DEEP SEA ELECTRONICS

DSEM835 Operator Manual

Document Number: 057-313

Author: Anthony Manton

Deep Sea Electronics Ltd
Highfield House
Hunmanby
North Yorkshire
YO14 0PH
ENGLAND

Sales Tel: +44 (0) 1723 890099
Sales Fax: +44 (0) 1723 893303

E-mail: sales@deepseaelectronics.com
Website: www. deepseaelectronics.com

DSEM835 Operator Manual

Revision History

Issue No.	Comments
1	Initial release of document
1.1	Added more detail about M835 specific items (font type and style, visualisation style, debug function)
2	Additional detail within *Connecting to CODESYS* section and addition of *Troubleshooting* section.
3	Update to cover M835-003-xx hardware (150 Hz Frequency Input)
3.1	Corrected typo in Freq input specification.
3.2	Added detail how to create *boot application* file for use with DSEServicetool. Corrected typos in Connection Details. Added more entries to Troubleshooting section.
3.3	Added new dimension drawing to show depth protruding from panel. Added note about batch download in several places and serial number for batch download to Specifications | Comms | CAN. Added additional IP rating data for new case housing (M835-004-01 onwards)
3.4	Updated Applicable Standards Added simple tutorial for Custom Gauges

TABLE OF CONTENTS

1 INTRODUCTION

This document details the operation and setup requirements of the DSEM835 Mobile Controller and Display, part of the DSEControl® range of products. The powerful processor coupled with the 3.5" full colour display allow the device to be utilised in many applications including J1939 engine control/display.

The manual forms part of the product and should be kept for the entire life of the product. If the product is passed or supplied to another party, ensure that this document is passed to them for reference purposes.
This is not a *controlled document.* DSE do not automatically inform on updates. Any future updates of this document are included on the DSE website at www.deepseaelectronics.com

Observe the operating instructions. Non-observance of the instructions, operation not in accordance with use as prescribed below, wrong installation or incorrect handling seriously affects the safety of operators and machinery.

A robust moulded case designed for fascia (panel) mounting houses the module. Connections are via locking plug and socket.

The controller is supplied with no application program. The equipment manufacturer is responsible for creating and managing the application program and installing it in the controller. This is achieved using CODESYS V3.5 programming. Contact DSE Technical Support for further details.

1.1 CLARIFICATION OF NOTATION

Clarification of notation used within this publication.

⚠ **NOTE:**	**Highlights an essential element of a procedure to ensure correctness.**
⚠ **CAUTION!**	**Indicates a procedure or practice, which, if not strictly observed, could result in damage or destruction of equipment.**
☠ **WARNING!**	**Indicates a procedure or practice, which could result in injury to personnel or loss of life if not followed correctly.**

1.2 GLOSSARY OF TERMS

Term	Description
Application	The application is the program that allows the DSEM835 to control the machine to which it is connected. The Application within the DSEM835 is designed and provided by the manufacturer of the complete machine (OEM).
Bootloader	The Bootloader is the program within the DSEM835 responsible for loading the Operating System and handling firmware updates.
CAN	Control Area Network. A high-speed data transmission system used extensively within the Automotive and Off-Highway industries.
CAN2.0A	CAN with 11 bit ID
CAN2.0B	CAN with 29 bit ID
CODESYS (Previously stylised as CoDeSys)	Integrated Development Environment for programming controller applications according to the international industrial standard IEC 61131-3. DSEM835 supports CODESYS V3.5
DSE	Deep Sea Electronics Ltd. www.deepseaelectronics.com
ECU	Electronic Control Unit. For example, the DSEM835 device.
EMC	Electromagnetic Compatibility. Part of the CE compliance testing for the device and is measure of the level of electromagnetic disturbance the device causes to other nearby electrical devices.
Firmware	The Firmware of the DSEM835 is the Operating System of the DSEM835 that reads and executes the Application program.
FSD	Full Scale Deflection. For example, 0 mA to 20 mA is the Full Scale Deflection of a current sink input.
I/O	Input / Output. For example, “The I/O is taken out to an external terminal strip in the user panel”.
IDE	Integrated Development Environment. For example, the CODESYS V3.5 application that runs on the host PC is an IDE.
IEC	International Electrotechnical Commission. The standards body responsible for IEC 61131-3 the standard to which CODESYS conforms.
Ixyyy	An Input, where x is the connector and yyy is the input number. For example, IA003 means Input 3 on Connector A.
J1939	High Level CAN specification dictated by S.A.E. and used by many Engine ECUs and other CAN devices.
Library	Refers to CODESYS library, a collection of items (functions, function blocks, enums, structs and more) extending the base functionality of CODESYS. A example includes the *DSE Standard* library, provided within the *DSE CODESYS Package*. This provides functions to operate the DSE Mseries range of controllers.
MSeries	Range of CODESYS devices manufactured by Deep Sea Electronics Ltd www.deepseaelectronics.com.

Term	Description
Namespace	When utilising functions from libraries a unique *Namespace* is used to identify the library within the program code. For example, *DSE Standard* library uses namespace *DSE*, meaning that all items within are referenced with *DSE.xxxx* where *xxxx* is the name of the item being referenced.
Off-Highway	An industrial vehicle used primarily “off road”. For example, construction and farm machinery. A wider interpretation includes on road access platforms, emergency vehicles and other industrial machinery, used either on the road, or off road.
O.E.M.	Original Equipment Manufacturer. The manufacturer of the overall machine that the DSEMSeries device is a part of. The OEM is also responsible for programming the device though they may subcontract this to System Integrators or Programming Houses.
PCAN-USB	USB to CAN converter interface for Windows™ PCs. See section entitled *Cables, Connectors, Harnesses and Spare Parts* elsewhere in this document for part number details.
PLC	Programmable Logic Controller. Industrial computer used primarily for the automation of electromechanical machinery.
Pin	A male or female pin connection in a housing (plug or socket).
Qxyyy	An Output, where x is the connector and yyy is the output number. For example, QA002 means Output 2 on Connector C.
Runtime	Refers to the program during execution. For example, a *Runtime error* is one that occurs during program execution that is not detected by the *Build* process. One such example is the statement `A:=B/C;` In itself there is no issue with this statement. However, if C takes on the value 0, this statement generates a *Exception Error: Divide by zero* and the application code is halted.
SAE	Society of Automotive Engineers (USA)

1.3 RELATED INFORMATION

This document refers to and is referred by the following DSE publications which are obtained from the DSE website: www.deepseaelectronics.com or by contacting DSE technical support: support@deepseaelectronics.com.

1.3.1 TECHNICAL INFORMATION

DSE Part	Description
055-266	DSEM835 Datasheet
053-250	DSEM835 Installation Instructions
057-265	DSEServicetool PC Software Manual
057-270	DSEM240 Operator Manual
057-328	DSEM5xx Keypads Operator Manual

1.4 SAFETY INSTRUCTIONS

1.4.1 GENERAL

- These instructions are for authorised persons according to the EMC and low-voltage directives. The device must be installed, connected, and put into operation by a qualified electrician.

- It is not permissible to open the controller or to modify or repair the controller. Modification or repairs to the wiring could result in dangerous malfunctions. Repairs to the controller must be performed by DSE. Contact your original equipment supplier in the case of malfunction.

- Do not insert or remove the connector when powered. Remove all sources of supply before insertion or removal.

- When the device is unpowered, ensure that no connection pins are connected to a voltage source. Thus, when the supply is switched off, the supply for the electronics, the power outputs and the external sensor supply must be switched off together.

- The customer is responsible for performing risk analysis of the mobile working machine and determining the possible safety related functions. The user is responsible for the safe function of the application programs created. If necessary, they must additionally carry out an approval test by corresponding supervisory and test organisations according to the national regulations.

- All connectors must be unplugged from the electronics during electrical welding and painting operations.

1.4.2 INSTALLATION NOTES

- Follow the instructions of the connector manufacturer, specifically with respect to preventing water from entering the device. See Section entitled *Cables, Connectors, Harnesses and Spare Parts* for details of DSE Part Numbers.

- Do not insert or remove the connector when powered. Remove all sources of supply before insertion or removal.

- To maintain IP67 rating where connectors have unused pins, ensure the use of a suitable blanking insert as recommended by the connector manufacturer.

2 SPECIFICATIONS

2.1 PROCESSOR

Description	Specification
Type	STM32h743
Speed	400 MHz

2.2 MEMORY

Description	Specification
Flash	8184 kB available for user CODESYS application
RAM	1 MB total 288 kB for CODESYS including application variables and fonts.
Non-Volatile FRAM	16 kB for device settings and application persistent variables

2.3 DC SUPPLY

Description	Specification
Operating Voltage (Pin A7)	8 V to 32 V
Maximum Current (Full Backlight, no External Loads)	<80 mA at 24 V <150 mA at 12 V
LCD Backlight Operation	Backlight switches off when Operating Voltage < 6 V and switches on when operating voltage > 8 V.

2.3.1 FUSING

Description	Specification
DC Supply (Pin A7) Supplies DSEM835 and High Current Outputs Fuse as Required by Output Loads (Pins A14, A15, A16)	3 A Min (to supply DSEM835)

2.4 ENVIRONMENTAL

Description	Specification	
Operating Temperature	-40 ºC to +85 ºC (-40 ºF to 185 ºF)	
Storage Temperature	-40 ºC to +85 ºC (-40 ºF to 185 ºF)	
Degrees of Protection Provided by Enclosure (With All Mating Connectors Fitted)	DSEM835-004-01 onwards	IP67 (NEMA 6)
	DSEM835-001-01 DSEM835-002-01 DSEM835-003-01	IP65 Front IP41 Rear

2.5 USER INTERFACE

2.5.1 CONTROLS

NOTE: For applications requiring additional control buttons, third party CAN Keypads are available. Contact support@deepseaelectronics.com if you require further advice.

Description	Specification
Push Buttons	5

2.5.2 DISPLAY

NOTE: LCD Backlight switches off when Operating Voltage < 6 V and switches on when Operating Voltage > 8 V.

Description	Specification
Size (Across Diagonal)	88.9 mm (3.5”)
Size (W x H)	320 px X 240 px
Type	Optically Bonded TFT
Lifetime	> 30,000 hours
Colour	24-bit

2.6 INPUTS

2.6.1 DIGITAL INPUTS

NOTE: Ensure Maximum and Minimum Thresholds are below 10 V as measurement above approx. 12 V is not supported).

2.6.1.1 DIGITAL

Description	Specification
Applicable Pins	Pins A5, A6, A12, A18
Maximum Measurable Voltage of Input Voltages above Maximum Measurable Voltage are read as approx. 12 V.	Typical 12 V
Minimum Voltage for High Level	Configurable (0 V to 10 V) Recommended 6 V
Maximum Voltage for Low Level	Configurable (0 V to 10 V) Recommended 2 V

2.6.1.2 FREQUENCY

NOTE: Frequency input functionality available from firmware version V1.1. Consult table below for Maximum Frequency Measurement as it varies between device hardware versions.

Description	Specification
Applicable Pin	Pin A6
Minimum Frequency Measurement	1 Hz
Maximum Frequency Measurement DSEM835-003-xx onwards	150 Hz
Maximum Frequency Measurement DSEM835-001-01 and DSEM835-002-01	50 Hz
Minimum Voltage for Frequency Measurement	8 V
Maximum Measurable Voltage of Input Voltages above Maximum Measurable Voltage are read as approx. 12 V.	Typical 12 V
Maximum Voltage of Frequency Signal	As ECU Supply Voltage Pin A7 (Max 32 V)
Duty Cycle Range for Frequency Measurement	25 % to 75 % Typical See Below for Detail Table.
Minimum Rise / Fall Time for Frequency Measurement	1.25 ms @ 12 V 0.5 ms @ 24 V

Duty Cycle Requirements (DSEM835-003-xx onwards only)

Input Voltage	Input Frequency	Min Duty Cycle	Typical Duty Cycle	Max Duty Cycle
8 V	150 Hz	25 %	50 %	75 %
12 V	150 Hz	15 %	50 %	85 %
24 V	150 Hz	10 %	50 %	90 %
32 V	150 Hz	5 %	50 %	95 %

2.6.2 ANALOGUE INPUTS

2.6.2.1 VOLTAGE

Description	Specification
Applicable Pins	Pins A5, A6, A12, A18
Range	0 V to 10 V
Input Resistance	>=7.5 kΩ
Sampling Rate	500 Hz
Resolution	12 bits
Accuracy	±1 % F.S.D. ±0.1 V

2.6.2.2 CURRENT

Description	Specification
Applicable Pins	Pins A5, A6, A12, A18
Configurable Ranges	0 mA to 20 mA 4 mA to 20 mA
Input Type	Current sink only
Input Sink Resistance	150 Ω ± 1%
Sampling Rate	500 Hz
Resolution	12 bits
Accuracy	± 1 % Full Scale Deflection 0.2 mA

2.6.2.3 RESISTIVE

Description	Specification
Applicable Pins	Pins A5, A6, A12, A18
Measurement Range	0 Ω to 3400 Ω
Measurement Source Voltage	12 V maximum
Measurement Source Current	3 mA
Sampling Rate	500 Hz
Resolution	12 bits
Accuracy	± 1 % Full Scale Deflection 34 Ω

2.7 OUTPUTS

NOTE: Outputs have integral *freewheel* (also known as *flyback*, *snubber*, *clamp* etc) diodes to protect the DSE device, however if elevated levels of back E.M.F. are generated by the de-energising of the external circuit, it is good practice to quench this at source. This too prevents it being induced into other cables where it may affect other circuits such as CAN communications or other devices entirely.

2.7.1 POSITIVE SWITCHING

Description	Specification
Applicable Pins	Pins A14, A15, A16
Maximum Current	1 A resistive
Digital Output Active Low 'ON' State Maximum Voltage at Rated Current	<100 mV
Digital Output Active Low 'OFF' State Leakage Current	<10 µA at 24 V output supply

2.7.2 VREF OUTPUT

VREF output is a configurable voltage output source, intended for use as a voltage reference and/or voltage source for input devices. Care must be taken not to exceed the maximum current rating.

Description	Specification
Applicable Pins	Pin A17
Voltage	5 V / 10 V selectable
Maximum Current	100 mA
Voltage Accuracy	±5 %

2.8 COMMUNICATIONS

2.8.1 CAN

NOTE: CANopen protocol is not provided directly by DSEM835. Application programming must be used to handle the messages of CANopen. Contact DSE Technical Support for additional advice.

NOTE: Software configurable termination resistor is internally fitted (factory setting: disabled). A complete CAN network must have 120 Ω terminators at each end of the network.

NOTE: Screened 120 Ω impedance cable specified for use with CAN must be used for the CAN links.
DSE stock and supply Belden cable 9841 which is a high quality 120 Ω impedance cable suitable for CAN use (DSE part number 016-030).

NOTE: DSEM835 sends CODESYS detection messages at device startup. To disable this, set *ID* to zero as detailed in the section entitled *Device Settings* elsewhere in this document.

NOTE: For details using DSEServicetool PC Software, including for Batch Programming of ten devices at the same time, see DSE publication *057-265 DSE Servicetool PC Software Manual.*

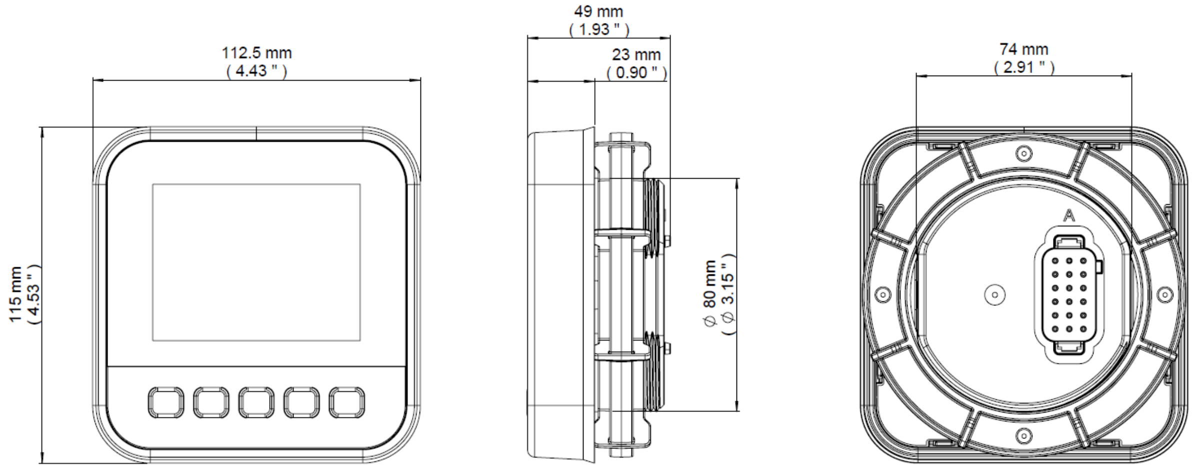
Description	Specification
Number of CAN Interfaces	1
Supported Protocols	J1939 Raw CAN (CAN2.0A or CAN2.0B) CODESYS connection for application download and debugging.
Supported Baud Rates (Factory Setting is 250 kbit/s)	50 kbit/s, 100 kbit/s, 125 kbit/s, 250 kbit/s, 500 kbit/s, 800 kbit/s, 1 Mbit/s
Internal Termination Resistor (Factory Setting is OFF)	Configurable between *On* and *Off* using *Device Settings* from within CODESYS or by user application code using *DSE Standard* library (namespace DSE).
Batch Download Capability Using DSEServicetool PC Software	From serial number 8958811 onwards. For details, see DSE publication *057-265 DSE Servicetool PC Software Manual.*

3 INSTALLATION

3.1 DIMENSIONS AND MOUNTING

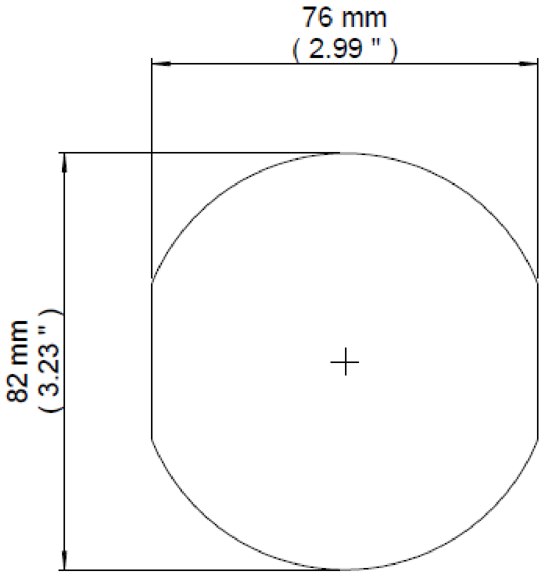
3.1.1 DIMENSIONS

NOTE: Prior to part number M835-004-01, depth protruding from panel was approximately 20 mm. From M835-004-01 onwards, depth protruding from panel is 23 mm as shown below.



3.1.2 PANEL CUTOUT

82 mm (3.23 “) hole is suitable. Maximum Panel Thickness: 6 mm (0.24 “)
If a punch or milling machine is available, adding ‘flats’ at 74 mm (2.91 “) spacing serves to prevent rotation of the device in the panel cut-out.



3.1.3 WEIGHT

<1 kg
(<35.3 oz)

3.1.4 FIXING

> ⚠**NOTE: In conditions of excessive vibration, mount the housing on suitable anti-vibration mountings.**

The device is held into the panel fascia using the large nut provided.

- Place the controller into the panel aperture.
- Attached the supplied fixing nut and hand tighten to provide adequate fixing (see below).
- Attach the connector plug (Plug A) and push home to click into place.

3.1.4.1 TORQUE SETTING OF FIXING NUT

8 Nm
5.9 ft-lb

3.2 APPLICABLE STANDARDS

<table>
<tr><th>Category</th><th colspan="2">Description</th><th>Standard</th></tr>
<tr><td>Minimum Temperature</td><td colspan="2">-40 °C (-40 °F)</td><td>BS EN 60068-2-1</td></tr>
<tr><td>Maximum Temperature</td><td colspan="2">+85 °C (185 °F)</td><td>BS EN 60068-2-2</td></tr>
<tr><td rowspan="3">CE marking</td><td colspan="2">Electromagnetic compatibility (EMC) noise immunity</td><td>EN 61000-6-2:2005</td></tr>
<tr><td colspan="2">Electromagnetic compatibility (EMC) emission standard</td><td>EN 61000-6-4:2007 +A1:2011</td></tr>
<tr><td colspan="2">Safety of information technology equipment, general requirements</td><td>BS EN 61010-1:2010</td></tr>
<tr><td rowspan="3">EMC for Road Vehicles</td><td colspan="2">Component test methods for electrical disturbances from narrowband radiated electromagnetic energy – Part 2 - 100 V/m</td><td>ISO 11452-2</td></tr>
<tr><td colspan="2">Component test methods for electrical disturbances from narrowband radiated electromagnetic energy – Part 4 – BCI 100mA</td><td>ISO 11452-4</td></tr>
<tr><td colspan="2">Road vehicles – Electrical disturbances from conduction and coupling</td><td>BS ISO 7637-3</td></tr>
<tr><td rowspan="5">Climatic tests</td><td colspan="2">Cold testing – Low temperature storage</td><td>BS EN 60068-2-1</td></tr>
<tr><td colspan="2">Dry heat – Max temperature storage</td><td>BS EN 60068-2-2</td></tr>
<tr><td colspan="2">Dry heat – Max operating temperature</td><td>BS EN 60068-2-2</td></tr>
<tr><td colspan="2">Damp heat testing – High humidity surface condensation</td><td>BS EN 60068-2-30</td></tr>
<tr><td colspan="2">Damp heat, steady state</td><td>BS EN 60068-2-78</td></tr>
<tr><td rowspan="3">Vibrations tests</td><td colspan="2">Vibration (Sinusoidal)</td><td>BS EN 60068-2-6</td></tr>
<tr><td colspan="2">Test Fh: Vibration, broadband random and guidance</td><td>IEC 60068-2-64</td></tr>
<tr><td colspan="2">Shock resistance</td><td>BS EN 60068-2-27</td></tr>
<tr><td>Electrical</td><td colspan="2">Road vehicles – Electrical loads</td><td>ISO 16750-2</td></tr>
<tr><td rowspan="2">Water and Dust</td><td>M835-004-01 onwards:</td><td>IP67 front and rear (NEMA 6)</td><td rowspan="2">BS EN 60529</td></tr>
<tr><td>M835-001-01
M835-002-01
M835-003-01</td><td>IP65 Front
IP21 Rear</td></tr>
</table>

In line with our policy of continual development, Deep Sea Electronics reserve the right to change specification without notice.

3.2.1 ENCLOSURE CLASSIFICATIONS

3.2.1.1 IP CLASSIFICATIONS

NOTE: To maintain IP67 rating where connectors have unused pins, ensure the use of a suitable blanking insert as recommended by the connector manufacturer.

The specification under *BS EN 60529 Degrees of protection provided by enclosures* is IP67 as below (DSEM835-004-01 onwards).

First Digit Protection against contact and ingress of solid objects		Second Digit Protection against ingress of water	
0	No protection	0	No protection
1	Protected against ingress solid objects with a diameter of more than 50 mm. No protection against deliberate access, e.g., with a hand, but large surfaces of the body are prevented from approach.	1	Protection against dripping water falling vertically. No harmful effect must be produced (vertically falling drops).
2	Protected against penetration by solid objects with a diameter of more than 12 mm. Fingers or similar objects prevented from approach.	2	Protection against dripping water falling vertically. There must be no harmful effect when the equipment (enclosure) is tilted at an angle up to 15° from its normal position (drops falling at an angle).
3	Protected against ingress of solid objects with a diameter of more than 2.5 mm. Tools, wires etc. with a thickness of more than 2.5 mm are prevented from approach.	3	Protection against water falling at any angle up to 60° from the vertical. There must be no harmful effect (spray water).
4	Protected against ingress of solid objects with a diameter of more than 1 mm. Tools, wires etc. with a thickness of more than 1 mm are prevented from approach.	4	Protection against water splashed against the equipment (enclosure) from any direction. There must be no harmful effect (splashing water).
5	Protected against harmful dust deposits. Ingress of dust is not totally prevented but the dust must not enter in sufficient quantity to interface with satisfactory operation of the equipment. Complete protection against contact.	5	Protection against water projected from a nozzle against the equipment (enclosure) from any direction. There must be no harmful effect (water jet).
6	Protection against ingress of dust (dust tight). Complete protection against contact.	6	Protection against heavy seas or powerful water jets. Water must not enter the equipment (enclosure) in harmful quantities (splashing over).
		7	No ingress of water in harmful quantity when the enclosure is immersed in water under defined conditions of pressure and time (up to 1 m of submersion)

3.3 FUSING

The individual electric circuits must be protected in order to protect the whole system. Select appropriate fuses to protect the outputs being supplied.

Pin	Description	Comments	Recommended Fuse Size
A7	ECU Supply	Supplies DSEM835 CPU and Outputs	3 A to 10 A Max

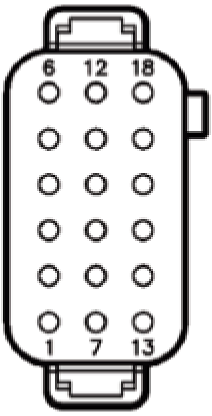
3.4 USER CONNECTIONS

NOTE: For suitable connection looms from DSE, see sections entitled *DSEM835 Connector Harness* and *DSEM835 Configuration Harness*.

NOTE: If a prewired connection cable is used, remove the cores with unused signal inputs and outputs. Unused cores, in particular core loops, lead to interference coupling that can influence the connected controller.

NOTE: Screened 120 Ω impedance cable specified for use with CAN must be used for the CAN links.
DSE stock and supply Belden cable 9841 which is a high quality 120 Ω impedance cable suitable for CAN use (DSE part number 016-030).

NOTE: Terminals A11, and A13 are internally connected to A1.

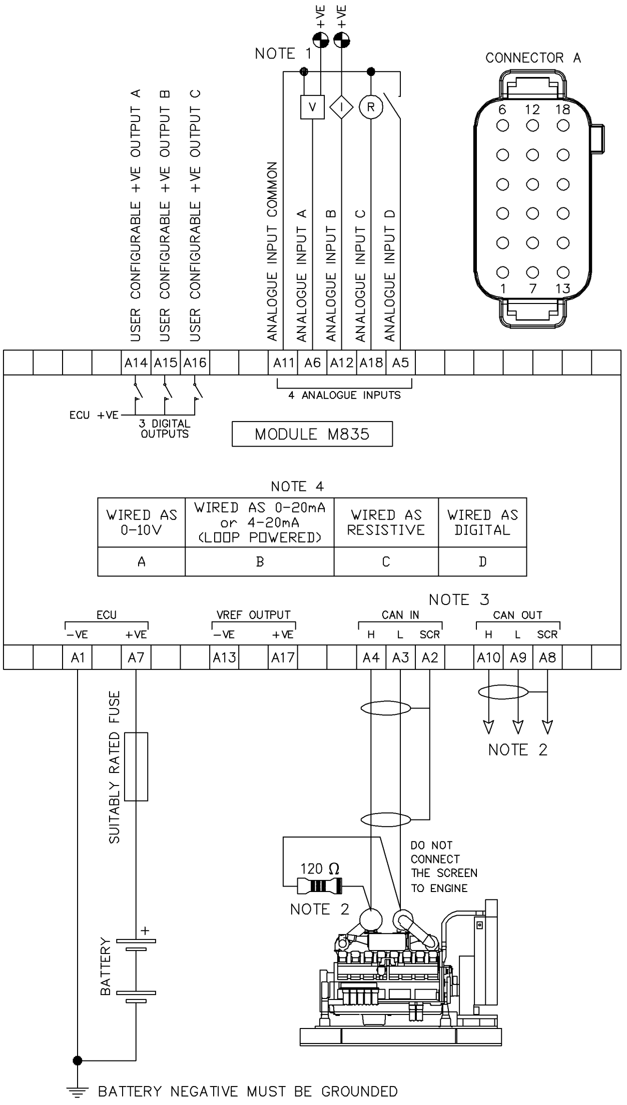
	Pin	Description	Notes
	A1	DC Plant Supply Input (Negative)	Battery negative supply for the device.
	A2	CAN SCR	Connect to Engine ECU CAN SCR
	A3	CAN L (in)	Connect to Engine ECU CAN L
	A4	CAN H (in)	Connect to Engine ECU CAN L
	A5	Input D	Analogue / Digital input D (IA004)
	A6	Input A (	Analogue / Frequency / Digital input A IA001)
	A7	DC Plant Supply input (Positive)	Battery positive supply for the device.
	A8	CAN SCR	Optional to connect to additional CAN devices. Internally connected to A2
	A9	CAN L (out)	Optional to connect to additional CAN devices. Internally connected to A3
	A10	CAN H (out)	Optional to connect to additional CAN devices. Internally connected to A4
	A11	Input Common GND	Provides ground reference to all inputs (A5, A6, A12, A18)
	A12	Input B	Analogue / Digital input B (IA002)
	A13	VREF Negative	Negative connection for the VREF Output voltage.
	A14	Output A	(QA001) Max 1 A at the voltage applied to A7
	A15	Output B	(QA002) Max 1 A at the voltage applied to A7
	A16	Output C	(QA003) Max 1 A at the voltage applied to A7
	A17	VREF Positive OUT	Positive connection for the VREF Output voltage.
	A18	Input C	Analogue / Digital input C (IA003)

3.5 TYPICAL WIRING DIAGRAM

> **NOTE: As systems have differing requirements, these diagrams show only a typical system and do not intend to show a complete system.**

> **NOTE: This diagram shows connection to a Negative Earth system. For Positive Earth and Floating Earth systems, see section entitled *Earth Systems* elsewhere in this document.**

> **NOTE: Screened 120 Ω impedance cable specified for use with CAN must be used for the CAN links.**
> **DSE stock and supply Belden cable 9841 which is a high quality 120 Ω impedance cable suitable for CAN use (DSE part number 016-030).**



WIRED AS 0-10V	WIRED AS 0-20mA or 4-20mA (LOOP POWERED)	WIRED AS RESISTIVE	WIRED AS DIGITAL
A	B	C	D

NOTE 1. ANALOGUE INPUT COMMON MUST NOT BE GROUNDED, IT IS LINKED INTERNALLY TO TERMINAL A1.

NOTE 2. 120 Ω TERMINATING RESISTOR MAY BE REQUIRED EXTERNALLY, SEE EXTERNAL EQUIPMENT MANUFACTURERS LITERATURE FOR FURTHER DETAILS.

NOTE 3. CAN IN AND CAN OUT ARE LINKED INTERNALLY.

NOTE 4. ANALOGUE INPUTS CAN BE CONFIGURED AS EITHER A DIGITAL INPUT, RESISTIVE INPUT, 0–10V INPUT, 0–20mA, 4–20mA INPUT, OR ANY COMBINATION OF THE ABOVE.

3.5.1 EARTH SYSTEMS

3.5.1.1 NEGATIVE EARTH

The typical wiring diagrams located within this document show connections for a negative earth system (the battery negative connects to Earth).

3.5.1.2 POSITIVE EARTH

When using a DSE module with a Positive Earth System (the battery positive connects to Earth), the following points must be followed:

Follow the typical wiring diagram as normal for all sections ***except*** the earth points.
All points shown as Earth on the typical wiring diagram should connect to ***battery negative*** (not earth).

3.5.1.3 FLOATING EARTH

Where neither the battery positive nor battery negative terminals are connected to earth the following points must be followed:

Follow the typical wiring diagram as normal for all sections ***except*** the earth points.
All points shown as Earth on the typical wiring diagram should connect to ***battery negative*** (not earth).

3.5.2 TYPICAL ARRANGEMENT OF CAN LINK

NOTE: As systems have differing requirements, these diagrams show only a typical system and do not intend to show a complete system.

NOTE: Screened 120 Ω impedance cable specified for use with CAN must be used for the CAN connection.

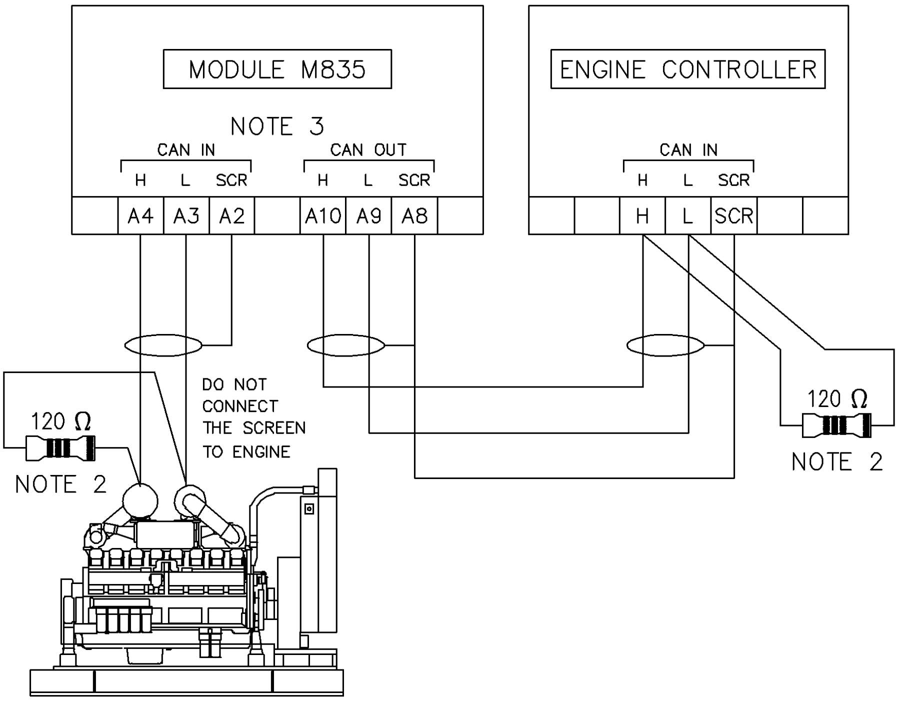
DSE stock and supply Belden cable 9841 which is a high quality 120 Ω impedance cable suitable for CAN use (DSE part number 016-030)

NOTE: A termination resistor MUST be fitted to the first and last unit on the CAN link. An internal 120 Ω termination resistor is fitted. This is configurable within CODESYS *Device Settings.*

NOTE: For details regarding Batch Programming of ten devices at the same time, see DSE publication *057-265 DSE Servicetool PC Software Manual.*

The following image shows a typical connection between the DSEM835 and an engine ECU. Additionally, an *Engine Controller* is included to show how to connect another device on the same CAN. For example, this could be a DSE Genset Controller or a DSEM240 CAN Expansion device providing additional Inputs/Outputs.

While many applications require connection to an engine ECU or other CAN device this is not a requirement for use of the DSEM835 and is included as an example of one typical use only.



NOTE 2. 120 Ω TERMINATING RESISTOR MAY BE REQUIRED EXTERNALLY, SEE EXTERNAL EQUIPMENT MANUFACTURERS LITERATURE FOR FURTHER DETAILS.

NOTE 3. CAN IN AND CAN OUT ARE LINKED INTERNALLY.

4 CONNECTING TO CODESYS

NOTE: DSEM835 supports CODESYS version 3.5.12.0. Ensure Compiler version, Visualisation version and the versions of any 3S libraries used in the project are no later than version 3.5.12.0. Contact support@deepseaelectronics.com if assistance is required.

NOTE: DSE Stock and supply PCAN-USB IPEH-002021 from Peak Systems (https://www.peak-system.com). DSE Part number 016-179. Contact sales@deepseaelectronics.com.

NOTE: DSEM835 supports CODESYS TargetVisuLight (TVL) and does not support CODESYS VisuElems. See section entitled *Using the Display in the Project* elsewhere in this document for guidance in using the features of TVL.

DSEM835 communicates with, and is programmed by, the CODESYS V3.5 Integrated Development Environment (IDE).

To connect the device to a PC, a CAN interface is required as detailed above.

4.1 INSTALL THE PCAN DRIVER

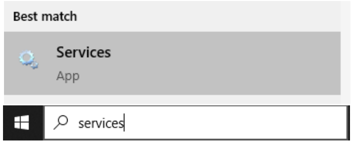
Before proceeding you must install the PCAN Driver supplied with the PCAN Interface, or downloaded from https://www.peak-system.com/

4.2 SETUP THE CODESYS GATEWAY

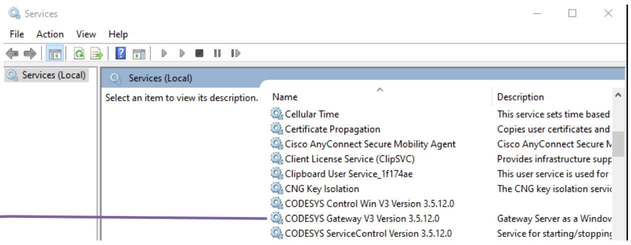
CODESYS Gateway requires configuration to add CAN support.
DSE provide the necessary configuration within *Gateway.cfg,* that is available at www.deepseaelectronics.com. This file must be downloaded and used to replace the existing file already in use on the PC where CODESSYS is installed.
The location of the file varies depending upon your PC setup. Where only one version CODESYS IDE is installed, this is within the *GatewayPLC* folder of that installation. For PCs that have multiple CODESYS IDE versions installed, use the below method to locate the file.

4.2.1 LOCATING WHERE TO STORE THE GATEWAY CONFIGURATION FILE

- In Windows, use search to locate and open the Windows Services application.



- Click Name (to sort by Name) and then scroll down to locate CODESYS Gateway. The version number shown differs depending upon the last version of CODESYS you installed on the PC



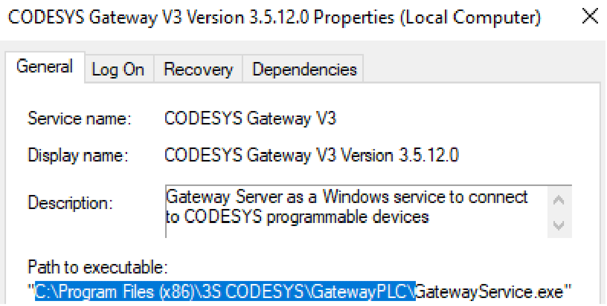
Select an item to view its description.

Name	Description
Cellular Time	This service sets time based
Certificate Propagation	Copies user certificates and
Cisco AnyConnect Secure Mobility Agent	Cisco AnyConnect Secure M
Client License Service (ClipSVC)	Provides infrastructure supp
Clipboard User Service_1f174ae	This user service is used for
CNG Key Isolation	The CNG key isolation servic
CODESYS Control Win V3 Version 3.5.12.0	
CODESYS Gateway V3 Version 3.5.12.0	Gateway Server as a Windov
CODESYS ServiceControl Version 3.5.12.0	Service for starting/stopping

CODESYS Gateway

Continued Overleaf…

- *Right-Click* the CODESYS Gateway and select *Properties.*
- *Path to executable* shows the location of the GatewayService File.
- Use the mouse to select only the path.
- Ensure to omit the leading " and omit the trailing *GatewayService.exe*
- *Right-Click* the selection and select *Copy* (or press *CTRL-C).*
- Open *Windows File Explorer* and *Paste* the path just copied.
- This opens *File Explorer* showing the contents.
- Copy the new *Gateway.cfg* file into this folder, overwriting the old one. If you prefer you can take a backup of the old one should you wish to revert to it in future.



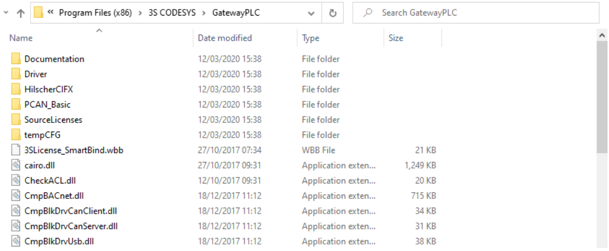
CODESYS Gateway V3 Version 3.5.12.0 Properties (Local Computer)

Service name: CODESYS Gateway V3

Display name: CODESYS Gateway V3 Version 3.5.12.0

Description: Gateway Server as a Windows service to connect to CODESYS programmable devices

Path to executable:
"C:\Program Files (x86)\3S CODESYS\GatewayPLC\GatewayService.exe"

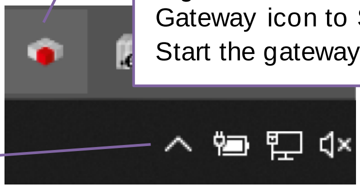


« Program Files (x86) › 3S CODESYS › GatewayPLC

Search GatewayPLC

Name	Date modified	Type	Size
Documentation	12/03/2020 15:38	File folder	
Driver	12/03/2020 15:38	File folder	
HilscherCIFX	12/03/2020 15:38	File folder	
PCAN_Basic	12/03/2020 15:38	File folder	
SourceLicenses	12/03/2020 15:38	File folder	
tempCFG	12/03/2020 15:38	File folder	
3SLicense_SmartBind.wbb	27/10/2017 07:34	WBB File	21 KB
cairo.dll	27/10/2017 09:31	Application exten...	1,249 KB
CheckACL.dll	12/10/2017 09:31	Application exten...	20 KB
CmpBACnet.dll	18/12/2017 11:12	Application exten...	715 KB
CmpBlkDrvCanClient.dll	18/12/2017 11:12	Application exten...	34 KB
CmpBlkDrvCanServer.dll	18/12/2017 11:12	Application exten...	31 KB
CmpBlkDrvUsb.dll	18/12/2017 11:12	Application exten...	38 KB

- The Gateway must now be stopped and restarted. You can either do this from within *Services*, or you can follow the instructions detailed.

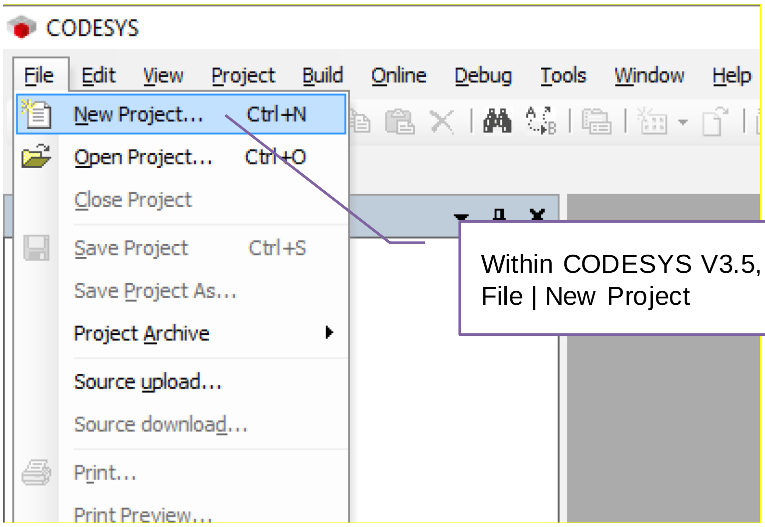


Right-Click the CODESYS Gateway icon to Stop and Start the gateway.

Click the UP arrow at the bottom right of Windows™ to access the system tray icons.

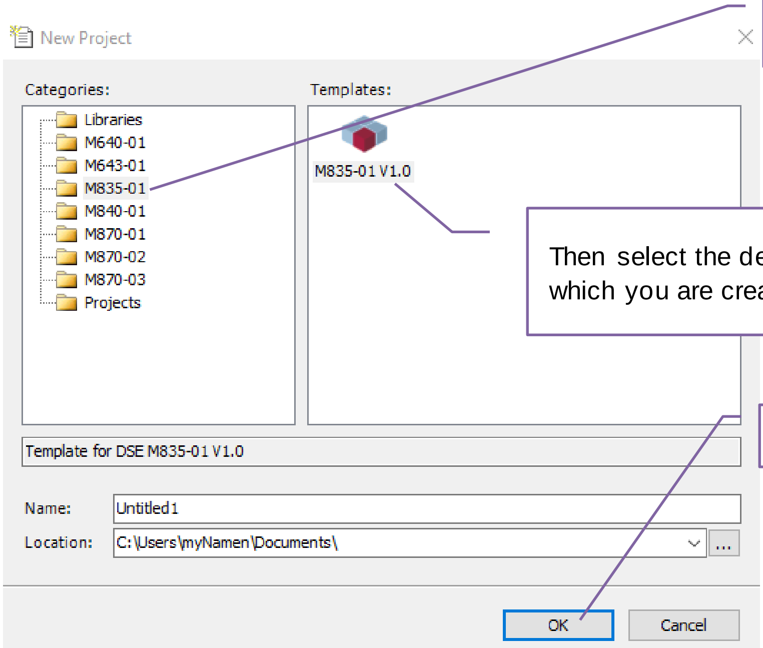
4.3 START NEW PROJECT

To begin, start a new project as shown with the supplied template.



Within CODESYS V3.5, select File | New Project

The default project is created from the template:



Template for DSE M835-01 V1.0

Name: Untitled1

Location: C:\Users\myNamen\Documents\

Select *M835*

Then select the device version for which you are creating the project.

Then click OK

4.4 CAN CONNECTION

NOTE: Ensure previous subsections *Install the PCAN Driver* and *Setup the CODESYS Gateway* have been followed before attempting *CAN Connection*.

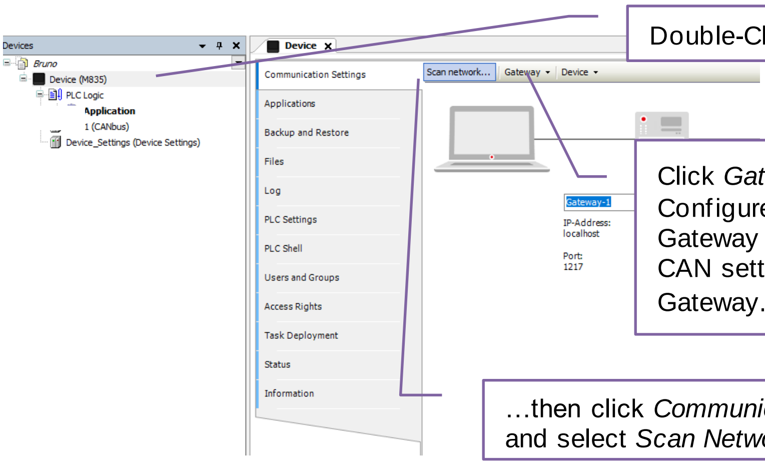
NOTE: DSE Stock and supply PCAN-USB IPEH-002021 from Peak Systems (https://www.peak-system.com). DSE Part number 016-179. Contact sales@deepseaelectronics.com.

NOTE: Ensure the Gateway is set to operate the CAN port at the same rate that the DSEM835 is set to. Factory Setting is 250 kbit/s though this may have been changed by the OEM or system programmer.

NOTE: Before writing the application to the DSEM835, power cycle the device. This ensures memory is cleared ready to accept the new application. Failure to do this may result in a failed application write, requiring this process to be repeated.

NOTE: Ensure CODESYS Gateway is running. For details, see section entitled *Stopping and Starting the Gateway* elsewhere in this manual.

Select *Device* | *Communication Settings* in the CODESYS V3.5 IDE:



Double-Click *Device…*

Click *Gateway* and then Configure the Local Gateway to check the CAN settings of the Gateway.

…then click *Communication Settings* and select *Scan Network.*

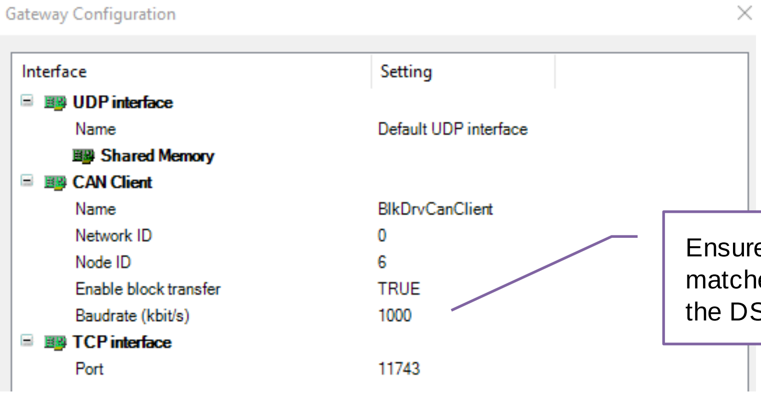
4.4.1 CONFIGURE THE LOCAL GATEWAY

NOTE: If *CAN Client* is not visible within the *Gateway Configuration* then CODESYS is not correctly setup for use with the CAN device. Ensure the latest *DSE CODESYS PACKAGE* is installed. For additional support, contact support@deepseaelectronics.com.

NOTE: Where the baud rate of the device is not known, try all available options. Where no connection is made, use DSEServicetool PC Software to set the device back to factory settings (250 kbit/s). See DSE Publication *057-265 DSEServicetool PC Software Manual* for further details.

NOTE: After changing *Gateway Configuration* settings, the gateway must be stopped and restarted before changes take effect.

NOTE: If only the *UDP interface* is showing (*CAN Client* is missing), replace *Gateway.cfg* with a new one as detailed in the section entitled *Setup the CODESYS Gateway* elsewhere in this document.



Gateway Configuration

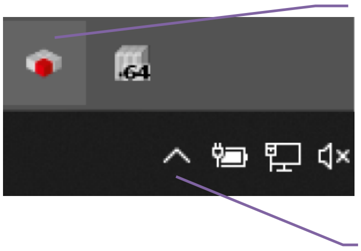
Interface	Setting
UDP interface	
Name	Default UDP interface
Shared Memory	
CAN Client	
Name	BlkDrvCanClient
Network ID	0
Node ID	6
Enable block transfer	TRUE
Baudrate (kbit/s)	1000
TCP interface	
Port	11743

Ensure *Baudrate* matches the setting of the DSEM835.

4.4.1.1 STOPPING AND STARTING THE GATEWAY

NOTE: After changing *Gateway Configuration* settings, the gateway must be stopped and restarted before changes take effect.

NOTE: Remember to start the gateway again if you previously stopped it to enable CAN connection to DSEServicetool. For details of DSEServicetool PC Software see DSE Publication *057-265 DSEServicetool PC Software Manual.*

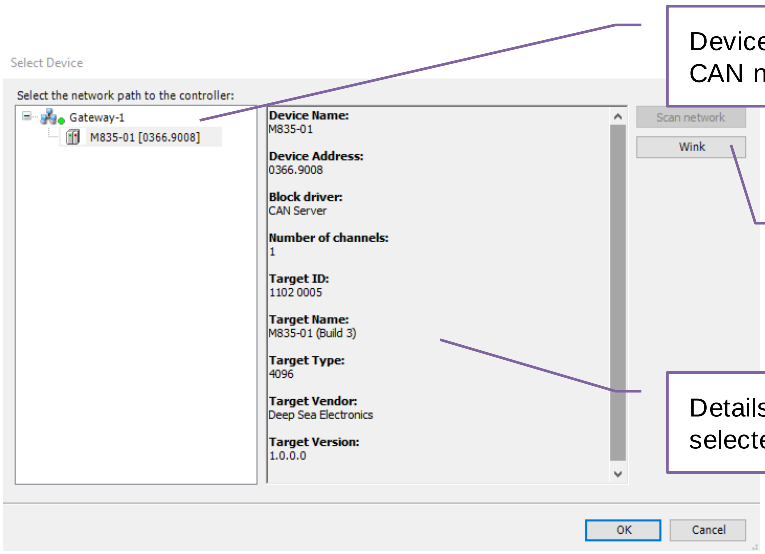


Right-Click the CODESYS Gateway icon to Stop and Start the gateway.

Click the UP arrow at the bottom right of Windows™ to access the system tray icons.

4.4.2 SCAN NETWORK

> ⚠ **NOTE: Before writing the application to the DSEM835, power cycle the device. This ensures memory is cleared ready to accept the new application. Failure to do this may result in a failed application write, requiring this process to be repeated.**



Select Device

Select the network path to the controller:

- Gateway-1
  - M835-01 [0366.9008]

Device Name:
M835-01

Device Address:
0366.9008

Block driver:
CAN Server

Number of channels:
1

Target ID:
1102 0005

Target Name:
M835-01 (Build 3)

Target Type:
4096

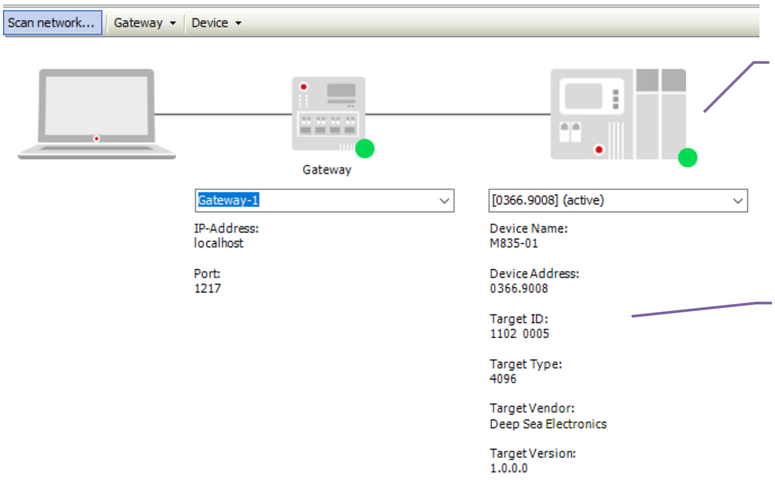
Target Vendor:
Deep Sea Electronics

Target Version:
1.0.0.0

Devices found on the CAN network.

Wink is not supported by DSEM835.

Details about the selected device.



Gateway-1

IP-Address:
localhost

Port:
1217

[0366.9008] (active)

Device Name:
M835-01

Device Address:
0366.9008

Target ID:
1102 0005

Target Type:
4096

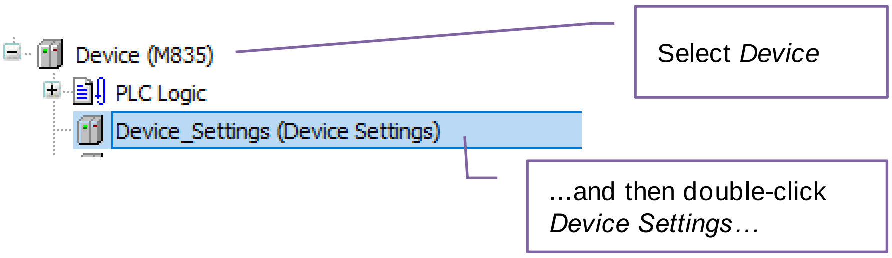
Target Vendor:
Deep Sea Electronics

Target Version:
1.0.0.0

Successful connection is shown by the green icon.

Details of the connected device.

4.5 DEVICE SETTINGS

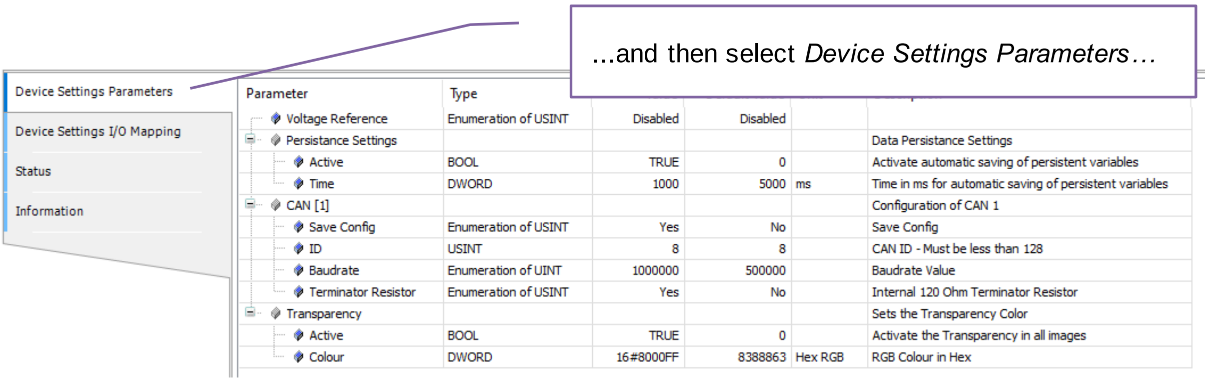


Select *Device*

…and then double-click *Device Settings…*

See the following subsections for details of the Device Settings pages.

4.5.1 DEVICE SETTINGS PARAMETERS



…and then select *Device Settings Parameters…*

Parameter	Type				
Voltage Reference	Enumeration of USINT	Disabled	Disabled		
Persistance Settings					Data Persistance Settings
Active	BOOL	TRUE	0		Activate automatic saving of persistent variables
Time	DWORD	1000	5000	ms	Time in ms for automatic saving of persistent variables
CAN [1]					Configuration of CAN 1
Save Config	Enumeration of USINT	Yes	No		Save Config
ID	USINT	8	8		CAN ID - Must be less than 128
Baudrate	Enumeration of UINT	1000000	500000		Baudrate Value
Terminator Resistor	Enumeration of USINT	Yes	No		Internal 120 Ohm Terminator Resistor
Transparency					Sets the Transparency Color
Active	BOOL	TRUE	0		Activate the Transparency in all images
Colour	DWORD	16#8000FF	8388863	Hex RGB	RGB Colour in Hex

4.5.1.1 VOLTAGE REFERENCE

Parameter	Description
Voltage Reference	Selects the level of the Voltage Reference output (Pin A17). This may be used to supply external sensors or other auxiliary devices. ***Disabled:* Vref is disabled** ***5V:*** Output is 5 V DC. ***10V:*** Output is 10 V DC.

4.5.1.2 PERSISTENCE SETTINGS

NOTE: For details of *Persistent Variables*, see section entitled *Connecting to CODESYS | Using Persistent Variables in the Project* elsewhere in this document.

Parameter	Description
Active	***FALSE:*** Persistent Variables are not saved. ***TRUE:*** Persistent Variables are cyclically saved at the period of *Time.*
Time	Applicable only when *Active* is set to *TRUE.* Select the period (milliseconds) of the cyclic saving of Persistent Vars. ***Minimum Value:*** 1000 ms. Values below this must not be used. This allows time for the device to write to the Non-Volatile memory block. ***Recommended Value:*** 1000 ms to 2000 ms.

4.5.1.3 CAN

⚠NOTE: DSEM835 has one CAN port. This is referred to as *Network0*. With DSE_CAN library the enum *DSE_CAN.eCANList.CAN1* is available.

Parameter	Description
Save Config	**⚠NOTE: After changing *CAN* settings, the device must be power cycled before changes take effect.** ***No:*** Changes to the CAN are not applied when the application is sent to the device. ***Yes:*** Changes to CAN settings are saved to the device. Powercycle the device after the application is sent to apply the new settings.
ID	**⚠NOTE: Setting this to zero (0) or above 128 disables CAN communication with CODESYS. To re-enable use DSEServicetool PC software to reset the device to factory settings.** ID of the device on the CAN network used during application transfer (ie CODESYS connection)
Baudrate	Baudrate of the device on the CAN network used during application transfer and application communcation with CAN (ie Engine ECUs and other external devices).
Terminator Resistor	**⚠NOTE: When enabled, this allows the device to act as *J1939 Type II (2) Device.* As such this device must be located at the end of the CAN. Only two devices of this type can exist on a CAN bus, one at each end.** **⚠NOTE: When enabled, the resistor is *switched in* when the device is powered up and CODESYS begins operation. Before CODESYS operation begins, the resistor is not in circuit. Hence DSEServicetool operation requires an externally terminated CAN network as CODESYS is not operating during the DSEServicetool Scan.** ***No:*** Disabled. ***Yes:*** Internal 120 Ω termination resistor is enabled on the CAN port.

4.5.1.4 TRANSPARENCY

NOTE: DSEM835 supports only uncompressed bmp (bitmap) images.

Parameter	Description
Active	***TRUE:*** Visualisation transparency for images is enabled. ***FALSE:*** Visualisation transparency for images is disabled.
Colour	Enter the RGB value of the colour to be used as the transparent part of images in the Visualisation. This setting is applied to all images within the project. ***Factory Setting:*** 8388863 (16#8000FF) It is recommended to view the hexadecimal value. For example: 16#8000FF is interpreted as follows ***Red:*** 16#80 ***Green:*** 16#00 ***Blue:***16#FF

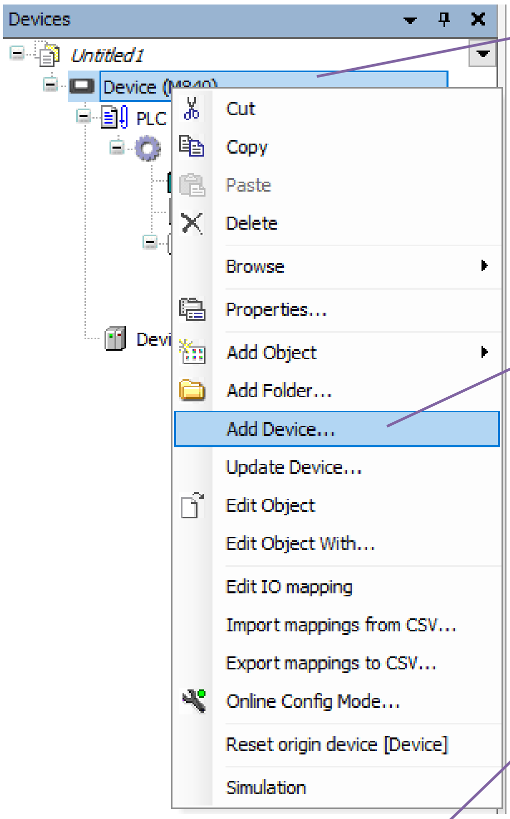
4.5.2 DEVICE SETTINGS I/O MAPPING

Allows mapping of the following parameters to program variables,

Parameter	Description
Error Code	Displays internal Error Code when available. For details, see section entitled *DSEM835 CODESYS Error Codes*.
Battery Voltage	ECU supply of the DSEM835 device.
Voltage Reference	Configuration of the Vref voltage output.
Backlight	**NOTE: LCD Backlight switches off when Operating Voltage < 6 V and switches on when Operating Voltage > 8 V.** Configuration of the LCD backlight.
Keyboard Backlight	**NOTE: Keyboard Backlight switches off when Operating Voltage < 6 V and switches on when Operating Voltage > 8 V.** Configuration of the backlight of the fascia button surround.

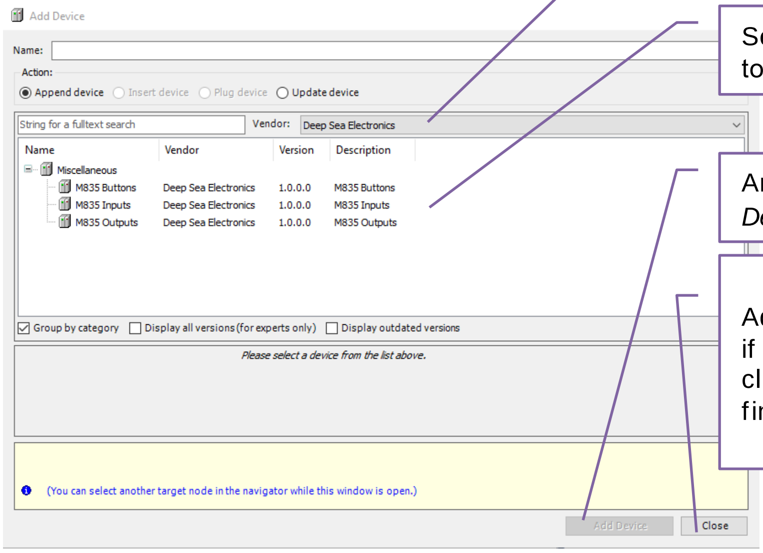
4.6 ADD INPUTS, OUTPUTS AND BUTTONS TO THE PROJECT

> **NOTE: Many programmers prefer to use the Inputs, Outputs and Buttons programmatically (using code). This is achieved using the library *DSE Standard* (Namespace: DSE). Additionally, *M835_Buttons* function is available within the library *DSE Utils* (Namespace: DSE_UTILS. Both libraries are available within the *DSE CODESYS Package* from www.deepseaelectronics.com.**



Right-Click *Device…*

And select *Add Device.*



Name:

Action:

● Append device ○ Insert device ○ Plug device ○ Update device

String for a fulltext search Vendor: Deep Sea Electronics

Name	Vendor	Version	Description
Miscellaneous			
M835 Buttons	Deep Sea Electronics	1.0.0.0	M835 Buttons
M835 Inputs	Deep Sea Electronics	1.0.0.0	M835 Inputs
M835 Outputs	Deep Sea Electronics	1.0.0.0	M835 Outputs

☑ Group by category □ Display all versions (for experts only) □ Display outdated versions

Please select a device from the list above.

(You can select another target node in the navigator while this window is open.)

Select *Deep Sea Electronics*

Select the device to add…

And click *Add Device.*

Add other devices if required, then click *Close* when finished.

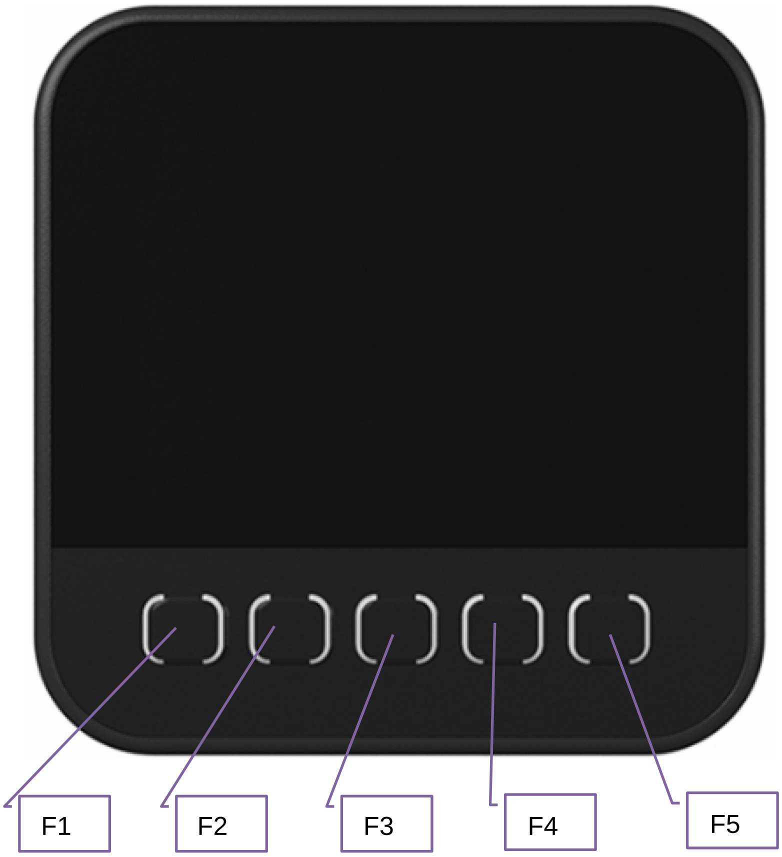
4.7 BUTTONS

4.7.1 BUTTON LOCATION

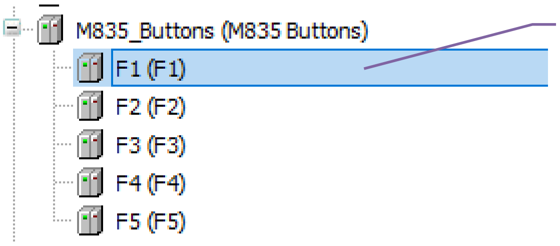
NOTE: DSE_UTILS compiled library provides *DSEM835_Buttons* function block to aid the use of the fascia buttons within projects by providing (among others), rising, and falling edge detection.

NOTE: For applications requiring additional control buttons, DSEM5xx CAN Keypads are available, along with DSE_KEYPADS CODESYS Compiled Library to aid their integration with the CODESYS project.

The below image shows the location of the buttons along with their default names within the CODESYS environment.



4.7.2 BUTTON SETTINGS



Double-Click the Button to configure the *Hold Time* and variable mappings.

4.7.2.1 PARAMETERS

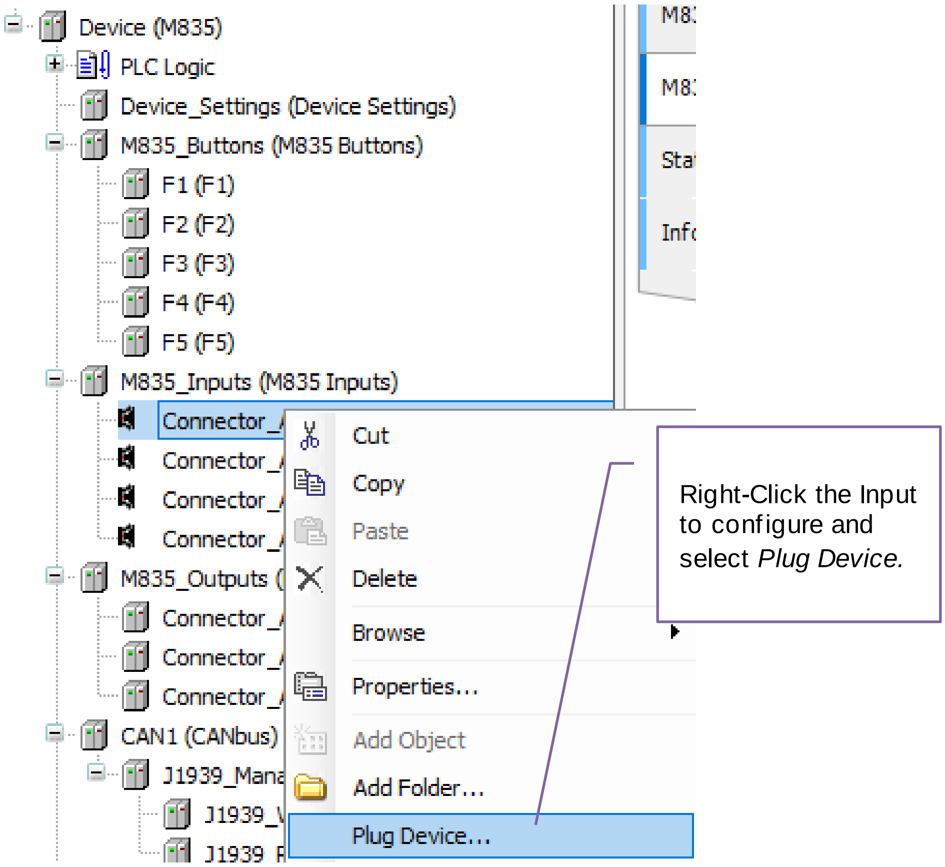
Parameter	Description
Hold Threshold	Amount of time (in milliseconds) that the button must be pressed before it is considered ‘held down’.

4.7.2.2 I/O MAPPING

Parameter	Description
State	Indicates if the button is pressed (TRUE) or not pressed (FALSE).
Held	Indicates if the button has been held for longer than the duration of the *Hold Threshold* (TRUE) or not (FALSE).
Hold Time	The amount of time (in milliseconds) that the button has been pressed for (zero if not currently pressed).

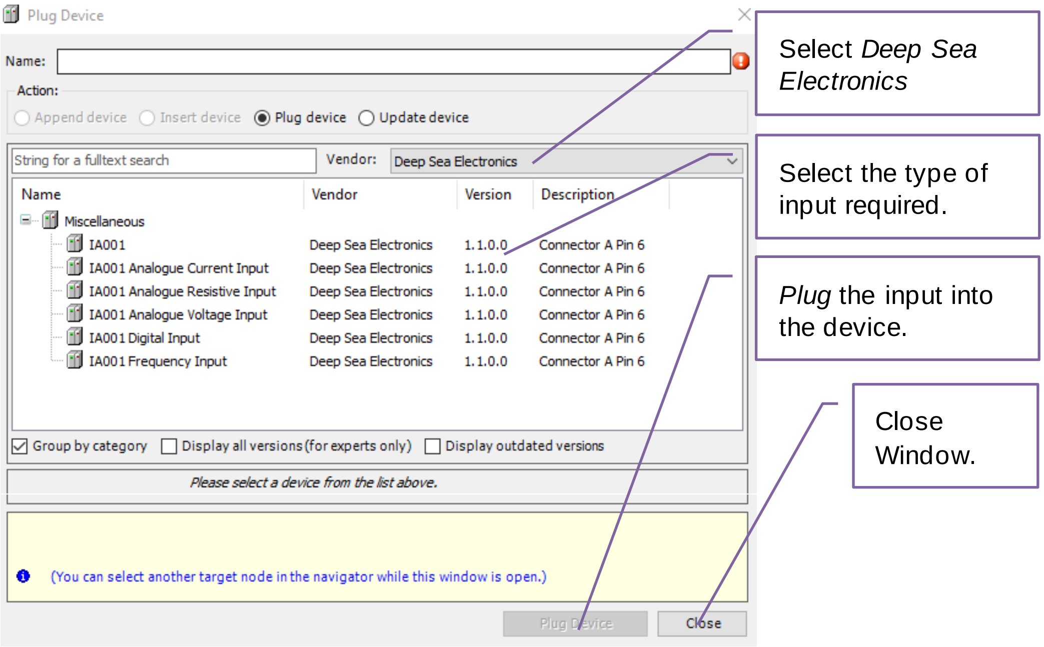
4.8 INPUTS

To select between the different input types (digital or analogue), we must *Plug* a device into the input slot.



Right-Click the Input to configure and select *Plug Device.*

NOTE: Frequency Input selection is available only for IA001 on DSEM835-003-xx onwards.



Plug Device

Name:

Action:

○ Append device ○ Insert device ● Plug device ○ Update device

String for a fulltext search Vendor: Deep Sea Electronics

Name	Vendor	Version	Description
Miscellaneous			
IA001	Deep Sea Electronics	1.1.0.0	Connector A Pin 6
IA001 Analogue Current Input	Deep Sea Electronics	1.1.0.0	Connector A Pin 6
IA001 Analogue Resistive Input	Deep Sea Electronics	1.1.0.0	Connector A Pin 6
IA001 Analogue Voltage Input	Deep Sea Electronics	1.1.0.0	Connector A Pin 6
IA001 Digital Input	Deep Sea Electronics	1.1.0.0	Connector A Pin 6
IA001 Frequency Input	Deep Sea Electronics	1.1.0.0	Connector A Pin 6

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

Please select a device from the list above.

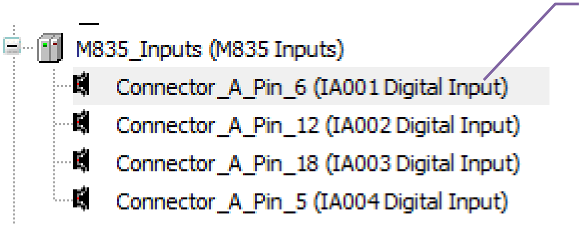
(You can select another target node in the navigator while this window is open.)

Select *Deep Sea Electronics*

Select the type of input required.

Plug the input into the device.

Close Window.



M835_Inputs (M835 Inputs)

Connector_A_Pin_6 (IA001 Digital Input)

Connector_A_Pin_12 (IA002 Digital Input)

Connector_A_Pin_18 (IA003 Digital Input)

Connector_A_Pin_5 (IA004 Digital Input)

Double-Click the Input to configure parameters and mappings.

4.8.1 DIGITAL INPUTS

NOTE: When reading the state of the digital inputs, the value is returned as TRUE (meaning input High) or FALSE (meaning input Low). The application must interpret these values, bearing in mind the configuration of the input *Resistor* to determine if the input is 'Active' or 'Inactive'.

4.8.1.1 PARAMETERS

Parameter	Description
Resistor	***Float***: The input is floating when no connection is made. Used where the external sensor (NPN Sinking or PNP Sourcing type) has an integrated pull-up or pull-down resistor. ***Pull Up***: An internal pull up resistor biases the input to the positive supply rail when no connection is made. Commonly used with NPN (Sinking) type switched sensors and volt-free contacts. ***Pull Down***: An internal pull-down resistor biases the input to the negative supply rail when no connection is made. Commonly used with PNP (Sourcing) type switched sensors and volt-free contacts.
Higher Threshold	The input is detected as being High (TRUE) when above this threshold with respect to the negative supply rail. Recommended 6 V (DSEM835 cannot measure above 10 V, application of voltage above this is seen as 10 V).
Lower Threshold	The input is detected as being Low (FALSE) when below this threshold with respect to the negative supply rail. Recommended 2 V.

4.8.2 FREQUENCY INPUT

NOTE: Frequency Input selection is available only for IA001 on DSEM83 V1.1 onwards.

This input type is used to measure the frequency of GND referenced signals of low frequency (150 Hz for M835-003-xx onwards, 50 Hz for prior hardware revisions).

4.8.2.1 PARAMETERS

Parameter	Description
Resistor	***Float***: The input is floating when no connection is made. Used where the external sensor (NPN Sinking or PNP Sourcing type) has an integrated pull-up or pull-down resistor. ***Pull Up***: An internal pull up resistor biases the input to the positive supply rail when no connection is made. Commonly used with NPN (Sinking) type switched sensors and volt-free contacts. ***Pull Down***: An internal pull-down resistor biases the input to the negative supply rail when no connection is made. Commonly used with PNP (Sourcing) type switched sensors and volt-free contacts.

4.8.3 ANALOGUE INPUTS

4.8.3.1 CURRENT

Parameter	Description
Current Range	***0..20 mA*:** The input is 0 mA to 20 mA. ***4..20 mA*:** The input is 4 mA to 20 mA.

I/O Mapping	Description
Value	The current value of the input (µA, microamps) is mapped to the selected program variable.
Error Code	The current error state of the input is mapped to the selected program variable.

4.8.3.2 RESISTIVE

I/O Mapping	Description
Value	The current value of the input (Ω) is mapped to the selected program variable.
Error Code	The current error state of the input is mapped to the selected program variable.

4.8.3.3 VOLTAGE

I/O Mapping	Description
Value	The current value of the input (mV, milli Volts) is mapped to the selected program variable.
Error Code	The current error state of the input is mapped to the selected program variable.

4.9 OUTPUTS

NOTE: Outputs have integral *freewheel* (also known as *flyback*, *snubber*, *clamp* etc) diodes to protect the DSE device, however if elevated levels of back E.M.F. are generated by the de-energising of the external circuit, it is good practice to quench this at source. This too prevents it being induced into other cables where it may affect other circuits such as CAN communications or other devices entirely.

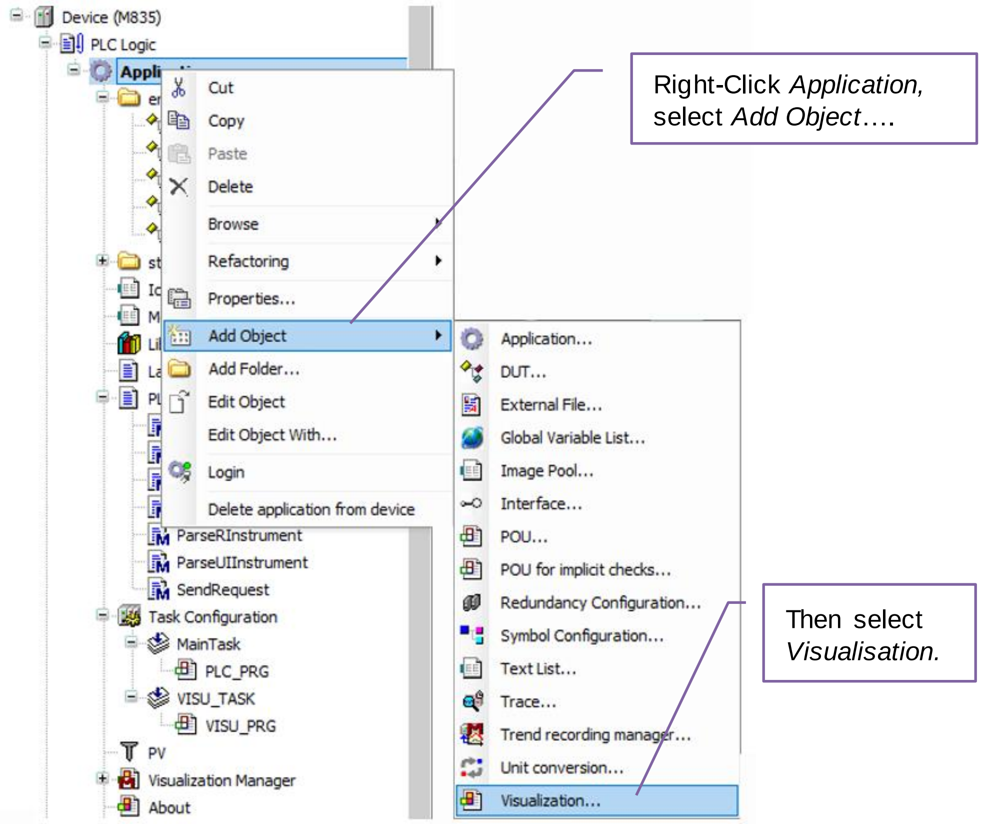
I/O Mapping	Description
Enable	***FALSE:*** The output is inactive. ***TRUE:*** The output is active.

4.10 USING THE DISPLAY IN THE PROJECT (VISUALISATION)

NOTE: Visualisation Style *DSEStyle* or *None* must be selected as detailed below. Other Visualisation Styles are not supported by DSEM835.

NOTE: DSEM835 uses CODESYS *Target Visu Light* (TVL). Ensure your project does not include *VIsuElems* libraries linked in the *Library Manager* and that VisuElems objects such as CODESYS *TextLists* not used in the project.

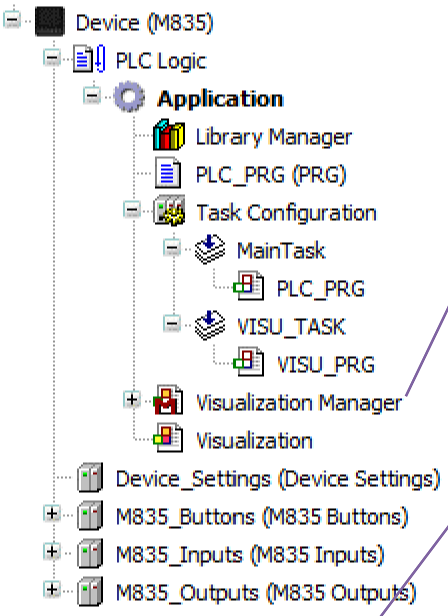
CODESYS 3.5 includes the facility to design and manipulate the LCD of the device. While the operation of the CODESYS environment is detailed within the CODESYS online document, this section provides a quick start guide to using the *Visualisation* component of CODESYS 3.5. DSEM835 visualisation options differ with some of DSE's other displays as it uses *TargetVisuLight* (CODESYS TVL) Library to economise resources.



Right-Click *Application*, select *Add Object*….

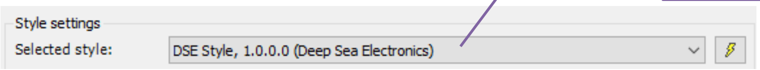
Then select *Visualisation.*

Continued overleaf…



```
Device (M835)
  PLC Logic
    Application
      Library Manager
      PLC_PRG (PRG)
      Task Configuration
        MainTask
          PLC_PRG
        VISU_TASK
          VISU_PRG
      Visualization Manager
      Visualization
  Device_Settings (Device Settings)
  M835_Buttons (M835 Buttons)
  M835_Inputs (M835 Inputs)
  M835_Outputs (M835 Outputs)
```

Double-Click *Visualisation Manager*



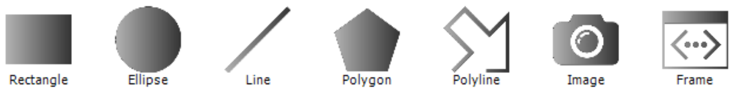
Style settings

Selected style: DSE Style, 1.0.0.0 (Deep Sea Electronics)

And ensure selection of *DSEStyle. Other styles are not supported by DSEM835.*

4.10.1 VISUALISATION ELEMENTS

The following Visualisation Elements are available for DSEM835. Select the element required and place them in the Visualisation.



Polygon and Polyline support *internal rotation.* This is useful for creating meter needles for custom gauges.

The following subsections provides insight as to how to meet most common requirements of a device Visualisation.

Advice regarding the suggested solutions is available from support@deepseaelectronics.com.

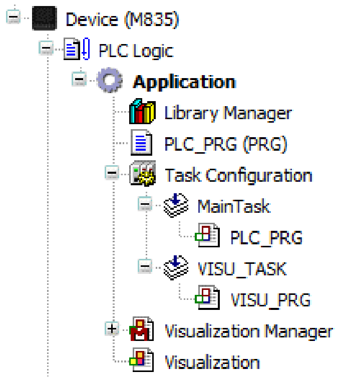
4.10.1.1 STYLES

NOTE: Style definition is not detailed within this manual. Consult CODESYS Help for details.

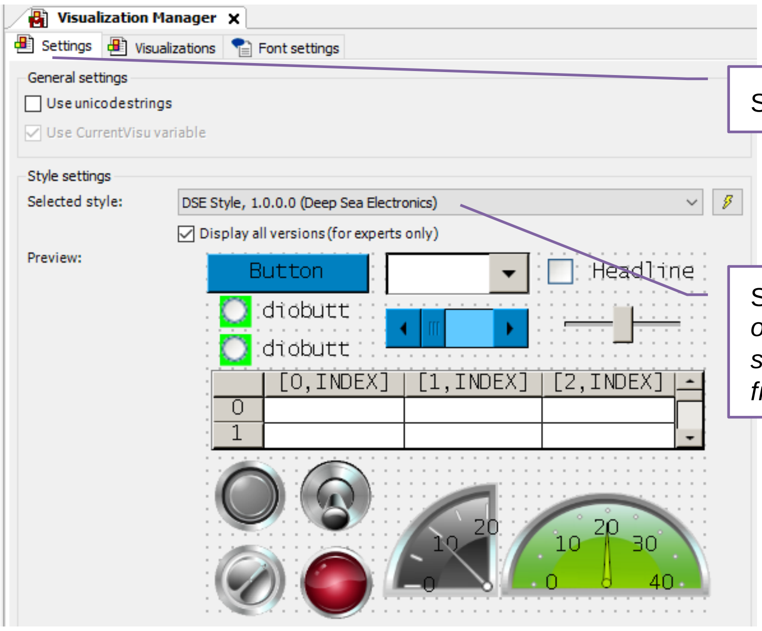
NOTE: When creating your own *Style* ensure to *Copy and Edit* Style, copying from *DSEStyle*. This ensures full compatibility with DSEM835.

NOTE: After changing styles, select *Build* | *Clean All* within CODESYS to clear the build cache and force a full rebuild of the application and the next login to the device. Failure to do so may result in unwanted (previously cached) colours being used.

While *Styles* are useful, if you have not defined your own, use *DSEStyle*. Features such as colour gradients that may be present in other selectable styles are not supported in DSEM835.



Double Click *Visualization Manager*



□ Use unicodestrings

☑ Use CurrentVisu variable

Style settings

Selected style: DSE Style, 1.0.0.0 (Deep Sea Electronics)

☑ Display all versions (for experts only)

Select *Settings*

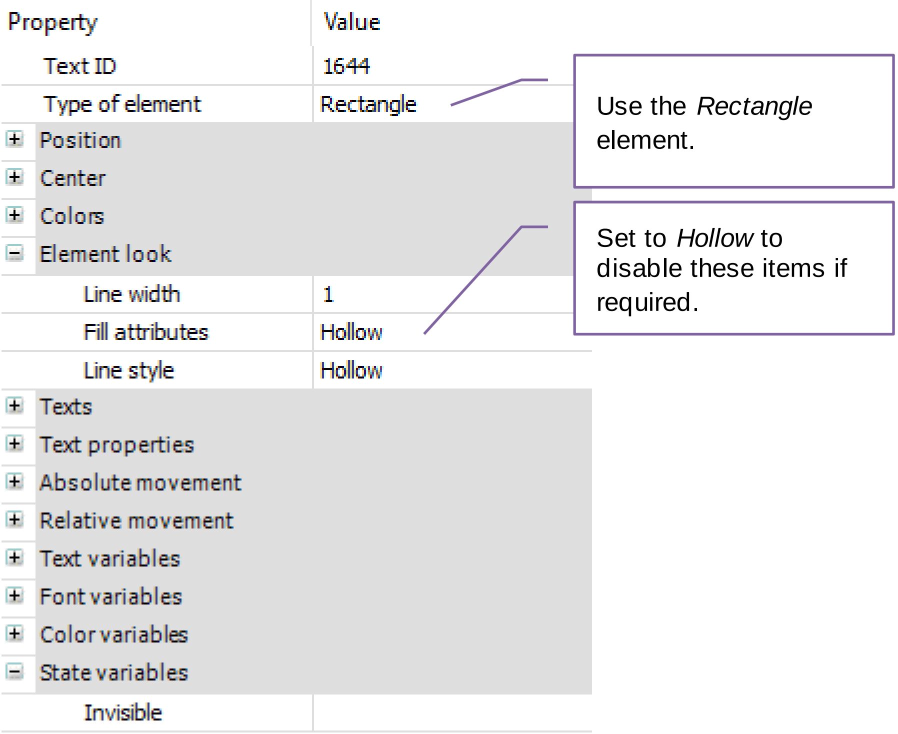
Select *DSEStyle*, *or your own style, derived from DSEStyle*

4.10.2 TEXT (FIXED OR DYNAMIC)

⚠NOTE: Text *Fonts* occupy a large amount of system memory and make the application code larger. To save resources, and ensure a smaller file for transfer, ensure to follow the instructions in the subsection entitled *Using Fonts*.

⚠NOTE: Variable type STRING is used for ASCII strings. For UNICODE, see subsection entitled *UNICODE* elsewhere in this document.

Use the flexible *Rectangle* element which provides the functionality of a rectangle (with optional border and fill), and an integral Text Field that may be fixed, or programmatically altered using the Text Variable property.
Border and Fill settings are disabled by setting the *Fill Attributes* and *Line Style* properties to *Hollow*.



Property	Value
Text ID	1644
Type of element	Rectangle
⊞ Position	
⊞ Center	
⊞ Colors	
⊟ Element look	
Line width	1
Fill attributes	Hollow
Line style	Hollow
⊞ Texts	
⊞ Text properties	
⊞ Absolute movement	
⊞ Relative movement	
⊞ Text variables	
⊞ Font variables	
⊞ Color variables	
⊟ State variables	
Invisible	

Use the *Rectangle* element.

Set to *Hollow* to disable these items if required.

4.10.2.1 RECOMMENDED FONTS

⚠NOTE: Also see subsection entitled *Installing Fonts on the DSEM835* elsewhere in this manual.

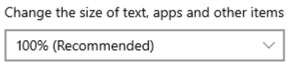
Windows TrueType™ Fonts are used by CODESYS and the DSEM835. While other fonts may be used, DSE recommend the following fonts for use with DSEM835:

Arial
Arial Rounded MT Bold
Calibri
Segoe UI (this font is designed for User Interfaces and has many sizes usable with DSEM835)
Lucida Console
Lucida San Typewriter

4.10.2.2 TEXT SIZE

NOTE: Use of assorted sizes of font within the project requires additional font characters to be transferred to the device, occupying more memory. To use less space, select sizes carefully. For example, using one font size for titles, and another for instrumentation keeps to only two font sizes, making efficient use of font memory.

NOTE: Within Windows Settings *System* | *Display* | *Scale and Layout*, ensure the size is set to *100 %*, even if this is not the Recommended size. Failure to do so results in CODESYS sending a scaled font to the device, which appears different to that selected within CODESYS:



Change the size of text, apps and other items

100% (Recommended)

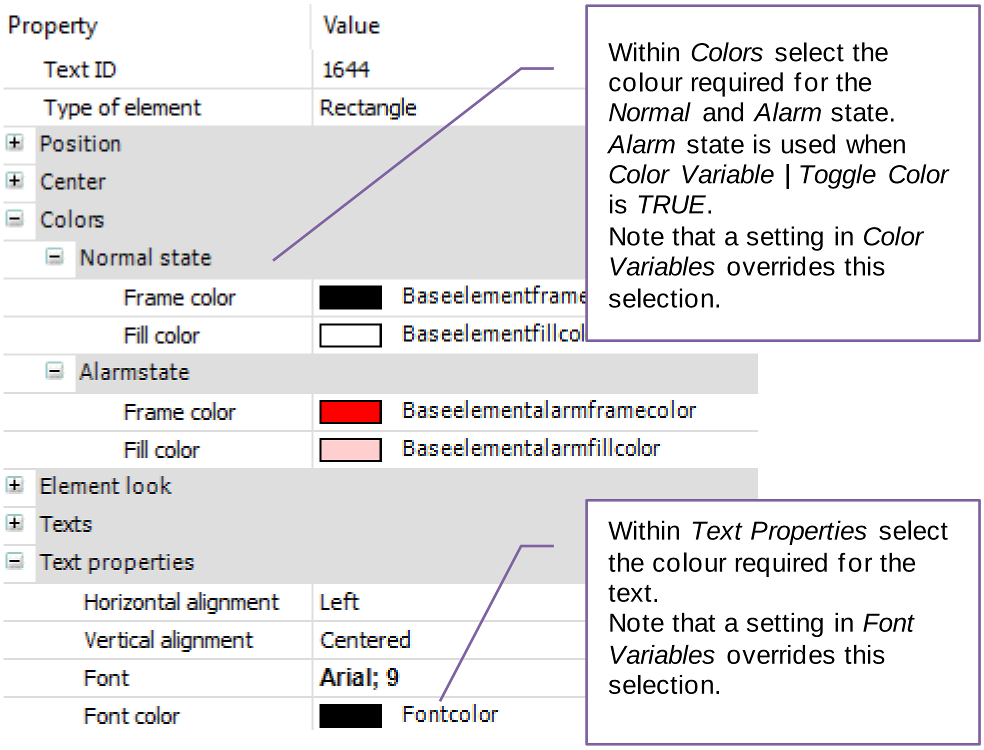
When selecting Text (font) size, only *Integer* font sizes are possible. Where other sizes are mistakenly used, the font appears on the M835 display in a default (small sized) font.
CODESYS allows selection of font sizes that are not appropriate for DSEM835.
For example, it is possible to select Arial 7.

After closing the font selector, it appears correctly set as Arial; 7 in the *Property* window. However, when selecting it with the mouse, the font size is not an integer: Arial, 6.75pt .
The font selector and property window ‘round up’ the font size until selected with the mouse.

To minimise occurrences of this error it is recommended to select Text Size from the following options (all multiples of 3: *6, 9, 12, 15, 18, 21, 24…* etc. However, it is still important to check for the *Integer size* as detailed above as this functionality differs between fonts.

4.10.2.3 TEXT COLOUR

Fixed Colour



Property	Value
Text ID	1644
Type of element	Rectangle
⊞ Position	
⊞ Center	
⊟ Colors	
⊟ Normal state	
Frame color	■ Baseelementframe
Fill color	□ Baseelementfillcol
⊟ Alarmstate	
Frame color	🟥 Baseelementalarmframecolor
Fill color	▭ Baseelementalarmfillcolor
⊞ Element look	
⊞ Texts	
⊟ Text properties	
Horizontal alignment	Left
Vertical alignment	Centered
Font	**Arial; 9**
Font color	■ Fontcolor

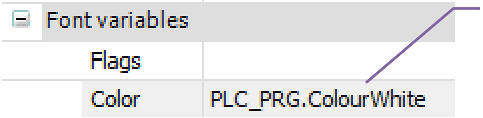
Within *Colors* select the colour required for the *Normal* and *Alarm* state.
Alarm state is used when *Color Variable* | *Toggle Color* is *TRUE*.
Note that a setting in *Color Variables* overrides this selection.

Within *Text Properties* select the colour required for the text.
Note that a setting in *Font Variables* overrides this selection.

<u>Programmatically Changing Colour</u>

> ⚠**NOTE: When using the *Color* parameter to programmatically change the colour at *runtime*, the colour within the PC Visualisation Editor remains as the colour selected in *Text Properties*.**

If required to change the text colour during application runtime, *Font Variables* is used as follows:



⊟ Font variables	
Flags	
Color	PLC_PRG.ColourWhite

Within *Font Variables* select the variable containing the colour required for the element.

Colour is defined as a DINT variable containing the RGB values for the colour (ie 16#RRGGBB)

Example to define the colours :

```
ColourWhite        :     DINT:=16#FFFFFF;
ColourBlack        :     DINT:=16#000000;
ColourRed          :     DINT:=16#FF0000;
ColourGreen        :     DINT:=16#00FF00;
ColourBlue         :     DINT:=16#0000FF;
```

Using Hexadecimal makes it easier to identify the values for each colour.
For example,
16#8000FF is made up from:
Red: 16#80
Green: 16#00
Blue: 16#FF

4.10.2.4 STRING ESCAPE CHARACTER $

Certain characters within strings have special meanings. For example, ‘ (single quote) is used as the character denoting the start and end of the string. Where it is required to include this character within the string for display, the *escape* character is used to inform CODESYS that the ‘ (single quote) character should be displayed and not considered as a string start/end. This escape character is the dollar sign ($).
In the event we need to display a dollar sign in the string we again prefix it with the *escape* character ($).
While the CODESYS inbuilt help file contains a comprehensive list of *escape codes*, the following details the most common.

String	Displayed as
$’	‘
$$	$
$t or ST	Control Character TAB

Examples:

```
VAR
      MyString    :     STRING:=’This $’fee$’ is $$50’;
END_VAR
```

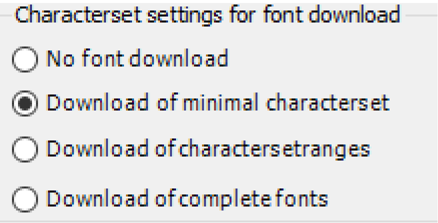
The string contains the message: **This ‘fee’ is $50**

4.10.2.5 INSTALLING FONTS ON THE DSEM835

> **NOTE: If the *Font Settings* tab is not visible within the Visualization Manager, close the tab, then again double click Visualization Manager to open it again and view the settings.**

Text fonts are resource hungry, occupying large parts of the system RAM. To make efficient use of the memory, and ensure a smaller file size for transfer to the device, CODESYS allows us to choose how the font is sent to the device:

Double Click Visualization Manager in the project tree and select the Font settings tab:



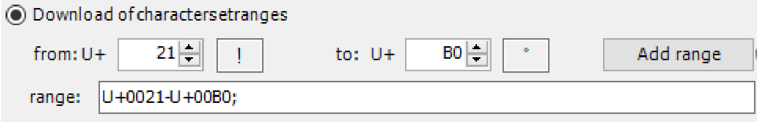
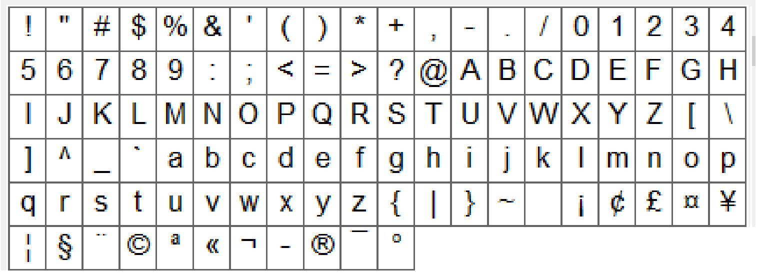
Characterset settings for font download

○ No font download

● Download of minimal characterset

○ Download of charactersetranges

○ Download of complete fonts

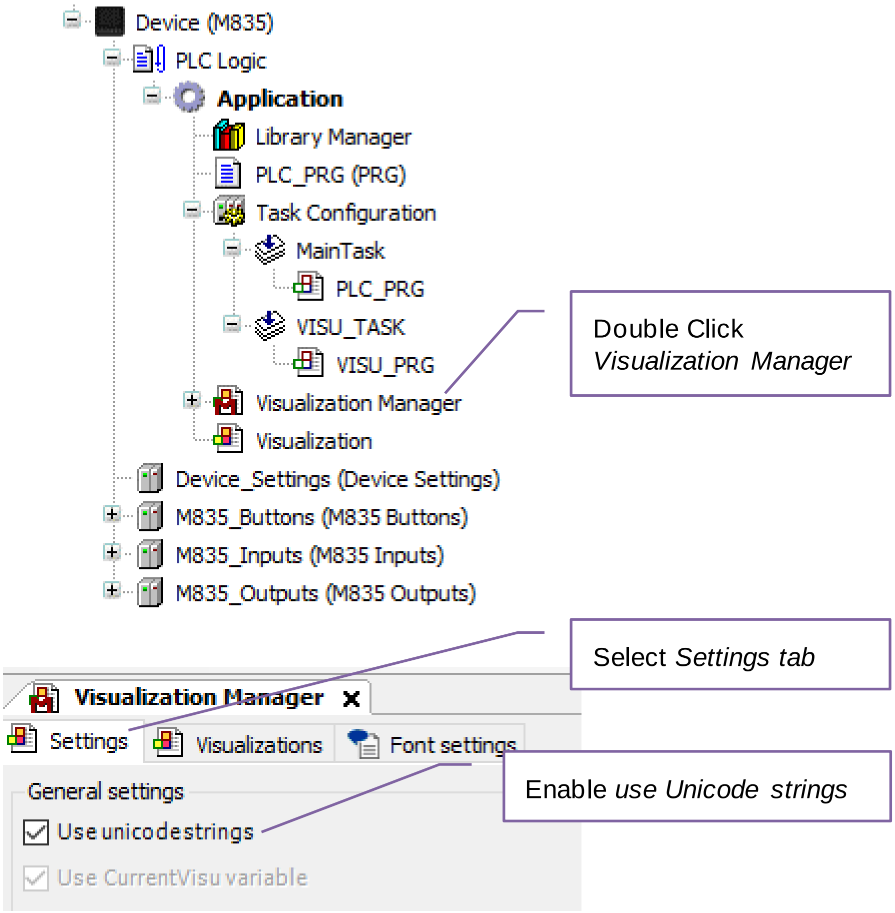
Parameter	Description
Character Settings for Font Download	***No Font Download:*** (Not Recommended) No font is sent to the device. ***Download of Minimal Character Set:*** (Recommended). All characters used in the visualisation are downloaded to the device. However, any characters not used in the visualisation but used in variables (ie Strings) are NOT sent to the device. To 'force' characters to be sent to the device: • Use a hidden visualisation page containing all the font/character selection required for the application. • Use *Download of Character Set Ranges.* ***Download of Character Set Ranges:*** (Advanced). The user selects which character range or ranges to send to the device. Windows Character Map is a useful tool for identifying the range(s) of characters you may need. Adding the range *U+0021-U+00B0;*  ● Download of charactersetranges from: U+ 21 ! to: U+ B0 ° Add range range: U+0021-U+00B0; ensures that most used characters are sent to the device (a to z, A to Z, 0 to 9 and many punctuation characters):  ! " # $ % & ' ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [\ ] ^ _ ` a b c d e f g h i j k l m n o p q r s t u v w x y z { | } ~ ¡ ¢ £ ¤ ¥ ¦ § ¨ © ª « ¬ - ® ¯ ° ***Download of Complete Fonts:*** (Not Recommended). The complete character set of all used fonts is sent to the device occupying a large amount of memory in the device.

4.10.2.6 UNICODE

NOTE: At the time of writing, text within shapes appears *centred* within the shape when *Use unicode strings* is selected, irrespective of the *Horizontal Alignment* setting.

NOTE: Within CODESYS, UNICODE strings must be stored in variable type WSTRING. Type STRING is used only for ASCII strings

To allow the use of UNICODE characters within the project visualisation:



Double Click *Visualization Manager*

Select *Settings tab*

Enable *use Unicode strings*

General settings

☑ Use unicodestrings

☑ Use CurrentVisu variable

4.10.3 ALTERNATIVE TO TEXTLISTS

CODESYS *TargetVisuLight* does not include support for the *TextLists* object available with CODESYS *VisuElems* library. However, alternatives are listed below.

4.10.3.1 DYNAMIC TEXT

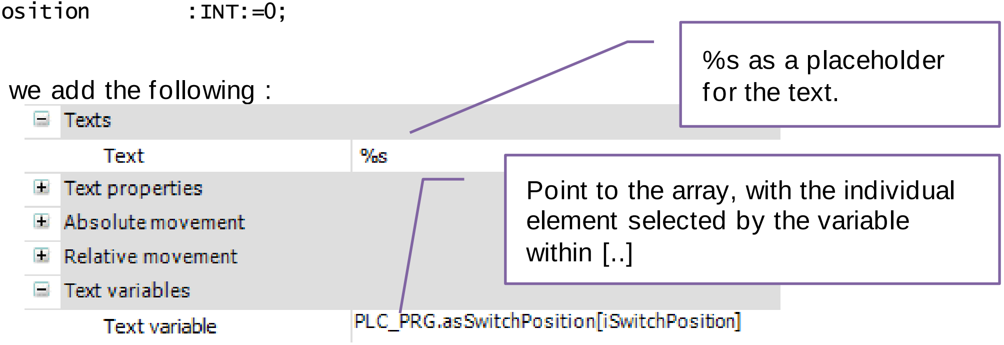
STRING ARRAY

Dynamic Text is using a Visualisation element to display text that changes depending upon some condition within the application. For example, this could be indication the position of a control switch. Numerous methods exist, this being the recommended option, having the advantage of the text being easily located at the top of the POU in the VAR area.

In this example `iSwitchPosition` is the physical position of the switch. We use this to index into an array to show the position in text, rather than a number.

```
VAR
      asSwitchPosition        :ARRAY[0..3] of STRING:=[‘Off’, ‘Manual’, ‘Auto’, ‘Test’];
      iSwitchPosition         :INT:=0;
END_VAR
```

Then in the Visu, we add the following :



⊟ Texts
Text %s
⊞ Text properties
⊞ Absolute movement
⊞ Relative movement
⊟ Text variables
Text variable PLC_PRG.asSwitchPosition[iSwitchPosition]

%s as a placeholder for the text.

Point to the array, with the individual element selected by the variable within [..]

CASE Statement

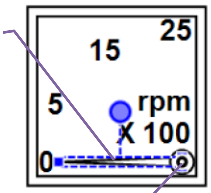
An alternative is to use a CASE statement. It is recommended to use string variables in the VAR section of the POU. This example uses string constants in the body of the code to set a STRING variable that is used in the code or in the Visu. This option allows for non-contiguous values of the index variable, however it is important to add the ELSE at the end to ensure all values are handled

```
CASE iSwitchPosition OF
      0     :     sSwitchPosition:=’Off’;
      1     :     sSwitchPosition:=’Manual’;
      2     :     sSwitchPosition:=’Auto’;
      3     :     sSwitchPosition:=’Test’;
      5     :     sSwitchPosition:=’Semi-Auto’;
      22    :     sSwitchPosition:=’Diagnostics’;
      255   :     sSwitchPosition:=’Switch fault’;
ELSE  sSwitchPosition:=’!! UNKNOWN !!’
END_CASE
```

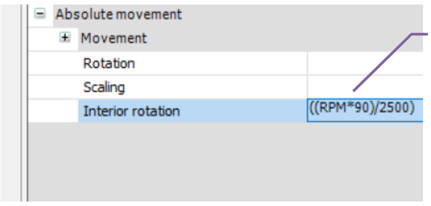
4.10.4 CUSTOM GAUGES

4.10.4.1 DRAWN BACKGROUND ROTARY GAUGE

Polygon used to create the needle. Draw it in the *zero* position

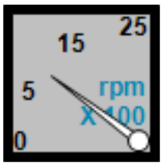


Place the ‘centre’ of the needle at the centre of rotation.



⊟ Absolute movement	
⊞ Movement	
Rotation	
Scaling	
Interior rotation	((RPM*90)/2500)

Scale the variable so that 2500 rpm = 90°

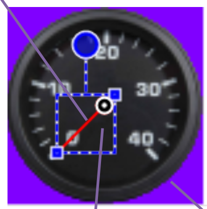


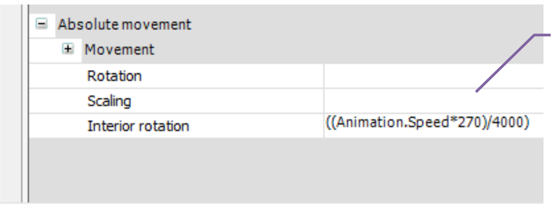
How it appears on the DSEM835

4.10.4.2 IMAGE BACKGROUND ROTARY GAUGE

Place an image on the visu, with the needle *in fron*t. On top of this is a small circle used to create a more realistic looking needle.

Polyline used to create the needle. Draw it in the *zero* position



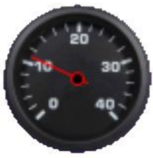


⊟ Absolute movement	
⊞ Movement	
Rotation	
Scaling	
Interior rotation	((Animation.Speed*270)/4000)

Scale the variable so that 4000 rpm = 270°

Place the ‘centre’ of the needle at the centre of rotation.

The purple colour is selected as the *transparent* colour in *Device Settings.*



How it appears on the DSEM835

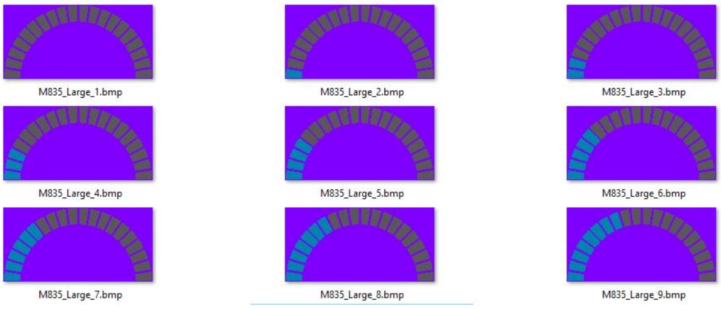
4.10.4.3 MULTIPLE IMAGES GAUGES

Multiple images are used to create a ‘stop frame’ animation used to represent a gauge.

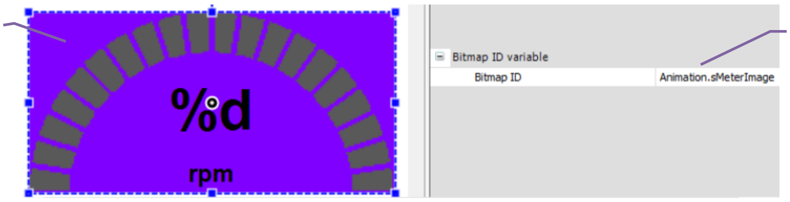
For example:
If we have 20 images to show a variable up to the value of 200, we can name the images *meter1, meter2, meter3* etc. Then use arithmetic to build the string with the desired image.

```
sMeterImage:=CONCAT(‘meter’,TO_STRING(TO_INT(myVariable / 10)));
```

example showing images in a folder on the PC:

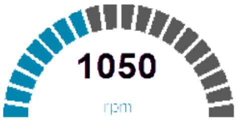


Place one of the bitmaps on the screen.



Bitmap ID variable	
Bitmap ID	Animation.sMeterImage

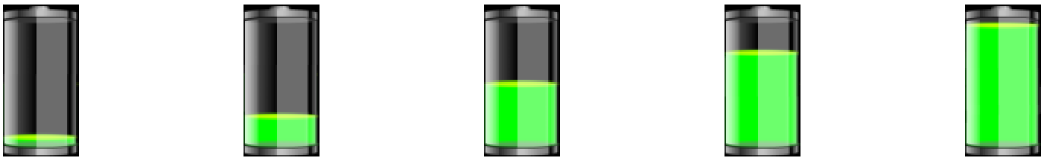
Map a STRING variable to *Bitmap ID.*



How it appears on the DSEM835

Examples of other kinds of image meters.

Battery Level:



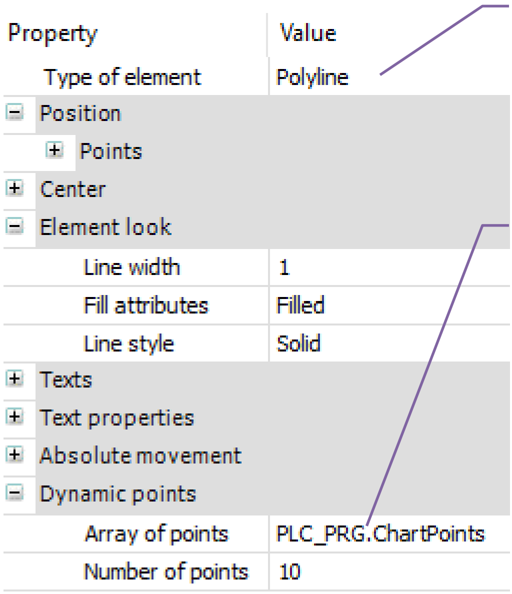
Mains Voltage Level:

Signal Strength:

4.10.5 LINE CHART OR ANIMATED SHAPES

NOTE: When using this feature within a Frame element, populate the Array of points every task cycle. Failure to do this results in the iX,iY variables incrementing, moving the position of the shape. This does not occur where Frames are not used.

The *Polyline* element is added, using *Dynamic Points* to allow the line to be manipulated using an array of points. In the same way a *Polygon* element is used with *Dynamic Points* to animate an arbitrary shape, defined, and modified in the array of points.



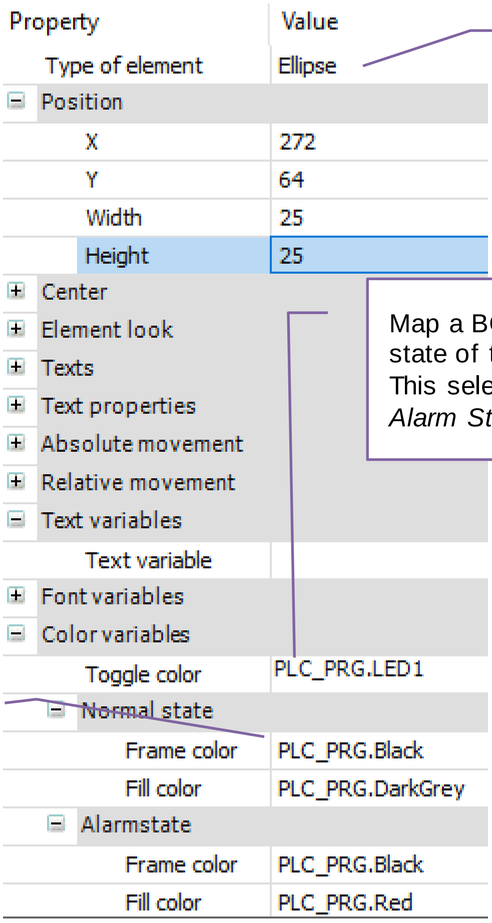
Property	Value
Type of element	Polyline
⊟ Position	
⊞ Points	
⊞ Center	
⊟ Element look	
Line width	1
Fill attributes	Filled
Line style	Solid
⊞ Texts	
⊞ Text properties	
⊞ Absolute movement	
⊟ Dynamic points	
Array of points	PLC_PRG.ChartPoints
Number of points	10

Use the *Polyline* element.

Map the array of points containing the coordinates of the line points.

4.10.6 LAMP OR LED

Use the *Ellipse* element to add a circular shape. Use either the *Color* property to change the fill colour or use the *Toggle Color* property to select between the *Normal State* and *Alarm State* colour configurations.



Property	Value
Type of element	Ellipse
⊟ Position	
X	272
Y	64
Width	25
Height	25
⊞ Center	
⊞ Element look	
⊞ Texts	
⊞ Text properties	
⊞ Absolute movement	
⊞ Relative movement	
⊟ Text variables	
Text variable	
⊞ Font variables	
⊟ Color variables	
Toggle color	PLC_PRG.LED1
⊟ Normal state	
Frame color	PLC_PRG.Black
Fill color	PLC_PRG.DarkGrey
⊟ Alarmstate	
Frame color	PLC_PRG.Black
Fill color	PLC_PRG.Red

Use the *Ellipse* element.

Map a BOOL used to change the state of the LED/Lamp indicator. This selects between *Normal* and *Alarm State.*

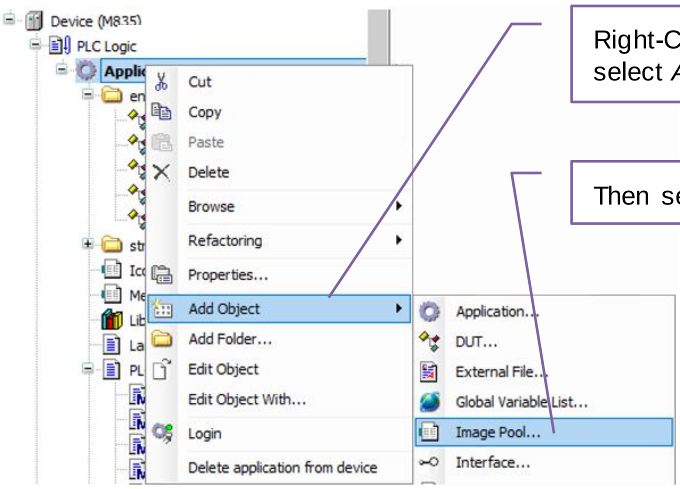
Map to the variables that define which colours to use (RGB colour)

ie
Red : DWORD:= 16#FF0000

4.10.7 USING CUSTOM IMAGES ON THE DISPLAY

Many applications require custom images to be placed on the DSEM835 display. This is controlled using an *Image Pool* within CODESYS. The Image Pool acts as a container for the images, which are then selected for display.

4.10.7.1 ADDING AN IMAGE POOL



Right-Click *Application*, select *Add Object*….

Then select *Image Pool*

4.10.7.2 ADDING IMAGES TO THE IMAGE POOL

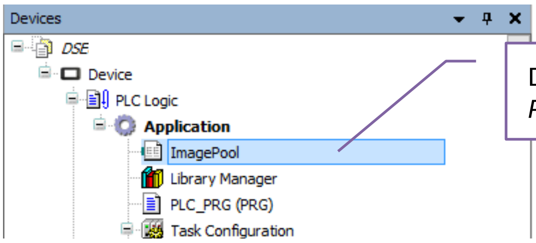
NOTE: Images must be Windows Bitmap (.bmp) type with 8-bit, 16-bit or 24-bit colour depth. Other image types are not supported.

NOTE: It is recommended to use 24 bit (or 16 bit) colour images. DSEM835 has limited memory for 8 bit image types, suitable only for *small* (icon type) images.

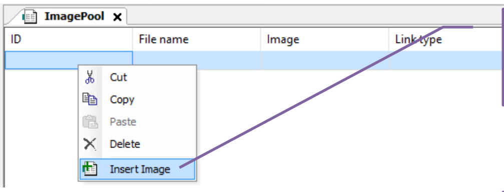
NOTE: Images MUST be scaled externally before being imported to the image pool as DSEM835 supports *Scaling Type: Fixed* for elements within the Visualisation.

NOTE: For image transparency see section entitled *Device Settings | Transparency.*

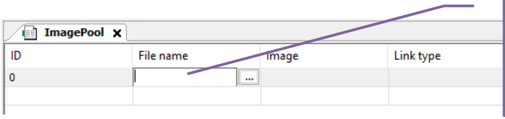
NOTE: Multiple Image Pools may be used to keep images of similar type or usage together.



Double-Click the *Image Pool.*



Right-Click the empty line and select *Insert Image.*



Then Double-Click the empty space under *File name* and click …

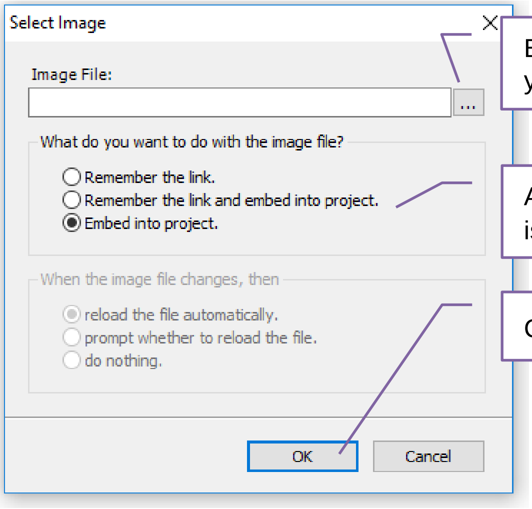

Select Image

Image File:

What do you want to do with the image file?

○ Remember the link.
○ Remember the link and embed into project.
● Embed into project.

When the image file changes, then

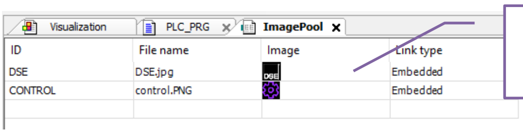
● reload the file automatically.
○ prompt whether to reload the file.
○ do nothing.

Browse to the image on your computer…

And select how the image is stored in the project.

Click *OK* when done.

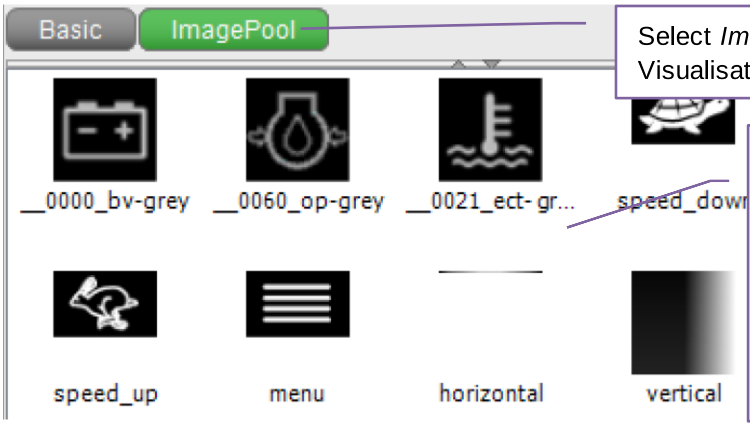
NOTE: Ensure the *ID* in the image pool contains only alphanumerical characters, - (minus), and _ (underscore). Many other characters, including . (full stop) and spaces are not supported and may result in the image not appearing on the display.


ID	File name	Image	Link type
DSE	DSE.jpg	DSE	Embedded
CONTROL	control.PNG		Embedded

Example entries in the *Image Pool*.

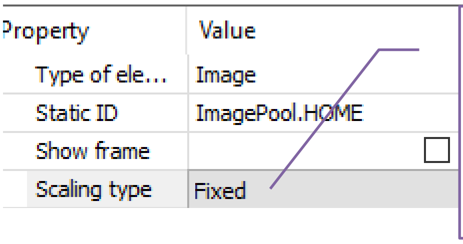
4.10.7.3 USING THE IMAGE POOL ON THE DISPLAY

Entries within the Image Pool are automatically detected by the CODESYS Visualisation Toolbox and are available for placing on the Visualisation.



Select *Image Pool* in the Visualisation Toolbox

Example entries in the *Image Pool* ready for placing on the Visualisation.



Property	Value
Type of ele...	Image
Static ID	ImagePool.HOME
Show frame	□
Scaling type	Fixed

You MUST select *Scaling Type: FIXED.*
Scaling within CODESYS is not supported.
After pressing *ENTER* this property is no longer displayed.

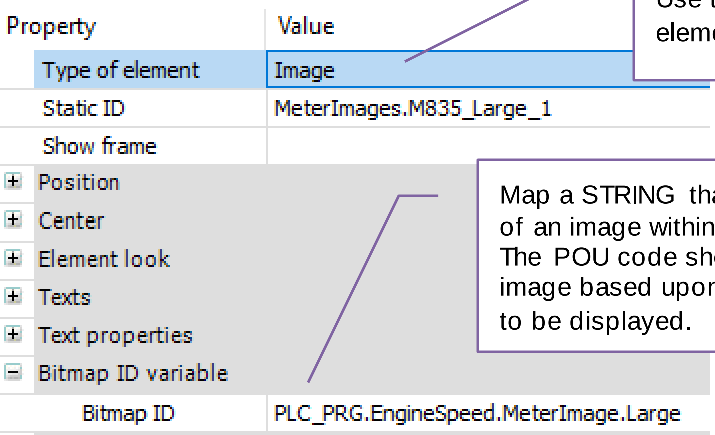
4.10.7.4 GRAPHICAL GAUGES OR IMAGE SWITCHING

NOTE: For a example of a gauge created from multiple images, see section entitled *Multiple Image Gauges* elsewhere in this document.

Multiple Images

Multiple images may be used to provide animation. This method has several images, each one different to the others. Swapping between images gives the illusion of animation.

Use the Image Pool to load in the required images. Add an *Image* to the visualisation and change the image reference using the *DynamicBitmap* property.



Property	Value
Type of element	Image
Static ID	MeterImages.M835_Large_1
Show frame	
⊞ Position	
⊞ Center	
⊞ Element look	
⊞ Texts	
⊞ Text properties	
⊟ Bitmap ID variable	
Bitmap ID	PLC_PRG.EngineSpeed.MeterImage.Large

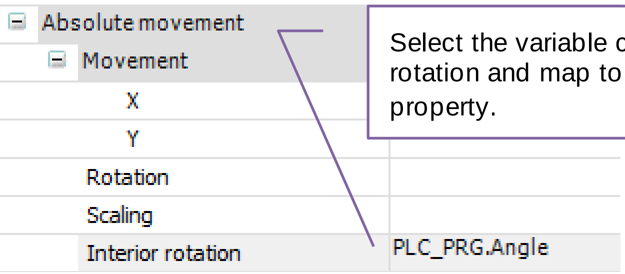
Use the *Image* element.

Map a STRING that matches the *ID* of an image within the image pool. The POU code should select the image based upon the value that is to be displayed.

Element Rotation

NOTE: *Interior Rotation* on DSEM835 is available only with the *Polyline and Polygon* Visualisation elements.

Visualisation Element *Polyline* may be rotated at application runtime. For example, this technique is used to give motion to a meter needle, or to show a valve opening/closing.



⊟ Absolute movement	
⊟ Movement	
X	
Y	
Rotation	
Scaling	
Interior rotation	PLC_PRG.Angle

Select the variable containing the angle of rotation and map to the *Interior rotation* property.

4.10.8 CHANGING VISUALISATION PAGES

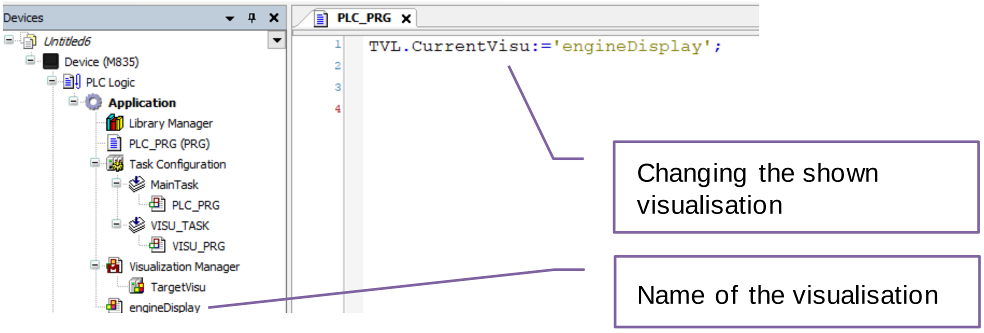
CODESYS provides differing methods to change visualisations within a project, described in the following subsections. This useful when making *multipage* applications.

For further details, contact Deep Sea Electronics Technical Support.

4.10.8.1 CURRENTVISU VARIABLE

This method allows the application to display any visualisation within the project.
The displayed visualisation may also include one or more frames.

Example:



```
TVL.CurrentVisu:='engineDisplay';
```

Changing the shown visualisation

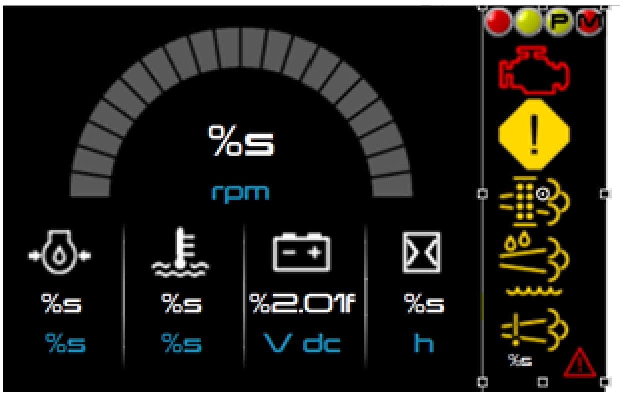
Name of the visualisation

4.10.8.2 FRAMES

This method allows for sections of the page to remain unchanged while other areas change to reflect the context of the application. For example, it is common to keep the bottom of the screen as indication of the button usage, while changing the screen section above.
Often this method is used in conjunction with manipulation of the *CurrentVisu* variable.

Example.

In this example, the icons in the right pane are a single *Frame* visu element*,* with the instruments to the left another *Frame.* Within the application, either can be changed, for example to show additional instruments, without affecting the display of the other frame. In this example, the visualisation contains only the two *Frame* elements.
We can change to other visualisations by manipulating the *CurrentVisu* variable, combining two visualisation selection methods.



4.11 DEBUG

⚠ **NOTE: Before writing the application to the DSEM835, power cycle the device. This ensures memory is cleared ready to accept the new application. Failure to do this may result in a failed application write, requiring this process to be repeated.**

When using the CODESYS *Debug* function, ensure to close all editor tabs except the one being used for the debug. This minimises the number of variables being displayed by CODESYS and ensures the fastest update rate. Additionally, closing all editor tabs and utilising the *Watch* function within CODESYS gives satisfactory results as only the *Watched* variables are updated, hence CAN use is minimised.

Update rate of the debugger varies upon application, but experienced CODESYS users should note that the CAN interface gives a slower update rate than that provided by other devices utilising an Ethernet connection. Typically, variables update no faster than once per second.

4.12 CAN

NOTE: DSEM835 has one CAN port. This is referred to as *Network0*. With DSE_CAN library the enum *DSE_CAN.eCANList.CAN_1* is available.

NOTE: To program the device using CAN, see section entitled *Cables, Connectors, Harnesses and Spare Parts* for details of a suitable PC USB CAN interface.

NOTE: DSE_CAN compiled library, supplied as part of the DSE CODESYS Package is available from www.deepseaelectronics.com.

DSEM835 CAN interface is used to configure the device (using CODESYS) and to connect to external CAN devices such as engine ECUs and DSEM240 J1939 (Input/Output expansion).
CAN2.0A or CAN2.0B is supported.

CAN Protocol	Options
SAE J1939	• CODESYS J1939 (see section entitled *CODESYS J1939)* • DSE_CAN compiled library
Raw CAN (custom protocols)	• DSE_CAN compiled library
CANopen	CANopen protocol is not provided directly by DSEM835. Application programming must be used to handle the messages of CANopen. Contact support@deepseaelectronics.com for additional advice
Other CAN protocols	Application programming must be used to handle the messages of other CAN protocols. Contact support@deepseaelectronics.com

4.12.1 DSE_CAN

NOTE: DSEM835 has one CAN port. This is referred to as *Network0*. With DSE_CAN library the enum *DSE_CAN.eCANList.CAN_1* is available.

DSE_CAN compiled library, supplied as part of the *DSE CODESYS Package* available from www.deepseaelectronics.com allows standard CAN messages to be sent and received. This may be used (with customer application code) to interface with J1939, custom protocols and other CAN protocols as required.

While CODESYS Library Manager includes help for the libraries, for additional information, contact support@deepseaelectronics.com.

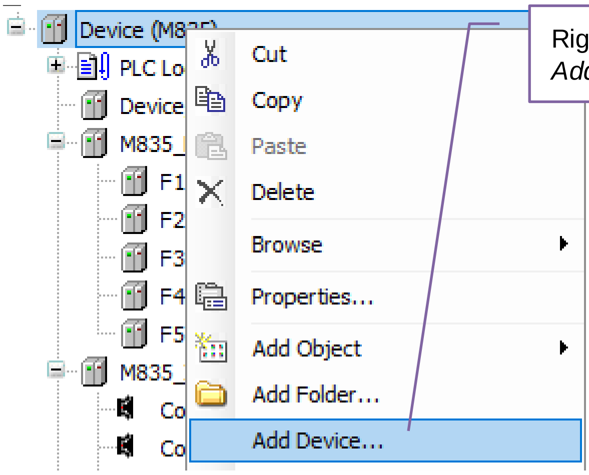
4.12.2 CODESYS J1939

⚠ NOTE: DSEM835 has one CAN port. Within the CODESYS CAN device, set *Network* to *0*,

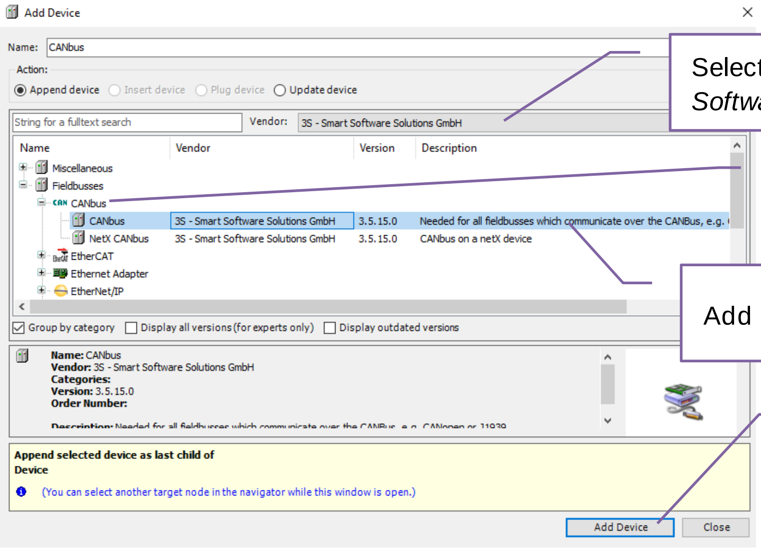
⚠ NOTE: CODESYS CAN devices are not required within the project unless attaching a J1939 CAN Manager to it.

CODESYS J1939 is a complete J1939 implementation for sending and receiving J1939 CAN messages. Conversion to/from actual values and CAN data is automatic (when enabled).

First add the CAN device to the project:



Right-Click *Device*, select *Add Device*….



Name: CANbus

Action:

● Append device ○ Insert device ○ Plug device ○ Update device

String for a fulltext search

Vendor: 3S - Smart Software Solutions GmbH

Name	Vendor	Version	Description
Miscellaneous			
Fieldbusses			
CANbus			
CANbus	3S - Smart Software Solutions GmbH	3.5.15.0	Needed for all fieldbusses which communicate over the CANBus, e.g.
NetX CANbus	3S - Smart Software Solutions GmbH	3.5.15.0	CANbus on a netX device
EtherCAT			
Ethernet Adapter			
EtherNet/IP			

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

Name: CANbus
Vendor: 3S - Smart Software Solutions GmbH
Categories:
Version: 3.5.15.0
Order Number:

Description: Needed for all fieldbusses which communicate over the CANBus, e.g. CANopen or J1939

Append selected device as last child of
Device

(You can select another target node in the navigator while this window is open.)

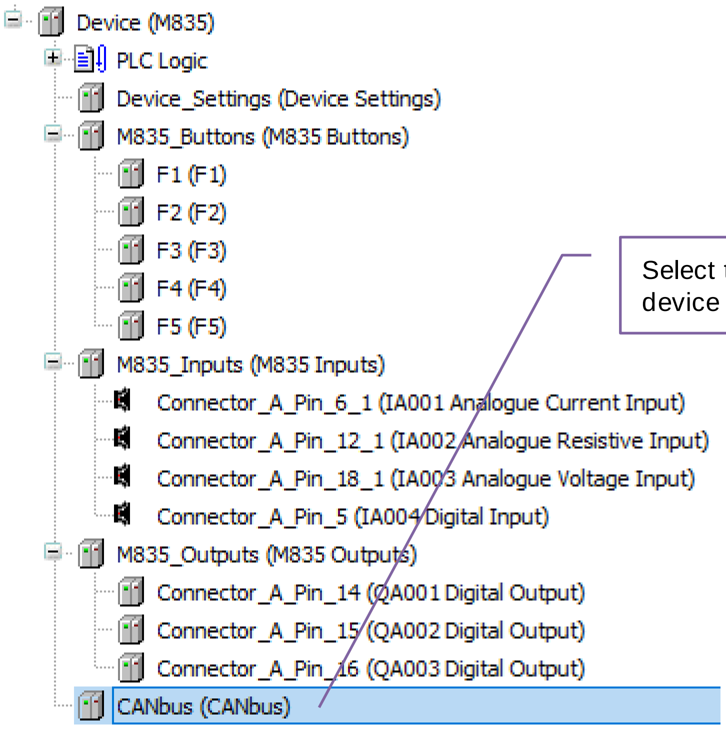
Select *3S – Smart Software Solutions GmbH*

Select *CANbus*

Add CAN to the project.

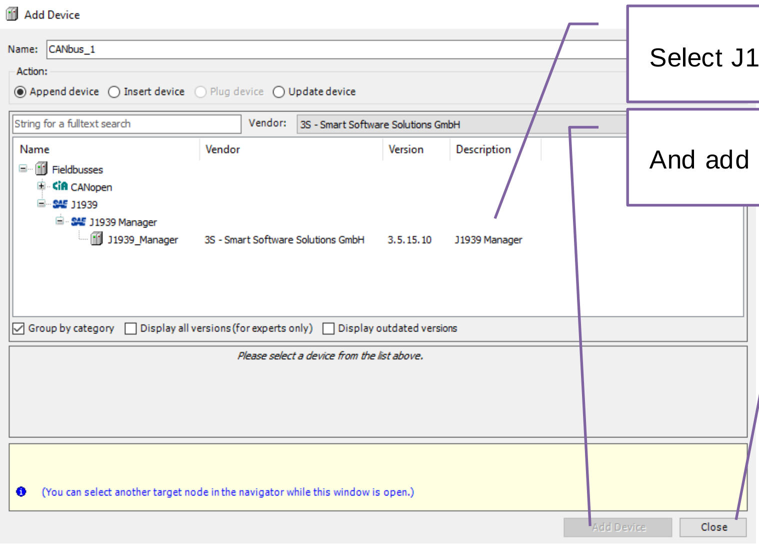
DO NOT close the window yet……

Keep the *Add Device* window open.



Device (M835)
- PLC Logic
- Device_Settings (Device Settings)
- M835_Buttons (M835 Buttons)
  - F1 (F1)
  - F2 (F2)
  - F3 (F3)
  - F4 (F4)
  - F5 (F5)
- M835_Inputs (M835 Inputs)
  - Connector_A_Pin_6_1 (IA001 Analogue Current Input)
  - Connector_A_Pin_12_1 (IA002 Analogue Resistive Input)
  - Connector_A_Pin_18_1 (IA003 Analogue Voltage Input)
  - Connector_A_Pin_5 (IA004 Digital Input)
- M835_Outputs (M835 Outputs)
  - Connector_A_Pin_14 (QA001 Digital Output)
  - Connector_A_Pin_15 (QA002 Digital Output)
  - Connector_A_Pin_16 (QA003 Digital Output)
- CANbus (CANbus)

Select the CANbus device in the project tree



Add Device

Name: CANbus_1

Action:

● Append device ○ Insert device ○ Plug device ○ Update device

String for a fulltext search   Vendor: 3S - Smart Software Solutions GmbH

Name	Vendor	Version	Description
Fieldbusses			
CANopen			
J1939			
J1939 Manager			
J1939_Manager	3S - Smart Software Solutions GmbH	3.5.15.10	J1939 Manager

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

Please select a device from the list above.

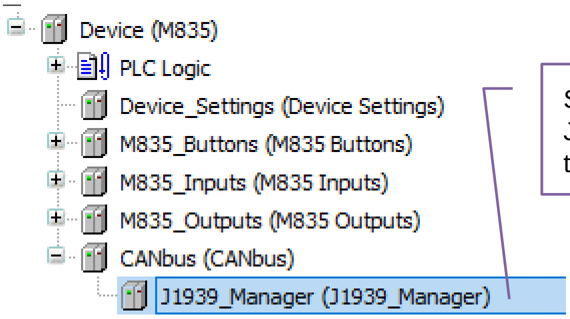
(You can select another target node in the navigator while this window is open.)

Select J1939 Manager

And add it to the project

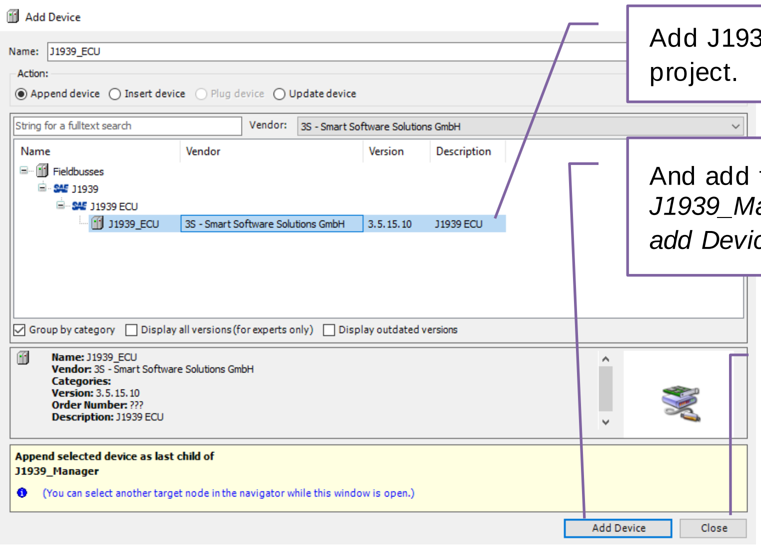
DO NOT close the window yet……

Keep the *Add Device* window open.



Device (M835)
- PLC Logic
- Device_Settings (Device Settings)
- M835_Buttons (M835 Buttons)
- M835_Inputs (M835 Inputs)
- M835_Outputs (M835 Outputs)
- CANbus (CANbus)
  - J1939_Manager (J1939_Manager)

Select the J1939_Manager device in the project tree



Add Device

Name: J1939_ECU

Action:

● Append device ○ Insert device ○ Plug device ○ Update device

String for a fulltext search | Vendor: 3S - Smart Software Solutions GmbH

Name	Vendor	Version	Description
Fieldbusses			
J1939			
J1939 ECU			
J1939_ECU	3S - Smart Software Solutions GmbH	3.5.15.10	J1939 ECU

☑ Group by category ☐ Display all versions (for experts only) ☐ Display outdated versions

Name: J1939_ECU
Vendor: 3S - Smart Software Solutions GmbH
Categories:
Version: 3.5.15.10
Order Number: ???
Description: J1939 ECU

Append selected device as last child of
J1939_Manager

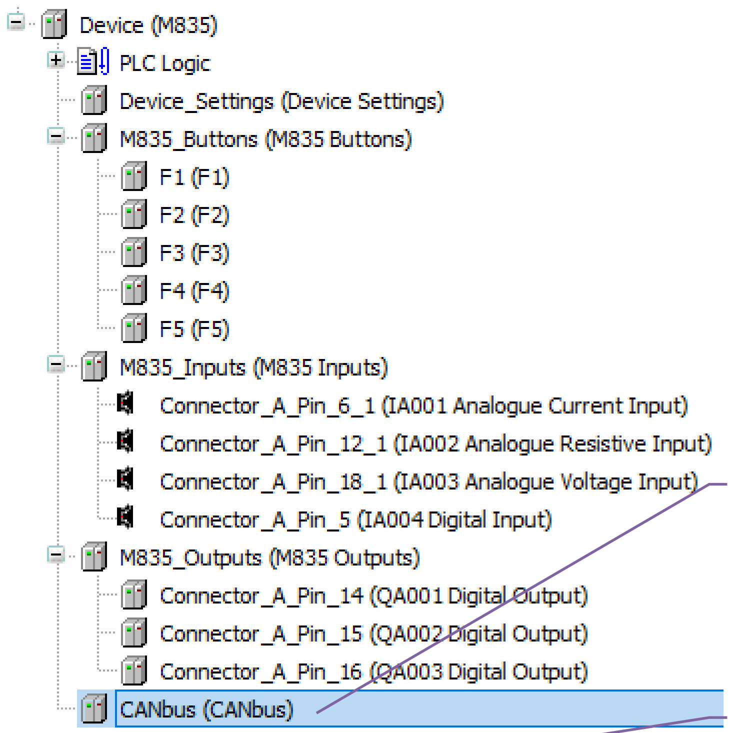
(You can select another target node in the navigator while this window is open.)

Add J1939 ECU to the project.

And add to it the *J1939_Manager* from the *add Device* window.

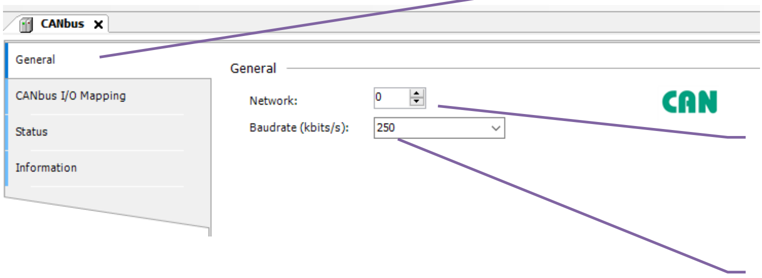
Close Window

NOTE: DSEM835 has one CAN port. Within the CODESYS CAN device, set *Network* to *0,*



- Device (M835)
  - PLC Logic
  - Device_Settings (Device Settings)
  - M835_Buttons (M835 Buttons)
    - F1 (F1)
    - F2 (F2)
    - F3 (F3)
    - F4 (F4)
    - F5 (F5)
  - M835_Inputs (M835 Inputs)
    - Connector_A_Pin_6_1 (IA001 Analogue Current Input)
    - Connector_A_Pin_12_1 (IA002 Analogue Resistive Input)
    - Connector_A_Pin_18_1 (IA003 Analogue Voltage Input)
    - Connector_A_Pin_5 (IA004 Digital Input)
  - M835_Outputs (M835 Outputs)
    - Connector_A_Pin_14 (QA001 Digital Output)
    - Connector_A_Pin_15 (QA002 Digital Output)
    - Connector_A_Pin_16 (QA003 Digital Output)
  - CANbus (CANbus)

Double Click the CANbus device.



General

Network: 0

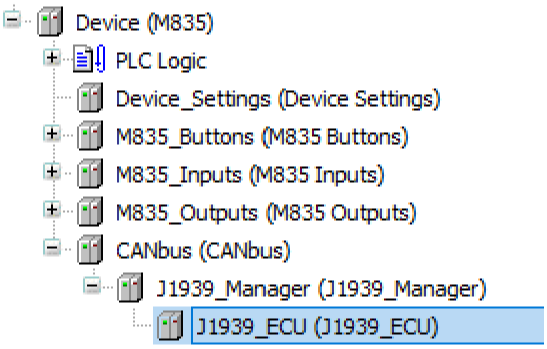
Baudrate (kbits/s): 250

Select the *General* tab

Ensure *Network: 0*

Set *Baudrate* as required.

The project is now able to communicate with one J1939 ECU. Other devices may be added as required to allow communication with other J1939 ECUs on the same CAN.



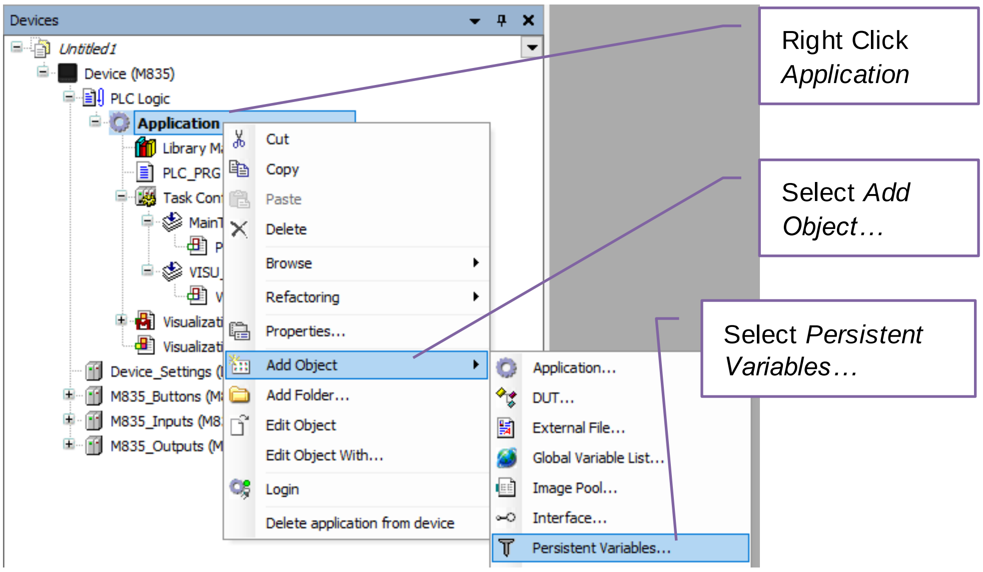
Device (M835)
- PLC Logic
- Device_Settings (Device Settings)
- M835_Buttons (M835 Buttons)
- M835_Inputs (M835 Inputs)
- M835_Outputs (M835 Outputs)
- CANbus (CANbus)
  - J1939_Manager (J1939_Manager)
    - J1939_ECU (J1939_ECU)

For further information, please contact support@deepseaelectronics.com.

4.13 USING PERSISTENT VARIABLES IN THE PROJECT

> **NOTE: For details how to enable and configure the storage interval of Persistent Variables, see section entitled *Connecting to CODESYS* | *Device Settings* | *Persistence Settings* elsewhere in this document.**

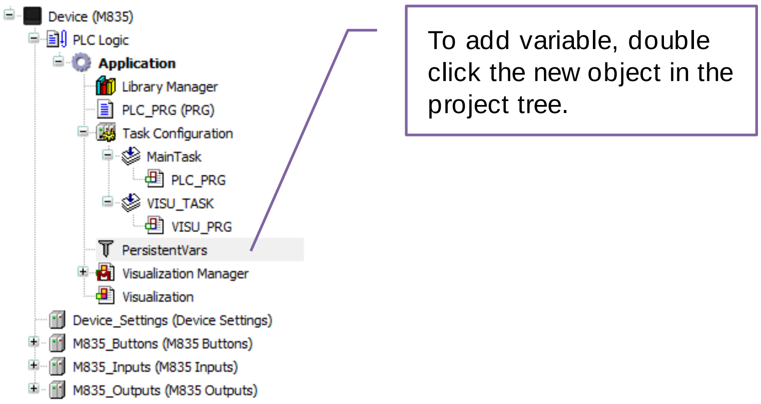
Variables placed in a Persistent Variables object are stored at regular intervals and maintained when the device is powered off. The variables are automacally loaded when the device powers up.



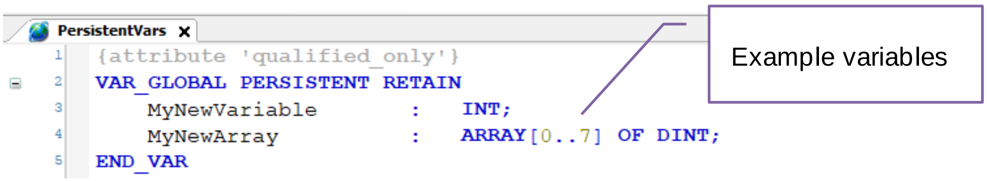
Right Click *Application*

Select *Add Object…*

Select *Persistent Variables…*



To add variable, double click the new object in the project tree.



```
{attribute 'qualified_only'}
VAR_GLOBAL PERSISTENT RETAIN
    MyNewVariable       :   INT;
    MyNewArray          :   ARRAY[0..7] OF DINT;
END_VAR
```

Example variables

Within the POU the variables are addressed as normal. In this example we've named the object *PersistentVars* however any IEC compatible name can be used. Multiple *Persistent Variable* objects can be used as required.

Example:

```
PersistentVars.MyNewVariable:=2512;
PersistentVars.MyNewArray[2]:=32167;
```

4.14 ALTERNATIVE METHODS TO LOAD THE APPLICATION

While a live connection to the device by Ethernet is the most common method to load the application, an alternative exists as detailed below. This requires the creation of a 'Boot Application' and the use of *DSEServicetool PC Software.*

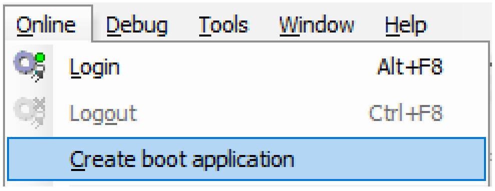
4.14.1 CREATION OF THE BOOT APPLICATION

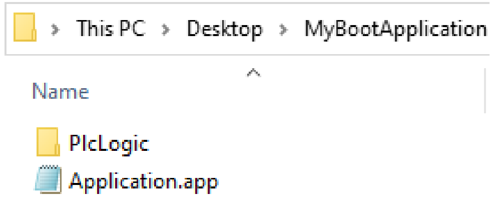
NOTE: For details using DSEServicetool PC Software, including for Batch Programming of ten devices at the same time, see DSE publication *057-265 DSEServicetool PC Software Manual.*

NOTE: The boot application includes the folder *PlcLogic*. This contains all images, and associated visualisation files.

Within CODESYS select *Online* | *Create boot application*

Browse to select the location to store the files. It is convenient to put the files into a new (empty) folder. The example, to the right shows the boot application files in the new folder.
All these files (including the sub folder) make up the boot application and must be kept together.
You can change the name of Application.app but all other files (including the sub folder) must not be altered.



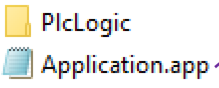


4.14.2 UPDATE USING DSESERVICETOOL AND ETHERNET CONNECTION

NOTE: For details using DSEServicetool PC Software, including for Batch Programming of ten devices at the same time, see DSE publication *057-265 DSEServicetool PC Software Manual.*

NOTE: Batch Download capability is present from DSEM835 Serial Number 8958811 onwards.

Within DSEServicetool PC Software scan for and connect to the device, then select the file (default name is *Application.app* though this may have been changed) and transfer it to the device.



.app (Application) file created by CODESYS Online | Create Boot Application

5 DSEM835 CODESYS ERROR CODES

DSEM835 returns error codes to CODESYS when appropriate. Individual bits are set within the returned value to indicate one or more error conditions. This is mapped to a variable if required and is available to view within CODESYS under the *Device Settings I/O Mapping*.

Example:

A *Device* error value of 2 (00000010 in binary) indicates *Over Temperature.*

5.1 DEVICE

MSB	Bit						LSB
8	7	6	5	4	3	2	1
Output Reference Outside Limits	Reserved	Reserved	Reserved	Reserved	Under Voltage Supply	Over Temperature	Error

5.2 INPUTS

Input Configuration	MSB			Bit				LSB
	8	7	6	5	4	3	2	1
Digital	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Error
Frequency	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Freq Over Range	Reserved	Error

5.3 DIGITAL OUTPUTS

Output Configuration	MSB			Bit				LSB
	8	7	6	5	4	3	2	1
Digital	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Error

6 FIRMWARE UPDATE

▲**NOTE: While the device firmware is updatable, the device bootloader is fixed during production and cannot be updated.**

▲**NOTE: For full details of firmware and bootloader version, see *M Series Release Notes* available from the DSE Website (*PC Software/Software Manuals* page of the relevant controller).**

DSEM835 firmware updates are performed using DSEServicetool PC Software in conjunction with PCAN USB device. For further details how to perform this operation, see DSE publication *057-265 DSE Servicetool PC Software Manual.*

7 CABLES, CONNECTORS, HARNESSES AND SPARE PARTS

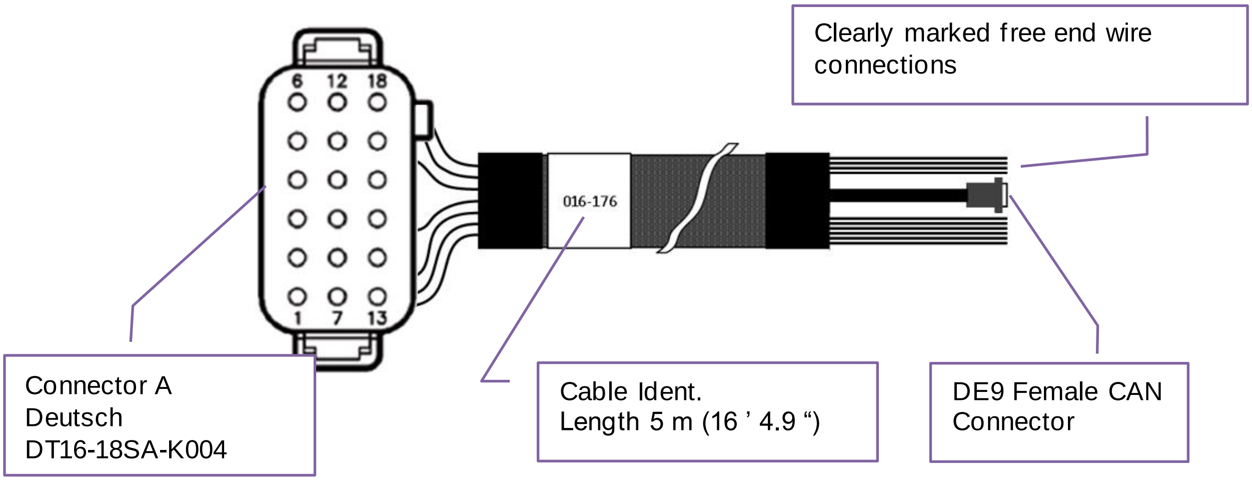
Description	DSE Part	Manufacturer Part	Manufacturer
DSEM835 Connector A	007-850	DT16-18SA-K004	TE / Deutsch
Connector Female Pin Crimp (0.5 mm^2 to 1.0 mm^2)	N/A	0462-201-16	TE
Connector Female Pin Crimp (2 mm^2)		0462-209-16	TE
DSEM835 Connector Harness Kit (See Below for Details)	016-176	N/A	DSE
DSEM835 Connector Configuration Harness. (See Over for Details)	016-177	N/A	DSE
Pin Blank Inserts (Seals unused connector pins)	N/A	114017	TE
Belden 9841 (CAN Cable)	016-030	9841	Belden
PCAN-USB PC Configuration Interface	016-179	IPEH-002021 or IPEH-002022	PEAK-System Technik GmbH
Rear Case Ring Nut	020-1058-01	N/A	DSE

7.1 DSEM835 CONNECTOR HARNESS (016-176)

NOTE: As the CAN connection of this harness is designed to be a spur onto an existing CAN backbone, there is no CAN termination resistor included in this harness.

DSE Part 016-176 consists of a cable with connector fitted at one end, with cable marking to identify the wires at the other end.

Item	Connector A
Assembly Ident	007-850
AMP Connector	DT16-18SA-K004
No of Connections	18
Wire size	0.5 mm^2 (AWG 20)
Wire Colour	Black
Wire Idents	1 to 18
Connector Female Pin Crimp (0.5 mm^2 to 1.0 mm^2)	0462-201-16
Connector Female Pin Crimp (2 mm^2)	0462-209-16



Clearly marked free end wire connections

Connector A
Deutsch
DT16-18SA-K004

Cable Ident.
Length 5 m (16 ’ 4.9 “)

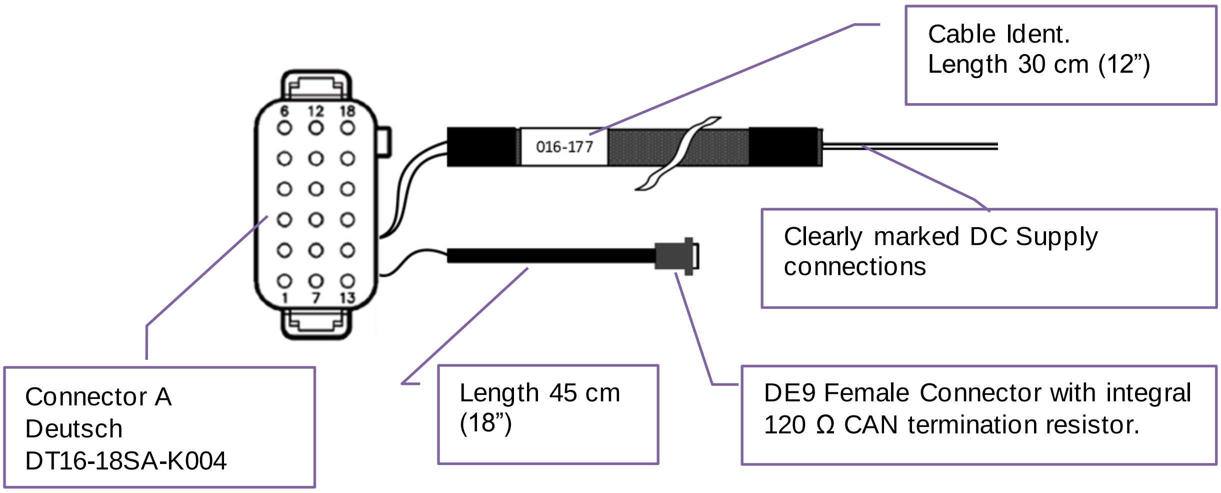
DE9 Female CAN Connector

7.2 DSEM835 CONFIGURATION HARNESS (016-177)

NOTE: DSEM835 Configuration Harness 016-177 must be used in conjunction with Peak Systems USB CAN interface IPEH-002021 or IPEH-002022 from https://www.peak-system.com

NOTE: During configuration by CAN using this harness, DSEM835 should be connected solely to the PC CAN. To assist this connection, *DSEM835 Configuration Harness* includes integral 120 Ω CAN termination resistor.

DSE Part 016-177 consists of a minimal cable with connector fitted at one end with DC supply wires and DE9 CAN connector at the other end. This is designed to connect to the DE9 connector of the USB CAN interface.



Cable Ident.
Length 30 cm (12”)

Clearly marked DC Supply connections

Connector A
Deutsch
DT16-18SA-K004

Length 45 cm
(18”)

DE9 Female Connector with integral 120 Ω CAN termination resistor.

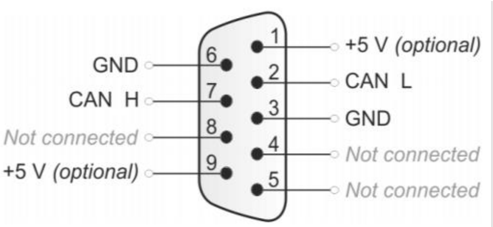
Description	Connector A	Termination
DC Positive	7	Single wire
DC Negative	1	Single wire
CAN SCR	2	DE9 Pin 5
CAN L	3	DE9 Pin 2
CAN H	4	DE9 Pin 7

7.3 PCAN-USB CONNECTION DETAILS

▲**NOTE: Where a DSE Configuration or Connection Harness is not used, connection details for PCAN-USB are shown below. PCAN-USB supplier details are given in the section entitled *Maintenance Spares, Repair and Servicing* elsewhere in this document.**

▲**NOTE: DSE Stock and supply PCAN-USB IPEH-002021. DSE Part number 016-179. Contact sales@deepseaelectronics.com.**

Connect PCAN-USB to DSEM835 using connections for CAN H, CAN L and GND. For suitable connection looms from DSE, see sections entitled *DSEM835 Connector Harness* and *DSEM835 Configuration Harness*.



Shown looking at the PCAN-USB interface from the ‘outside’

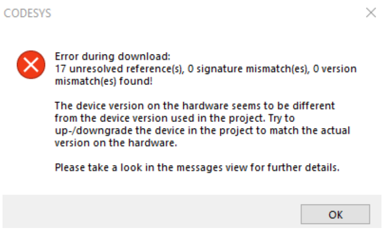
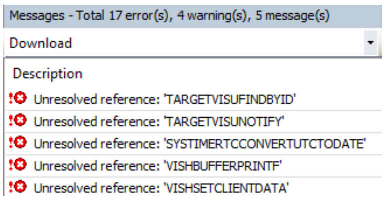
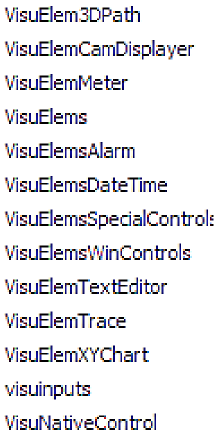
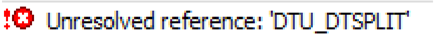
8 TROUBLESHOOTING

The below are suggestions answering frequently questions about the device.

8.1 COMMUNICATION AND APPLICATION TRANSFER

Symptom	Suggestion
'Network Error. No Route to Host' is shown when scanning for the device.	• Ensure CODESYS Gateway is enabled. Perhaps it has been previously disabled for use with DSEServicetool?
DSEM835 is not communicating with CODESYS	• Ensure correctly terminated 120 Ω impedance CAN rated cable is used with correctly fitted 120 Ω termination resistors. • Where possible, use DSEM835 Configuration Harness, DSE part 016-177. This has integral termination resistor. • Ensure no other devices are communicating on the CAN. It is recommended to use a dedicated CAN link for application transfer where possible. Otherwise disable (power off) other CAN devices on the bus. • Carefully follow the procedure in the section entitled *Connecting to CODESYS | CAN Connection* elsewhere in this document. It is important that each step is followed correctly.
Transfer of files to and from DSEM835 is slow. (CAN issues?)	First Check for CAN issues: • Ensure no other devices are connected to the CAN. Where removal is not practical, ensure they are disabled. A bus with no additional CAN traffic results in a faster data transfer with DSEM835. • Ensure correctly 120 Ω impedance CAN rated cable is used with correctly fitted 120 Ω termination resistors. • Where possible, use DSEM835 Configuration Harness, DSE part 016-177. This has integral termination resistor. • A correctly connected system allows a transfer of a 2 MB CODESYS application in ten minutes. If transfer appears excessively slower than this, there is an issue with the CAN connection, or other devices are transmitting on the bus, leaving less bandwidth available for CODESYS-M835 communications. If all the above has been checked, see Application Size below.

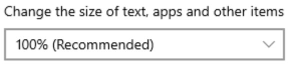
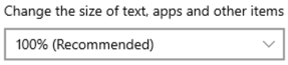
Symptom	Suggestion
Transfer of files to and from DSEM835 is slow. (Application Size?)	If CAN issues have been checked as above, the slow transfer is due to the size of the application. Tips for economising on the application size include: • Remove any unneeded images from the project Image Pools. Uncompressed bitmaps take up valuable resources and increase project file size, hence slowing project transfer. • Try to avoid full screen images. Instead use several smaller images placed upon a solid colour Visu background. This usually occupies less project space than a full screen image, hence the project is smaller (faster) to transfer. • Where using preinitialised STRING variables, consider the length of the STRING to keep the application size down. By default, CODESYS assigns 80 bytes per string, these are padded by empty bytes if not filled, increasing the size of the application. For example use myVariable : STRING(11):=”Hello World”; (11 bytes) instead of myVariable : STRING:=’Hello World’; (80 bytes). Use VAR CONSTANT for variables that are not altered in the code to ensure that the string cannot be increased in size incorrectly later.

Symptom	Suggestion
Unresolved references errors appear *after* the application has been sent to the device and do not appear during the build process. **NOTE: The exact message differs depending upon the specific issue.** Example error message:  CODESYS Error during download: 17 unresolved reference(s), 0 signature mismatch(es), 0 version mismatch(es) found! The device version on the hardware seems to be different from the device version used in the project. Try to up-/downgrade the device in the project to match the actual version on the hardware. Please take a look in the messages view for further details. Example from *Messages* view:  Messages - Total 17 error(s), 4 warning(s), 5 message(s) Download Description Unresolved reference: 'TARGETVISUFINDBYID' Unresolved reference: 'TARGETVISUNOTIFY' Unresolved reference: 'SYSTIMERTCCONVERTUTCTODATE' Unresolved reference: 'VISHBUFFERPRINTF' Unresolved reference: 'VISHSETCLIENTDATA'	The answer to this issue is often determined by the statements in the *Messages* window with *Download* selected. Solutions are (but are not limited to) the following: • Where messages mention *Vis, Visu or VisuElems,* this is caused by *VisuElems* libraries being erroneously included in the project. Visit the Library Manager and **delete** the following libraries from the project. These are not supported by DSEM835 and may be left within your project if you have incorrectly converted a project meant for a different device: If deleting them causes additional errors in the project code, then these errors show use of a function within one of these unsupported functions. This must be removed. Contact support@deepseaelectronics.com if you need assistance finding an alternative function.  VisuElem3DPath VisuElemCamDisplayer VisuElemMeter VisuElems VisuElemsAlarm VisuElemsDateTime VisuElemsSpecialControls VisuElemsWinControls VisuElemTextEditor VisuElemTrace VisuElemXYChart visuinputs VisuNativeControl **DSEM835 requires TVL (Target Visu Light) library for Visualisations so do not delete that library by mistake.** • If the *Messages* | *Download* view shows errors relating to *DT, DateAndTime,* or *RTC* such as (but not limited to)  Unresolved reference: 'DTU_DTSPLIT' Then ensure the project has no function calls attempting to read the (non-existent) Real Time Clock of the device. M835 does not include a Real Time Clock. • Ensure the project has no function attempting to utilise a CODESYS VisuElems Text List. DSM835 does not support Text Lists, errors are given should your project attempt to use them. • Ensure CODESYS libraries used are no later than version 3.5.12.0. Attempting to use a later version library may generate an error message caused by DSEM835 containing only CODESYS 3.5.12.0 (and earlier) functionality.
The device fails to respond after transferring the application, but before the application is started.	• Use DSEServicetool to *Erase Application* and *Erase Settings* to return the device to factory settings and restore normal operation.

Symptom	Suggestion
	• Ensure that all RETAIN variables are correctly placed in a CODESYS Persistent Variables object as shown in the section entitled *Using Persistent Variables* elsewhere in this document. If placed within a POU or other place, the application cannot be started.
A full download to the device takes place upon connection, despite no changes being made to the application code. This impedes the use of the debugger.	• This is prevented from within CODESYS by selecting *Build* | *Generate code* prior to logging in to the device.
Debugging with CODESYS is slow to update.	CAN provides a slower debugger update when compared with Ethernet based devices. You can help to keep data transfer to a minimum, and increase update rates of the debugger by adopting the following techniques: • Ensure all unnecessary project windows are closed. • For fastest operation, add important variables to the ‘Watch’ window within CODESYS. Then close all project windows except the ‘Watch’ window. • Ensure all unnecessary CAN traffic is stopped. This releases bandwidth for use with the CODESYS debugger. • Ensure CAN network is correctly made, utilising 120 Ω impedance cable and correctly terminated with 120 Ω termination resistors.

8.2 VISUALISATION AND DISPLAY

Issues with Visualisation and Display are often caused by unfamiliarity with the TVL (Target Visu Light) functionality supported by DSEM835. TVL differs from VisuElems utilised by other DSE M Series displays.

Symptom	Suggestion
Text appears on the display smaller than selected.	• This can occur when a 'decimal' font size is unknowingly used or when the font itself is not present on the device. • DSEM835 uses CODESYS TVL (*TargetVisuLight)* that does not support the resizing of fonts. Integer sizes must be selected. For further details, see section entitled *Connecting to CODESYS* | *Text* elsewhere in this document. • Within Windows Settings *System* | *Display* | *Scale and Layout,* ensure the size is set to *100 %*, even if this is not the Recommended size. Failure to do so results in CODESYS sending a scaled font to the device, which appears different to that selected within CODESYS:  Change the size of text, apps and other items 100% (Recommended)
Text appears on the display larger (or smaller) than selected	Within Windows Settings *System* | *Display* | *Scale and Layout,* ensure the size is set to *100 %*, even if this is not the Recommended size. Failure to do so results in CODESYS sending a scaled font to the device, which appears different to that selected within CODESYS:  Change the size of text, apps and other items 100% (Recommended)
Some visualisation characters are missing	This occurs where the selected font character is not present on the device. Adjusting the options for *Font Download* will correct this. For further details, see section entitled ***Installing Fonts on the M835*** elsewhere in this document.
Images are not appearing on the device display.	• See section entitled *Using Custom Images on the Display* elsewhere in this document. • It is possible that the device memory is full and there is no room for more images. To ensure pre-existing, unneeded files are not present, use DSEServicetool to erase the device and then try to send the program again.
Colours of shapes and borders are not appearing as selected.	• Ensure *DSEStyle* is selected, or a style derived from it. For details, see section entitled *Styles* elsewhere in this manual. • After changing style or colours, select *Build* | *Clean All* to clear the build cache and force a full compile/build at the next login to the device.
Parts of the application visualisation do not appear correctly after application transfer.	• Select *Build* | *Clean All* to clear the build cache and force a full compile/build at the next login to the device. • Power cycle the device before transferring the application. This ensures the memory is clear before starting. • Power cycle the device after transferring the application. This ensures the memory is clear, not used by the CODESYS debugger and ensures the CODESYS debugger is not affecting the CAN or processor operation of the application.

8.3 PERSISTENT VARIABLES

Symptom	Suggestion
How do I use Persistent Variables?	To configure and utilise Persistent variables: • Ensure *Persistence Settings* are configured as required. See section entitled *Connecting to CODESYS* | *Device Settings* | *Persistence Settings* • Add a Persistent Variables object to the project. Right Click *Application* and select *Add Object* | *Persistent Variables.* Variables placed here are stored as selected above and persist through a power off.

8.4 INPUTS AND OUTPUTS

Symptom	Suggestion
Inputs are not measuring correctly.	DSEM835 inputs can measure 0 V to 10 V maximum. This includes inputs configured as *digital.* Where a higher voltage is applied, it is clamped. No damage will occur so long as the voltage applied is not above the DC supply voltage on Pin A7. Ensure thresholds for digital inputs are set below 10 V. For example, 2 V low and 6 V high thresholds are suitable for most applications where the input is either from a 12 V or 24 V supply. VREF output is provided for use as a voltage reference for input devices such as sensors and switches (max 100 mA). For details how to enable and configure this output source, see section entitled *Connecting to CODESYS* | *Device Settings* | *Voltage Reference* elsewhere in this document.
How can I connect an RTD or thermocouple?	DSEM835 resistive Inputs have a range greater than that required for measurement of RTDs or thermocouples. Suggested alternatives are: • Temperature sensors with a 0 V to 5 V, 0 V to 10 V, or 4 mA to 20 mA output across the range of temperatures to be measured • Sensors with CAN support. • Convert the RTD or thermocouple signal to one of the above mentioned alternatives.

8.5 MISCELLANEOUS

Symptom	Suggestion
CAN is not operating correctly after application transfer.	• Select *Build* | *Clean All* to clear the build cache and force a full compile/build at the next login to the device. • Power cycle the device before transferring the application. This ensures the memory is clear before starting. • Power cycle the device after transferring the application. This ensures the memory is clear, not used by the CODESYS debugger and ensures the CODESYS debugger is not affecting the CAN or processor operation of the application.
Tasks are taking longer to complete than usual with the same application.	

9 MAINTENANCE AND WARRANTY

The device is *Fit and Forget*. As such, there are no user serviceable parts within the controller.
In the case of malfunction, you should contact your original equipment manufacturer (OEM).

DSE Provides limited warranty to the equipment purchaser at the point of sale. For full details of any applicable warranty, refer to the original equipment supplier (OEM).

10 DISPOSAL

10.1 WEEE (WASTE ELECTRICAL AND ELECTRONIC EQUIPMENT)

If you use electrical and electronic equipment you must store, collect, treat, recycle and dispose of WEEE separately from your other waste

This Page is Intentionally Blank

This Page is Intentionally Blank