



DSEControl



DEEP SEA ELECTRONICS

DSEM812 CODESYS Manual

Document Number: 057-318

Author: Anthony Manton



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

Sales Tel: +44 (0) 1723 890099

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

DSEM812 CODESYS Manual

© Deep Sea Electronics Ltd.

All rights reserved. No part of this publication may be reproduced in any material form (including photocopying or storing in any medium by electronic means or other) without the written permission of the copyright holder except in accordance with the provisions of the Copyright, Designs and Patents Act 1988.

Applications for the copyright holder's written permission to reproduce any part of this publication must be addressed to Deep Sea Electronics Ltd at the address above.

The DSE logo and the name DSEControl® are UK registered trademarks of Deep Sea Electronics Ltd.

Any reference to trademarked product names used within this publication is owned by their respective companies.

Deep Sea Electronics Ltd reserves the right to change the contents of this document without prior notice.

Revision History

Issue No.	Comments
1	First Release
2	Added WebBrowser toolbox item
3	Updated to add support for M812 V2 (CODESYS 3.5.17)
4	Updated to add support for M812 V3 (CODESYS 3.5.19)

TABLE OF CONTENTS

Section	Page
1 Introduction	5
1.1 CLARIFICATION OF NOTATION.....	6
1.2 GLOSSARY OF TERMS.....	6
1.3 RELATED INFORMATION	8
1.3.1 TECHNICAL INFORMATION.....	8
1.4 SAFETY INSTRUCTIONS	9
1.4.1 GENERAL	9
1.4.2 INSTALLATION NOTES	9
2 CONNECTING TO CODESYS.....	10
2.1 CODESYS VERSION	10
2.2 START NEW PROJECT	11
2.3 CONNECT BY AUTOMATIC SCAN	12
2.4 CONNECT BY IP ADDRESS	13
2.5 CONFIGURE SETTINGS AND MONITOR THE DEVICE	14
2.5.1 DEVICE SETTINGS PARAMETERS	15
2.5.1.1 NEW NETWORK CONFIGURATION	15
2.5.1.2 VOLTAGE REFERENCE	16
2.5.1.3 SCREEN ROTATION.....	16
2.5.1.4 PERSISTENCE SETTINGS.....	17
2.5.1.5 MANUAL SHUTDOWN	17
2.5.1.6 OTHER PARAMETERS.....	17
2.5.2 DEVICE SETTINGS I/O MAPPING.....	18
2.6 ADD INPUTS, OUTPUTS AND BUTTONS TO THE PROJECT (GUI).....	19
2.6.1 BUTTONS	20
2.6.1.1 BUTTON LOCATION	20
2.6.1.2 BUTTON POSITION ON THE VISUALISATION	20
2.6.1.3 BUTTON SETTINGS	21
2.6.2 INPUTS AND OUTPUTS	22
2.6.3 DIGITAL INPUT PARAMETER CONFIGURATION	23
2.7 USING THE DISPLAY IN THE PROJECT	24
2.7.1 CLIENT ANIMATIONS	25
2.7.2 INSTALLING FONTS ON THE DEVICE	26
2.7.3 USING CUSTOM IMAGES ON THE DISPLAY	26
2.7.3.1 ADDING AN IMAGE POOL.....	26
2.7.3.2 ADDING IMAGES TO THE IMAGE POOL	27
2.7.3.3 USING THE IMAGE POOL ON THE DISPLAY	28
2.7.4 UTILISING THE TOUCHSCREEN AND WEBVISU	29
2.7.4.1 SWITCHES, SLIDERS, COMBO BOXES, RADIO BUTTONS.....	29
2.7.4.2 NUMBER AND TEXT ENTRY.....	30
2.7.5 WEBVISU ACCESS.....	32
2.7.5.1 INTERACTIONS WITH CURRENTVISU VARIABLE.	32
2.7.6 USING THE WEBBROWSER	33
2.7.6.1 WEBBROWSER VIRTUAL KEYBOARD	34
2.7.7 DSE ACTIVEX COMPONENTS.....	35
2.7.7.1 ONLINE DOCUMENTATION	36
2.7.7.2 DISPLAYING AN ANALOGUE (COMPOSITE) CAMERA.....	37
2.7.7.3 DISPLAYING A TCP/IP CAMERA	37
2.7.7.4 USING THE MOVIE PLAYER.....	38
2.7.7.5 USING THE CAMERA / VIDEO OVERLAY	40
2.8 USING THE USB DRIVE WITHIN APPLICATIONS.....	42
2.9 USING PERSISTENT VARIABLES IN THE PROJECT	43
2.11 USING NETWORK VARIABLES IN THE PROJECT	44
2.12 USING CAN IN THE PROJECT	45
2.12.1 CODESYS CAN	45
2.12.2 DSE_CAN COMPILED LIBRARY	45

2.13	USING MODBUS IN THE PROJECT	46
2.13.1	MODBUS TCP MASTER (CLIENT).....	46
2.13.1.1	DSE GENCOMM MODBUS PROTOCOL.....	46
2.13.2	MODBUS TCP SERVER.....	46
2.14	USING GPS IN THE PROJECT	47
2.14.1	OBTAINING A POSITION.....	47
2.14.2	DISPLAYING THE POSITION ON THE MAP.....	48
2.14.2.1	MAP STRUCTS.....	49
2.14.2.2	POI (POINTS OF INTEREST).....	50
2.15	ALTERNATIVE METHODS TO LOAD THE APPLICATION	51
2.15.1	USB.....	51
2.15.2	DSESERVICETOOL PC SOFTWARE.....	51
2.15.2.1	CREATION OF THE BOOT APPLICATION.....	51
2.15.2.2	UPDATE USING DSESERVICETOOL AND ETHERNET CONNECTION.....	51
3	DSEM812 CODESYS ERROR CODES	52
3.1	DEVICE.....	52
3.2	ANALOGUE INPUTS.....	52
3.3	DIGITAL INPUTS.....	53
3.4	DIGITAL OUTPUTS.....	53
4	TROUBLESHOOTING	54
5	MAINTENANCE AND WARRANTY	55
6	DISPOSAL	55
6.1	WEEE (WASTE ELECTRICAL AND ELECTRONIC EQUIPMENT).....	55
7	MISCELLANEOUS	55

1 Introduction

This document details the operation and setup requirements of the DSEM812 CODESYS Controller and Display, part of the DSEControl® range of products.

Legacy DSEM812 V1 is 'greyed out' to draw attention to its deprecation.

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 the product, operators and machinery.




A robust metal case designed for chassis mounting houses the module. Connections are via locking plug and sockets.

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 DSEM812 to control the machine to which it is connected. The Application within the DSEM812 is designed and provided by the manufacturer of the complete machine.
Boot Application	A special file created by CODESYS that allows a device firmware to be updated without requiring CODESYS to perform the update. See section entitled <i>Alternative Methods to Load the Application</i> elsewhere in this document.
Bootloader	The Bootloader is the program within the DSEM812 responsible for loading the Operating System and handling updates by .pkg files. Bootloader operations are carried out in <i>Recovery</i> mode.
CAN	Control Area Network. A high-speed data transmission system used extensively within the Automotive and Off-Highway industries.
CODEC	(COder / DECoder) Video CODEC refers to the format of the video data within the video file.
CODESYS (Previously stylised as CoDeSys)	Integrated Development Environment for programming controller applications according to the international industrial standard IEC 61131-3. DSEM812 supports CODESYS V3.5
ECU	Electronic Control Unit. For example, the DSEM812 device.
Firmware	The Firmware of the DSEM812 is the Operating System of the DSEM812 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.
Ixyyy	An Input, where x is the connector and yyy is the input number. For example, IC003 means Input 3 on Connector C.
MODBUS TCP Client	<i>Client</i> has replaced the unacceptable legacy wording previously used. A MODBUS client device makes queries of a MODBUS server device and receives the replies. An example of a client device is a DSEM812 display making queries of a DSE Genset or DSE Engine Only controller.
MODBUS TCP Server	<i>Server</i> has replaced the unacceptable legacy wording previously used. A MODBUS server device receives queries from a MODBUS client device and replies to it. Examples of server devices are a DSE Genset or DSE Engine Only controller receiving queries from a SCADA system, DSE M812 display etc
Pkg file	A pkg file is read by the device when in <i>Recovery</i> mode. The file may contain a firmware update, compiled boot application file or other update.

Introduction

Term	Description
PLC	Programmable Logic Controller. Industrial computer used primarily for the automation of electromechanical machinery.
PWM PWMi	A digital signal is used to represent an analogue value by using Pulse Width Modulation. The mark-space ratio of a square wave changes to represent the value. Used for many control applications including proportional valves. PWM= Duty Cycle control. PWMi = Current control.
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.
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, QC002 means Output 2 on Connector C.

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
055-267	DSEM812 Datasheet
057-317	DSEM812 Installation and Operation Manual
057-336	DSE_CAN Compiled Library Software 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.
- 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 controller heatsink at the rear heats up beyond normal ambient temperature during operation. To avoid danger caused by high temperatures, protect against contact.
- 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.
- To maintain IP67 rating where connectors have unused pins, ensure the use of a suitable Blanking Insert. In the case of a completely unused connector, the plug must be inserted, fully populated with Pin Blanking Inserts. See Section entitled *Cables, Connectors, Harnesses and Spare Parts* for details.
- M12 protection plugs (supplied) must be installed in both the USB and Ethernet interfaces to ensure IP67 rating when the connectors are not in use. Tighten to 0.8 Nm (0.6 lbf ft). Where IP protection is required when the interfaces are in use, suitable O-rings must be fitted.
- The heatsink must be wired to vehicle ground to comply with EMC guidelines. A screw connection point is provided for this purpose. A metallic screw must be used to create an electrical connection to vehicle / machine ground.

2 CONNECTING TO CODESYS

 **HINT: CODESYS Online help provides instructions for the CODESYS IDE and programming languages (<https://help.codesys.com>). This manual section describes useful hints and tips specifically targeted at the DSE M Series device.**

2.1 CODESYS VERSION

DSEM812 V3+ requires CODESYS 3.5.19 IDE. While the IDE has similarities to previous versions (CODESYS 3.5.x), there are many changes in the underlying code. After installation of CODESYS 3.5.19, run CODESYS Installer from the Windows Start Menu and ensure all components are updated.

DSEM812 V2+ requires CODESYS 3.5.17 IDE. While the IDE has similarities to previous versions (CODESYS 3.5.x), there are many changes in the underlying code. After installation of CODESYS 3.5.17, run CODESYS Installer from the Windows Start Menu and ensure all components are updated.

 **NOTE: DSEM812 firmware can be Updated or Dwndated as required. Latest firmware version is available at www.deepseaelectronics.com for website registered users.**

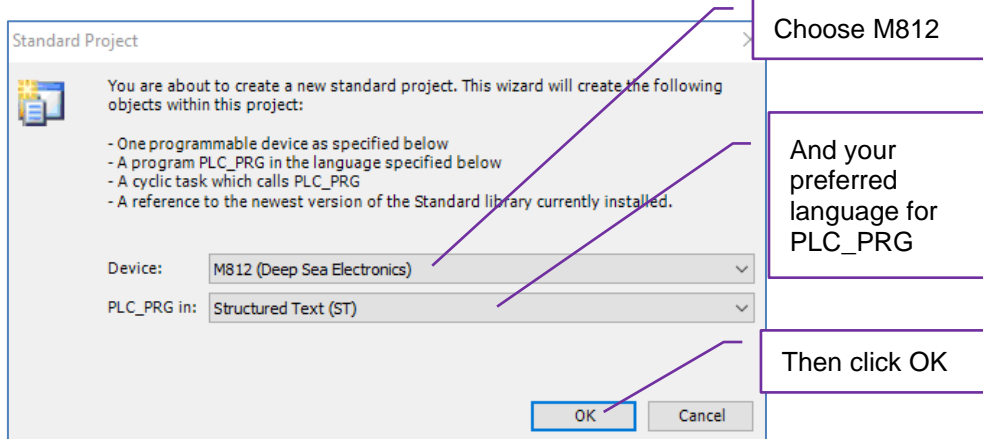
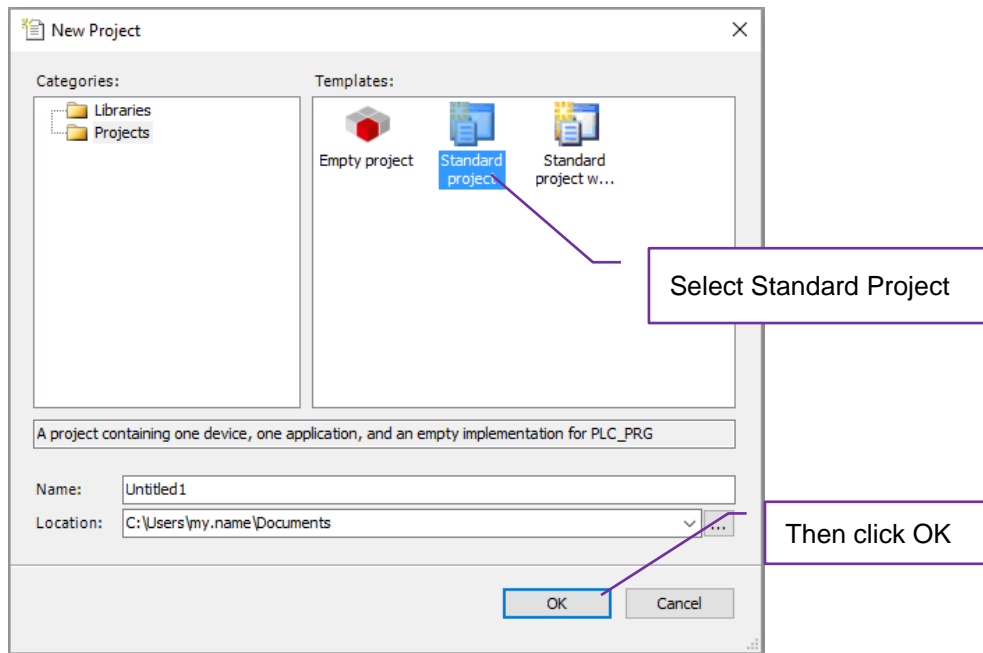
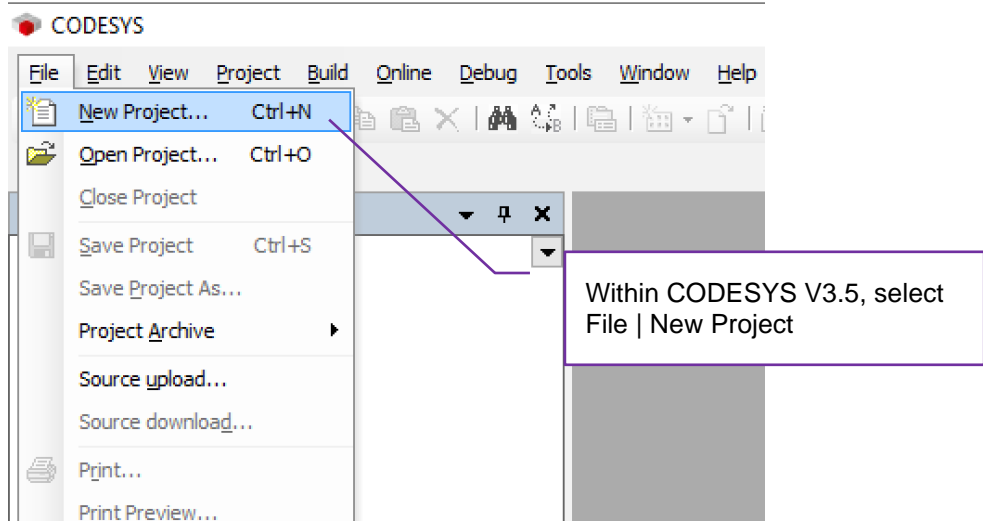
DSEM812 communicates with, and is programmed by, the CODESYS V3.5 Integrated Development Environment (IDE).

Ensure to install and use the correct CODESYS IDE depending on the version of M812 firmware you are using. Multiple CODESYS versions can exist on the same PC.

DSEM812 Firmware Version	CODESYS Runtime Version in use
V1.x.x	CODESYS 3.5.12
V2.x.x	CODESYS 3.5.17
V3.x.x	CODESYS 3.5.19

2.2 START NEW PROJECT

To begin, start a new project as shown.



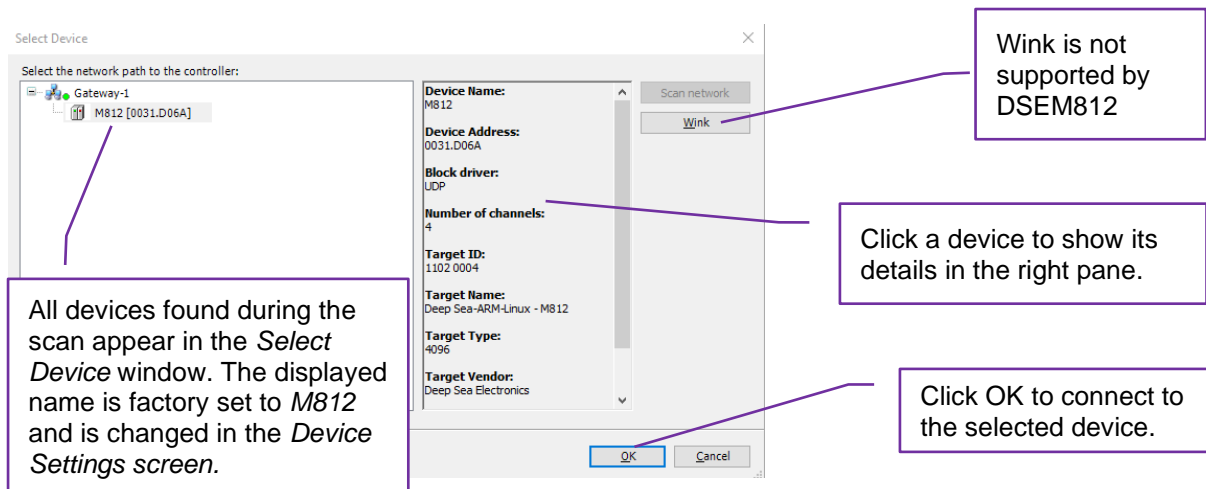
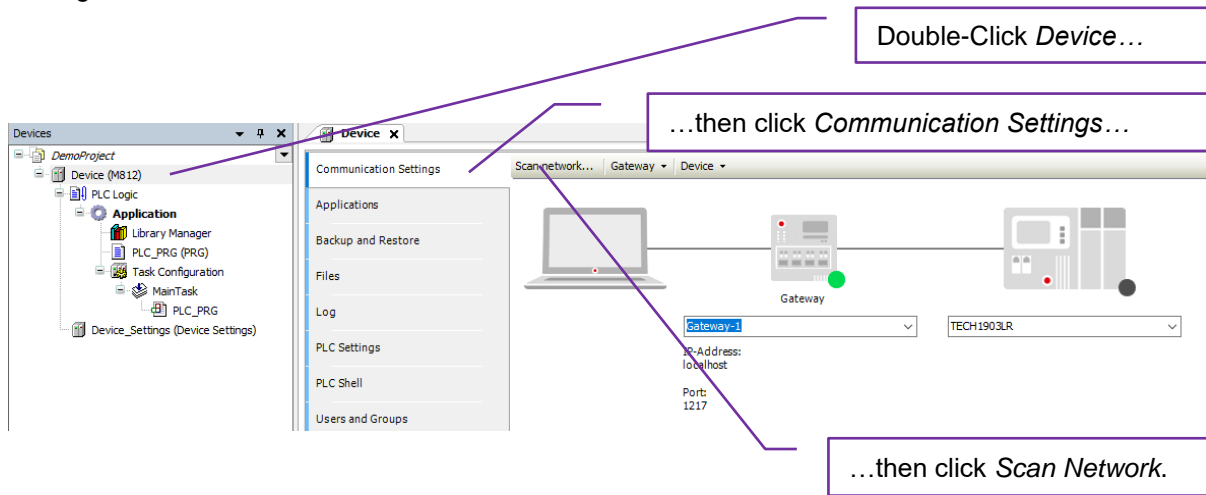
2.3 CONNECT BY AUTOMATIC SCAN

NOTE: A device in *Configuration* or *Recovery* mode cannot be found in the *Scan* and cannot connect to CODESYS.

NOTE: The *Scan* requires the device to be on the same subnet as the PC. If this is not possible, locate the IP address of the device and connect a detailed in the section entitled *Connect by IP Address* elsewhere in this document.

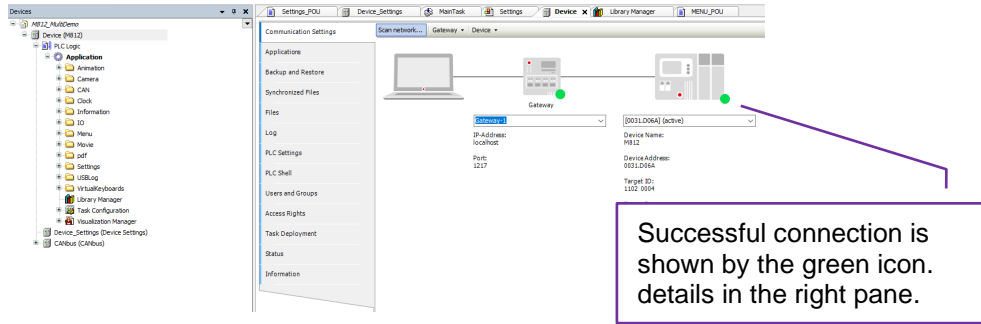
NOTE: If the IP address of the device is known, connection may also be achieved manually as detailed in the section entitled *Connect by IP Address* elsewhere in this document.

This method of connection is suitable only for devices in the same subnet as the PC. With the device connected to the same Ethernet network as the PC, Select *Device | Communication Settings* in the CODESYS V3.5 IDE.



All devices found during the scan appear in the *Select Device* window. The displayed name is factory set to *M812* and is changed in the *Device Settings* screen.

Connecting to CODESYS



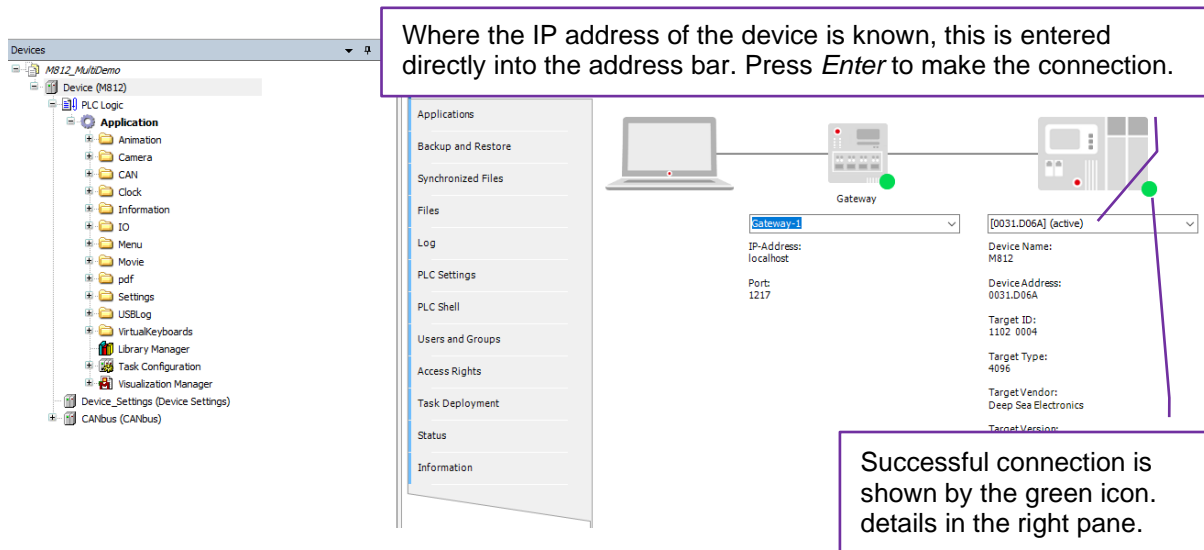
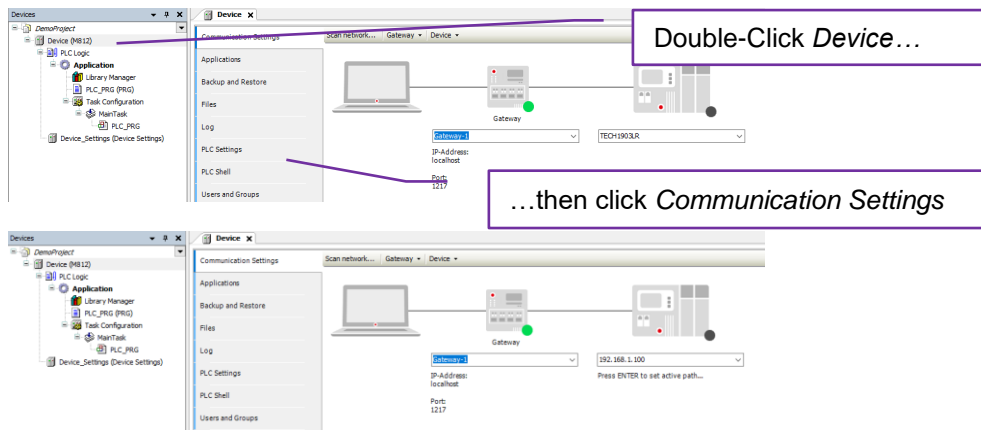
2.4 CONNECT BY IP ADDRESS

NOTE: A device in *Configuration* or *Recovery* mode cannot connect to CODESYS.

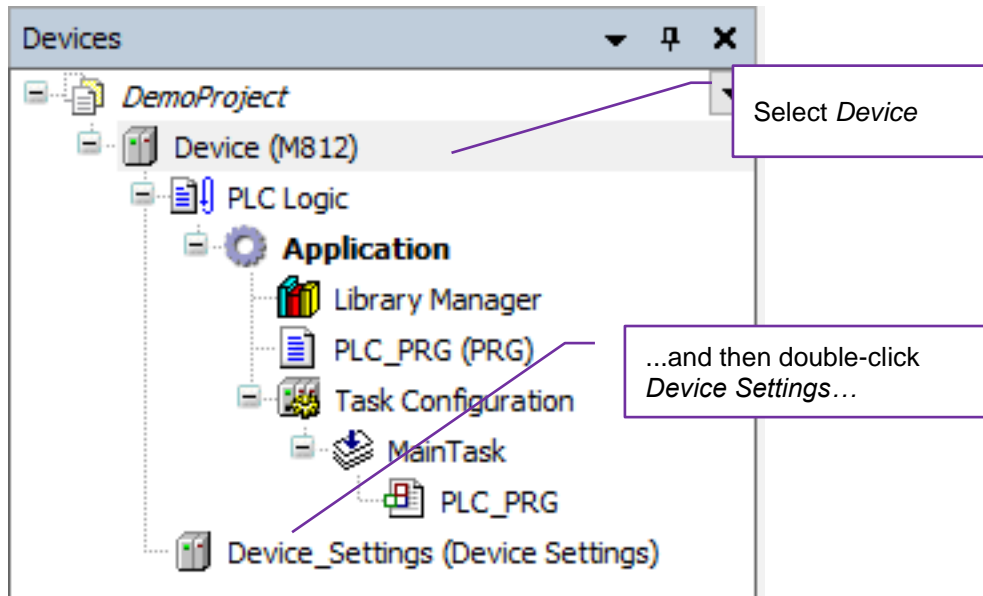
NOTE: If the IP address of the device is not known, see the section entitled *Connect by Automatic Scan* elsewhere in this document.

This method of connection is particularly suited for devices on other subnets or networks, providing the PC has a routable connection to the device.

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



2.5 CONFIGURE SETTINGS AND MONITOR THE DEVICE

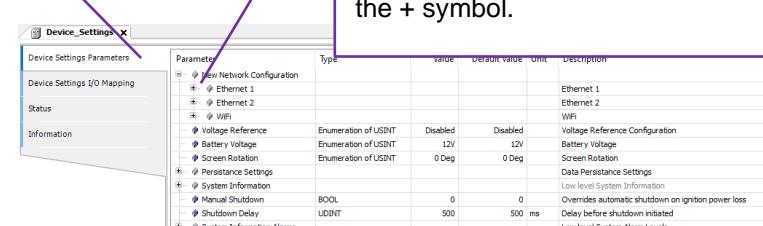


See the following subsections for details of the Device Settings pages.

2.5.1 DEVICE SETTINGS PARAMETERS

NOTE: After changing the setting of *Screen Rotation*, *DSEM812* must be power cycled for the change to take effect.

Select
Device
Settings
Parameters
...



...and then expand sections as required by clicking the + symbol.



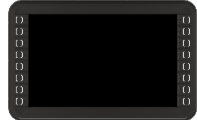

2.5.1.1 NEW NETWORK CONFIGURATION

Parameter	Description
Save Config	<p>NOTE: After making changes, choose Yes in the Save Config section to save the changes made when the project is downloaded to the DSEM812 device.</p>
Enable	<p>NOTE: Disabling Ethernet Ports that are not required within the Application improves (decreases) the start-up time of the device.</p> <p>No: This Ethernet port is disabled. Yes: This Ethernet port is disabled.</p>
Hidden SSID (Wifi only)	<p>No: The Wifi SSID is not broadcast so cannot be seen during a wifi network scan. Yes: The Wifi SSID is not broadcast so cannot be seen during a wifi network scan.</p>
DHCP Server (Wifi only)	<p>No: Dynamic Host Configuration Protocol Server is not enabled. Yes: Dynamic Host Configuration Protocol Server is enabled. DSEM812 is able to configure devices that connect to the Wifi server providing they are set to <i>DHCP</i>.</p>
Channel (Wifi only)	Wifi channel to use (1 to 11)
IP Mode (Ethernet 1,2 only)	<p>DHCP: Enables Dynamic Host Configuration Protocol. Static: Network setup is manual.</p>
IP Address	The IP address used when <i>IP Mode</i> is set to <i>Static</i> .
Subnet Mask	The Subnet Mask applied to <i>IP address</i> used when <i>IP Mode</i> is set to <i>Static</i> .
Gateway Address	The Default Gateway used when <i>IP Mode</i> is set to <i>Static</i> .
DNS Address	The IP address of the Domain Name Service used when <i>IP Mode</i> is set to <i>Static</i> .
Hostname	Hostname by which the device makes itself known on the network.
SSID (Wifi only)	The SSID of the Wifi network served by the DSEM812.
Password	Password required by clients wishing to connect to the SSID of the Wifi network served by the DSEM812.

2.5.1.2 VOLTAGE REFERENCE

Parameter	Description
Voltage Reference	<p>Enables the <i>VREF</i> output. This is an auxiliary output supply used for example, to supply external input sensors.</p> <p>Disabled: The outputs is OFF. 5V: The output gives 5 V dc. 10V: The output gives 10 V dc.</p>


2.5.1.3 SCREEN ROTATION

Parameter	Description
Screen Rotation	<p>⚠ NOTE: After changing the setting of <i>Screen Rotation</i>, the device must be power cycled for the change to take effect.</p> <p>⚠ NOTE: For Screen Rotation <i>ROT_90</i> and <i>ROT_270</i> (portrait mode) the Visualisation must be created accordingly to fit to the full screen size as detailed below. CODESYS does not automatically adjust the Visualisation Size upon changes to the Screen Rotation setting.</p> <p>Sets the device screen rotation.</p> <p>ROT_0: Visualisation is not adjusted, to suit the 'normal' device orientation with connectors A & C on the left when viewed from the rear. (Landscape 1280 px X 800 px)</p>  <p>ROT_90: Visualisation is rotated 90° clockwise to suit the device orientation with connectors A & C on the top when viewed from the rear. (Portrait 800 px X 1280 px).</p>  <p>ROT_180: Visualisation is rotated 180° suit the device orientation with connectors A & C on the right when viewed from the rear. (Landscape 1280 px X 800 px).</p>  <p>ROT_270: Visualisation is rotated 270° clockwise to suit the device orientation with connectors A & C on the bottom when viewed from the rear. (Portrait 800 px X 1280 px).</p> 

2.5.1.4 PERSISTENCE SETTINGS

Parameter	Description
Active	FALSE: Persistent Variables are not saved. TRUE: Persistent Variables are cyclically saved at the period of <i>Time</i> .
Time	Applicable only when <i>Active</i> 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.

2.5.1.5 MANUAL SHUTDOWN

 **NOTE:** Ensure *Ignition* is not active when calling *DSE.SystemShutdown*.

Parameter	Description
Manual Shutdown	0: Removal of the <i>Ignition</i> pin begins the shutdown process after <i>Shutdown Delay</i> has expired. 1: Removal of the <i>Ignition</i> pin does not begin the shutdown process. Instead, the application can monitor the <i>Ignition</i> pin, performing a graceful machine shutdown, before programmatically instructing DSEM812 to shutdown using the function: <code>DSE.SystemShutdown()</code> ; Ignition pin is monitored either by mapping <i>Ignition Switch</i> within the <i>Device Settings I/O Mapping</i> page, or by using: <code>DSE.SystemGetIgnition(Ignition => bIgnitionState);</code> Where <code>bIgnitionState</code> is a BOOL variable used to store the state of the system <i>Ignition</i> pin. To restart the DSEM812 application, reapply <i>Ignition</i> pin.
Shutdown Delay	Delay before shutdown is initiated.

2.5.1.6 OTHER PARAMETERS

Parameter	Description
System Information	Allows monitoring of Memory, Storage and CPU usage.
System Information Alarms	Allows user alarms to be set for Memory, Storage and CPU usage.

2.5.2 DEVICE SETTINGS I/O MAPPING

This page is used to monitor the device, and if required, to map the monitored values to program variables.

Variable	Mapping	Channel	Address	Type	Unit	Description
Device Error Code			%IW0	UINT		Error Code: Check Manual for more information
Device Temperature			%ID1	REAL	°C	Value of the Device Temperature
Battery Voltage			%ID2	DINT	mV	Battery Voltage
Supply Voltage 1			%ID3	DINT	mV	Supply Voltage 1
Ignition Switch			%IX16.0	BIT		Ignition Switch
Voltage Reference			%IW9	INT	mV	Voltage Reference
Wakeup Pulse			%IX20.0	BIT		Wakeup Pulse
Backlight			%QW0	UINT		Backlight Level
Keypad Backlight Colour			%QD1	UDINT		Keypad Backlight Colour (RGB value)

Parameter	Description
Device Error Code	<p>A bitfield to allow the error code to be mapped to a variable. This enables the application to decode and display internal errors.</p> <p>See Section Entitled <i>DSEM812 CODESYS ERROR CODES</i> elsewhere in this document for more details.</p>
Device Temperature	The actual temperature (°C) as measured within the device. Typical temperatures are more than 25°C above ambient temperature and vary depending upon device usage.
Battery Voltage	The voltage measured at the <i>ECU Power</i> terminals, PINs A1 (+ve) and A9, A16 (-ve).
Supply Voltage 1	The voltage measured at the <i>Supply Voltage 1</i> terminals, PINs B1, B9 (+ve) and B4, B5, B8, B11, B12, B15, B23 (-ve). This voltage is used to supply the device outputs on PINS B2, B3, B6, B7, B19, B20.
Ignition Switch	Contains the state of the <i>Ignition Switch</i> on PIN A3. Upon ignition removal the variable changes from TRUE to FALSE. A brief time later the device completes the shutdown process, and the application is shutdown.
Voltage Reference	Provides the voltage of the <i>VREF</i> output on PINS B10 (+ve) and B16 (-ve). This output is used as an auxiliary DC supply for example, to power input sensor devices.
Wakeup Pulse	With DC power applied, activating the <i>Wakeup Pulse</i> (minimum 1 second) silently begins the device start up process. Subsequent application of <i>Ignition</i> results in a faster application start-up. A typical example, is to apply this pulse upon the vehicle door opening.
Backlight	Sets the percentage of the display LCD Backlight. (0 = minimum, 100 = maximum).
Keyboard Backlight Colour	<p>Sets the colour of the Keyboard surround Backlight as an RGB Value. Increasing values for each colour increases the brightness of that colour.</p> <p>Examples: 16#FF0000 = Red 0xff, Green 0, Blue 0 (full brightness Red) 16#008000 = Red 0xff, Green 0x80, Blue 0 (half brightness Green) 16#C0C0C0= Red, Green Blue 0xC0 (3/4 brightness White) 16#FFFFFF = Red 0xff, Green 0xFF, Blue 0 (full brightness White)</p>

2.6 ADD INPUTS, OUTPUTS AND BUTTONS TO THE PROJECT (GUI)

NOTE: This section shows how to use Inputs, Outputs and Buttons using settings editors in the project tree. An alternative (advanced) method is to control/read them using the *DSE Standard* library (namespace *DSE*).

Right-Click Device...

And select Add Device.

Select Deep Sea Electronics

Select the device to add...

And click Add Device.

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

Name	Vendor	Version	Description
Miscellaneous			
M812 Buttons	Deep Sea Electronics	1.0.0.0	M812 Buttons
M812 Inputs	Deep Sea Electronics	1.0.0.0	M812 Inputs
M812 Outputs	Deep Sea Electronics	1.0.0.0	M812 Outputs

Name: M812_Buttons
Vendor: Deep Sea Electronics
Categories:
Version: 1.0.0.0
Order Number: ???
Description: M812_Buttons

Append selected device as last child of Device
 (You can select another target node in the navigator while this window is open.)

2.6.1 BUTTONS

NOTE: DSE_UTILS library includes *DSE_UTILS.M812buttons* functions. In addition to providing the state of the button presses, it also includes integral rising and falling edge detection. For many users / applications, *DSE_UTILS.M812buttons* function is more suitable than mapping variables in the project tree and applying rising edge detection externally.

2.6.1.1 BUTTON LOCATION



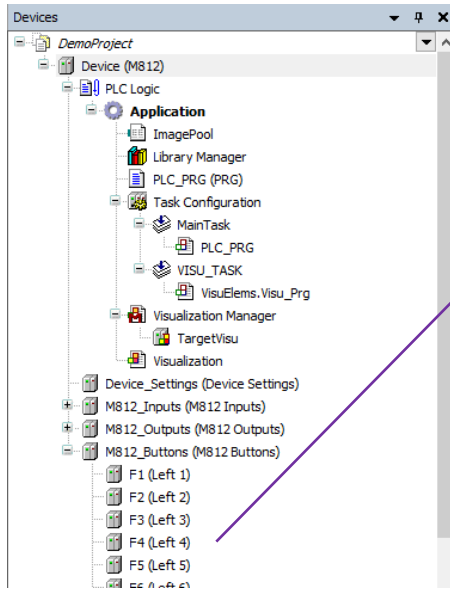
2.6.1.2 BUTTON POSITION ON THE VISUALISATION

To aid positioning of icons on the visualisation, the vertical position of the centre of each button is as follows.

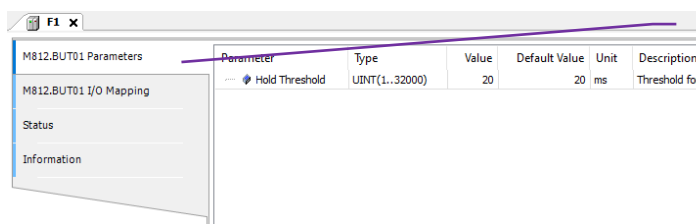
Button	Pixel Position From Top of Display
F1 & F9 (centre)	40
F2 & F10 (centre)	142
F3 & F11 (centre)	244
F4 & F12 (centre)	346
F5 & F13 (centre)	448
F6 & F14 (centre)	550
F7 & F15 (centre)	652
F8 & F16 (centre)	754

2.6.1.3 BUTTON SETTINGS

NOTE: DSE_UTILS library includes *DSE_UTILS.M812buttons* functions. In addition to providing the state of the button presses, it also includes integral rising and falling edge detection. For many users / applications, *DSE_UTILS.M812buttons* function is more suitable than mapping variables in the project tree and applying rising edge detection externally.

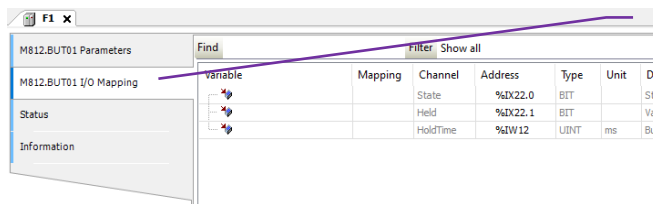


Double-Click the Button to configure the *Hold Time* and variable mappings.



Click *Parameters* to edit the *Hold Threshold*

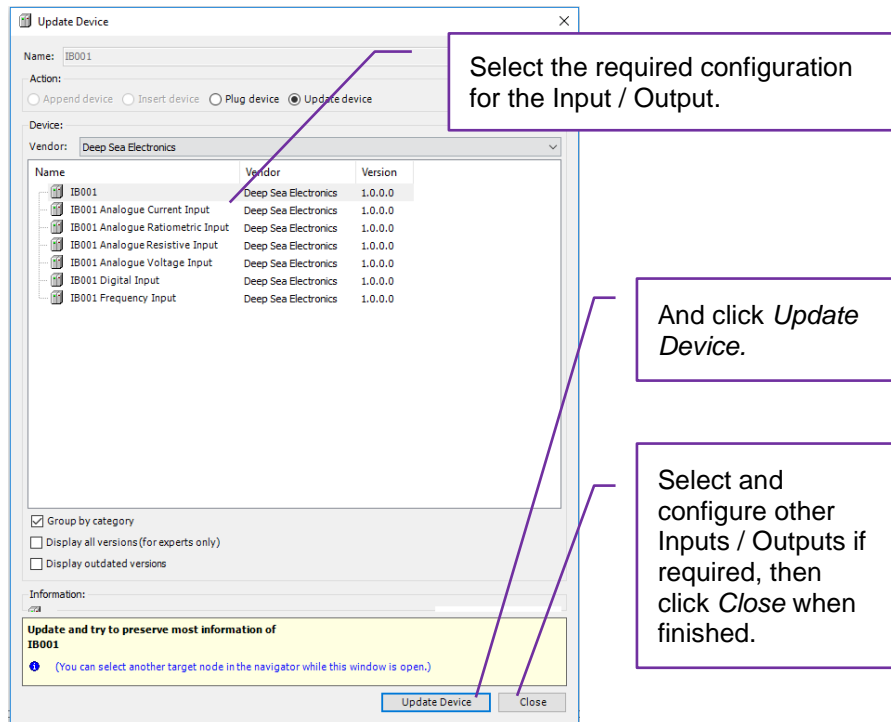
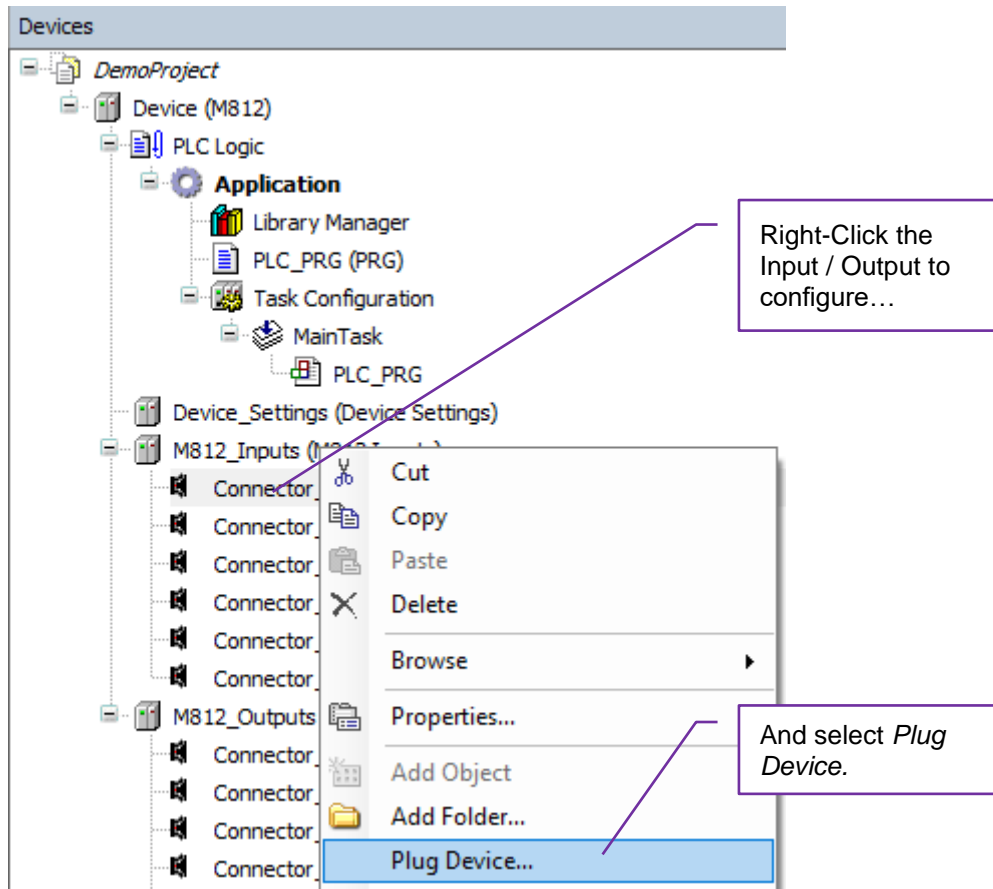
Parameter	Description
Hold Threshold	Amount of time (in milliseconds) that the button must be pressed before it is considered 'held down'.



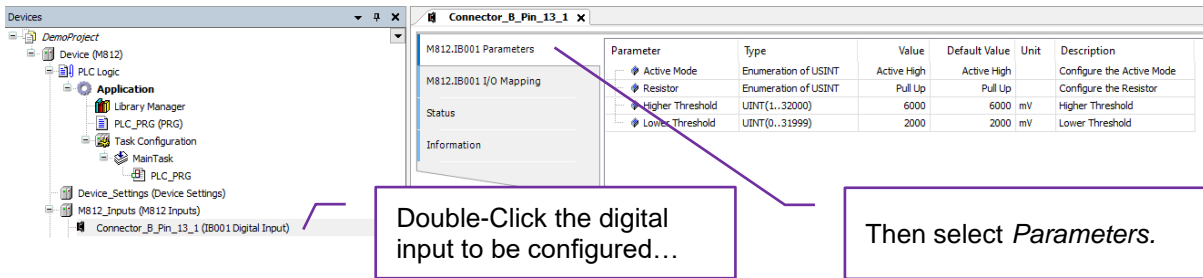
Click *Mapping* to view and edit the variable mapping.

Parameter	Description
State	Indicates if the button is pressed (1) or not pressed (0).
Held	Indicates if the button has been held for longer than the duration of the <i>Hold Threshold</i> (1) or not (0).
Hold Time	The amount of time (in milliseconds) that the button has been pressed for (zero if not currently pressed).

2.6.2 INPUTS AND OUTPUTS



2.6.3 DIGITAL INPUT PARAMETER CONFIGURATION



Parameter	Description
Active Mode	Active High: The input connects to the positive supply rail when activated. Active Low: The input connects to the negative supply rail when activated.
Resistor	Float: The input is floating when no connection is made. Commonly used with PNP (Sourcing) type switched sensors. 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 volt-free contacts.
Higher Threshold	For Active High inputs, the input is detected as being active when above this threshold with respect to the negative supply rail.
Lower Threshold	For Active Low inputs, the input is detected as being active when below this threshold with respect to the negative supply rail.

2.7 USING THE DISPLAY 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.

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

select *Add Object*....

Then select *Visualisation*.


Double-Click *Target* to select which *Visualisation* is shown at application start up. This is automatic if there is only one in the project.

Double-Click *Visualisation* to begin editing. Refer to CODESYS online documentation for details.

2.7.1 CLIENT ANIMATIONS

 **NOTE:** Client Animations are supported on M812 V2+. Prior to this, animations must be handled by the user application code.

 **NOTE:** A simple example of *Client Animations* is available from support@deepseaelectronics.com

 **NOTE:** Client Animations are required to be enabled on M812 V2+ to enable use of the Visualisation *Invisible* property.

Client Animations allow certain parts of the visualisation to be animated automatically by the DSE MSeries Device.

The following (non-exhaustive) list gives some examples:


- A visu element fades in/out upon the changing of the *Invisible* property of the element.
- Dialogue boxes and menus open and close in an animated way, rather than instantly appearing.
- Absolute Movement or rotation of a visu element takes place over a specified period rather than occurring instantly.

2.7.2 INSTALLING FONTS ON THE DEVICE

DSEM812 includes a *default* font, ready installed on the device. To use other fonts, first the font must be installed on the device using the *System Pages*. For details of this, see DSE Publication 057-317 *DSEM812 Installation and Operation Manual*.

 **HINT:** You can also upload fonts using *CODESYS Device / Files* and place them in the */Fonts* folder. At the next device power-up they'll be installed automatically.

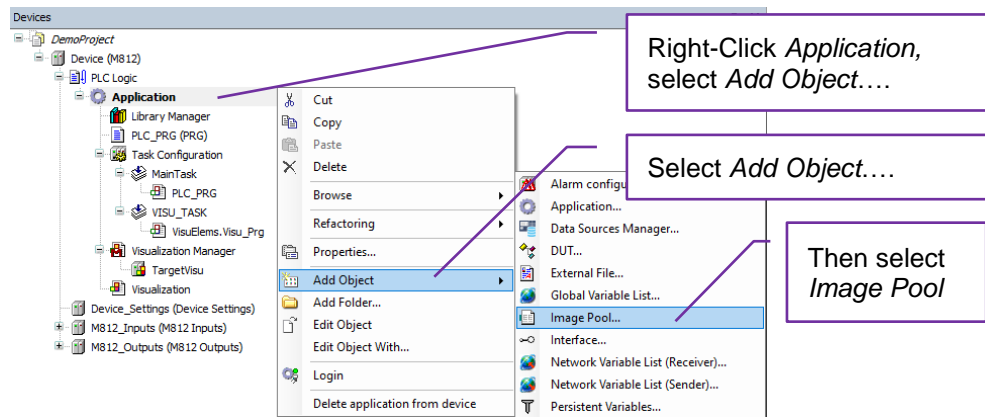
2.7.3 USING CUSTOM IMAGES ON THE DISPLAY

 **NOTE:** It is recommended that image filenames contain only alphanumeric characters and the underscore character. Use of other characters may lead to incompatibility with the Linux filesystem in use on the device and/or incompatibility with the automatic ID generation of the image in the *CODESYS ImagePool*.

 **HINT:** Multiple Image Pools can be used to group images into meaningful groups.

Many applications require custom images to be placed on the DSEM812 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.

2.7.3.1 ADDING AN IMAGE POOL

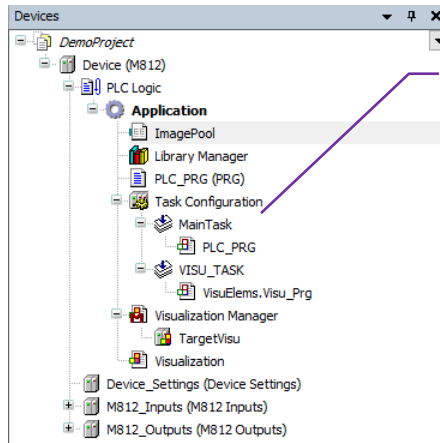


Right-Click *Application*, select *Add Object...*

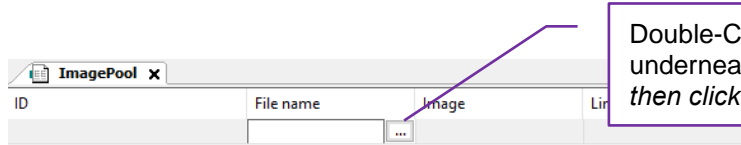
Select *Add Object...*

Then select *Image Pool*

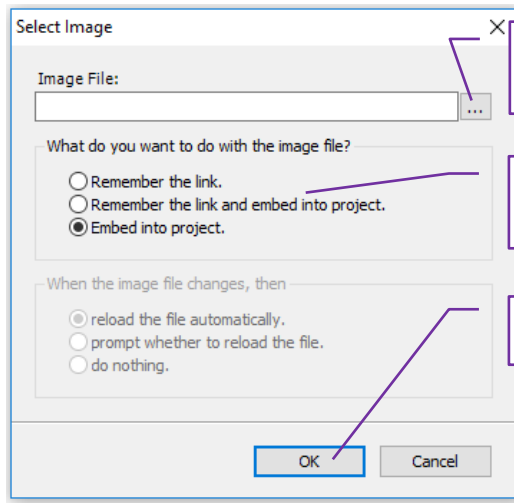
2.7.3.2 ADDING IMAGES TO THE IMAGE POOL



Double-Click the *Image Pool*.



Double-Click an empty row underneath *File name* and then click '...'

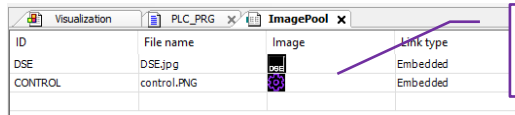


Browse to the image on your computer...

And select how the image is stored in the project.

Click OK when done.

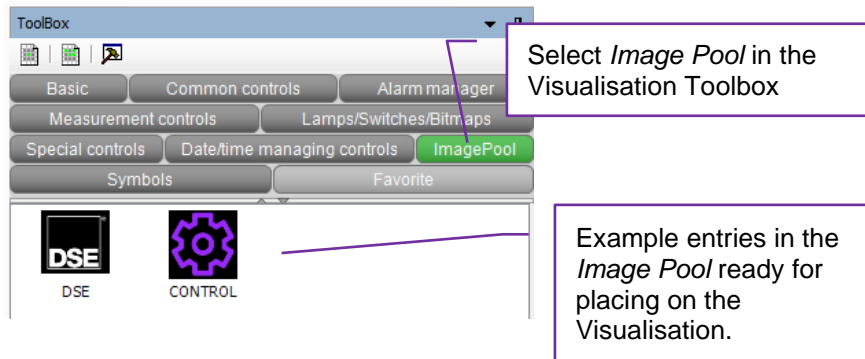
NOTE: Ensure the *ID* in the image pool contains only alphanumerical characters, - (minus), and '_' (underscore). Many other characters, including '.' (full stop) and spaces are not supported and may result in the image not appearing on the display.



Example entries in the *Image Pool*.

2.7.3.3 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.



2.7.4 UTILISING THE TOUCHSCREEN AND WEBVISU

NOTE: WebVisu is supported on M812 V3+.

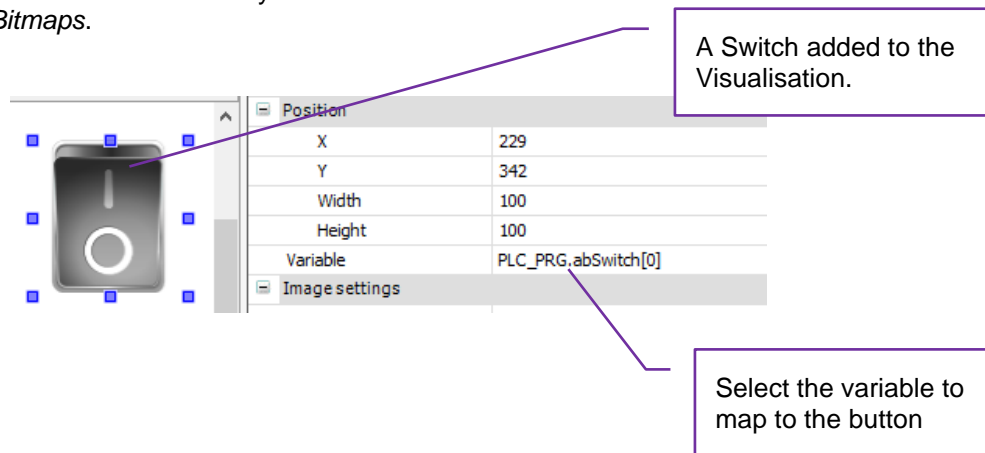
The capacitive touchscreen of DSEM812 mimics the operation of a mouse within the CODESYS environment. The following examples show how to utilise this functionality within the Visualisation. There are two ways to utilise the touchscreen. Either method is standalone (if required) or works alongside the other method. They are described in the following subsections.

2.7.4.1 SWITCHES, SLIDERS, COMBO BOXES, RADIO BUTTONS

Many CODESYS visualisation elements require no additional coding or settings to enable them for use with the Touchscreen. These include Switches, Sliders, Combo Boxes and Radio Buttons. Simply add them to the Visualisation and map a variable to them. Touch them on the screen to operate them.

Example:

Add a switch to the Visualisation. Usually these are found in the *Toolbox* under *Lamps/Switches/Bitmaps*.



A Switch added to the Visualisation.

Select the variable to map to the button

During the execution of the application, simply touch the switch on the screen to operate it.

2.7.4.2 NUMBER AND TEXT ENTRY

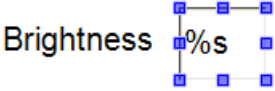
NOTE: *M812Multidemo* application contains a UK keyboard. To obtain this, contact support@deepseaelectronics.com. CODESYS supplied virtual keyboard is German layout.

HINT: Use Client Animation (M812 V2+) to animate the opening and closing of dialogue boxes.

Number entry via the DSEM812 Touchscreen requires the use of a *Virtual Numpad*.
Text entry via the DSEM812 Touchscreen requires the use of a *Virtual Keypad*.

CODESYS includes a simple number pad and a German layout keypad as a *Dialog* and are utilised as follows.

A *TextField* added to the Visualisation.



Locate *Input configuration* in the properties and click *Configure* next to *OnClick*

Select the Dialog to use. *VisuDialogs.Numpad* is supplied with CODESYS. Other dialogs may be available depending upon libraries you may have installed.

Input Configuration

OnClick

- Close Dialog
- Open Dialog
- Change the language
- Change shown Visualization
- Execute command
- Switch Framevisualization
- Write a Variable
- Execute ST-Code
- Toggle a variable
- File transfer

Write a Variable

Input type: VisuDialogs.Numpad

Choose variable to edit

Use text output variable

Use another variable

PLC_PRG.udBacklight

Initial display format:

Min: 10

Max: 100

Dialog Title: Enter display brightness

Password field

Position to open input dialog:

Use global setting (from Visualization manager)

Centered

OK Cancel


DoubleClick *Write a Variable* to add this entry to the action list

Select which variable to store the entry into

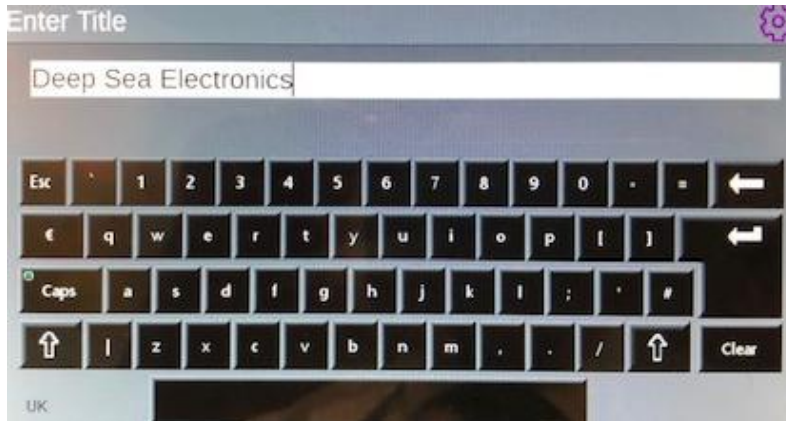
Example:

Upon touching the screen on the Visualisation element, CODESYS displays the selected Dialog. Enter the value required.

ESC: Exit without saving the value.

OK or : Exit and update the variable with the text or value entered.

Should the value be outside the configured range, this is indicated, and the dialog remains on screen to allow the correction to be made.



Virtual Keypad




Virtual Numpad

2.7.5 WEBVISU ACCESS

 **NOTE: WebVisu is supported on M812 V3+.**

 **NOTE: As *User Management* is mandatory on DSEM812, a valid password must be entered when connecting by WebVisu.**

 **NOTE: To ensure device security, it is not possible to upload (or download) files to (or from) the device using the WebVisu interface.**

DSEM812 V3+ includes CODESYS WebVisu functionality, an integral webpage server, whereby a project visualisation is available for viewing and control by an external device supported by a web browser. This WebVisu could be an exact duplication of what is shown on the DSEM812 display, or it can serve a different visualisation, for example a settings page, not accessible from the DSEM812 device itself.

Example:

Connect a PC to ethernet. Ensure the PC can “route” through to the DSEM812. You may need to consult your IT department to set up the network to allow this.

On the web browser, point it to port 8080 of the DSEM812. For example, if the DSEM812 IP address is 192.168.1.100 then point the web browser to `http://192.168.1.100:8080`.

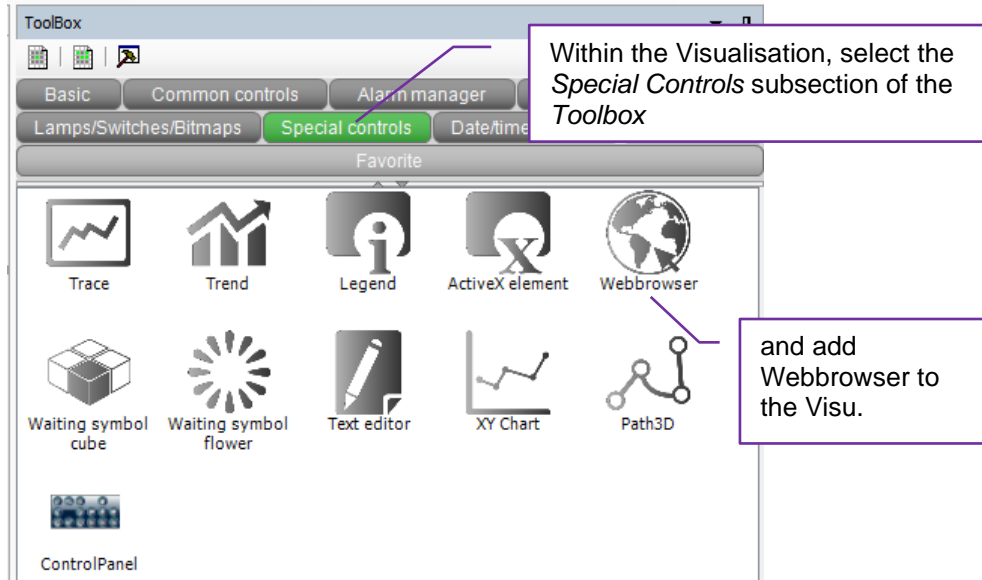
2.7.5.1 INTERACTIONS WITH CURRENTVISU VARIABLE.

In applications utilising `VisuElems.CurrentVisu` variable to change the shown visualisation on the device, this affects the WebVisu also. Hence where it is required to show differing visualisations on the device and on the WebVisu, other display switching methods must be used (eg Frames).

2.7.6 USING THE WEBBROWSER

NOTE: Ensure to provide a method of accessing the *Back* and *Forward* parameters of the *Webbrowser* component to allow simple navigation through the webpage.

The WebBrowser visualisation component allows the application to display and interact with a web page.



Control Variables of the visu component allow navigation of the pages. The following example shows mapping of them to variables in the user application.

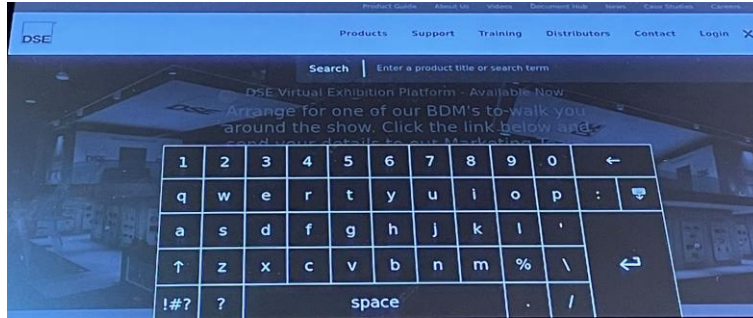
Control variables	
URL	PLC_PRG.url
Show	PLC_PRG.show
Back	PLC_PRG.back
Forward	PLC_PRG.forward

Parameter	Description
URL	STRING containing the full path to the webpage to display.
Show	BOOL. TRUE: The webpage is accessed and displayed. FALSE: The webpage is not shown.
Back	BOOL Upon rising edge, the previous page is shown (if applicable)
Forward	BOOL Upon rising edge, next page is shown (if applicable)

2.7.6.1 WEBBROWSER VIRTUAL KEYBOARD

When required, the DSE virtual keypad appears automatically on the display. This allows entry of data into webpages. For example, entry into search boxes or adjustment of settings where devices have internal webservers for settings pages.

Example, showing DSE website. The search icon has been pressed and the virtual keyboard is displayed to enter the search parameters.



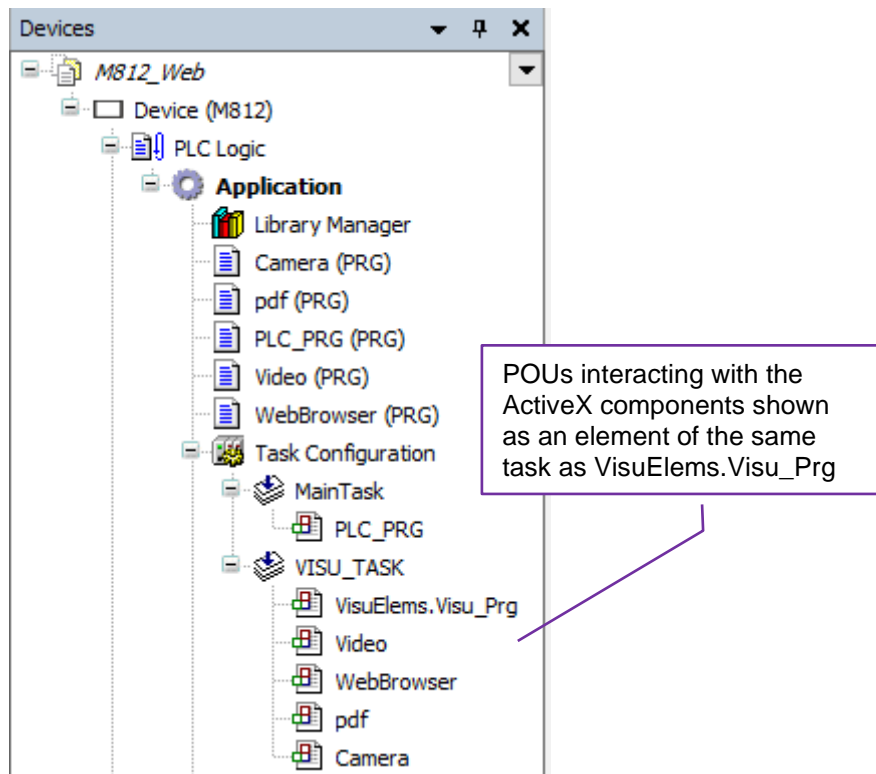
2.7.7 DSE ACTIVEX COMPONENTS

NOTE: Example applications are available from Deep Sea Electronics Technical Support (support@deepseaelectronics.com).

NOTE: Only one pdf viewer may be used in the application.

NOTE: DSEM812 V1 does not support 'on the fly' changes to camera or video position/size. This is supported only with M812 V2+ (DSE_MULTIMEDIA library).

Camera, Video, pdf, and Webbrowser visu components utilise ActiveX. Where the application interacts with these elements, it is strongly recommended to execute the interaction with the same CODESYS application task as the execution of the CODESYS Visu (ie VisuElems.Visu_Prg). The following image shows this by utilising a separate POU for each of the possible ActiveX components, however each need not be placed within its own POU.



2.7.7.1 ONLINE DOCUMENTATION

The screenshot shows the Library Manager interface. On the left, a tree view displays the library structure under 'DSE_Multimedia, 1.0.0.0 (Deep Sea Electronics)'. The 'CompositeCamera' folder is expanded, showing sub-folders like 'CompositeCameraViewSettings'. The main window displays the documentation for 'CompositeCameraViewSettings (STRUCT)'. Below the title, it states 'TYPE CompositeCameraViewSettings : STRUCT' and 'InOut:'. A table lists the fields of the structure:

Name	Type	Comment
Input	UDINT	If the camera inputs are muxed (e.g. M870-01), this variable selects the muxed, set to 0.
Rotation	UDINT	Normally 0, 90, 180 or 270. Depending on implementation, other values are an error.
WFlip	BOOL	TRUE to flip the image horizontally, FALSE for no horizontal flip.
VFlip	BOOL	TRUE to flip the image vertically, FALSE for no vertical flip.
Contrast	USINT	0 to 255
Brightness	USINT	0 to 255
Hue	USINT	0 to 255

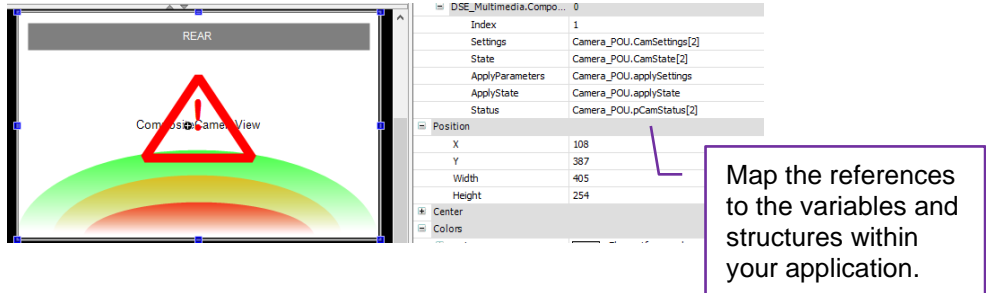
To the right of the table, a text box explains: 'Structures are used to hold the control variables and status of the Multimedia objects. They are documented in the project Library Manager online help as shown in this example.'

2.7.7.2 DISPLAYING AN ANALOGUE (COMPOSITE) CAMERA

DSEM812 V2+

NOTE: Examples showing single and multiple camera use are available from support@deepseaelectronics.com

Use the *CompositeCameraView* included in the DSE_MULTIMEDIA library.



DSEM812 V1

NOTE: An example is included in DSEM812Multidemo (V1) available from support@deepseaelectronics.com

Use the M870Camera included in the DSE_LINUX library.

2.7.7.3 DISPLAYING A TCP/IP CAMERA

NOTE: Examples are available from support@deepseaelectronics.com

TCP/IP Cameras are cameras that stream a video file that can be displayed by the M Series device. For best results, use H264 CODEC. The Movie Player element is used, with the `rtsp://<PATH>` specified as the URL for the video. Each camera manufacturer has its own specifications for the file path. Here is an example of a stream path, however you must refer to the documentation provided by your camera supplier:

```
'rtsp://<username>:<password>@<IP_ADDRESS>/stream/h264/'
```

2.7.7.4 USING THE MOVIE PLAYER

NOTE: Code interaction with the Movie Player is best performed in the Visu task. For details, see section entitled *DSE ActiveX Components*.

NOTE: DSEM812 blends the Movie overlay with CODESYS visualisation and the camera display allowing all to be visible at the same time. For details see section entitled *Using the Overlay*.

Movie Player is required for both the display of movie files and TCP/IP Camera streams. H264 is recommended for both.

Location of Movie Player library:

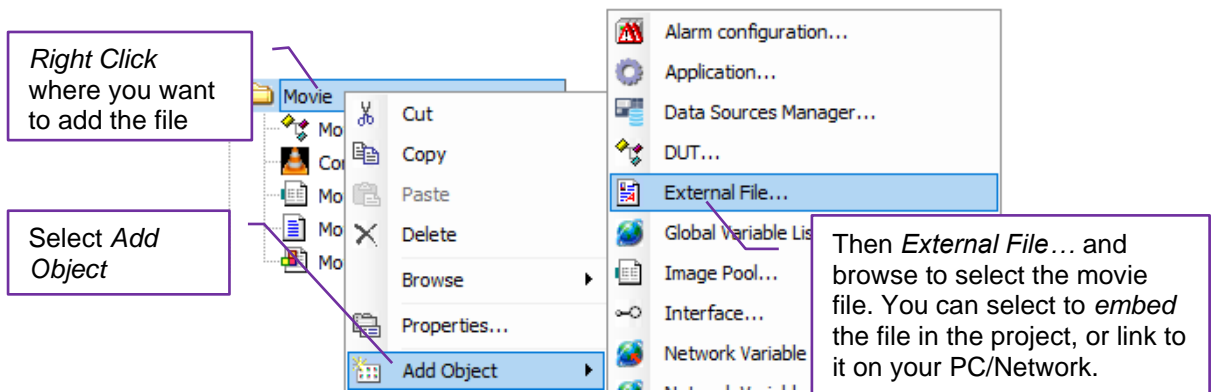
Device Version	Library Required
DSEM812 V1	DSE_LINUX
DSEM812 V2+	DSE_MULTIMEDIA

Adding the Movie File

NOTE: DSEM812 supports H264 (recommended), H263, XviD, MPEG2, and MPEG4 video CODEC formats.

NOTE: An example, using a video file is available. Contact support@deepseaelectronics.com for further details.

NOTE: For details how to specify the path to the file, see section entitled *Movie Player File Path* elsewhere in this document.



Movie Player File Path

A file added to the project using *Add Object | External File...* is placed by CODESYS into the following location in the device file system:

`'PlcLogic/Application/MyFileName'`

This is the path to the file when using functions from the *SysFile* library.

For the purposes of the movie player, a file added as *External File* has the additional prefix:

`'file://'`

Making the complete path for use with the movie player:

`'file://PlcLogic/Application/MyFileName'`



HINT: You can use *Device | Files* in CODESYS IDE to browse the file system of the device and check on the location of your files.

2.7.7.5 USING THE CAMERA / VIDEO OVERLAY

DSEM812 V2+ and DSEM812 V1 differ in this regard, please see below for details of each method. Legacy DSEM812 V1 is 'greyed out' to draw attention to its deprecation.

Location of components required for Camera / Video Overlay:

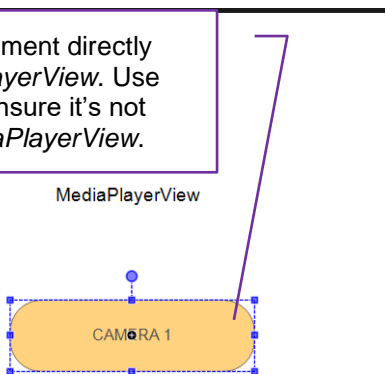
Device Version	Library Required
DSEM812 V1	DSE_LINUX
DSEM812 V2+	DSE_MULTIMEDIA

DSEM812 V2+

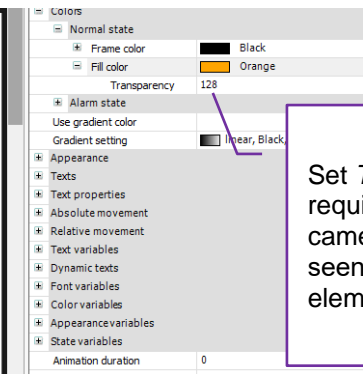
NOTE: DSEM812 V2+ allows CODESYS Visualisation elements to be placed directly over the camera/video viewer. The transparency of the image (alpha channel) or Transparency Property (Visu Element) is used to allow the camera/video to be seen 'through' the image/visu element.

Overlaying a CODESYS Visu Element

Place the Visu element directly onto the *MediaPlayerView*. Use *BringToFront* to ensure it's not 'behind' the *MediaPlayerView*.



Set *Transparency* as required to allow the camera/video to be seen 'through' the element.



Overlaying an Image

Use your preferred paint package to create the overlay. The *transparency* used within the image allows you to specify which parts of the camera/video can be seen 'through' the image. A fully transparent section of the image acts as a clear window through to the camera/video. A fully opaque section does not allow the camera/video to be seen through it. Varying the transparency between the two extremes allows differing areas of transparency (opaqueness).

DSEM812 V1



 **NOTE:** DSEM812 V1 blends the camera/video overlay with CODESYS visualisation and the camera/video display allowing all to be visible at the same time.

Overlay Settings


 **NOTE:** DSEM812 V1 utilises existing DSEM870 Camera functions. Therefore, some items in the DSE_LINUX library reference DSEM870.

Structs *M870CameraOverlaySettings* and *VideoOverlaySettings_t*, contains options to Enable/Disable the overlay and control the transparency.

Structure used for DSEM812 overlay display.

Variable	Type	Description
Enable	BOOL	Turns overlay ON / OFF <input type="checkbox"/> : Camera/video is displayed. <input checked="" type="checkbox"/> : Camera is displayed according to the settings of <i>Transparency</i> and <i>FileName</i> below.
Transparency	UDINT	<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">  NOTE: This parameter is NOT applicable if <i>FileName</i> is specified. </div> Specifies the transparency of the overlay over the camera/video display. <i>0</i> : Overlay is not visible, camera/video Only. <i>1 to 254</i> : Overlay is 'more visible' the higher the value. <i>128</i> : Overlay and camera/video are equally visible. <i>255</i> : Camera is not visible.
FileName	STRING	<div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">  NOTE: Leave this parameter empty if an overlay image is not required. In this case, use <i>Transparency</i> to control the blending of camera/video and Visualisation. </div> Overlay Image Filename (png, jpg, bmp). The image should be sized to match the size of the camera/video display within the application. If not the same size, it is scaled to fit. <i>Transparency</i> setting is not used, and the overlay image transparency is taken from the <i>Alpha Channel</i> value contained within the image. Some PC 'Paint' applications may call this <i>Transparency</i> or use similar wording.

2.8 USING THE USB DRIVE WITHIN APPLICATIONS

 **NOTE:** M812MultiDemo application available from Deep Sea Electronics Technical Support (support@deepseaelectronics.com) includes a simple demonstration writing to a csv file.

Using the *Sysfile* library, applications can read/write data from/to files on the USB device. A typical use for this is storing data in csv (comma separated value) files for datalogging. Within CODESYS, the device is accessed via the following path.

'USBDRIVE/Partition_Name'

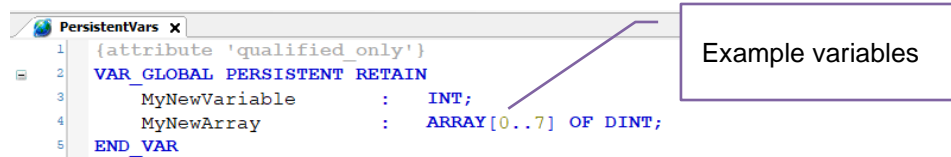
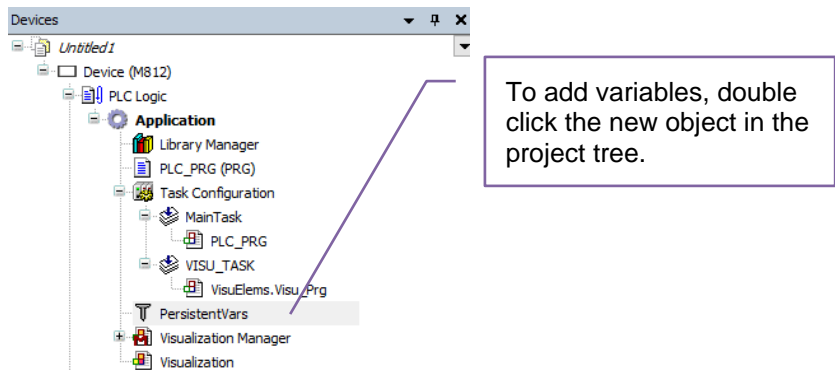
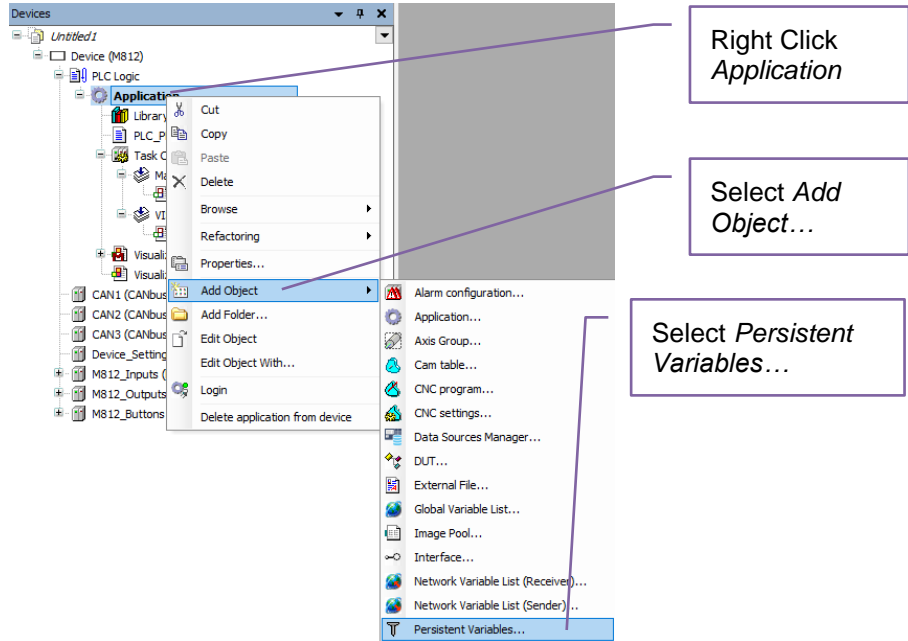
Example:

```
// Open the file and get the handle  
hFile := SysFileOpen(szFile:= 'USBDRIVE/DSE_PENDRIVE/Test.csv', am := AM_WRITE, pResult := ADR(eResult));
```

2.9 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 automatically loaded when the device powers up.



Within the POU the variables are address 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;

2.11 USING NETWORK VARIABLES IN THE PROJECT

NOTE: Network Variables are not supported in DSEM812 V1.

NOTE: Sender and Receiver Network Variable Lists MUST match.

NOTE: A suitable M12 to RJ45 cable is DSE Part 016-160.

Network Variables are a special type of Global Variable List and allow multiple devices to share a common variable list, utilising Ethernet (UDP) to communicate. To facilitate this, the simplest method of connection is to utilise a generic ethernet network switch as shown below. Whilst it is possible to connect two devices by a single M10 to M10 ethernet cable, the below method also provides for connection of the PC to another port of the network switch.



One device is the *Sender*, with other devices being a *Receiver*. The variable list from the application of the sender is exported to allow the receiver applications to receive them.

2.12 USING CAN IN THE PROJECT

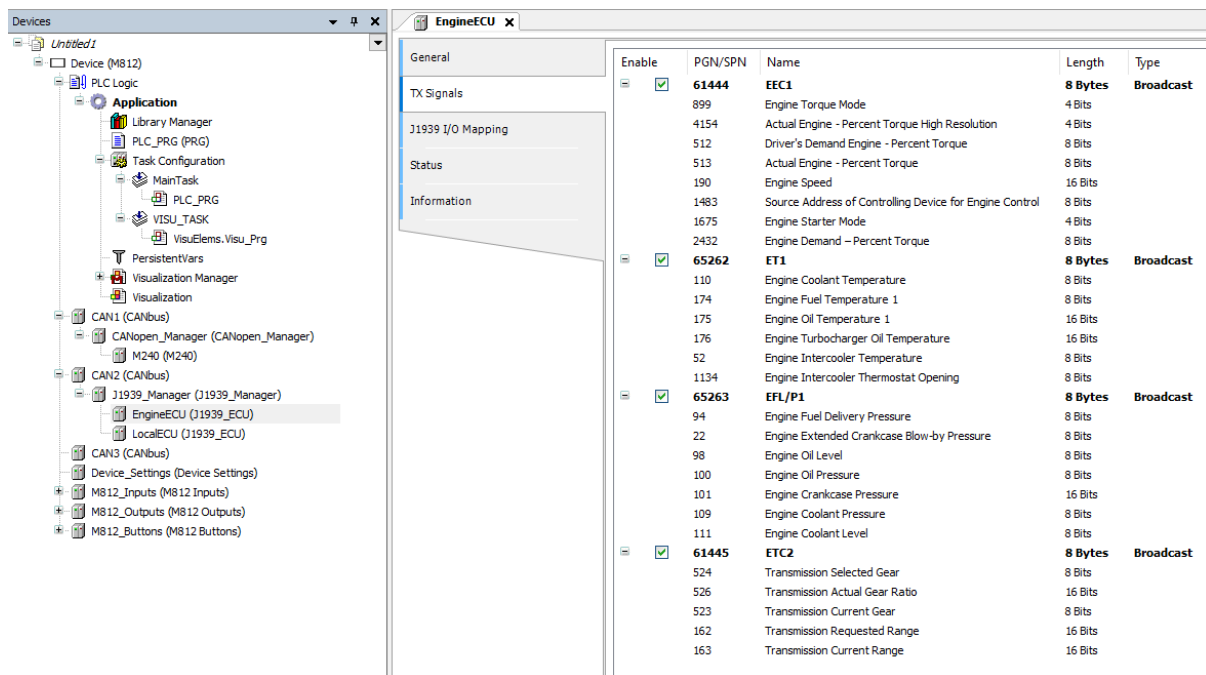
DSEM812 is provided with two alternative CAN libraries as discussed in the following subsections.

2.12.1 CODESYS CAN

NOTE: CODESYS J1939 Manager within CODESYS 3.5.17 requires a third party J1939 database (dbc) file. Contact support@deepseelectronics.com for more information.

Provides a graphical interface to setup the CAN utilising high level protocols such as J1939 and/or CANopen.

Example showing the use of CANopen (CAN1) and J1939 (CAN2).



2.12.2 DSE_CAN COMPILED LIBRARY

NOTE: For full details of DSE_CAN Compiled Library and example code, see *DSE Publication 057-336 DSE_CAN Compiled Library*.

Provides functions for both CAN Raw (useful for creating custom CAN protocols, for example to send data between application devices) and a selection of higher level J1939 functions. DSE_CAN Compiled library is for advanced users, or users demanding more flexible control of the CAN functions.

2.13 USING MODBUS IN THE PROJECT

2.13.1 MODBUS TCP MASTER (CLIENT)

DSEM812 supports CODESYS MODBUS Client, enabling the device to act as MODBUS TCP Client and make queries of an externally connected MODBUS TCP Server.

Examples of MODBUS TCP Server devices are (but not limited to):

- DSEM640 / DSEM840 configured (programmed) to provide MODBUS TCP Server functionality.
- Controllers in the DSEGenSet / DSE EngineOnly (E) range with integral ethernet ports.
- Controllers in the DSEGenSet / DSE EngineOnly (E) range with integral USB ports (with added DSE855 USB to Ethernet adaptor).
- Renewable Power Inverters (for example Solar Farm installations).

2.13.1.1 DSE GENCOMM MODBUS PROTOCOL

To discuss your requirements when connecting DSEM812 to a device supporting DSE Gencomm protocol, contact support@deeseapseaelectronics.com. Depending upon the application requirements, differing solutions are available.

2.13.2 MODBUS TCP SERVER



NOTE: CODESYS MODBUS TCP Server requires M812 V3+

DSEM812 supports CODESYS MODBUS Server, enabling the device to act as MODBUS TCP Server and respond to queries of an externally connected MODBUS TCP Client.

2.14 USING GPS IN THE PROJECT

▲ NOTE: GPS requires the use of an external antenna that has a clear view of the sky. At least four satellites must be seen to get a positional fix. Typically a fix cannot be obtained with a device where the antenna is located indoors.

2.14.1 OBTAINING A POSITION

- Include DSE_GPS library in the project. This library is included within the *DSE CODESYS PACKAGE*.
- Initialise the GPS as follows:

```
IF NOT init THEN
  Init:=TRUE;
  DSE_GPS.GpsInit(); // initialise the GPS
END_IF
```

- Call the GPSTimer function to retrieve the last fix obtained by the GPS receiver. Do not call this function too often. It is recommended to call it no more than once per second. In the following example, the GPS position is received once per second and the position stored in a persistent variables object. If the GPS signal is lost (NOT isValid) then the map position is set to a 'default' location, however the last known position is retained in the persistent variables object.

For example:

```
GPSTimer(IN:=TRUE, PT:=T#1S);
IF GPSTimer.Q THEN
  GPSTimer(IN:=FALSE);
  DSE_GPS.GpsLastPos(GeoPos=>GV_Map.GPSPosition);
  IF GV_Map.GPSPosition.isValid THEN
    PV.Position.Lat:=TO_REAL(GV_Map.GPSPosition.latitude);
    PV.Position.Lon:=TO_REAL(GV_Map.GPSPosition.longitude);
    PV.Position.ASL:=TO_REAL(GV_Map.GPSPosition.altitude);
  ELSE // in this example we show the location as DSE if there is no GPS signal.
    GV_Map.SatMapSettings.Position.UseGPS:=FALSE;
    GV_Map.SatMapSettings.Position.Latitude:=DSE_UTILS.LOCATION_DSE.Lat;
    GV_Map.SatMapSettings.Position.Longitude:=DSE_UTILS.LOCATION_DSE.Lon;
  END_IF
END_IF
```

2.14.2 DISPLAYING THE POSITION ON THE MAP

NOTE: This feature is intended to show the location on a map, navigation is NOT provided. Where a 'breadcrumb' trail is required, you can programmatically add POIs at regular intervals or upon certain events (button press, input activation etc) and the use of a dedicated POI icon/category to show it as a historical waypoint.

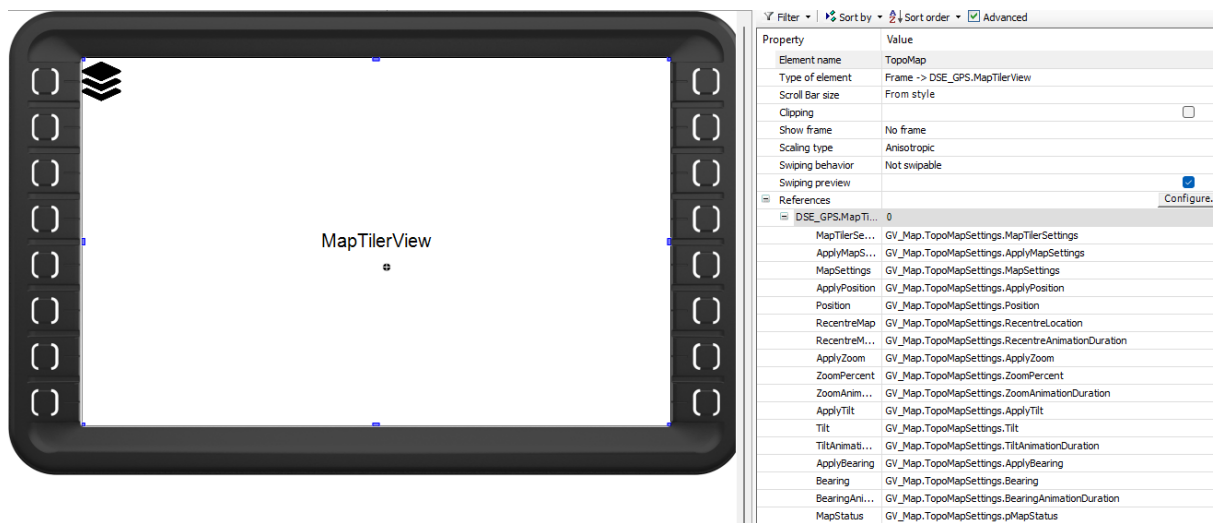
NOTE: Mapping is provided by maptiler.com. Obtain an API key from <http://www.maptiler.com> by registering and adding an API Key to your account.

NOTE: An example of MapTilerView is available on request from support@deepseaelectronics.com with library documentation available within the CODESYS Library Manager Online Help.

NOTE: To use the MapTilerView, an internet connection to the DSEM812 is required.

NOTE: Once initialised, the map 'layer' (map type) cannot be changed. To change between 'layers', use multiple MapTilerView objects, and hide (invisible) the currently unrequired views.

Add the MapTilerView to the visualisation and map the variables as required.
Example:



2.14.2.1 MAP STRUCTS

The following structs are used by the MapTilerView

MapTilerSettings_t

Name	Type	Description
MapName	STRING	The name of the map, e.g. "satellite", "hybrid" or "<UUID>" where <UUID> is the ID of a custom map
Key	STRING	API key for MapTiler

Example settings:

```
MapTilerSettings          :   DSE_GPS.MapTilerSettings_t := (
                           MapName := 'satellite',
                           Key := 'PutYourAPIKeyHere'
                           );
```

MapPosition_t

Name	Type	Description
UseGPS	BOOL	Set TRUE to use the device's GPS position, FALSE if not
Latitude	LREAL	Initial latitude (only used if UseGPS is false)
Longitude	LREAL	Initial longitude (only used if UseGPS is false)

MapStatus_t

Name	Type	Description
ZoomPercent	UDINT	The current zoom as a percentage
MinZoom	REAL	The allowed minimum zoom (absolute value)
MaxZoom	REAL	The allowed maximum zoom (absolute value)
Tilt	REAL	The current tilt angle in degrees
MinTilt	REAL	The minimum tilt angle in degrees
MaxTilt	REAL	The maximum tilt angle in degrees
Bearing	REAL	Between 0 and 360 degrees

MapSettings_t

Name	Type	Description
PositionActiveIconPath	STRING	file://<path> of the active position icon. Leave empty to use the default
PositionInactiveIconPath	STRING	file://<path> of the inactive position icon. Leave empty to use the default
PositionIconAlignment	MapAlignment_t	Icon alignment for the given coordinates
GPSTimeout	UDINT	Timeout in milliseconds for a the GPS to be considered inactive.
RecentreTimeout	UDINT	Timeout in milliseconds for a pan gesture to reset to the currently set location. Use 0 to disable.

2.14.2.2 POI (POINTS OF INTEREST)

POI Options

When active, the map can be *long pressed* (800 ms). A menu is shown containing POI options. This allows the placing of a POI onto the map. By default there is only one POI Option (*Point of Interest*). New categories (including description and icon) may be added via application code as shown below. These are not 'retained' through power off, so the code must add them at each application start up.

```
// example how to add POI categories. Use path ' ' if no icon is used.
DSE_GPS.MapAddPOIOption("Campsite", 'file:///home/codesys/PlcLogic/visu/campsite.bmp');
DSE_GPS.MapAddPOIOption("Caution", 'file:///home/codesys/PlcLogic/visu/caution.bmp');
DSE_GPS.MapAddPOIOption("Fishing", 'file:///home/codesys/PlcLogic/visu/fishingspot.bmp');
DSE_GPS.MapAddPOIOption("Marker", 'file:///home/codesys/PlcLogic/visu/custommarker.bmp');
DSE_GPS.MapAddPOIOption("Parking", 'file:///home/codesys/PlcLogic/visu/parking.bmp');
DSE_GPS.MapAddPOIOption("Viewpoint", 'file:///home/codesys/PlcLogic/visu/viewpoint.bmp');
```

Programmatically placing POIs

In addition to the manual placement of POIs (long press on the map > 800 ms), they can be added by program code as shown below. POIs stored by either of these methods are places in a json file on the device (/home/codesys/maps/poi.json), hence are retained through a power off.

```
// add the POIs. A user could, for example, store POIs in a file and read them at device power up, adding them to the map
DSE_GPS.MapAddPOI("DSE", 'file:///home/codesys/PlcLogic/visu/dse_black.png', 54.176148587969934, -0.31134968095104204);
```

Removing POIs

POIs are removed by either clicking on them on the map and selecting to remove them, or by program call to MapRemovePOI.

2.15 ALTERNATIVE METHODS TO LOAD THE APPLICATION

2.15.1 USB

NOTE: For details of the *Clone/Recover* process see DSE publication *057-317 DSEM812 Installation and Operation Manual*.

A *Clone File* of the device can then be created and used to *Recover* onto other devices of the same variant and firmware version.

2.15.2 DSESERVICETOOL PC SOFTWARE

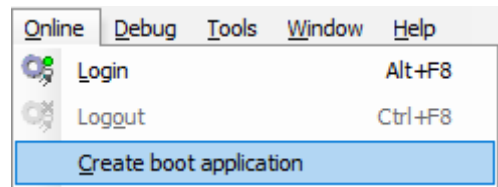
NOTE: For details using DSEServicetool PC Software see DSE publication *057-265 DSE Servicetool PC Software Manual*.

DSEServicetool PC Software is used to transfer applications to the device. This allows OEMs to update equipment application software without requiring a connection to CODESYS. However, CODESYS is required to create the *Boot Application* files in the first instance.

2.15.2.1 CREATION OF THE BOOT APPLICATION

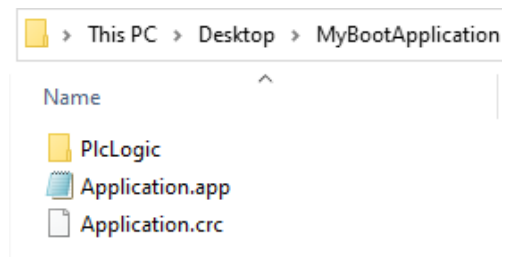
NOTE: The boot application includes the folder *PlcLogic*. This contains all images, text files and associated visualisation files.

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.



2.15.2.2 UPDATE USING DSESERVICETOOL AND ETHERNET CONNECTION

NOTE: For details using DSEServicetool PC Software see DSE publication *057-265 DSE Servicetool PC Software Manual*.

Within DSEServicetool PC Software scan for and select the device, then select the file *Application.app* and transfer it to the device.

3 DSEM812 CODESYS ERROR CODES

DSEM812 returns error codes to CODESYS when appropriate. Individual bits are set within the returned value to indicate one or more error conditions. This can be mapped to a variable if required and is available to view within CODESYS under the *Device Settings I/O Mapping* as shown below.

Variable	Mapping	Channel	Address	Type	Current Value	Prepared Value	Unit	Description
Error Code			%IW18	UINT	120			Error Code: Check Manual for more information
Device Temperature			%ID10	REAL	18.2		°C	Value of the Device Temperature
Battery Voltage			%IW22	UINT	15024		mV	Battery Voltage
Supply Voltage 1			%IW23	INT	247		mV	Supply Voltage 1
Supply Voltage 2			%IW24	INT	274		mV	Supply Voltage 2
Supply Voltage 3			%IW25	INT	185		mV	Supply Voltage 3
Supply Voltage 4			%IW26	INT	301		mV	Supply Voltage 4
Ignition Switch			%IX54.0	BIT	TRUE			Ignition Switch
Program Enable			%IX54.1	BIT	TRUE			Program Enable
Voltage Reference			%IW28	INT	-1217		mV	Voltage Reference

Examples:

A *Device* error value of 120 (01111000 in binary) indicates that all four *Output Supplies* are *Under Voltage*.

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

3.1 DEVICE

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

3.2 ANALOGUE INPUTS

Input Configuration	Bit							
	MSB 8	7	6	5	4	3	2	LSB 1
Digital	Invalid Parameter	Reserved	Reserved	Invalid Threshold	Reserved	Reserved	Reserved	Error
Voltage	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Over Range	Inverted Input (<10 mV)	Error
Current	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Over Range	Wire Break (<4 mA)	Error
Resistance	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Over Range	Reserved	Error
Ratiometric	Invalid Parameter	Reserved	Invalid Reference	Reserved	Reserved	Over Range	Reserved	Error

3.3 DIGITAL INPUTS

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

3.4 DIGITAL OUTPUTS

Output Configuration	Bit							
	MSB 8	7	6	5	4	3	2	LSB 1
Digital	Invalid Parameter	Reserved	Reserved	Reserved	Reserved	Over Current	Wire Break (Config)	Error

4 TROUBLESHOOTING

The below are suggestions answering frequently questions about the device.

Symptom	Suggestion
The Visualisation 'Invisible' property has no effect.	With CODESYS 3.5.17, as used by DSEM812 V2+, <i>Invisible</i> requires the enabling of <i>Client Animations</i> , accessible under Visualization Manager.
Text on the display differs from that selected.	This occurs where the selected font character is not present on the device, the <i>default</i> font is selected. The ttf (TrueType Font) file must be installed to the device using the <i>Configuration</i> mode. For further details, see DSE Publication 057-317 <i>DSEM812 Installation and Operation manual</i> .
Images are not appearing on the device display.	<ul style="list-style-type: none"> • See section entitled <i>Using Custom Images on the Display</i> elsewhere in this document. • Ensure the ID in the image pool contains only alphanumeric characters. • It is possible that the device memory is full and there is no room for more images. To ensure existing, unneeded files are not present, format the device before deploying
How do I use Persistent Variables?	<ul style="list-style-type: none"> • Ensure <i>Persistence Settings</i> are configured as required. See section entitled <i>Connecting to CODESYS Configure Settings and Monitor the Device Persistence Settings</i> • Add a Persistent Variables object to the project. Right Click <i>Application</i> and select <i>Add Object Persistent Variables</i>. Variables placed here are stored as selected above and persist through a power off.
How can I connect an RTD temperature sensor?	DSEM812 Resistive Inputs are accurate to 1 % of full measurement scale (3400 Ω). This makes RTDs unsuitable for connection to DSEM812. 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. Alternatively, sensors with CAN support are suitable.

5 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).

6 DISPOSAL

6.1 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



7 MISCELLANEOUS

This product includes copyrighted third-party software licensed under the terms of the GNU General Public License. A copy of the corresponding source code for all included third-party software is available on request, please contact DSE Technical Support for additional information.

This Page is Intentionally Blank