocked) state (answer with error 0xE + DM1: inconsistent CAN Control message).

### **Byte 4 – 5 : Extended Flow**

In extended resolution byte4 and byte5 are used. Byte1 is ignored but to avoid malfunction, It is recommended to complete the byte 1 with standard resolution.

For Example, with the command Extend 0%, Byte 4 and 5 are set to 0 and Byte 1 will be used in this case.

Extended resolution : 0 → No flow, neutral, 64000 is the highest flow 0.0015625% resolution,

0 : Use 8bit resolution

64001...64255 : 100%

64256...65534: Neutral

65535 : Use 8bit resolution

### **Bytes 6 to 8 are ignored by the ACU**

These bytes are reserved. They must be filled with 0xFF

### Example:

With DA = 0xA0 (ACU Address) / SA = 0x22 (ECU Address) / Standard resolution

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x0CC4A022	0x00	0xFF	0x30	0xFF	0xFF	0xFF	0xFF	0xFF	Neutral
0x0CC4A022	0xFA	0xFF	0x31	0xFF	0xFF	0xFF	0xFF	0xFF	Extend 100%
0x0CC4A022	0xFA	0xFF	0x32	0xFF	0xFF	0xFF	0xFF	0xFF	Retract 100%
0x0CC4A022	0x80	0xFF	0x31	0xFF	0xFF	0xFF	0xFF	0xFF	Extend 51.2%
0x0CC4A022	0x80	0xFF	0x32	0xFF	0xFF	0xFF	0xFF	0xFF	Retract 51.2%
0x0CC4A022	0x00 (Ignore)	0xFF	0x33	0xFF	0xFF	0xFF	0xFF	0xFF	Float
0x0CC4A022	0x00 (Ignore)	0xFF	0x3E	0xFF	0xFF	0xFF	0xFF	0xFF	Safety

With DA = 0xA0 (ACU Address) / SA = 0x22 (ECU Address) / Extended resolution

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x0CC4A022	0x00	0xFF	0x30	0x00	0x00	0xFF	0xFF	0xFF	Neutral
0x0CC4A022	0x00	0xFF	0x31	0x00	0x00	0xFF	0xFF	0xFF	Extend 0%
0x0CC4A022	0x00	0xFF	0x32	0x00	0x00	0xFF	0xFF	0xFF	Retract 0%
0x0CC4A022	0xFA (ignore)	0xFF	0x31	0x00	0xFA	0xFF	0xFF	0xFF	Extend 100%
0x0CC4A022	0xFA (ignore)	0xFF	0x32	0x00	0xFA	0xFF	0xFF	0xFF	Retract 100%
0x0CC4A022	0x80 (ignore)	0xFF	0x31	0x01	0x80	0xFF	0xFF	0xFF	Extend 51.2015625%
0x0CC4A022	0x80 (ignore)	0xFF	0x32	0x01	0x80	0xFF	0xFF	0xFF	Retract 51.2015625%
0x0CC4A022	0x00 (Not use)	0xFF	0x33	0x00 (Not use)	0x00 (Not use)	0xFF	0xFF	0xFF	Float
0x0CC4A022	0x00 (Not use)	0xFF	0x3E	0x00 (Not use)	0x00 (Not use)	0xFF	0xFF	0xFF	Safety

### Note :

If "ECU Address Filter" (@667) is set to disable (default value), this command accepts all addresses independent of "Destination Address" EEPROM value (@106).

Extend and Retract with regard to spool movement direction

**6.1.8.3. Auxiliary valve estimated flow 0...15**

Message ID	0C FE 10 "SA" (valve 0) 0C FE 11 "SA" (valve 1) ...
Default Priority	3
PGN	00FE10 (65040) (valve 0) ...
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	No
Message direction	Transmit (ACU to ECU).
Message length	8 bytes
Repetition rate	100ms
Timeout	N/A

N/A

Byte 1	MSB	LSB
Extend position	Extend Estimated Flow - Standard	
Byte 2	MSB	LSB
Retract position	Retract Estimated Flow - Standard	
Byte 3	MSB	LSB
Operation mode	Failsafe operation: 0      0 Reserved	Valve state
Byte 4	MSB	LSB
Reserved / Limit status	1      1      1      1      1      1      1      1	
Byte 5	MSB	LSB
Extend estimated flow LSB	1      1      1      1      1      1      D1      D0	
Byte 6	MSB	LSB
Retract estimated flow	1      1      1      1      1      1      D1      D0	
Byte 7	MSB	LSB
Reserved	1      1      1      1      1      1      1      1	
Byte 8	MSB	LSB
Reserved	1      1      1      1      1      1      1      1	

### **Byte 1: Extend position:**

Byte 1 value	Description
125	Neutral (No flow)
126 to 225	Proportional extend flow where 126 is the lowest flow and 225 is 100% flow
226 to 250	Not used (100...125% extend flow)
251 to 255	Not used (invalid values)
25 to 124	25 to 124 – Proportional negative extend flow where 124 is the lowest flow and 25 is -100% extend flow
0 to 24	Not used (-125...-100% extend flow)

Note: Estimated flow is updated as soon as Auxiliary valve commands is received

### **Byte 2: Retract position**

Byte 2 value	Description
125	Neutral (No flow)
126 to 225	Proportional retract flow where 126 is the lowest flow and 225 is 100% flow
226 to 250	Not used (100...125% retract flow)
251 to 255	Not used (invalid values)
25 to 124	25 to 124 – Proportional negative retract flow where 124 is the lowest flow and 25 is -100% retract flow
0 to 24	Not used (-125...-100% retract flow)

Note: Estimated flow is updated as soon as Auxiliary valve commands are received

### **Byte 3 (bit 7..8): Failsafe Operation**

For security reasons, the only safe operation permitted is "return to Neutral" although others possibilities are defined in the J1939 standard. Bits 6 and 7 will always be set to "0"

### **Byte 3 (bit 1..4): Valve States**

- 0x0: Block (neutral), the actuator is commanded to neutral position
- 0x1: Extend, the actuator is commanded to the extend side of the valve
- 0x2: Retract, the actuator is commanded to the retract side of the valve
- 0x3: Float, the actuator is commanded to the float position
- 0xE: Error

All other values are reserved.

### **Byte 4 : Limit status**

Bit 8 to 6 will be used to code the limit status in the next issue of ISO11783-7. These bits are filled with 111b to indicate that this parameter is not available

### **Byte 5, 6:**

These bytes contain the 2 LSB of estimated flows (extend flow for byte 5 and retract flow for byte 6) to get a more accurate estimated position if needed.

### **Byte 7, 8:**

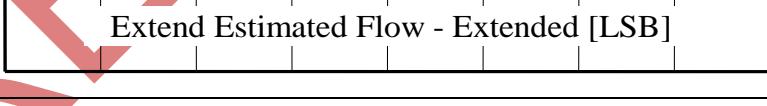
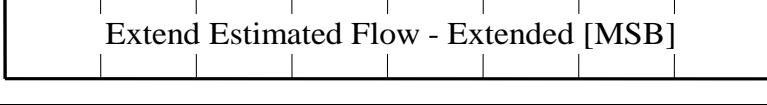
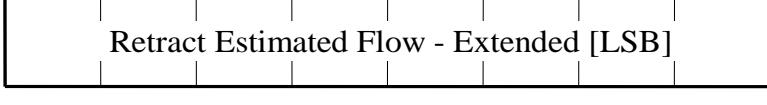
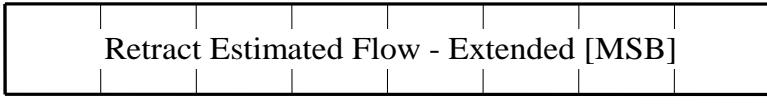
These bytes are reserved. They are filled with 0xFF.

#### **Note :**

Estimated flow is sent with Destination Address in EEPROM Value.

#### 6.1.8.4. General Purpose estimated flow

Message ID	0C C6 "DA" "SA"
Default Priority	3
PGN	00C600 (50688)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer
MP (multipacket)	No
Message direction	Transmit (ACU to ECU).
Message length	8 bytes
Repetition rate	100ms
Timeout	N/A

<b>Byte 1</b> Extend position Standard	MSB	LSB
		
<b>Byte 2</b> Retract position Standard	MSB	LSB
		
<b>Byte 3</b> Operation mode	MSB	LSB
		
<b>Byte 4</b> Reserved / Limit status	MSB	LSB
		
<b>Byte 5</b> Extend estimated flow Extended Resolution (LSB)	MSB	LSB
		
<b>Byte 6</b> Extend estimated flow Extended Resolution (MSB)	MSB	LSB
		
<b>Byte 7</b> Retract estimated flow Extended Resolution (LSB)	MSB	LSB
		
<b>Byte 8</b> Retract estimated flow Extended Resolution (MSB)	MSB	LSB
		

## Byte 1: Extend position – Standard resolution

Byte 1 value	Description
125	Neutral (No flow)
126 to 225	Proportional extend flow where 126 is the lowest flow and 225 is 100% extend flow
226 to 250	Not used (100...125% extend flow)
251 to 255	Not used (invalid values)
25 to 124	25 to 124 – Proportional negative extend flow where 124 is the lowest flow and 25 is -100% extend flow
0 to 24	Not used (-125...-100% extend flow)

These bytes contain the Extend estimated flows with standard resolution (1% / bit, -125% offset)

Note: Estimated flow is updated as soon as General purpose valve commands is received

## Byte 2: Retract position – Standard resolution

Byte 2 value	Description
125	Neutral (No flow)
126 to 225	Proportional retract flow where 126 is the lowest flow and 225 is 100% retract flow
226 to 250	Not used (100...125% retract flow)
251 to 255	Not used (invalid values)
25 to 124	25 to 124 – Proportional negative retract flow where 124 is the lowest flow and 25 is -100% retract flow
0 to 24	Not used (-125...-100% retract flow)

These bytes contain the Retract estimated flows with standard resolution (1% / bit, -125% offset)

Note: Estimated flow is updated as soon as General purpose valve commands is received

## Byte 3 (bit 7..8): Failsafe Operation

For security reasons, the only safe operation permitted is "return to Neutral" although others possibilities are defined in the J1939 standard. Bits 6 and 7 will always be set to "0"

## Byte 3 (bit 1..4): Valve States

- 0x0: Block (neutral), the actuator is commanded to neutral position
- 0x1: Extend, the actuator is commanded to the extend side of the valve
- 0x2: Retract, the actuator is commanded to the retract side of the valve
- 0x3: Float, the actuator is commanded to the float position
- 0xE: Error

All other values are reserved.

## Byte 4 : Limit status

Bit 8 to 6 will be used to code the limit status in the next issue of ISO11783-7. These bits are filled with 111b to indicate that this parameter is not available

### **Byte 5, 6 Extend Position – Extended resolution**

Byte 5-6 value	Description
31250	Neutral (No flow)
31251 to 56250	Proportional extend flow where 31251 is the lowest flow and 56250 is 100% extend flow
56251 to 62500	Not used (100...125% extend flow)
62501 to 65535	Not used (invalid values)
6250 to 31249	6250 to 31249 – Proportional negative extend flow where 31249 is the lowest flow and 6250 is -100% extend flow
0 to 6249	Not used (-125...-100% extend flow)

These bytes contain the Extend estimated flows with extended resolution (0.004% / bit, -125% offset) to get a more accurate estimated position if needed.

### **Byte 7, 8 Retract Position – Extended resolution**

Byte 5-6 value	Description
31250	Neutral (No flow)
31251 to 56250	Proportional retract flow where 31251 is the lowest flow and 56250 is 100% retract flow
56251 to 62500	Not used (100...125% retract flow)
62501 to 65535	Not used (invalid values)
6250 to 31249	6250 to 31249 – Proportional negative retract flow where 31249 is the lowest flow and 6250 is -100% retract flow
0 to 6249	Not used (-125...-100% retract flow)

These bytes contain the Retract estimated flows with extended resolution (0.004% / bit, -125% offset) to get a more accurate estimated position if needed.

### **Byte 7, 8:**

These bytes are reserved. They are filled with 0xFF.

### **Example:**

With DA = 0x22 (ECU Address) / SA = 0xA0 (ACU Address)

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x0CC622A0	0x7D	0x7D	0x00	0xFF	0x12	0x7A	0x12	0x7A	Start-up
0x0CC622A0	0x7D	0x7D	0x00	0xFF	0x12	0x7A	0x12	0x7A	Neutral
0x0CC622A0	0xE1	0x19	0x01	0xFF	0xBA	0xDB	0x6A	0x18	Extend CMD 100%
0x0CC622A0	0x19	0xE1	0x02	0xFF	0x6A	0x18	0xBA	0xDB	Retract CMD 100%
0x0CC622A0	0x7D	0x7D	0x03	0xFF	0x12	0x7A	0x12	0x7A	Float
0x0CC622A0	0x7D	0x7D	0x0E	0xFF	0x12	0x7A	0x12	0x7A	Error

### **Note :**

Value for Destination Address is the EEPROM Value (@106d)

Extend and Retract with regard to spool movement direction

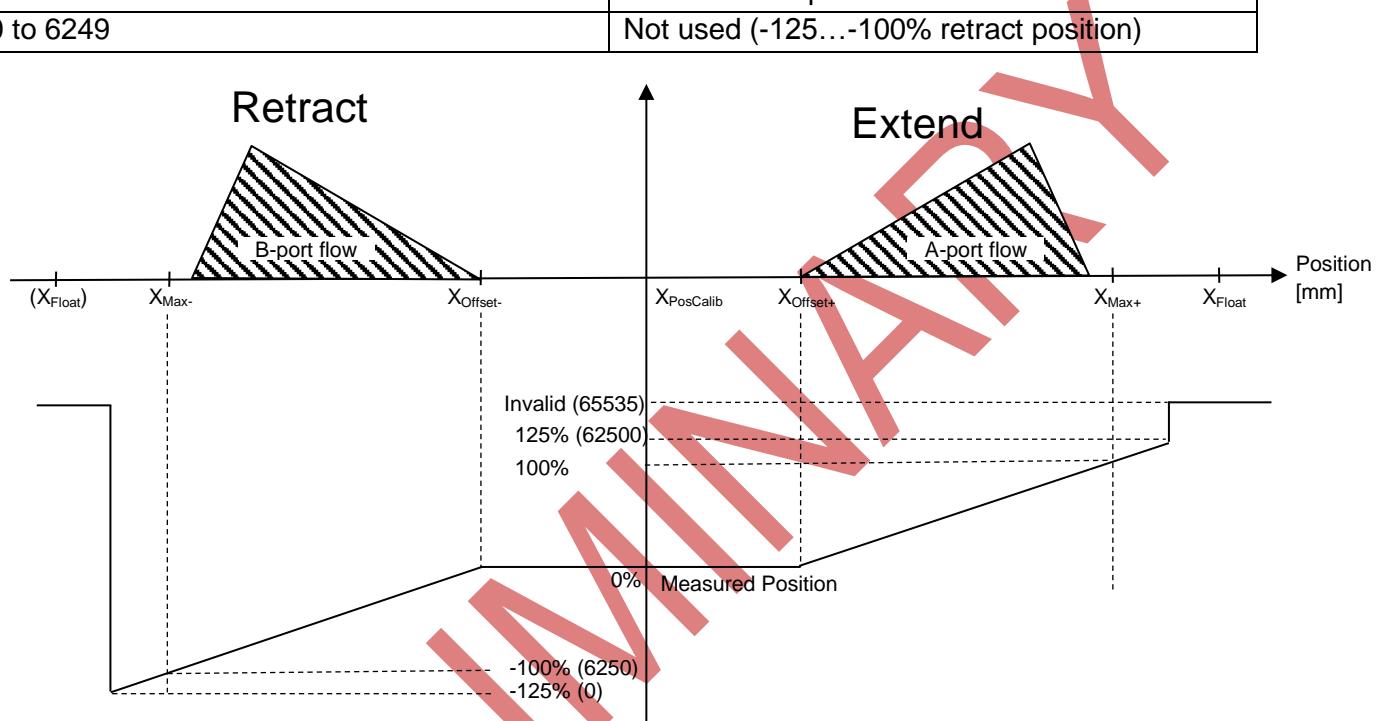
### 6.1.8.5. Auxiliary valve measured position 0...15

Message ID	0C FF 20 "SA" (valve 0) 0C FF 21 "SA" (valve 1) ...
Default Priority	3
PGN	00FF20 (65312) (valve 0) ...
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	No
Message direction	Transmit (ACU to ECU).
Message length	8 bytes
Repetition rate	100ms
Timeout	N/A

<b>Byte 1</b>	MSB	LSB
Measured position	Measured Position % [LSB]	
<b>Byte 2</b>	MSB	LSB
Measured position	Measured Position % [MSB]	
<b>Byte 3</b>	MSB	LSB
Operation mode	1    1    1    1	Sensor State
<b>Byte 4</b>	MSB	LSB
Measured position ( $\mu\text{m}$ )	Measured Position $\mu\text{m}$ [LSB]	
<b>Byte 5</b>	MSB	LSB
Measured position ( $\mu\text{m}$ )	Measured Position $\mu\text{m}$ [MSB]	
<b>Byte 6</b>	MSB	LSB
Reserved	1    1    1    1    1    1    1    1	
<b>Byte 7</b>	MSB	LSB
Reserved	1    1    1    1    1    1    1    1	
<b>Byte 8</b>	MSB	LSB
Reserved	1    1    1    1    1    1    1    1	

### Byte 1, 2: Measured position (%):

Byte 1, 2 value	Description
31250	Neutral (No flow)
31251 to 56250	Proportional extend position where 31251 is the lowest position and 56250 is 100% position
56251 to 62500	Not used (100...125% extend position)
62501 to 65535	Not used (invalid values)
6250 to 31249	6250 to 31249 – Proportional retract position where 31249 is the lowest position and 6250 is -100% retract position
0 to 6249	Not used (-125...-100% retract position)



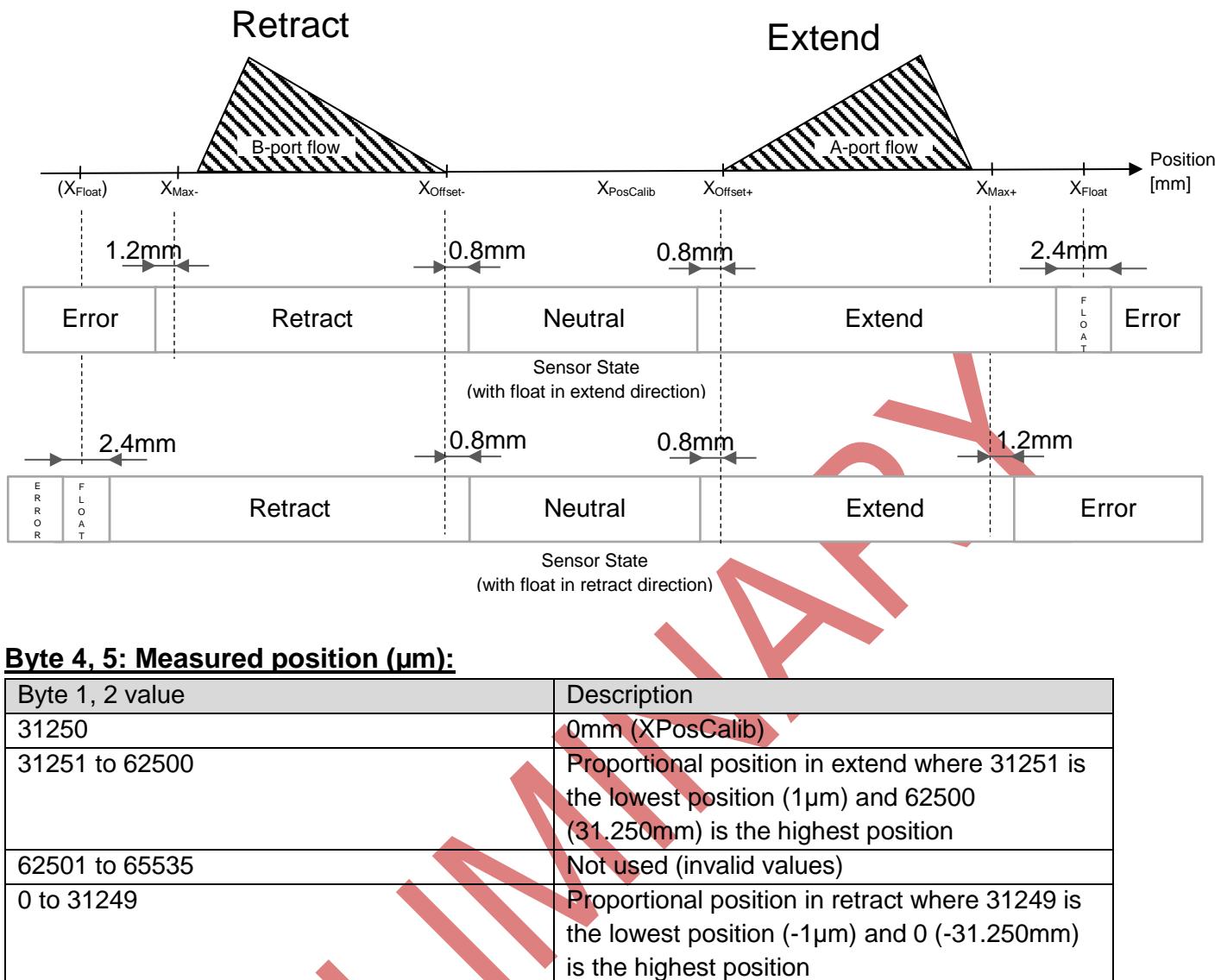
### Byte 3 (bit 1..4): Sensor States

Example with float in extend direction

- 0x0: Block (neutral), the actuator is in Neutral position  $(X_{offset-} - 0.8mm) \geq Position \leq (X_{offset+} - 0.8mm)$
  - 0x1: Extend, the actuator is in the extend side of the valve  
 $(X_{offset+} - 0.8mm) \geq Position \leq (X_{float} - 1.2mm)$  if  $XfloatDir = 0$  (Extend)  
 $(X_{offset+} - 0.8mm) \geq Position \leq (X_{max+} + 1.2mm)$  if  $XfloatDir = 1$  (Retract)
  - 0x2: Retract, the actuator is in the retract side of the valve  
 $(X_{max-} - 1.2mm) \geq Position \leq (X_{offset-} + 0.8mm)$  if  $XfloatDir = 0$  (Extend)  
 $(X_{float} + 1.2mm) \geq Position \leq (X_{offset-} + 0.8mm)$  if  $XfloatDir = 1$  (Retract)
  - 0x3: Float, the actuator is in the float position  $(X_{float} - 1.2mm \geq Position \geq X_{float} + 1.2mm)$
  - 0xE: Error, initialization error or out of range value
- All other values are reserved.

Note1 : For retract parameters, the values are defined as negative in the calculation

Note2 : The minimum value for detection of the Neutral State is +/- 0.8mm. If  $X_{offset} \leq 1.6mm$ , the value  $(X_{offset+} - 0.8mm)$  or  $(X_{offset-} + 0.8mm)$  will be forced to +/- 0.8mm.



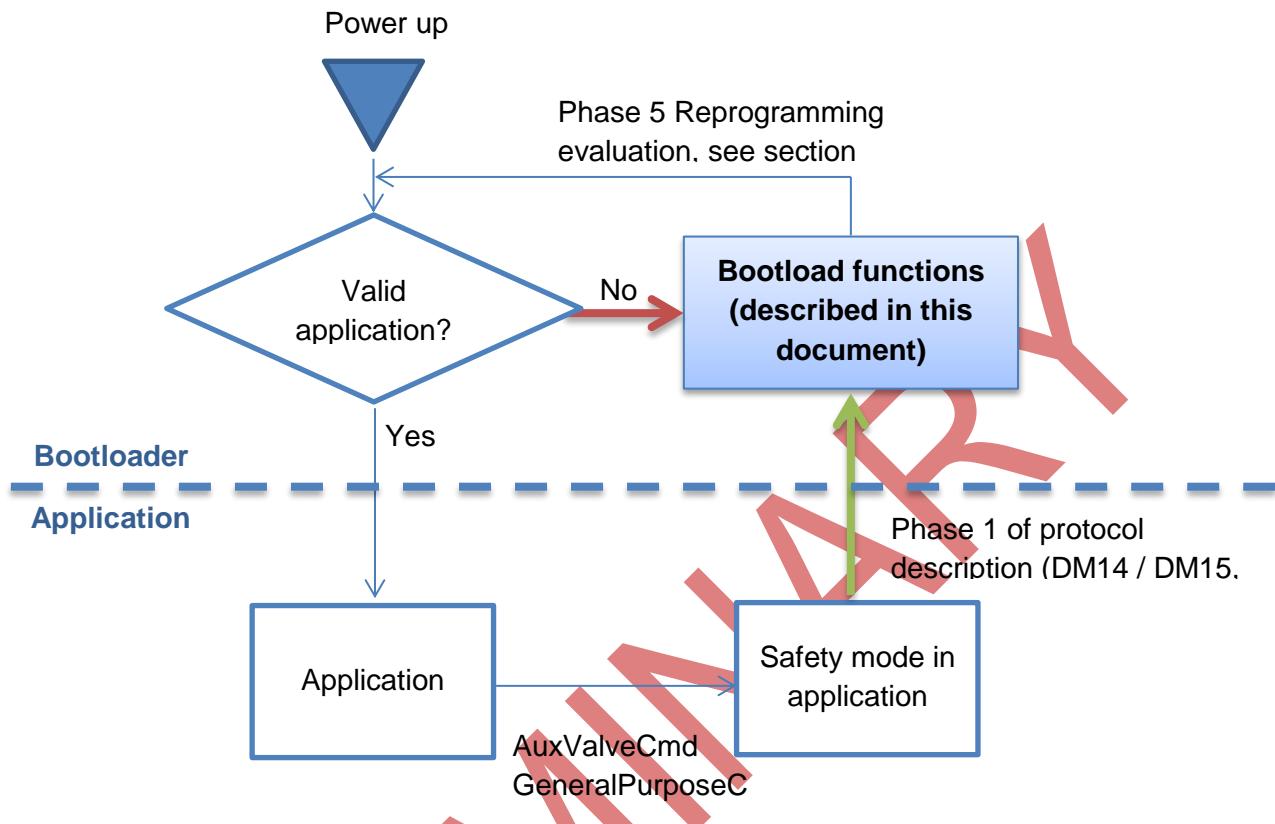
**Byte 4 to 8:**

These bytes are reserved. They are filled with 0xFF.

**PRELIMINARY**

## 6.2. CAN J1939 Bootloader Protocol

### 6.2.1. State diagram power up sequence



As seen in the figure above, there are two possible entry points to the bootloader functions.

**Scenario 1: The device has a valid application**

**Scenario 2: The device has no valid application (application is erased or not fully programmed)**

Scenario1 with valid application:

- The address claim procedure was already done by the application
- The programmer communicates with the application sets the application into the safety mode. After this, the programmer sends the reprogramming request. **The whole phase 1 of the protocol description is done by the application.** If successful, the application will jump into the bootloader code.
- The bootloader does not send an address claim
- The bootloader expects the beginning of **phase 2** of the protocol
- The (reduced) DM1 message informs about the flash state (ok before flash has been erased, erroneous while programming and ok after a successful "Request new program error check" message).

Scenario 2 with invalid application:

- The address claim procedure is done by the bootloader (reduced functionality)
- The bootloader expects the beginning of **phase 1** of the protocol (reprogramming request)
- The (reduced) DM1 message informs about the flash error

Security seed and key validation must be known by application and bootloader

## 6.2.2. Protocol Overview

The CANbus reprogramming protocol covers three major activities:

- initial handshaking and compatibility checks between the programmer and target module (Phase 1)
- reprogramming of the target module (repeated cycles of Phases 2,3,4)
- program error checks (Phase 5)

### *Phase 1: Initiate Reprogramming Request, Perform Security Check*

The programming device must issue a DM14 / DM15 memory request that includes a HWID code. This code verifies that the firmware to be programmed is compatible with the target module hardware version, and Boot Loader code version installed on the target module.

The initial sequence must also respond to a security seed with a preprogrammed algorithm to authenticate it has the authority to reprogram the target module.

### *Phase 2: Negotiate Reprogramming Parameters*

Compatibility established, the programmer will request permission to load new firmware within a specific memory range of the target module, and await permission to proceed.

### *Phase 3: Transfer Program Block*

The programmer may transfer up to 1784 bytes per memory range request. Important: Due to internal memory access in the target, the programmer must send an even number of bytes!

### *Phase 4: Acknowledge Transfer of Program Block*

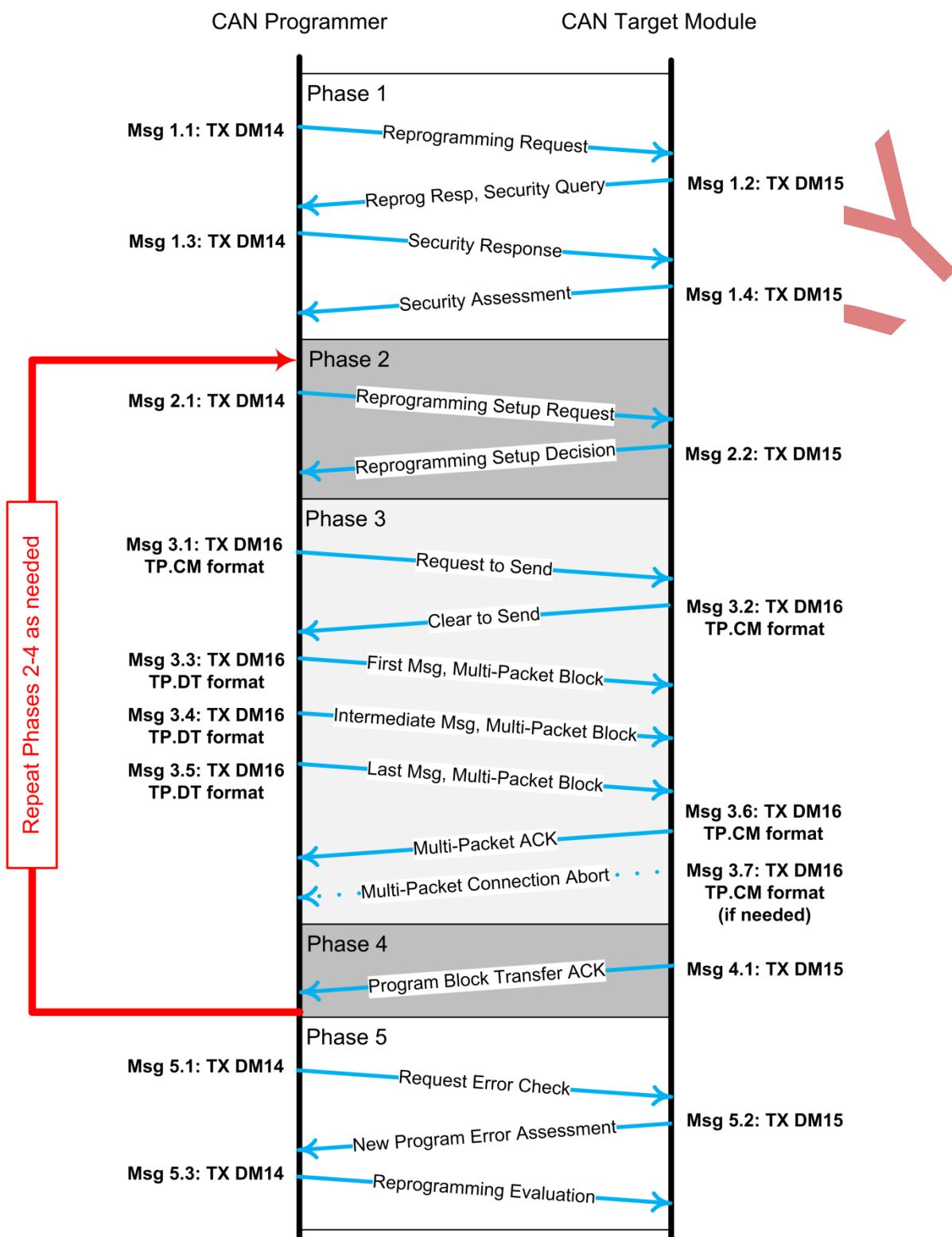
Once the program block transfer has completed without errors, the programmer may continue with a new address range request and data transfer without repeating the initial security exchange. This process may repeat until all of the code has been transferred, an error has occurred during the transfer, or the operation has timed out.

### *Phase 5: Program Error Check*

Once the programmer has transferred the entire firmware image, the target module verifies the integrity of the firmware with an adequate algorithm (e.g. checksum or CRC check). If the check is successful, the target module will restart using the new firmware. If not, the target module will default to a boot loader mode that will allow the programmer to retry the programming of the target module. A successful integrity check must occur before the target module will return to service.

### 6.2.2.1. Protocol diagram

## CAN Module Reprogramming Procedure



The following is a command by command description of the protocol exchange needed to request, secure, and set up the boot load, transfer the data, and reset enable the target module when the transfer is complete. Exceptions, timeouts, and the default boot loader mode are explained within this document. For a full description of each command used, refer to SAE documents J1939-73 and J1939-21.

Phase 1: Init. Reprog. Request, Perform Security Check

- Msg 1.1: Reprogramming Request
- Msg 1.2: Response to Reprog. Request, Security Query
- Msg 1.3: Security Response
- Msg 1.4: Security Assessment

-----Repeat Phases 2,3,4 as needed to transfer entire program-----

Phase 2: Negotiate Reprogramming Parameters

- Msg 2.1: Reprogramming Setup Request

**Long delay of bootloader when receiving the first Msg 2.1<sup>(1)</sup>**

- Msg 2.2: Reprogramming Decision (Proceed/Terminate)

Phase 3: Transfer Program Block

- Msg 3.1: Request To Send Multi-Packet Program Block
- Msg 3.2: Clear To Send Multi-Packet Program Block
- Msg 3.3: First Message of Multi-Packet Program Block Transfer
- Msg 3.4: Intermediate Message of Multi-Packet Program Block Transfer
- Msg 3.5: Last Data Message of Multi-Packet Program Block Transfer
- Msg 3.6: Multi-Packet Transfer Acknowledgement

**Erreur ! Source du renvoi introuvable.**

Phase 4: Acknowledge Transfer of Program Block

- Msg 4.1: Program Block Transfer Acknowledge

Phase 5: Program Error Check

- Msg 5.1: Request New Program Error Check
- Msg 5.2: New Program Error Assessment
- Msg 5.3: Reprogramming Evaluation

(1) When receiving the very first Msg 2.1 the bootloader will erase the application flash. Erasing the flash does typically last 12 seconds but can last longer. The programmer must wait up to 30 seconds before assuming a erase failure. The whole application flash is erased in one step. This means that this is necessary once per reprogramming. During the erase process the bootloader is not able to communicate!

### 6.2.3. Protocol details

#### 6.2.3.1. Phase 1: Init. Reprog. Request, Perform Security Check

##### 6.2.3.1.1. Msg 1.1: Reprogramming Request

**Source:** Programmer

**Receiver:** Target module

Message Type: J1939 - **DM14** Memory Access Request

Data Length	8
Data Page	0
PDU Format	217 (D9 h)
PDU Specific	Destination Address (Address of target module)
Default Priority	6
PGN	55552 (00D900 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1*	00h	Length of memory requested (least significant 8 bits) Length = 0(00) h = 0
2	LLLP CCCR b	LLL: Length of memory requested (most significant 3 bits) LLL = 000 b = 0  P: Pointer type P = 1: Directed spatial addressing  CCC: Command CCC = 110 b = <b>6: Boot load</b>  R: SAE Reserved R = 1
3	00h	Memory Pointer Low Byte
4	00h	Memory Pointer Mid Byte
5	00h	Memory Pointer High Byte Memory Pointer = 000000 h = 0
6	00 h	Memory Pointer Extension = 0
7	02 h	HWID Low Byte **
8	01 h	HWID High Byte **

\* Byte order: Byte 1 is the first byte of the message data to be transmitted. Each multi-byte data parameter is transmitted using Intel, little endian, format. Eg. HWID – low byte, high byte.

\*\* 1<sup>st</sup> step of security validation procedure.  
The HWID is taken from the S28 file (in S0 record).

The HWID is defined by Sonceboz as following:

HWID = 0xABCD

With:

- ABC corresponding to the three last digits of the Sonceboz part number
- D corresponding to the Sonceboz HW Version Identification (hard coded on the PCB)

Exemple:

SBZ part number = 8566R1234

Sonceboz HW Version Identification = 0x9

⇒ **HWID = 0x2349**

## 6.2.3.1.2. Msg 1.2: Response to Reprog. Request, Security Query

**Source: Target module**  
**Receiver: Programmer**

Message Type: J1939 - DM15 Memory Access Response

Data Length	8
Data Page	0
PDU Format	216 (D8 h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	6
PGN	55296 (00D800 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	00h	Length of memory allowed (least significant 8 bits) Length = 0(00) h = 0
2	LLL R SSS R b	LLL: Length of memory allowed (most significant 3 bits) LLL = 000 b = 0 R: SAE Reserved R = 0 SSS: Target module Status SSS = 000 b = <b>0: Proceed,</b> SSS = 101 b = <b>5: Request Failed,</b>
3		R: SAE Reserved R = 0
4		Error Code Low Byte
5		Error Code Mid Byte
5		Error Code High Byte
6	06 h	Error Code = 000000 h = 0: No Error Error Code = 001002 h = 4098: Invalid ID Error Code = 001007 h = 4103: Not Allowed
6		Error Indicator Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code (alternative: an EDC Parameter)
7	02 h*	Seed Low Byte
8	00 h*	Seed High Byte Seed (example) = 0002 h* (2 <sup>nd</sup> step of security validation procedure)

**The seed is a fixed number: 0x0002**

Error response:

If a DM14 with the following parameters was received:

- Length of memory requested ≠ 0
- HWID incorrect

A DM15 with the following parameters is transmitted:

- Length allowed = 0
- Status = 5: Operation failed
- Error Code = 001002 h = 4098: Invalid ID
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code
- Seed = FFFFh

### 6.2.3.1.3. Msg 1.3: Security Response

**Source: Programmer**

**Receiver: Target module**

Message Type: J1939 - **DM14 Memory Access Request**

Data Length	8
Data Page	0
PDU Format	217 (D9 h)
PDU Specific	Destination Address (Address of target module)
Default Priority	6
PGN	55552 (00D900 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	00 h	Length of memory requested (least significant 8 bits) Length = 0(00) h = 0
2	LLLP CCCR b	LLL: Length of memory requested (most significant 3 bits) LLL = 000 b = 0  P: Pointer type P = 1: Directed spatial addressing  CCC: Command CCC = 110 b = <b>6: Boot load</b>
3	00 h	R: SAE Reserved R = 1  Memory Pointer Low Byte
4	00 h	Memory Pointer Mid Byte
5	00 h	Memory Pointer High Byte Memory Pointer = 000000 h = 0
6	00 h	Memory Pointer Extension = 0
7	4Ah*	Seed Key Low Byte
8	6Ah*	Seed Key High Byte Seed Key (example) = 6A4A h* (Per Algorithm) (3 <sup>rd</sup> step of security validation procedure)

The algorithm to create the Key is:

**Key = (~seed) + 0x6A4D**

#### 6.2.3.1.4. Msg 1.4: Security Assessment

**Source: Target module**  
**Receiver: Programmer**

Message Type: J1939 - DM15 Memory Access Response

Data Length	8
Data Page	0
PDU Format	216 (D8 h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	6
PGN	55296 (00D800 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	00h	Length of memory allowed (least significant 8 bits) Length = 0(00) h = 0
2	LLL R SSS R b	LLL: Length of memory allowed (most significant 3 bits) LLL = 000 b = 0 R: SAE Reserved R = 0 SSS: Target module Status SSS = 100 b = 4: <b>Operation complete</b> if valid security state, SSS = 101 b = 5: <b>Operation failed</b> if invalid security state
3		R: SAE Reserved R = 0
4		Error Code Low Byte
5		Error Code Mid Byte
5		Error Code High Byte
6	06 h	Error Code = 000000 h = 0: No Error Error Code = 001003 h = 4103: Invalid Data Error Code = 001007 h = 4103: Not Allowed Error Indicator Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code (alternative: an EDC Parameter)
7	FF h	Seed Low Byte
8	FF h	Seed High Byte Seed = FFFF h (no seed – security check completed)

The programmer waits 500ms after getting this response before continuing the reprogramming. In this time the application jumps to the bootloader (or, if already in bootloader waits).

Error response:

If a DM14 with the following parameters was received:

- Source address = tool source address
- Command ≠ 6: Boot load
- Length of memory requested ≠ 0
- Key incorrect

A DM15 with the following parameters is transmitted:

- Length allowed = 0
- Status = 5: Operation failed
- Error Code = 001003 h = 4099: Invalid key
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code
- Seed = FFFFh

If a DM14 with the following parameters was received:

- Source address ≠ tool source address

A DM15 with the following parameters is transmitted:

- Received source address
- Length allowed = 0
- Status = 1: Busy
- Error Code = 001007 h = 4103: Not allowed
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code

Seed = FFFFh

### 6.2.3.2. Phase 2: Negotiate Reprogramming Parameters

#### 6.2.3.2.1. Msg 2.1: Reprogramming Setup Request

Source: Programmer

Receiver: Target module

Message Type: J1939 - DM14 Memory Access Request

Data Length	8
Data Page	0
PDU Format	217 (D9 h)
PDU Specific	Destination Address (Address of target module)
Default Priority	6
PGN	55552 (00D900 h)

Byte	Specific Data	Description
1	00 h	Length of memory requested (least significant 8 bits) Length = 4(00) h = 1024
2	LLLP CCCR b	LLL: Length of memory requested (most significant 3 bits) LLL = 100 b = 4 P: Pointer type P = 0: Direct memory addressing CCC: Command CCC = 010 b = 2: Write command

		R: SAE Reserved R = 1
3	00h	Memory Pointer Low Byte
4	10h	Memory Pointer Mid Byte
5	00h	Memory Pointer High Byte
		Memory Pointer = 001000 h = 4096
6	00h	Memory Pointer Extension = 0
7	FF h	Seed Key Low Byte
8	FF h	Seed Key High Byte
		Seed Key = FFFF h
		(no seed key – security check completed)

### 6.2.3.2.2. Msg 2.2: Reprogramming Decision (Proceed/Terminate)

**Source:** Target module

**Receiver:** Programmer

Message Type: J1939 - DM15 Memory Access Response

Data Length	8
Data Page	0
PDU Format	216 (D8 h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	6
PGN	55296 (00D800 h)

Byte	Specific Data	Description
1	00h	Length of memory allowed (least significant 8 bits) Length = 4(00) h = 1024
2	LLL R SSS R b	LLL: Length of memory allowed (most significant 3 bits) LLL = 100 b = 4 R: SAE Reserved R = 0 SSS: Target module Status SSS = 000 b = 0: Proceed SSS = 101 b = 5: Request Failed,
3		R: SAE Reserved R = 0
4		Error Code Low Byte
5		Error Code Mid Byte
5		Error Code High Byte Error Code = 000000 h = 0: No Error Error Code = 0A1000 h = 659456: General Error
6	06 h	Error Indicator Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code (alternative: an EDC Parameter)
7	FF h	Seed Low Byte
8	FF h	Seed High Byte Seed = FFFF h

(no seed – security check completed)

**Before sending this message the target will check if it has to erase the flash memory. If yes, the target will first erase the flash and then send the response. This means that the programmer must expect a delay up to 30 seconds before getting the response!**

Error response:

If a DM14 with the following parameters was received:

- Command = 2: Write
- Source address ≠ tool source address
- Pointer type ≠ 0
- Security not OK

A DM15 with the following parameters is transmitted:

- Length allowed = 0
- Status = 5: Operation failed
- Error Code = 0A1000 h = 659456: General Error
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code
- Seed = FFFFh

### 6.2.3.3. Phase 3: Transfer Program Block

At this point, all exchanges necessary to start the actual transfer of data to begin the reprogramming process are in place. The programmer and target module will now use the DM16 Transport Protocol, including the use of RTS and CTS packet exchanges, to transfer and flash the data to the address and length of data specified in the exchange above.

Once the program block has been exchanged and flashed, and no errors have occurred, the target module will remain in safe mode, and will be ready to flash another block of memory, once it exchanges a new DM14 / DM15 series of packets to specify the starting address, and length of the next flash segment.

The lack of timely communications from either the programmer or the target module is monitored by time-out functions detailed in SAE J1939-21 (exceptions are listed in this specification).

If the reprogramming fails for any reason (timeout, invalid address range, power fail, lost data packet, Checksum/CRC failure, etc) the target will reboot at next power cycle to a safe mode.

This allows multiple attempts to reprogram a target module that failed the initial attempt, without disassembly of the machine.

#### 6.2.3.3.1. Msg 3.1: Request To Send Multi-Packet Program Block

**Source: Programmer**

**Receiver: Target module**

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.CM Transport Protocol Connection Management

Data Length            8

Data Page	0
PDU Format	236 (EC h)
PDU Specific	Destination Address (Address of target module)
Default Priority	7
PGN	60416 (00EC00 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	10 h	Control Byte = 10 h = 16 (Destination specific RTS)
2		Total Message Size Low Byte
3		Total Message Size High Byte Max total msg size (in bytes) = 400 h + 1 h = 1024+1
4		Total Number of Packets to be Transmitted Overall DM16 max = 255 (1784 bytes/7 bytes per packet = 255)
5		Max Num of Packets Requested per Exchange (per CTS) DM16 max = 255 (1784 bytes/7 bytes per packet = 255)
6	00 h	PGN of Data Content Low Byte*
7	D7 h	PGN of Data Content Mid Byte*
8	00 h	PGN of Data Content High Byte* PGN of Data Content = 00D700 h = 55040 (This is the PGN for DM16 Binary Data Transfer)

\* Following SAE J1939 standards, all data consisting of more than 8 bytes must be sent using a universal J1939 transport protocol. The content transmitted using this protocol can pertain to any parameter type (PGN). Therefore, the initial messages of the exchange must identify the type of content (PGN) being transmitted. In this case, the content will belong to the DM16 PGN which covers Binary Data Transfer.

**Due to limitations of the bootloader, the programmer must send TotalNumberOfPackets = MaxNumberOfPackets. In the example above, 255 is set as number. The programmer can decide to send fewer packets as long as it sends all packets per CTS.**

**Due to internal memory access (memory width is 16 bit), the programmer must send always an even number of bytes in one block transfer.**

### 6.2.3.3.2. Msg 3.2: Clear To Send Multi-Packet Program Block

**Source: Target module**

**Receiver: Programmer**

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.CM Transport Protocol Connection Management

Data Length	8
Data Page	0
PDU Format	236 (EC h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	7
PGN	60416 (00EC00 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	11 h	Control Byte = 11 h = 17 (CTS)
2		Number of Packets Allowed per Exchange (ref byte 5 of RTS) DM16 max = 255 (1784 bytes / 7 bytes per packet = 255)
3		Number of Next Packet
4	FF h	SAE Reserved = FF h
5	FF h	SAE Reserved = FF h
6	00 h	PGN of Data Content Low Byte
7	D7 h	PGN of Data Content Mid Byte
8	00 h	PGN of Data Content High Byte PGN of Data Content = 00D700 h = 55040 (This is the PGN for DM16 Binary Data Transfer)

The bootloader will respect the number of packets the programmer wants to send and response with the demanded MaxNumberOfPackets (Byte 2 will be set to the value in RTS byte 5).

Error response:

If a TP.CM message with the following parameters was received:

- Control byte ≠ 10h = 16: RTS
- Source address ≠ tool source address
- PGN ≠ 55040: DM16
- Total message length ≠ (allowed data length form previous DM15 + 1)

A TP.CM message with the following parameters is transmitted:

- Received source address
- Control byte = FFh = 255: Connection abort
- PGN = 55040: DM16

### ~~DRY~~ 6.2.3.3.3. Msg 3.3: First Message of Multi-Packet Program Block Transfer

**Source: Programmer**

**Receiver: Target module**

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.DT Transport Protocol Data Transfer

Data Length	8
Data Page	0
PDU Format	235 (EB h)
PDU Specific	Destination Address (Address of target module)
Default Priority	7
PGN	60160 (00EB00 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	01 h	Sequence Number – Always 01 h = 1 for first message.
2	FF h	Number of Occurrences (Always FF h for Multi-Packet DM16)
3		Raw Binary Program - Byte 1
4		Raw Binary Program - Byte 2

5	Raw Binary Program - Byte 3
6	Raw Binary Program - Byte 4
7	Raw Binary Program - Byte 5
8	Raw Binary Program - Byte 6

\* This is never a stand-alone message since it makes no reference to the PGN of the data content.  
 It must always follow in the wake of TP.CM-based RTS, CTS initialization.

Error response:

If a TP.DT message with the following parameters was received:

- Source address ≠ tool source address
- Sequence number ≠ expected next packet number

A TP.CM message with the following parameters is transmitted:

- Received source address
- Control byte = FFh = 255: Connection abort
- All other bytes = FFh
- PGN = 55040: DM16

If a receive timeout occurs (no TP.DT Msg. received within 1250ms from CTS) a TP.CM message with the following parameters is transmitted:

- Tool source address
- Control byte = FFh = 255: Connection abort
- Byte2 = 3: Connection abort timeout
- All other bytes = FFh
- PGN = 55040: DM16

#### 6.2.3.3.4. Msg 3.4: Intermediate Message of Multi-Packet Program Block Transfer

**Source: Programmer**

**Receiver: Target module**

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.DT Transport Protocol Data Transfer

Data Length	8
Data Page	0
PDU Format	235 (EB h)
PDU Specific	Destination Address (Address of target module)
Default Priority	7
PGN	60160 (00EB00 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1		Sequence Number – in numerical order
2		Raw Binary Program - Byte n
3		Raw Binary Program - Byte n+1
4		Raw Binary Program - Byte n+2
5		Raw Binary Program - Byte n+3
6		Raw Binary Program - Byte n+4

7	Raw Binary Program - Byte n+5
8	Raw Binary Program - Byte n+6

\* This is never a stand-alone message since it makes no reference to the PGN of the data content and is content-dependent upon previous messages in the protocol.

Error response:

If a TP.DT message with the following parameters was received:

- Source address ≠ tool source address
- Sequence number ≠ expected next packet number

A TP.CM message with the following parameters is transmitted:

- Received source address
- Control byte = FFh = 255: Connection abort
- All other bytes = FFh
- PGN = 55040: DM16

If a receive timeout occurs (no TP.DT Msg. received within 750ms) a TP.CM message with the following parameters is transmitted:

- Tool source address
- Control byte = FFh = 255: Connection abort
- Connection Abort Reason (Byte2) = 3: Connection abort timeout
- All other bytes = FFh
- PGN = 55040: DM16

### 6.2.3.3.5. Msg 3.5: Last Data Message of Multi-Packet Program Block Transfer

**Source: Programmer**

**Receiver: Target module**

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.DT Transport Protocol Data Transfer

Data Length	8
Data Page	0
PDU Format	235 (EB h)
PDU Specific	Destination Address (Address of target module)
Default Priority	7
PGN	60160 (00EB00 h)

<u>Byte</u>	<u>Specific Data**</u>	<u>Description**</u>
1		Sequence Number – in numerical order
2		Raw Binary Program - Msg ByteLength - 4
3		Raw Binary Program - Msg ByteLength - 3
4		Raw Binary Program - Msg ByteLength - 2
5		Raw Binary Program - Msg ByteLength - 1
6		Raw Binary Program - Msg ByteLength
7	FF h	No Data**
8	FF h	No Data**

\* This is never a stand-alone message since it makes no reference to the PGN of the data content and is content-dependent upon previous messages in the protocol.

\*\* Example only. Data can terminate at any byte within the data portion of the message. Bytes containing no data are stuffed with FF h.

Error response:

If a TP.DT message with the following parameters was received:

- Source address ≠ tool source address
- Sequence number ≠ expected next packet number

A TP.CM message with the following parameters is transmitted:

- Received source address
- Control byte = FFh = 255: Connection abort
- All other bytes = FFh
- PGN = 55040: DM16

If a receive timeout occurs (no TP.DT Msg. received within 750ms) or too many TP.DT messages are received a TP.CM message with the following parameters is transmitted:

- Tool source address
- Control byte = FFh = 255: Connection abort
- Connection Abort Reason (Byte2) = 3: Connection abort timeout
- All other bytes = FFh
- PGN = 55040: DM16

### **6.2.3.3.6. Msg 3.6: Multi-Packet Transfer Acknowledgement**

**Source: Target module**

**Receiver: Programmer**

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.CM Transport Protocol Connection Management

Data Length	8
Data Page	0
PDU Format	236 (EC h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	7
PGN	60416 (00EC00 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	13 h	Control Byte = 13 h = 19 (EOM-ACK)
2		Total Message Size Low Byte
3		Total Message Size High Byte Max total msg size (in bytes) = 400 h + 1 h = 1024+1
4		Total Number of Packets Received Max Message Size: 1024 bytes = 147 packets 6 bytes “first packet” + 1015 bytes / 7 bytes per packet “intermediate packets” + 3 bytes “last packet” = 147 packets

5	FF h	SAE Reserved = FF h
6	00 h	PGN of Data Content Low Byte
7	D7 h	PGN of Data Content Mid Byte
8	00 h	PGN of Data Content High Byte
		PGN of Data Content = 00D700 h = 55040
		(This is the PGN for DM16 Binary Data Transfer)

### 6.2.3.3.7. Msg 3.7: Multi-Packet Connection Abort (on fault)

**Source:** Target module

**Receiver:** Programmer

Message Data Content: J1939 – **DM16** Binary Data Transfer

Message Structure: J1939 – TP.CM Transport Protocol Connection Management

Data Length	8
Data Page	0
PDU Format	236 (EC h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	7
PGN	60416 (00EC00 h)

Byte	Specific Data	Description
1	FF h	Control Byte = FF h = 255 (CONN-ABORT)
2		Connection Abort Reason – from J1939-21
3	FF h	SAE Reserved = FF h
4	FF h	SAE Reserved = FF h
5	FF h	SAE Reserved = FF h
6	00 h	PGN of Data Content Low Byte
7	D7 h	PGN of Data Content Mid Byte
8	00 h	PGN of Data Content High Byte
		PGN of Data Content = 00D700 h = 55040
		(This is the PGN for DM16 Binary Data Transfer)

If an error occurs, the bootloaders internal state will be set to an error state.

Power cycle is needed before retrying to reprogram.

### 6.2.3.4. Phase 4: Acknowledge Transfer of Program Block

Once the DM16 transport protocol has successfully transferred the all of the data beginning at the starting address specified by the Phase 2 reprogramming setup exchange, the target module sends a DM15 message (operation complete) as shown below.

#### 6.2.3.4.1. Msg 4.1: Program Block Transfer Acknowledge

**Source:** Target module

**Receiver: Programmer**Message Type: J1939 - **DM15** Memory Access Response

Data Length	8
Data Page	0
PDU Format	216 (D8 h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	6
PGN	55296 (00D800 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	00h	Length of memory allowed (least significant 8 bits) Length = 4(00) h = 1024
2	LLL R SSS R b	LLL: Length of memory allowed (most significant 3 bits) LLL = 100 b = 4 R: SAE Reserved R = 0 SSS: Target module Status SSS = 100 b = 4: Complete
3		R: SAE Reserved R = 0
4		Error Code Low Byte
5		Error Code Mid Byte
		Error Code High Byte
		Error Code = 000000 h = 0: No Error
		Error Code = 0A1000 h = 659456: General Error
6	06 h	Error Indicator Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code (alternative: an EDC Parameter)
7	FF h	Seed Low Byte
8	FF h	Seed High Byte Seed = FFFF h (no seed – security check completed)

The bootloader gets the whole multi-packet transfer into the RAM and does write to the flash as soon it has sent the multi packet ACK message. Writing 1024 bytes to the flash will last maximum 125ms. Therefore the ACK message will be slightly delayed, but still within the normal device response time defined by J1939 (200ms).

The target module can then accept another block of flash data at a new address from the programmer, without repeating the security/seed data exchange. This must occur within 100 mS to keep the session open. Any timeout will result in an error state which will need a power cycle before restarting the reprogramming.

## Error response:

If there was an error while downloading program data with DM16 TP a DM15 with the following parameters is transmitted:

- Length allowed = 0
- Status = 5: Operation failed

- Error Code = 0A1000 h = 659456: General Error
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code
- Seed = FFFFh

### 6.2.3.5. Phase 5: Program Error Check

If all of the data has now been transferred, the programmer will initiate the following exchange with the target module.

This command instructs the target module to check for the firmware's integrity.

The target module responds with either Operation Complete (checksum match) or EDC parameter not correct.

#### 6.2.3.5.1. Msg 5.1: Request New Program Error Check

**Source: Programmer**

**Receiver: Target module**

Message Type: J1939 - DM14 Memory Access Request

Data Length	8
Data Page	0
PDU Format	217 (D9 h)
PDU Specific	Destination Address (Address of target module)
Default Priority	6
PGN	55552 (00D900 h)

Byte	Specific Data	Description
1	00 h	Length of memory requested (least significant 8 bits) Length = 0(00) h = 0
2	LLLP CCCR b	LLL: Length of memory requested (most significant 3 bits) LLL = 000 b = 0 P: Pointer type P = 0: Directed spatial addressing CCC: Command CCC = 111 b = 7: EDCP Generation
3	00 h	R: SAE Reserved R = 1
4	10 h	Memory Pointer Low Byte
5	00 h	Memory Pointer Mid Byte
		Memory Pointer High Byte
		Example Memory Pointer = 001000 h = 4096
6	00 h	Memory Pointer Extension = 0
7	FF h	Seed Key Low Byte
8	FF h	Seed Key High Byte Seed Key = FFFF h (no seed key – security check completed)

## 6.2.3.5.2. Msg 5.2: New Program Error Assessment

**Source: Target module**  
**Receiver: Programmer**

Message Type: J1939 - DM15 Memory Access Response

Data Length	8
Data Page	0
PDU Format	216 (D8 h)
PDU Specific	Destination Address (Address of programmer)
Default Priority	6
PGN	55296 (00D800 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	00 h	Length of memory allowed (least significant 8 bits) Length = 0(00) h = 0
2	LLL R SSS R b	LLL: Length of memory allowed (most significant 3 bits) LLL = 000 b = 0 R: SAE Reserved R = 0 SSS: Target module Status SSS = 100 b = 4: Operation Complete
3		Error Code Low Byte
4		Error Code Mid Byte
5		Error Code High Byte Error Code = 000000 h = 0: No Error Error Code = 000020 h = 32: EDC (Checksum) Error
6	06 h	Error Indicator Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code (alternative: an EDC Parameter)
7	FF h	Seed Low Byte
8	FF h	Seed High Byte Seed = FFFF h (no seed – security check completed)

If the target module sends an operation completed with no error indication, the programmer will then return the target module to service by sending an operation complete DM14 message. This will cause the target module to reset and return to service.

Error response:

If a DM14 with the following parameters was received:

- Command = 7: EDCP Generation
- Source address ≠ tool source address
- Pointer type ≠ 0
- Security not OK

A DM15 with the following parameters is transmitted:

- Length allowed = 0
- Status = 5: Operation failed
- Error Code = 0A1000 h = 659456: General Error
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code
- Seed = FFFFh

If the Flash CRC check fails a DM15 with the following parameters is transmitted:

- Length allowed = 0
- Status = 5: Operation failed
- Error Code = 000020 h = 32: EDC (Checksum) Error
- Err Ind = 06 h = 6: Data bytes 3-5 are an Error Code
- Seed = FFFFh

### 6.2.3.5.3. Msg 5.3: Reprogramming Evaluation

**Source: Programmer**

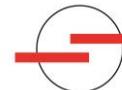
**Receiver: Target module**

Message Type: J1939 - **DM14 Memory Access Request**

Data Length	8
Data Page	0
PDU Format	217 (D9 h)
PDU Specific	Destination Address (Address of target module)
Default Priority	6
PGN	55552 (00D900 h)

<u>Byte</u>	<u>Specific Data</u>	<u>Description</u>
1	00 h	Length of memory requested (least significant 8 bits) Length = 0(00) h = 0
2	LLLP CCCR b	LLL: Length of memory requested (most significant 3 bits) LLL = 000 b = 0  P: Pointer type P = 0: Directed spatial addressing  CCC: Command CCC = 100 b = <b>4: Operation Complete</b>
		R: SAE Reserved R = 1
3	00 h	Memory Pointer Low Byte
4	00 h	Memory Pointer Mid Byte
5	00 h	Memory Pointer High Byte Memory Pointer = 000000 h = 0
6	00 h	Memory Pointer Extension = 0
7	FF h	Seed Key Low Byte
8	FF h	Seed Key High Byte Seed Key = FFFF h (no seed key – security check completed)

This command is used to restart the system after a successful reprogramming.



### 6.2.4. Address claim procedure

The bootloader does support a **reduced** address claim procedure only.

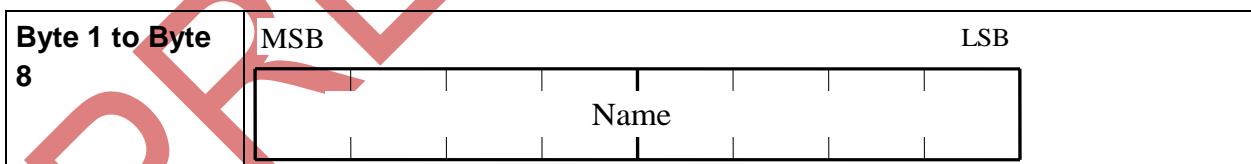
- It does not support the “commanded address” message
- It will not respond to a request for address claimed message.
- It does not support the evaluation of lower/higher priority in case of reception of an address claim message with the same address.
- If an address conflict arises, the bootloader will send a “cannot claim address” message and will stop communicating.

If this occurs while a data transfer is running, the data transmission will be aborted. This leads to an incomplete flash content. The boot load procedure will have to be repeated without any address conflict to get a running system.

**This means that any address conflict on the bus must be resolved before doing the boot load.**

#### 6.2.4.1. Address claimed

Message ID	18 EE FF "SA"
Default Priority	6
PGN	00EE00 (60928)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer (but becomes broadcast with DA = 0xFF)
MP (multipacket)	NO
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU)
Message length	8 bytes
Repetition rate	Sent at power on in case there was no valid application found. Sent after Bus Off Not sent if bootloader was started by application Not sent if requested (the request is not supported)
Timeout	N/A



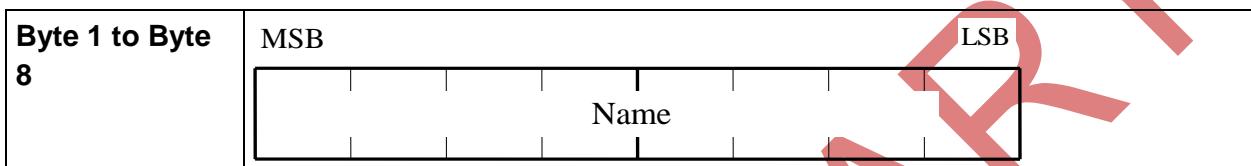
**Byte 1...8: NAME**

Example:

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x18EEFF81	0x00	0x00	0xE0	0x20	0x08	0x81	0x02	0x20	Address Claimed

#### 6.2.4.2. Cannot claim address

Message ID	18 EE FF FE
Default Priority	6
PGN	00EE00 (60928)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Peer to Peer (but becomes broadcast with DA = 0xFF)
MP (multipacket)	NO
Message direction	Transmit (ACU to ECU) and Receive (ECU to ACU)
Message length	8 bytes.
Repetition rate	N/A. Sent if address was not successfully claimed
Timeout	N/A



**Byte 1...8: NAME** (the name is read from the EEPROM and not described in this document)

Example:

ID	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Comment
0x18EEFFFE	0x00	0x00	0xE0	0x20	0x08	0x81	0x02	0x20	Cannot claim address

PRELIMARY

### 6.2.5. DM1 Active diagnostic trouble codes

Message ID	18 FE CA "SA"
Default Priority	6
PGN	00FECA (65226)
Type (Manufacturer/SAE)	SAE
Peer to Peer / Broadcast	Broadcast
MP (multipacket)	NO
Message direction	Transmit (ACU to ECU).
Message length	8 bytes
Repetition rate	1 per second Not sent on request (request is not supported)
Timeout	N/A

<b>Byte 1</b>  SAE Lamp Status	MSB	LSB				
	<table border="1"> <tr> <td>Malfunction Indicator</td> <td>Red Stop</td> <td>Amber Warning</td> <td>Protect Lamp</td> </tr> </table>	Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp	
Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp			
<b>Byte 2</b>  SAE Flash Lamp	MSB	LSB				
	<table border="1"> <tr> <td>Malfunction Indicator</td> <td>Red Stop</td> <td>Amber Warning</td> <td>Protect Lamp</td> </tr> </table>	Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp	
Malfunction Indicator	Red Stop	Amber Warning	Protect Lamp			
<b>Byte 3</b>  SPN	MSB	LSB				
	<table border="1"> <tr> <td>SPN 1 ... 8</td> </tr> </table>	SPN 1 ... 8				
SPN 1 ... 8						
<b>Byte 4</b>  SPN	MSB	LSB				
	<table border="1"> <tr> <td>SPN 9 ... 16</td> </tr> </table>	SPN 9 ... 16				
SPN 9 ... 16						
<b>Byte 5</b>  SPN / FMI	MSB	LSB				
	<table border="1"> <tr> <td>SPN 17 ... 19</td> <td>FMI</td> </tr> </table>	SPN 17 ... 19	FMI			
SPN 17 ... 19	FMI					
<b>Byte 6</b>  Reserved	MSB	LSB				
	<table border="1"> <tr> <td>0</td> <td>Occurrence count</td> </tr> </table>	0	Occurrence count			
0	Occurrence count					

## Byte 7 & 8: FFh

**Byte 1 & 2 :** For every 2 bits lamp : 00 Lamp OFF, 01 Lamp ON, 10 Reserved, 11 Reserved

**Byte 3 & 4:** Suspect Parameter Number 1..16 (LSB)

**Byte 5, Bit 6..8:** Suspect Parameter Number 17..19 (MSB)

**Byte 5, bit 1..5:** Failure Mode Identifier

**Byte 6, bit 8:** SPN conversion method (Always 0)

**Byte 6, bit 1..7:** The occurrence count is not managed in the DM1 message and is set to 127d for all SPN.

**Byte 7:** fixed FFh

**Byte 8:** fixed FFh

SPN	BYTE 1 SAE Lamp Status	BYTE 2 SAE Flash Lamp	FMI	SBZ fault code	Fault	ACU Action	ACU Restart
00h	00h	FFh	00h	00h	Normal operation		
628d (02 74 h) FLASH inconsistent	00h	FFh	2 (02h)	44d	Invalid application	No action	Can only be resolved by reprogramming a valid application

PRELIMINARY

## 6.2.6. Not supported J1939 functionality

To reduce complexity (size) of the bootloader code, it does only support a minimum of the J1939 protocol. All supported functions are listed in this document, not listed functions are not supported.

A not complete list of the limitations:

- Requests are not answered
- ACK / NACK messages are not sent by the bootloader
- The Address claim procedure is simplified (see 6.2.4)
- Commanded address not supported
- The peer-to-peer transport protocol is simplified (see 6.2.6.1)
- The DM1 message contains only information about the flash memory state and is sent periodically (see 6.2.5)
- BAM not supported

### 6.2.6.1. Reduced Transport Protocol functionality

The peer to peer transport protocol is used to reprogram a new application. To keep it as simple as possible the bootloader does add following rules which the programmer must be aware of.

- The bootloader does expect that the TP.DT packets are sent in the correct order starting with packet number 1.
- The bootloader does not support sending a CTS message to ask to resend data which were lost. The CTS is only used to confirm the first RTS of a transport (at the very beginning of Phase 3).
- The bootloader expects that the RTS message has set TotalNumberOfPackets = MaxNumberOfPackets set.

PRELIMINARY



## 7. Product validation

Feature	Value and units	Remark, comment
<b>Vibrations</b>		
Random	7.658 grms	Details : Cycle based on ISO15003
Sinus	10g peak	3 axis / 8h per axis
<b>Corrosion resistance</b>		
Salt spray test	192 h	DIN EN ISO 9227
<b>Protection class</b>		
Water /dust	IP6k9k	IEC60529 (tightened mounted on counterpart)
Gas Corrosive atmosphere	300 h	ASTM B827
Humidity	97 % RH at 55 °C	ISO15003
<b>Drop test</b>		
On concrete floor	0.45 m	IEC60068-2-32
<b>Solar radiation (UV)</b>		
	300 h	Refer to ISO 4892-2
<b>EMC, ESD</b>		
Validated for truck and off highway		More details on request Conform to CE/2009/64
<b>Temperature shocks</b>		
Shock	-40°C to 105 °C	100 cycles
<b>Mechanical shocks</b>		
Half sine puls	500 m/s <sup>2</sup> /11 ms	
<b>Altitude</b>		
	5000 m	ISO 15003

PRELIMINARY