# HDLC Router

## User Manual

A-HDLC

Document No. D118-007
03/2024
Revision 1.6

aparian

# CONTENTS


1. Preface ..... 5
   1.1. Introduction to the HDLC Router ..... 5
   1.2. Additional Information ..... 7
   1.3. Support ..... 7
2. Installation ..... 8
   2.1. Module Layout ..... 8
   2.2. Module Mounting ..... 10
   2.3. Power ..... 11
   2.4. HDLC Port ..... 11
   2.5. RS485 Termination ..... 13
   2.6. Ethernet Port ..... 13
3. Setup ..... 14
   3.1. Install Configuration Software ..... 14
   3.2. Network Parameters ..... 14
   3.3. Creating a New Project ..... 19
   3.4. HDLC Router Configuration ..... 21
   3.5. HDLC Mapping ..... 24
   3.6. Module Download ..... 26
   3.7. RSLogix 5000 Configuration ..... 29
   3.8. Modbus Configuration ..... 33
4. Operation ..... 35
   4.1. Master Mode ..... 35
   4.2. Slave Mode ..... 35
   4.3. Logix assemblies ..... 35
      4.3.1. Input Assembly ..... 36
      4.3.2. Output Assembly ..... 37
   4.4. Status Word ..... 39
   4.5. Modbus Registers ..... 39
5. Diagnostics ..... 41
   5.1. LEDs ..... 41
   5.2. Module Status Monitoring in Slate ..... 42

5.3. HDLC Packet Capture ..... 47
5.4. Module Event Log ..... 49
5.5. Status Report ..... 50
5.6. Web Server ..... 52
6. Technical Specifications ..... 53
6.1. Dimensions ..... 53
6.2. Electrical ..... 54
6.3. Ethernet ..... 54
6.4. HDLC (RS485) Port ..... 55
6.5. Certifications ..... 55
Index ..... 56

# Revision History

| Revision | Date | Comment |
|---|---|---|
| 1.0 | 24 November 2016 | Initial document |
| 1.1 | 29 April 2017 | Add ODVA Conformance mark<br>Add CE Conformance mark |
| 1.2 | 23 May 2017 | Add support for two additional interface modes.<br>Add RCM Conformance mark. |
| 1.3 | 13 June 2017 | Add Response Timeout and Enable Timeout Packet Repeat to HDLC Config |
| 1.4 | 26 June 2017 | Added configurable duplex for HDLC Router |
| 1.5 | 14 February 2018 | Added support for Modbus TCP |
| 1.6 | 8 March 2024 | Added more information regarding duplex settings with 4 wire systems. |

# 1. PREFACE

## 1.1. INTRODUCTION TO THE HDLC ROUTER

This manual describes the installation, operation, and diagnostics of the Aparian HDLC Router. The HDLC Router, (hereafter referred to as the **module**,) provides intelligent data routing between EtherNet/IP and an HDLC (RS485) network.

This allows the user to integrate HDLC devices into a Rockwell Logix platform (e.g. ControlLogix or CompactLogix) with minimal effort.

The module can be configured to act as either an HDLC master or slave, in either 2-wire or 4-wire RS485 mode. Data can be easily mapped to custom positions in the Logix input and output assemblies affording maximum user flexibility.

Two example applications are described below:

The first example application uses the HDLC Router to exchange data between a GEM80 MulitGEM controller via a STARNET card, and a ControlLogix controller.

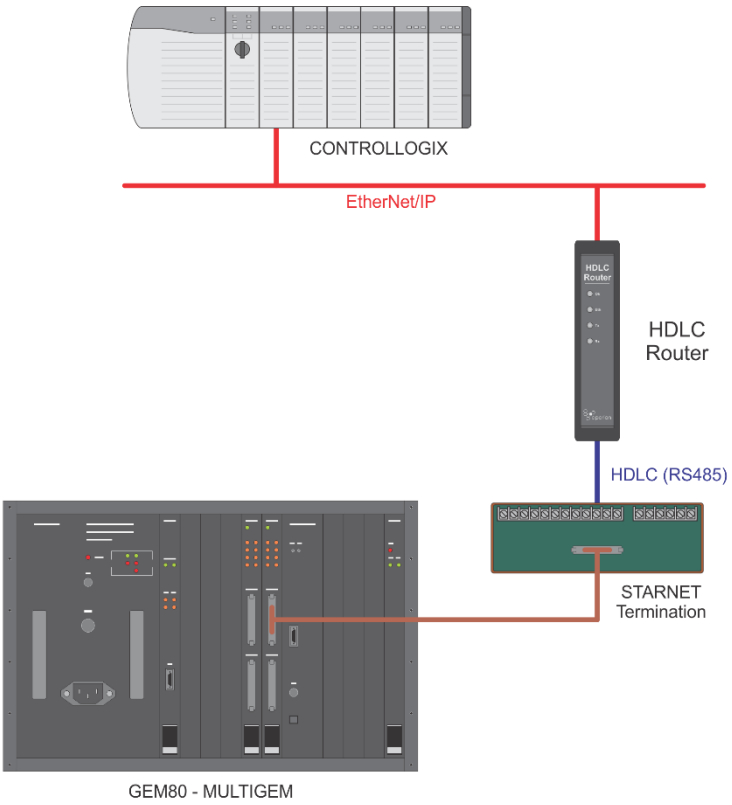



Figure 1.1. – Application Example 1

The HDLC Router can be configured to be either a Master or Slave, depending on the configuration of the GEM Starnet module. This application typically makes use of a 4-wire connection.

The second example has the HDLC Router configured as a slave device with a number of other slaves connected on the same HDLC segment. The HDLC Router is able to masquerade as multiple slave nodes, according to the pre-configured data mapping.

This configuration can make use of either 4-wire or 2-wire RS485 topologies.

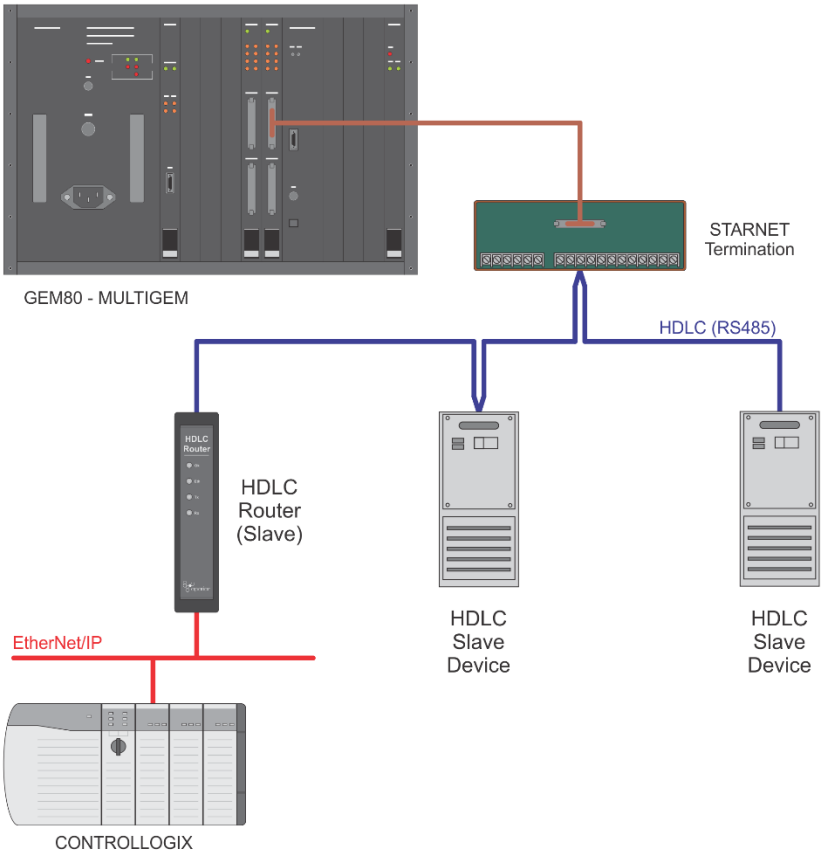



Figure 1.2. – Application Example 2

The HDLC Router is configured using the Aparian Slate application. This program can be downloaded from www.aparian.com free of charge.

The module also provides a range of statistics and a packet analyser to help fault find any problems. The HDLC RS485 port uses a terminal block for convenient installation.

A built-in webserver provides detailed diagnostics of system configuration and operation, including the display of HDLC operation and communication statistics, without the need for any additional software.

## 1.2. ADDITIONAL INFORMATION

The following documents contain additional information that can assist the user with the module installation and operation.

| Resource | Link |
|---|---|
| Slate Installation | http://www.aparian.com/software/slate |
| HDLC Router User Manual<br>HDLC Router Datasheet<br>Example Code & UDTs<br>Tech Notes | http://www.aparian.com/products/hdlcrouter |
| Ethernet wiring standard | www.cisco.com/c/en/us/td/docs/video/cds/cde/cde205_220_420/installation/guide/cde205_220_420_hig/Connectors.html |
| CIP Routing | The CIP Networks Library, Volume 1, Appendix C:Data Management |

Table 1.1. - Additional Information

## 1.3. SUPPORT

Technical support is provided via the Web (in the form of user manuals, FAQ, datasheets etc.) to assist with installation, operation, and diagnostics.

For additional support the user can use either of the following:

| Resource | Link |
|---|---|
| Contact Us web link | www.aparian.com/contact-us |
| Support email | support@aparian.com |

Table 1.2. – Support Details

# 2. INSTALLATION

## 2.1. MODULE LAYOUT

The module has three ports at the bottom of the enclosure as shown in the figure below. The ports are used for Ethernet, HDLC (RS485), and power. The power port uses a three-way connector which is used for the DC power supply positive and negative (or ground) voltage as well as the earth connection.

The Ethernet cable must be wired according to industry standards which can be found in the additional information section of this document.

The HDLC connector is used for both 2-wire and 4-wire connections. It also includes an earth connection, used for shielding the HDLC cable in high noise environments.

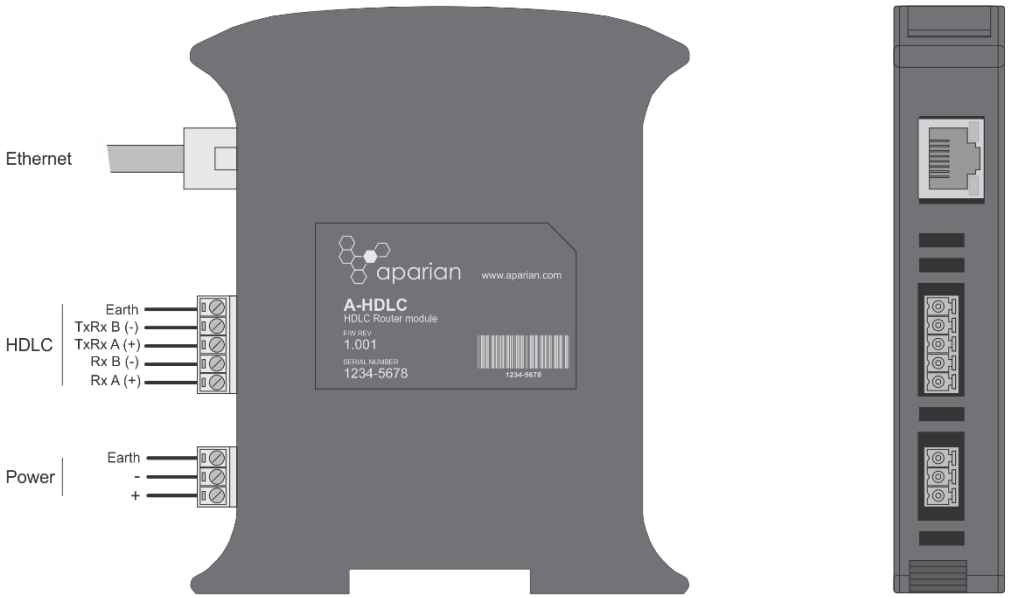



Figure 2.1 - HDLC Router side and bottom view

The module provides four diagnostic LEDs as shown in the front view figure below. These LEDs are used to provide information regarding the module system operation, the Ethernet interface, and the HDLC communication (transmit and receive.)

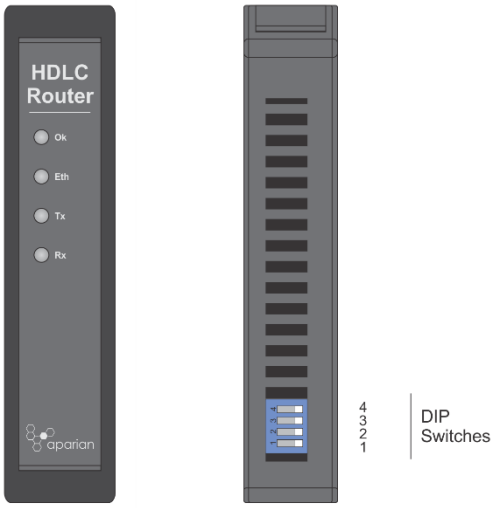



Figure 2.2 - HDLC Router front and top view

The module provides four DIP switches at the top of the enclosure as shown in the top view figure above.

| DIP Switch | Description |
|---|---|
| DIP Switch 1 | Used to force the module into “Safe Mode”. When in “Safe Mode” the module will not load the application firmware and will wait for new firmware to be downloaded. This should only be used in the rare occasion when a firmware update was interrupted at a critical stage. |
| DIP Switch 2 | This will force the module into DHCP mode which is useful when the user has forgotten the IP address of the module. |
| DIP Switch 3 | Enables the TxRx port’s (terminal 3 & 4) terminating resistor (120 Ω). |
| DIP Switch 4 | Enables the Rx port’s (terminal 1 & 2) terminating resistor (120 Ω). |

Table 2.1. - DIP Switch Settings

## 2.2. MODULE MOUNTING

The module provides a DIN rail clip to mount onto a 35mm DIN rail.

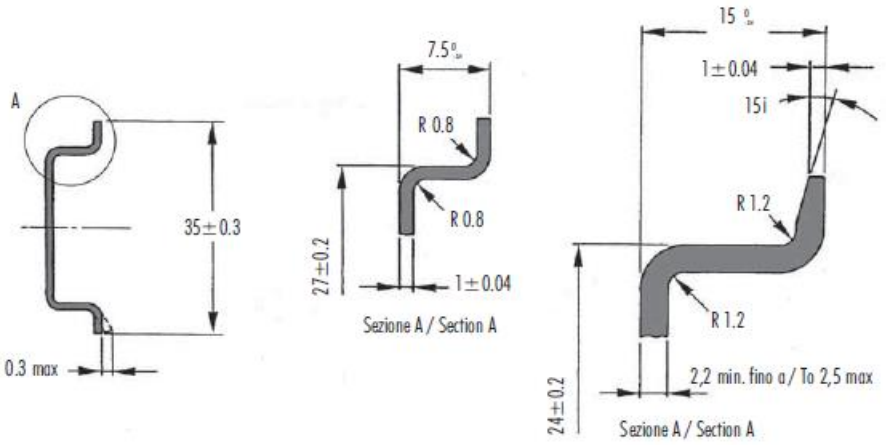



Figure 2.3 - DIN rail specification

The DIN rail clip is mounted on the bottom of the module at the back as shown in the figure below. Use a flat screw driver to pull the clip downward. This will enable the user to mount the module onto the DIN rail. Once the module is mounted onto the DIN rail the clip must be pushed upwards to lock the module onto the DIN rail.

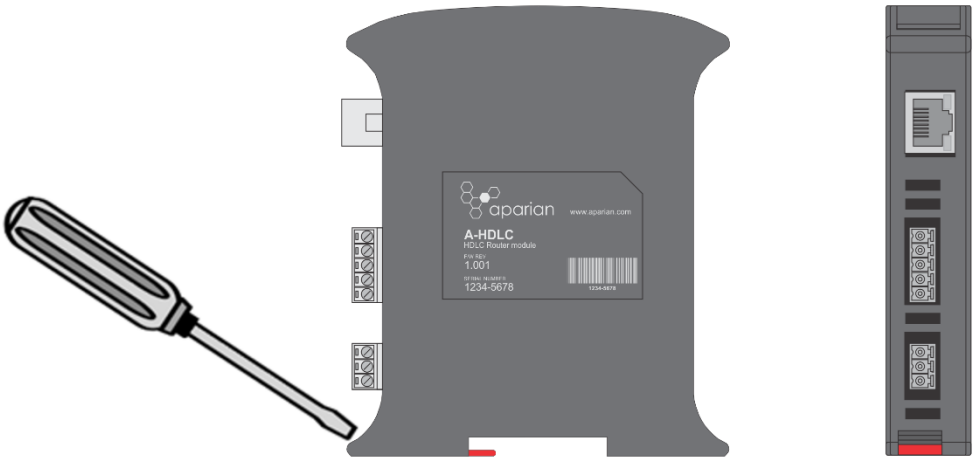



Figure 2.4 - DIN rail mounting

## 2.3. POWER

A 3-way power connector is used to connect Power+, Power– (GND), and earth. The module requires an input voltage of 10 – 28Vdc. **Refer** to the technical specifications section in this document.

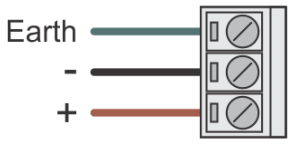



Figure 2.5 - Power connector

## 2.4. HDLC PORT

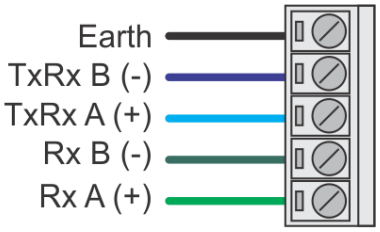



Figure 2.6 - HDLC (RS485) connector

The 5-way HDLC connector is used to connect to the HDLC network in either a 2-wire or 4-wire configuration. The earth terminal can be used for shielded cable in high noise environments.

**NOTE:** The earth of the RS485 port is internally connected to the power connector earth. When using a shield it is important to connect the Earth terminal on the power connector to a clean earth. Failing to do this can lower the signal quality of the HDLC (RS485) communication.

**NOTE:** When using a shielded cable, it is important that only one end of the shield is connected to earth to avoid current loops.

The HDLC connection depends on the mode and 2-wire or 4-wire configuration selection.

Connections for various configurations is illustrated in the following examples:

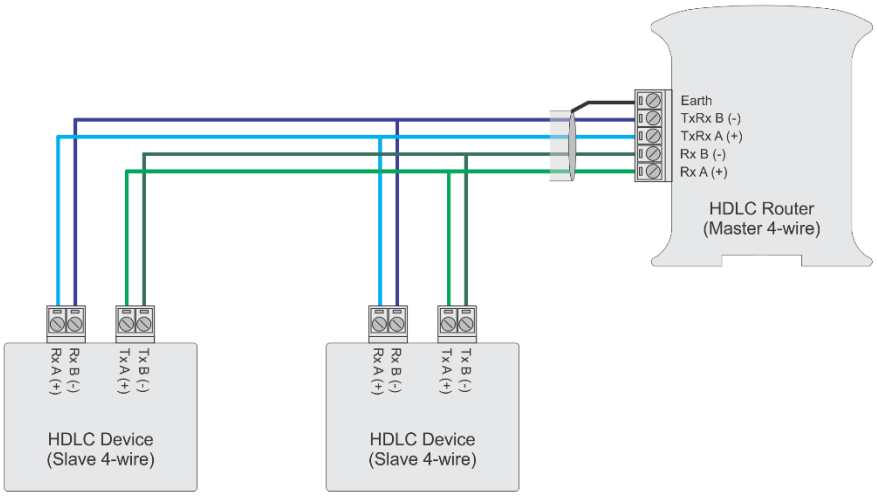



Figure 2.7 - HDLC Router as a 4-wire Master

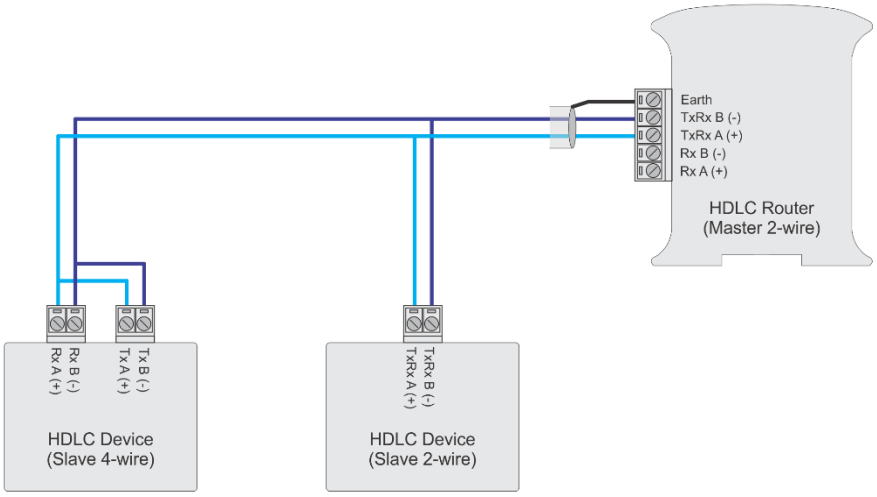



Figure 2.8 - HDLC Router as a 2-wire Master

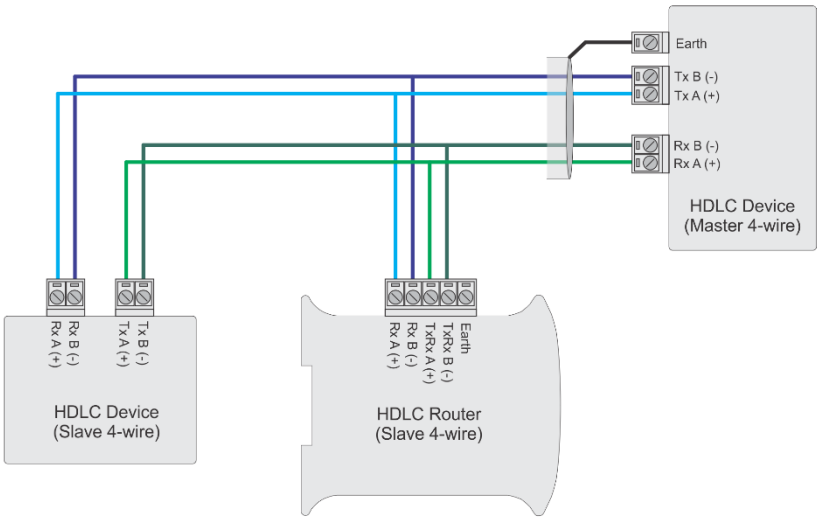



Figure 2.9 - HDLC Router as a 4-wire Slave

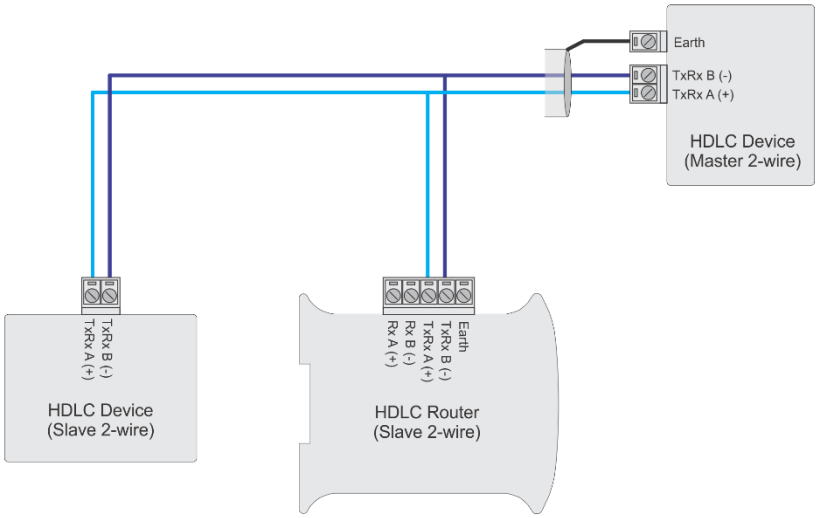



Figure 2.10 - HDLC Router as a 2-wire Slave

## 2.5. RS485 TERMINATION

All RS485 networks need to be terminated at the extremities (start- and end-points) of the communication conductor. The termination is achieved by placing a resistor between the positive and negative communication conductor. The value of the resistor will depend on the characteristic impedance of the cable chosen, but generally ranges from 100 Ohm to 150 Ohm.

The HDLC Router has on-board 120 Ω resistors for both the TxRx and the Rx ports, which can be enabled via DIP switch 3 and 4 respectively.

## 2.6. ETHERNET PORT

The Ethernet connector should be wired according to industry standards. **Refer** to the additional information section in this document for further details.

# 3. SETUP

## 3.1. INSTALL CONFIGURATION SOFTWARE

All the network setup and configuration of the module is achieved by means of the Aparian Slate device configuration environment. This software can be downloaded from http://www.aparian.com/software/slate.

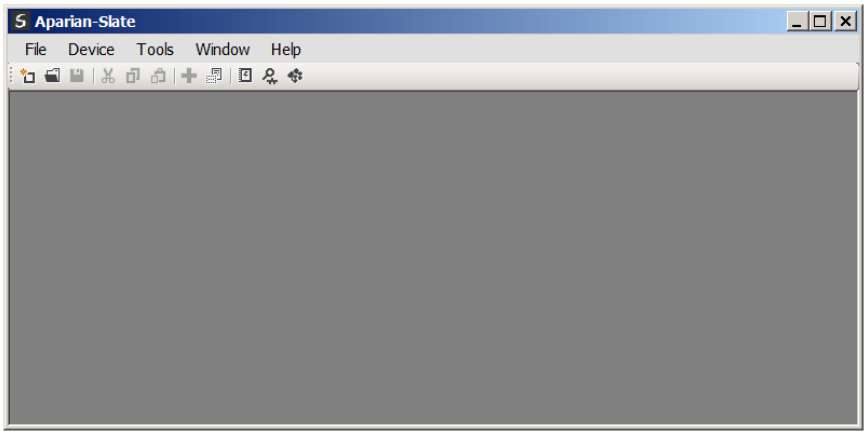



Figure 3.1. - Aparian Slate Environment

## 3.2. NETWORK PARAMETERS

The module will have DHCP (Dynamic Host Configuration Protocol) enabled as factory default. Therefore, a DHCP server must be used to provide the module with the required network parameters (IP address, subnet mask, etc.). There are a number of DHCP utilities available, however it is recommended that the DHCP server in Slate be used.

Within the Slate environment, the DHCP server can be found under the Tools menu.

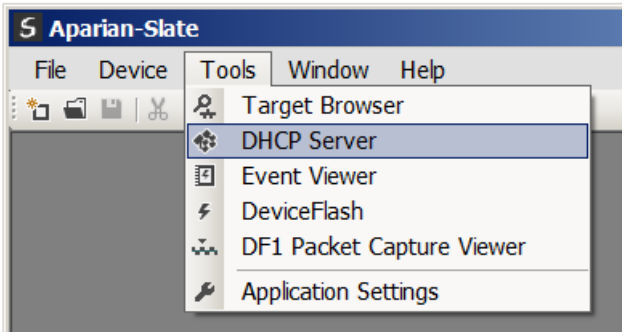



Figure 3.2. - Selecting DHCP Server

Once opened, the DHCP server will listen on all available network adapters for DHCP requests and display their corresponding MAC addresses.

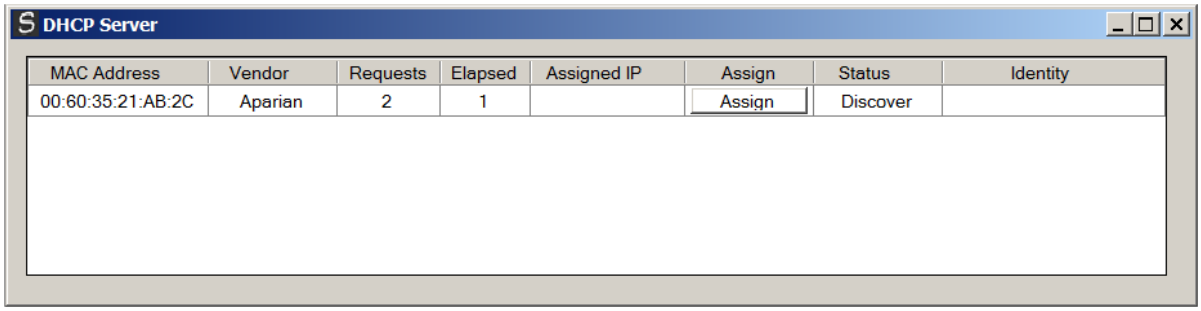



Figure 3.3. - DHCP Server

**NOTE:** If the DHCP requests are not displayed in the DHCP Server it may be due to the local PC's firewall. During installation the necessary firewall rules are automatically created for the Windows firewall. Another possibility is that another DHCP Server is operational on the network and it has assigned the IP address.

To assign an IP address, click on the corresponding "Assign" button. The IP Address Assignment window will open.

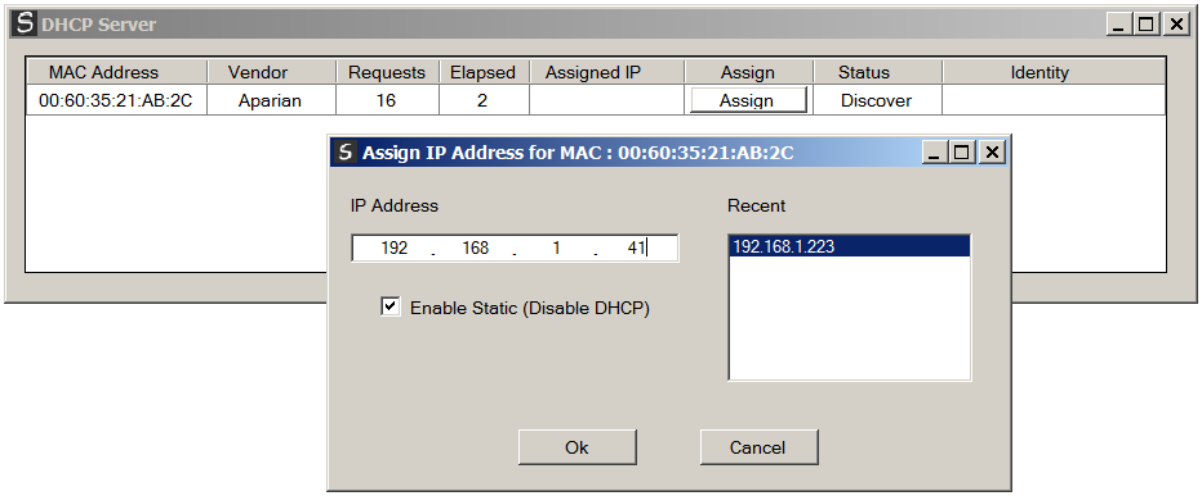



Figure 3.4. - Assigning IP Address

The required IP address can then be either entered, or a recently used IP address can be selected by clicking on an item in the Recent List.

If the "Enable Static" checkbox is checked, then the IP address will be set to static after the IP assignment, thereby disabling future DHCP requests.

Once the IP address window has been accepted, the DHCP server will automatically assign the IP address to the module and then read the Identity object Product name from the device.

The successful assignment of the IP address by the device is indicated by the green background of the associated row.

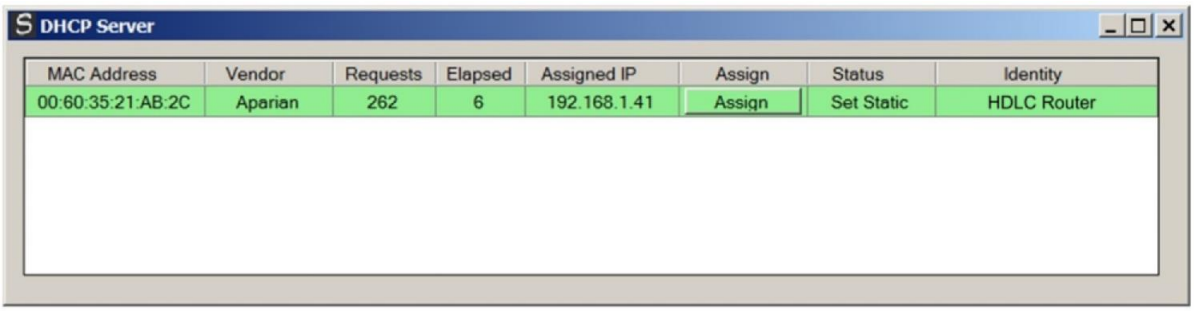



Figure 3.5. - Successful IP address assignment

It is possible to force the module back into DHCP mode by powering up the device with DIP switch 2 set to the On position.

A new IP address can then be assigned by repeating the previous steps.

**NOTE:** It is important to return DIP switch 2 back to Off position, to avoid the module returning to a DHCP mode after the power is cycled again.

If the module's DIP switch 2 is in the On position during the address assignment, the user will be warned by the following message.




Figure 3.6. - Force DHCP warning

In addition to the setting the IP address, a number of other network parameters can be set during the DHCP process. These settings can be viewed and edited in Slate's Application Settings, in the DHCP Server tab.

Once the DHCP process has been completed, the network settings can be set using the Ethernet Port Configuration via the Target Browser.

The Target Browser can be accessed under the Tools menu.

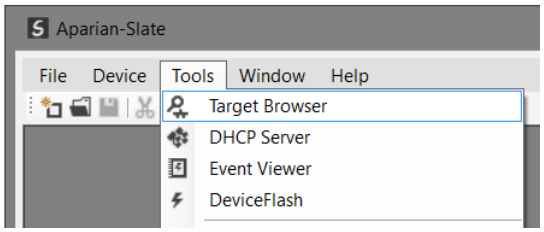



Figure 3.7. - Selecting the Target Browser

The Target Browser automatically scans the Ethernet network for EtherNet/IP devices.

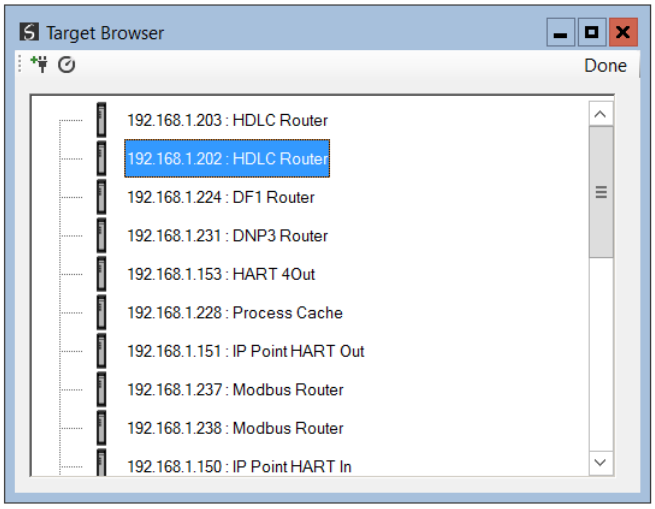



Figure 3.8. - Target Browser

Right-clicking on a device, reveals the context menu, including the Port Configuration option.

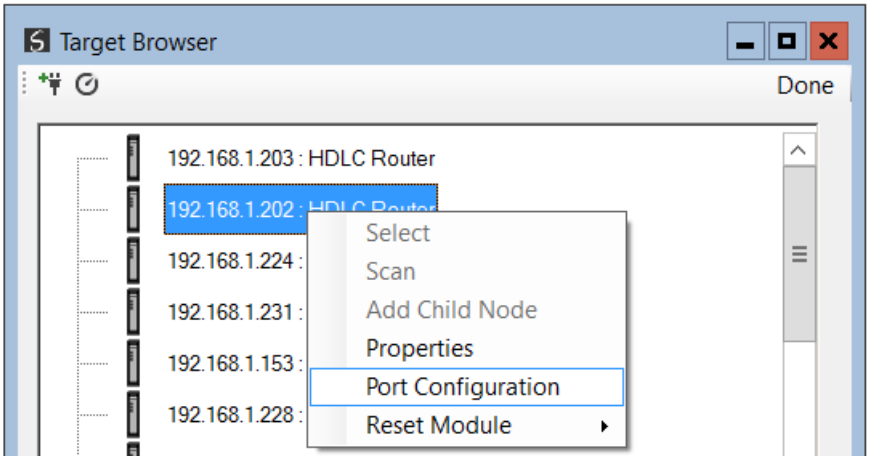



Figure 3.9. - Selecting Port Configuration

All the relevant Ethernet port configuration parameters can be modified using the Port Configuration window.

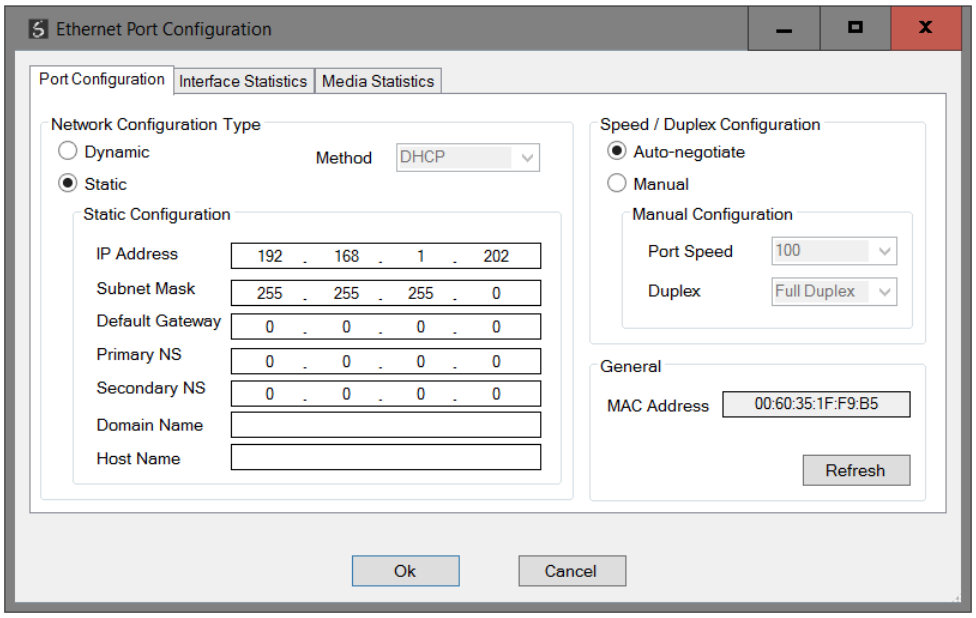



Figure 3.10. - Port Configuration

Alternatively, these parameters can be modified using Rockwell Automation's RSLinx software.

## 3.3. CREATING A NEW PROJECT

Before the user can configure the module, a new Slate project must be created. Under the ***File*** menu, select ***New***.

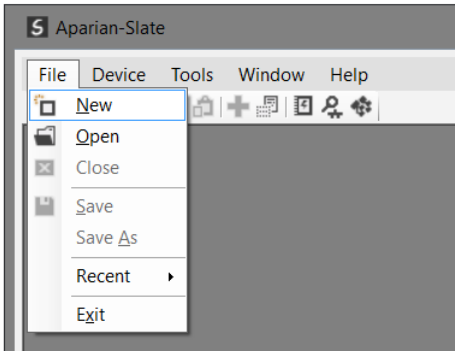



Figure 3.11. - Creating a new project

A Slate project will be created, showing the Project Explorer tree view. To save the project use the ***Save*** option under the File menu.

A new device can now be added by selecting ***Add*** under the Device menu.

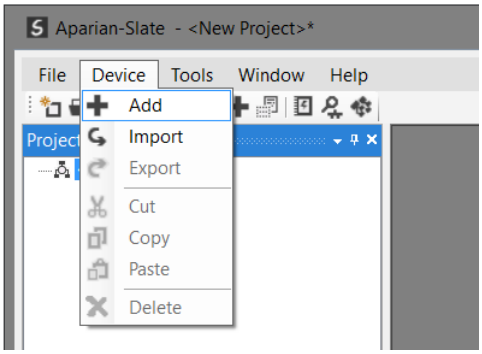



Figure 3.12. - Adding a new device

In the Add New Device window select the ***HDLC Router***, and click the Ok button.

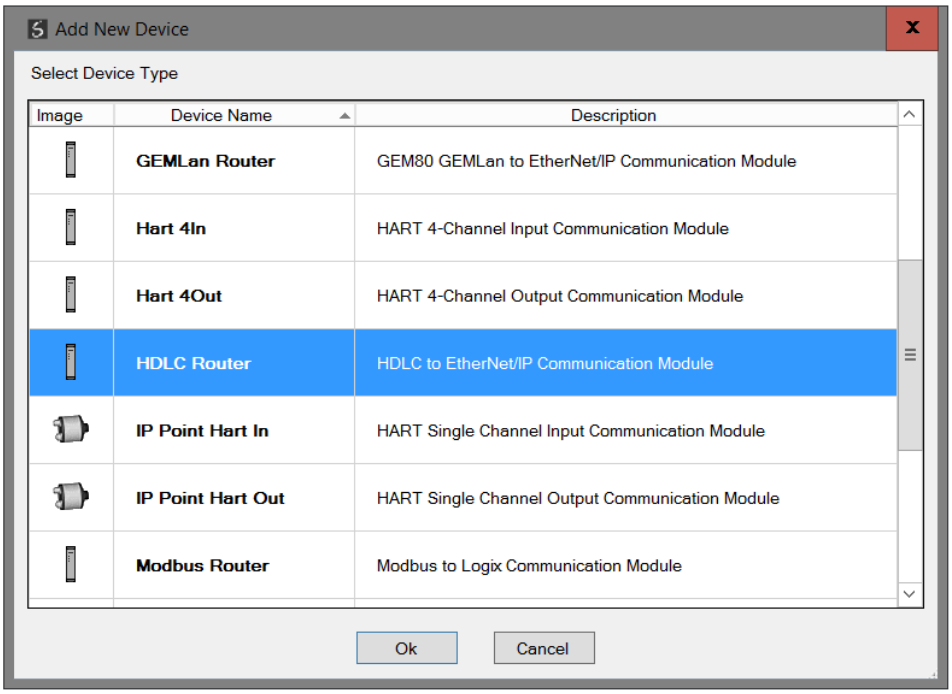



Figure 3.13 – Selecting a new HDLC Router

The device will appear in the Project Explorer tree as shown below, and its configuration window opened.

The device configuration window can be reopened by either double clicking the module in the Project Explorer tree or right-clicking the module and selecting *Configuration*.

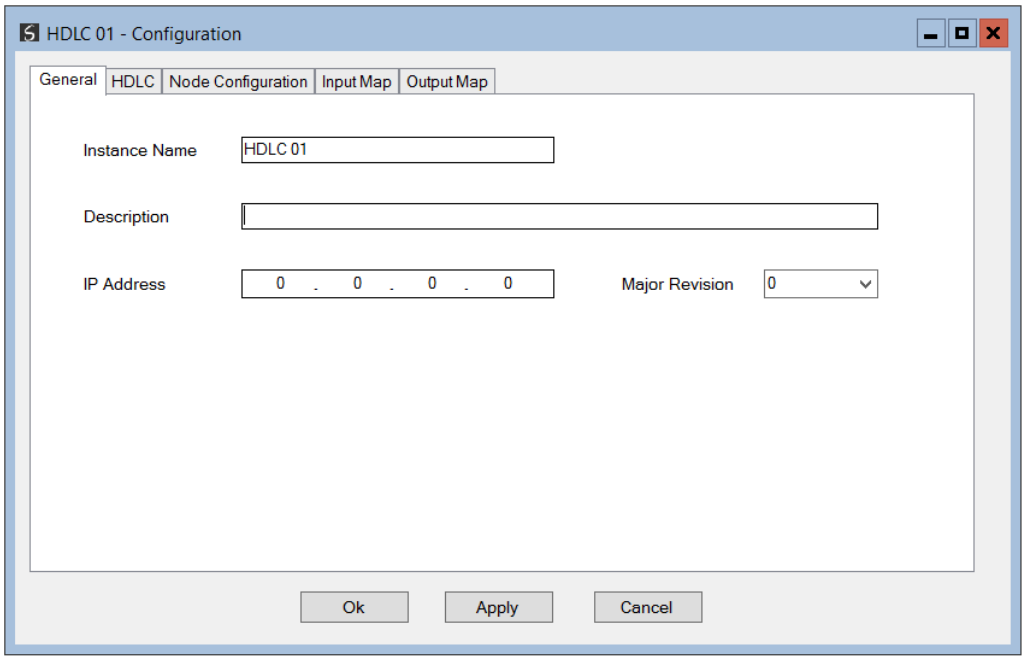



Figure 3.14. – HDLC Router configuration

Refer to the additional information section in this document for Slate's installation and operation documentation.

## 3.4. HDLC ROUTER CONFIGURATION

The HDLC Router configuration is configured using Slate. This includes the HDLC communication settings and the mapping to the Logix input and output assemblies.

When downloading this configuration into the module it will be saved in non-volatile memory that persists when the module is powered down.



**NOTE:** When a firmware upgrade is performed, the module will clear all configuration and mappings.

The general configuration consists of the following parameters:

| Parameter | Description |
|---|---|
| Instance Name | User defined name to identify between various HDLC Routers. |
| Description | User defined description for the module. |
| IP Address | The IP Address used to connect to the module. |
| Major Revision | The major revision of the module |

Table 3.1 - General configuration parameters

The general configuration is shown in the figure below. The HDLC general configuration window is opened by either double clicking on the module in the tree or right-clicking the module and selecting *Configuration*.

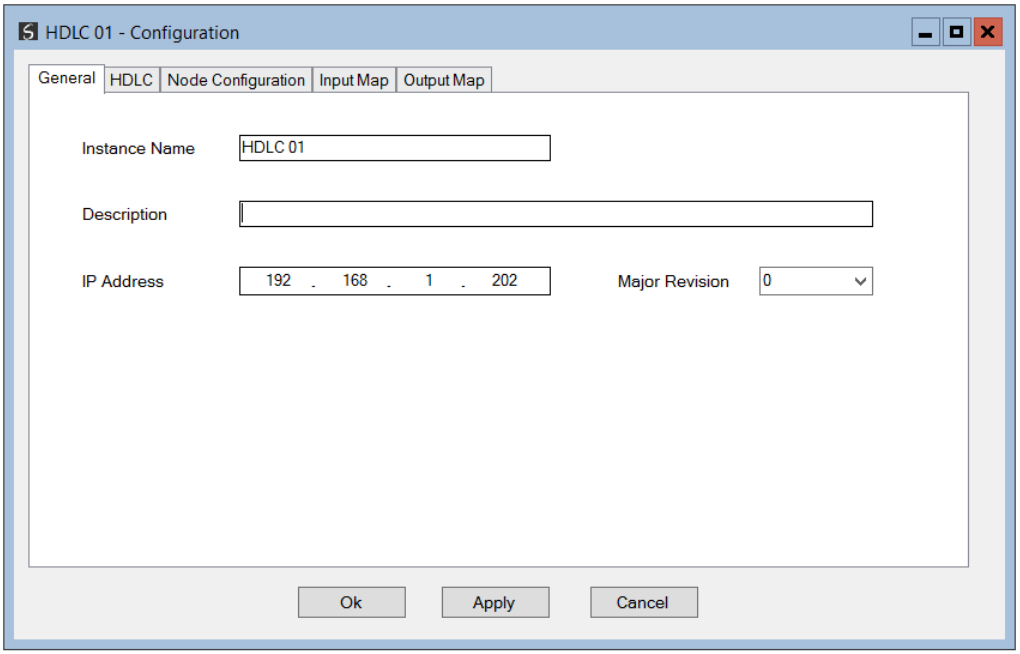



Figure 3.15 - General Configuration

The HDLC communication configuration consists of the following parameters:

| Parameter | Description |
|---|---|
| Mode | The mode in which the Router will operate, either:<br>• **Master** – Initiating requests<br>• **Slave** – Responding to requests |
| Connection | The physical topology for the HDLC (RS485) connection, either:<br>• **2-wire** – Use 2-wire Tx and Rx multiplexed connection<br>• **4-wire** – Use 4-wire with separate connections for Tx and Rx<br>Refer to section 2.4 for information on the choices. |
| Duplex | The Duplex can be setup in one of the following two modes:<br>• **Full Duplex** – When using point-to-point communication.<br>• **Half Duplex** – When using multidrop communication.<br><br>**NOTE:** When the HDLC Router is configured as a 4-wire HDLC Master in a multidrop slave network, then the communication duplex will still be setup as **Full duplex**. This is because the HDLC Master sends to all on one communication pair, while the HDLC Slaves share the return communication pair.<br><br>**NOTE:** When the HDLC Router is configured as a 4-wire HDLC Slave in a multidrop slave network (i.e., Interface mode – Broadcast with Response), then the communication duplex will still be setup as **Half Duplex**. This is because the HDLC Master sends to all |

| | |
|---|---|
| | on one communication pair, while the HDLC Slaves share the return communication pair. |
| BAUD Rate | The BAUD rate to be used, either:<br>• **48 kb/s**<br>• **180 kb/s** |
| Encoding | The bit encoding method to be used, either:<br>• **NRZI** – Non-return to zero inverse encoding<br>• **FM0** – Biphase space encoding<br>• **FM1** – Biphase mark encoding |
| Interface mode | The interface mode can be set to either of the following:<br>• **Table Exchange**<br>• **Broadcast with Response**<br>• **Broadcast without Response** |
| Master Rate | The rate at which the module send requests when in master mode, expressed in milliseconds. |
| Response Delay | When in Slave mode, this is the time delay (milliseconds) between when a request is received and the reply is sent.<br>In Master mode, this defines the time (milliseconds) between when a request is received before the next request is sent. |
| Node Timeout | The amount of time of inactivity for a specific node before the node is declared offline (de-active). The time is expressed in multiples of 50ms. |
| Response Timeout | The time in milliseconds the HLDC Router waits when there is no communication on the HDLC bus. |
| Enable Timeout Packet Repeat | When enabled the HDLC Router will resend the last packet sent when the Response Timeout is reached. |

Table 3.2 – HDLC communication configuration parameters

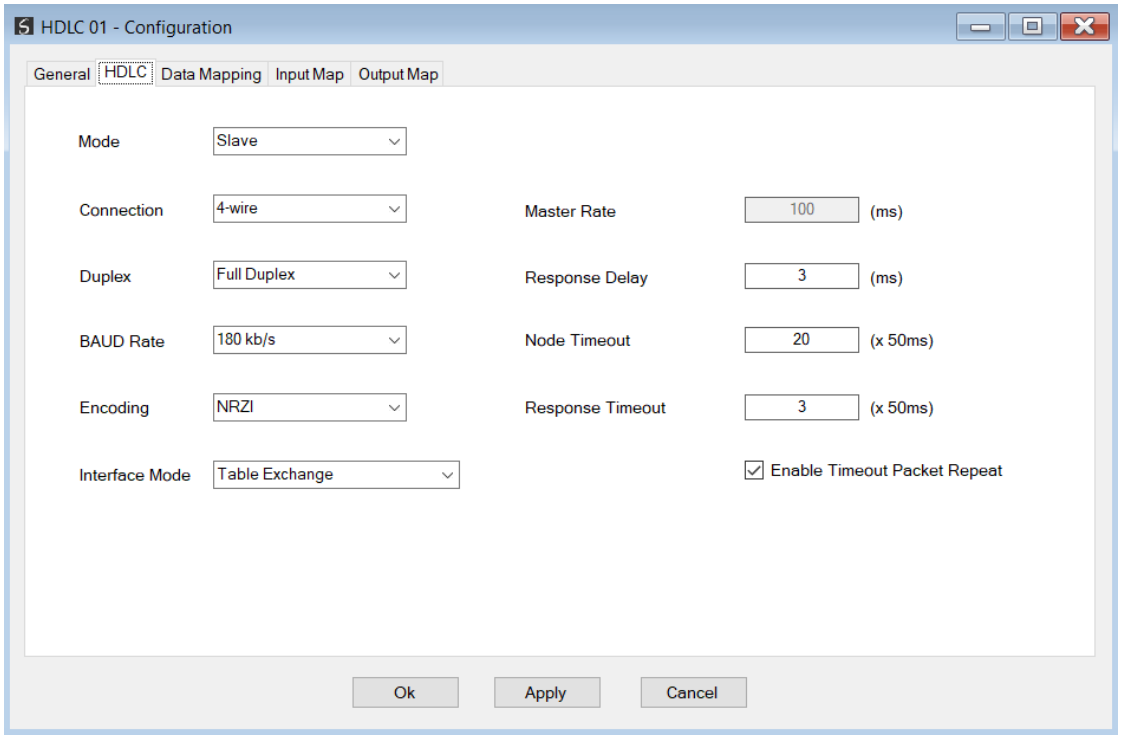



Figure 3.16 – HDLC communication configuration

## 3.5. HDLC MAPPING

The third tab is used to configure the data mapping between the remote HDLC devices and the Logix input and output data assemblies.

Each remote HDLC device is entered one row at a time with the corresponding input, output and status word mapping entries.

The following parameters are associated with each device:

| Parameter | Description |
|---|---|
| Description | User defined description for the remote device. |
| Node | The HDLC node address for the remote device. |
| Input Offset | The offset in the Data Table, within the Logix Input assembly, to which the data from the remote device will be transferred. |
| Input Size | The number of 16 bit words to be transferred to the Logix Input assembly. |
| Output Offset | The offset in the Data Table, within the Logix Output assembly, from which the data will be transferred to the remote device. |
| Output Size | The number of 16 bit words to be transferred from the Logix Output assembly. |
| Status Offset | The offset in the Data Table, within the Logix Input assembly, where the Status word will be written. |

| | |
|---|---|
| | Note: The status word contains information regarding the connection to the remote device. |

Table 3.3 - HDLC Data Mapping

**NOTE**: When choosing the offset for an item, care must be taken to ensure that the resulting mapping is within the 192-word data table, and that it does not overlap with any other item's data. This is valid for all options except when the HDLC Router is configured as a HDLC Master with broadcast interface.

**NOTE:** When the HDLC Router is configured as a HDLC Master with broadcast communication (with and without response), Slate will allow the user to duplicate the output mapping for multiple devices to ensure the same data is sent to all devices on the network. All other HDLC parameter configuration combinations will not allow the user to have output mappings that overlap. Refer to the *Master Broadcast Configuration* technical application note for further details on broadcast communication configuration.

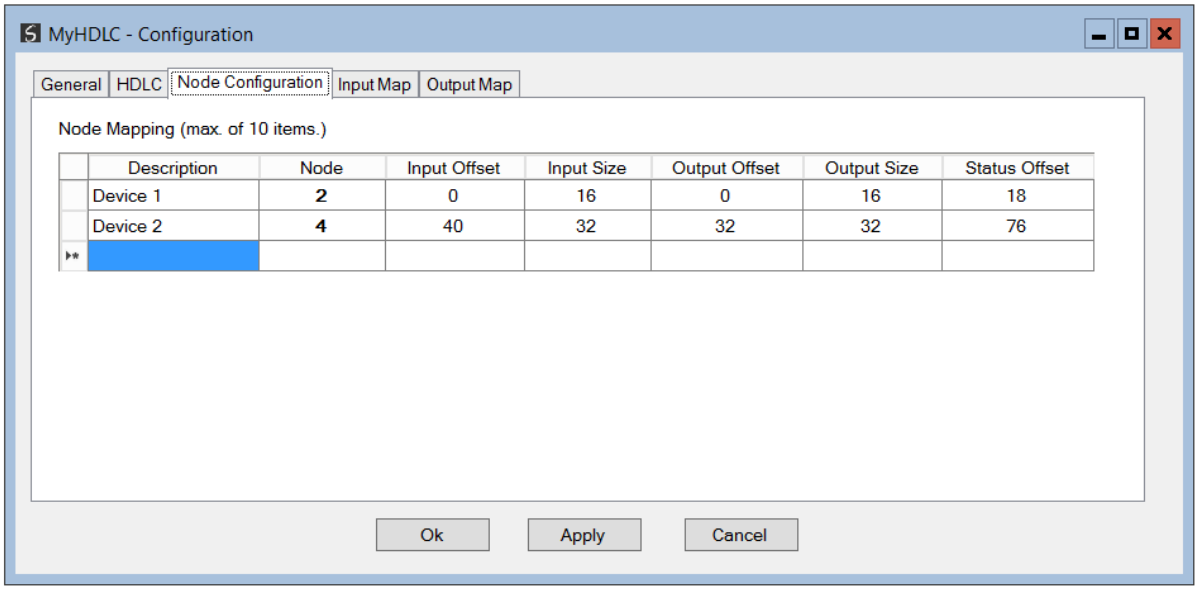



Figure 3.17 - HDLC Data Mapping tab

The Input Map and Output Map tabs are read-only tabs and provide a graphical display of the Input and Output Data Table respectively.

The node number occupying each data table word is placed inside the block. If the word is being used by a status word, then the node number as the letter 'S' appended to it.

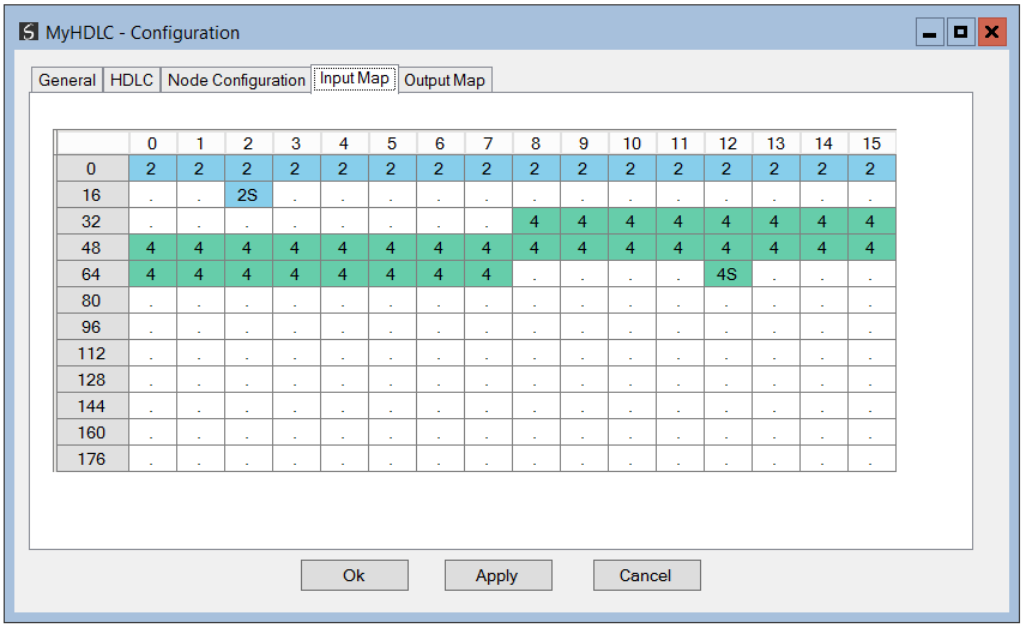



Figure 3.18 - Input Map

Should a word be assigned to more than one item, then this conflict will be represented by the letter 'X' with a red background.

**NOTE**: The downloading of the configuration will be blocked if any conflicts exist.

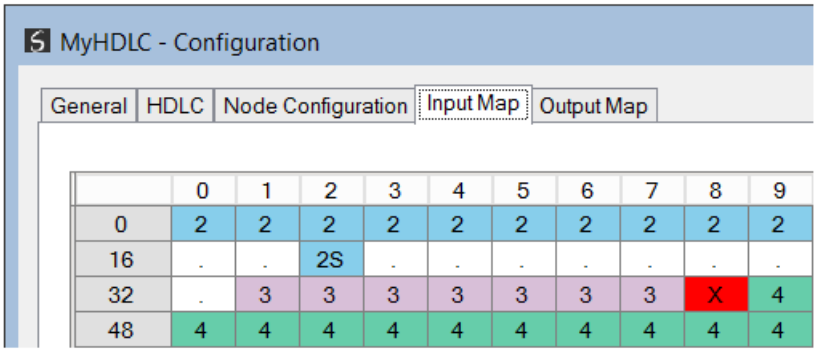



Figure 3.19 - Input Map Conflict

## 3.6. MODULE DOWNLOAD

Once the HDLC configuration has been completed, it must be downloaded to the module.

Before downloading the Connection Path of the module should be set. This path will automatically default to the IP address of the module, as set in the module configuration. It can however be modified, if the HDLC Router is not on a local network.

The Connection path can be set by right-clicking on the module and selecting the Connection Path option.

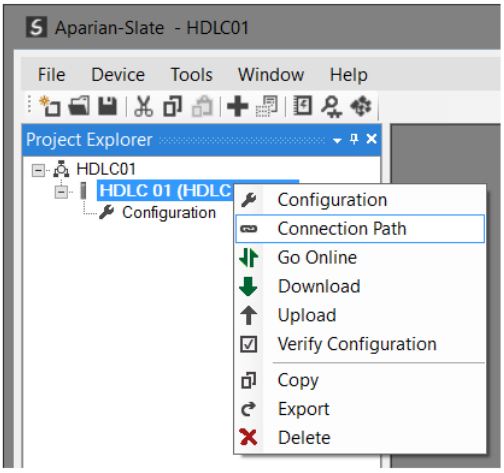



Figure 3.20 - Selecting Connection Path

The new connection path can then be either entered manually or selected by means of the Target Browser.

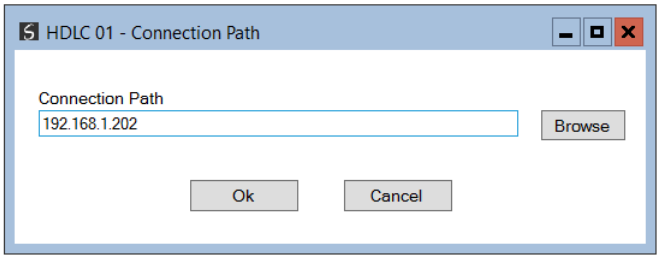



Figure 3.21 - Connection Path

To initiate the download, right-click on the module and select the Download option.

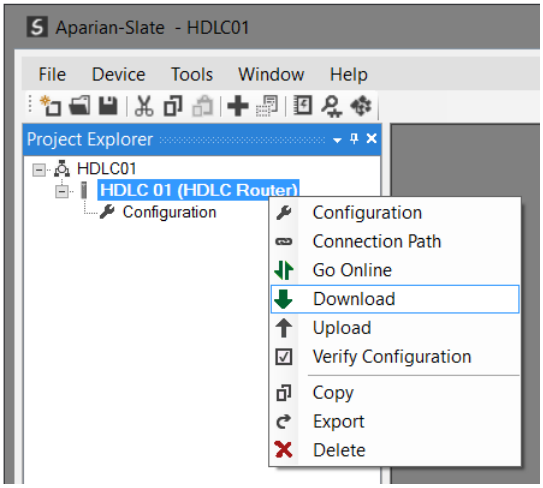



Figure 3.22 - Selecting Download

Once complete, the user will be notified that the download was successful.




Figure 3.23 - Successful download

The module is now configured and will start operating immediately.

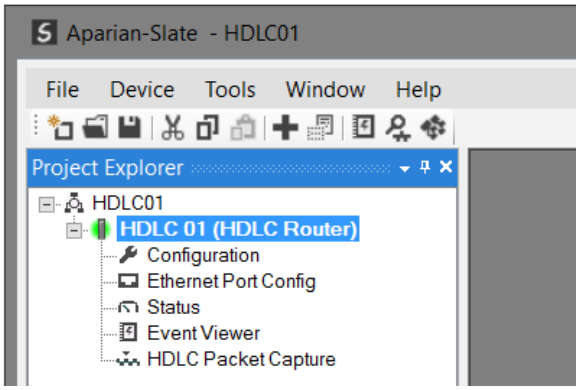



Figure 3.24 - Module online

# 3.7. RSLOGIX 5000 CONFIGURATION

The HDLC Router modules can be easily integrated with the Allen-Bradley Logix family of controllers. The HDLC Router will need to be configured for EtherNet/IP communication before it will connect to a Logix controller. This is done by setting the mode to EtherNet/IP under the general configuration (as shown below):

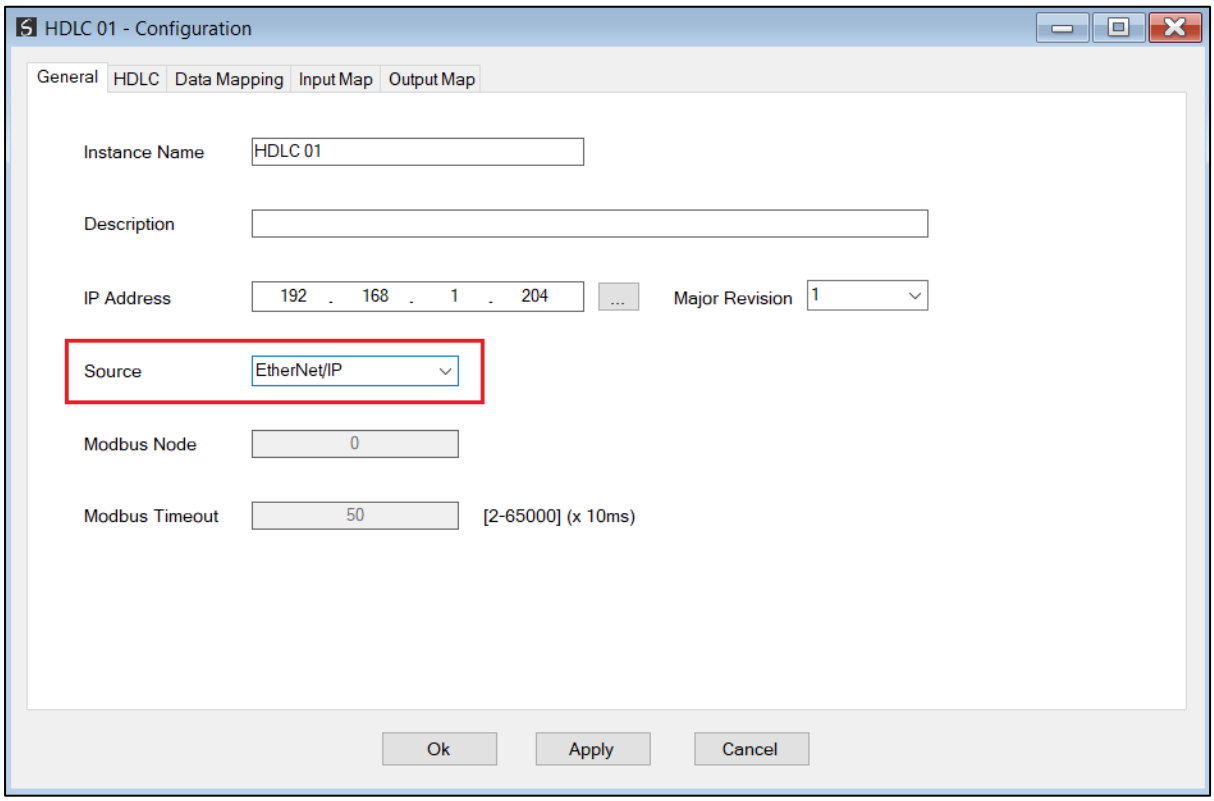



Figure 3.25. – EtherNet/IP source

#### 3.7.1.1. ADD MODULE TO I/O CONFIGURATION

Adding the module to the Logix IO tree makes use of the generic Ethernet module profile. This is done by right clicking on the Ethernet Bridge in the RSLogix 5000 and selecting *New Module* after which the *ETHERNET-MODULE* is selected to be added as shown in the figure below.

NOTE: See the next section for importing the configuration (L5X).

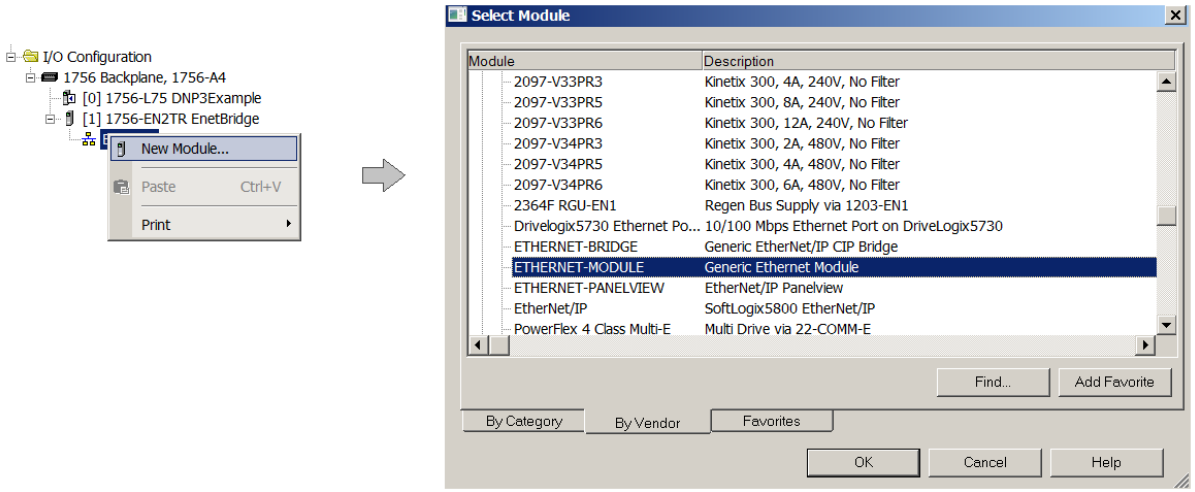



Figure 3.26 - Add a Generic Ethernet Module in RSLogix 5000

The user must enter the IP address of the HDLC Router that will be used. The assembly instance and size must also be added for the input, output, and configuration in the connection parameters section. Below are the required connection parameters.

| Connection Parameter | Assembly Instance | Size |
|---|---|---|
| Input | 119 | 480 (8-bit) |
| Output | 120 | 396 (8-bit) |
| Configuration | 102 | 0 (8-bit) |

Table 3.4 - RSLogix class 1 connection parameters for the HDLC Router

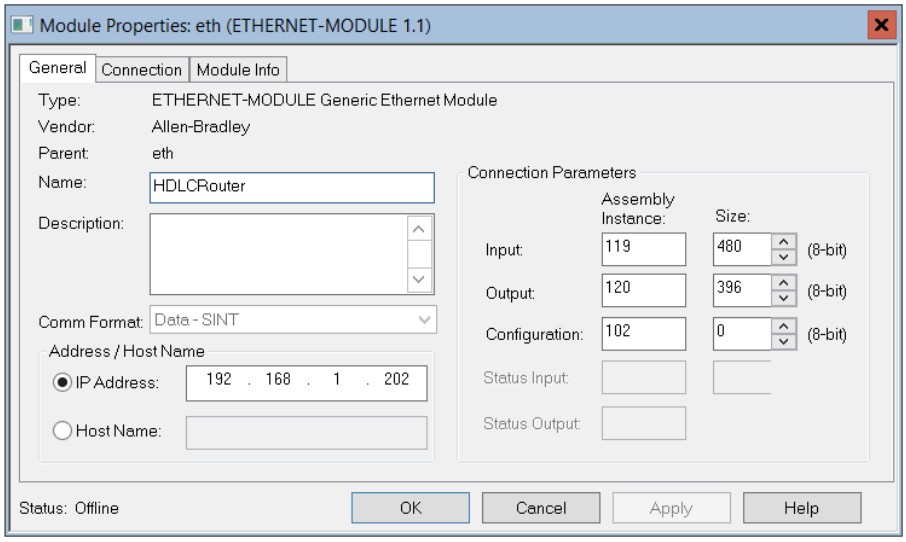



Figure 3.27 . - General module properties in RSLogix 5000

**NOTE:** The user will need to enter the exact connection parameters before the module will establish a class 1 connection with the Logix controller.

Next the user needs to add the connection requested packet interval (RPI). This is the rate at which the input and output assemblies are exchanged. The recommended value is 10ms. Refer to the technical specification section in this document for further details on the limits of the RPI.

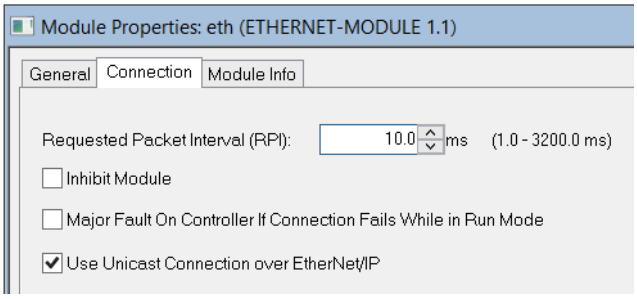



Figure 3.28 - Connection module properties in RSLogix 5000

Once the module has been added to the RSLogix 5000 IO tree the user must assign the User Defined Types (UDTs) to the input and output assemblies. The user can import the required UDTs by right-clicking on *User-Defined* sub-folder in the *Data Types* folder of the IO tree and selecting *Import Data Type*. The assemblies are then assigned to the UDTs with a ladder copy instruction (COP) as shown in the figure below.

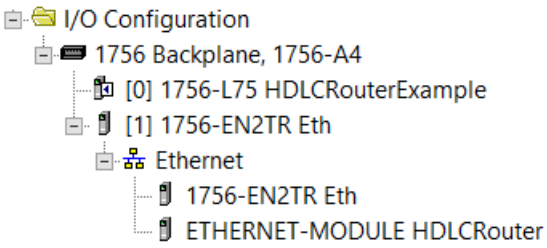



Figure 3.29 – RSLogix 5000 I/O module tree

#### 3.7.1.2. Importing UDTs and Mapping Routines

To simplify the mapping of the input image, an RSLogix 5000 Routine Partial Import (L5X) file is provided. This file can be imported by right-clicking on the required Program and selecting the Import Routine option.

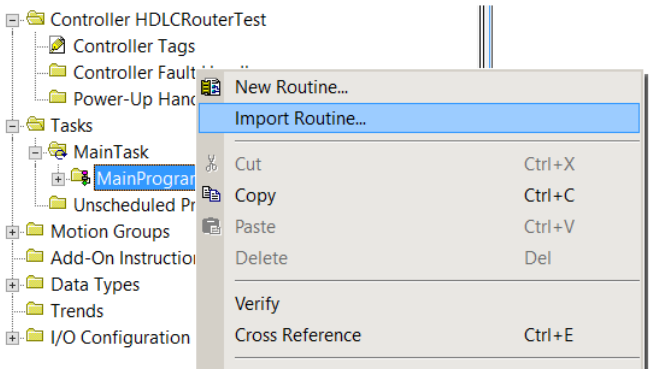



Figure 3.30 – RSLogix 5000 Importing HDLC Router specific routine and UDTs

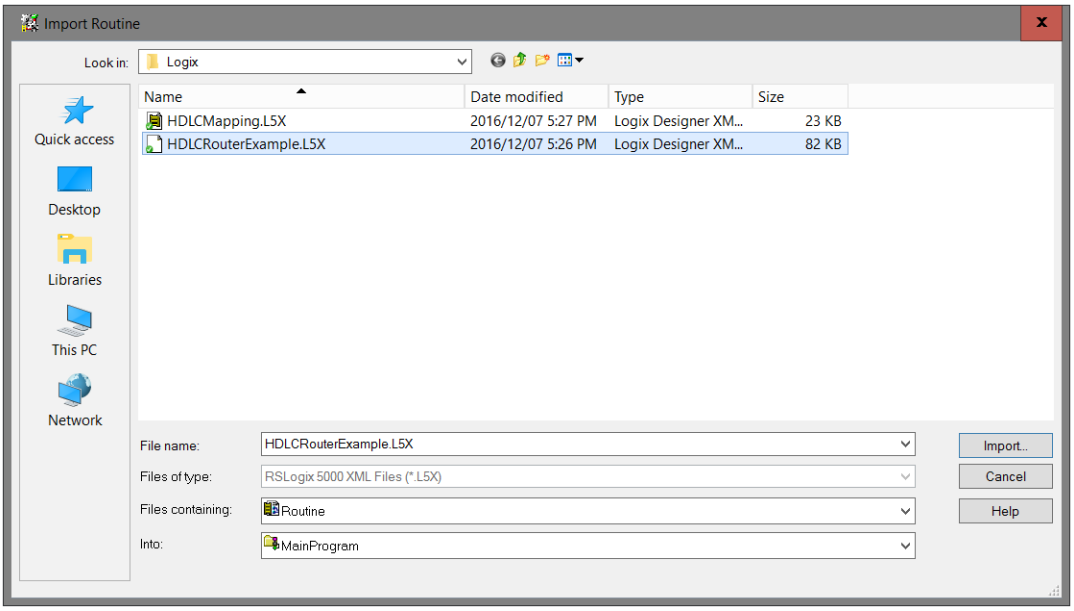



Figure 3.31 - Selecting partial import file

The import will create the following:

- The required UDTs (user defined data types)
- Two controller tags representing the Input and Output assemblies.
- A routine mapping the HDLC Router module to the aforementioned tags.

The user may need to change the routine to map to the correct HDLC Router module instance name, and make sure that the mapping routine is called by the Program's Main Routine.

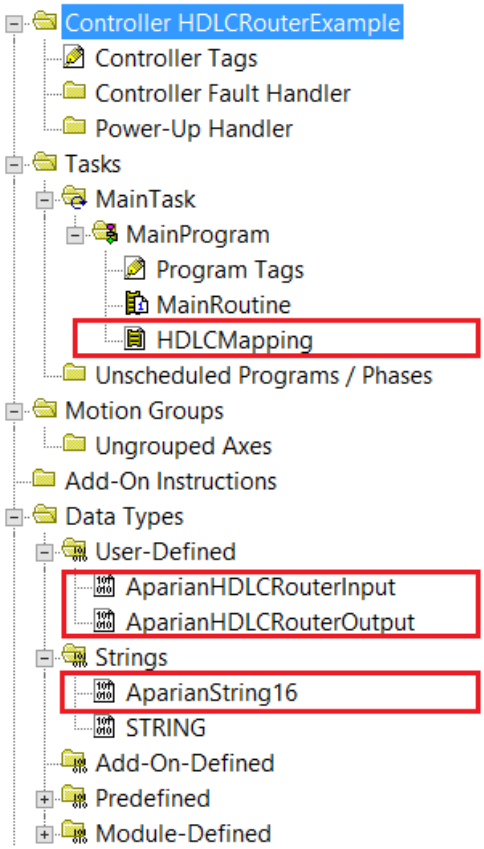



Figure 3.32 - Imported RSLogix 5000 objects

Refer to the additional information section of this document for an example RSLogix 5000 project as well as the required UDTs.

## 3.8. MODBUS CONFIGURATION

The HDLC Router will need to be configured for Modbus TCP communication before a Modbus Master will be able to communicate with it (the HDLC Router will operate as a Modbus Slave). This is done by setting the mode to Modbus TCP under the general configuration (as shown below).

The user will also need to configure a Modbus Node address for the HDLC Router which will be used by the Modbus Master to communicate with the HDLC Router.

The Modbus Timeout is used to determine the Modbus Holding Register write interval required to avoid a HDLC communication fault. The user will need to write to the Modbus Holding register section required for the Output Map at an interval less than the configured Modbus Timeout.

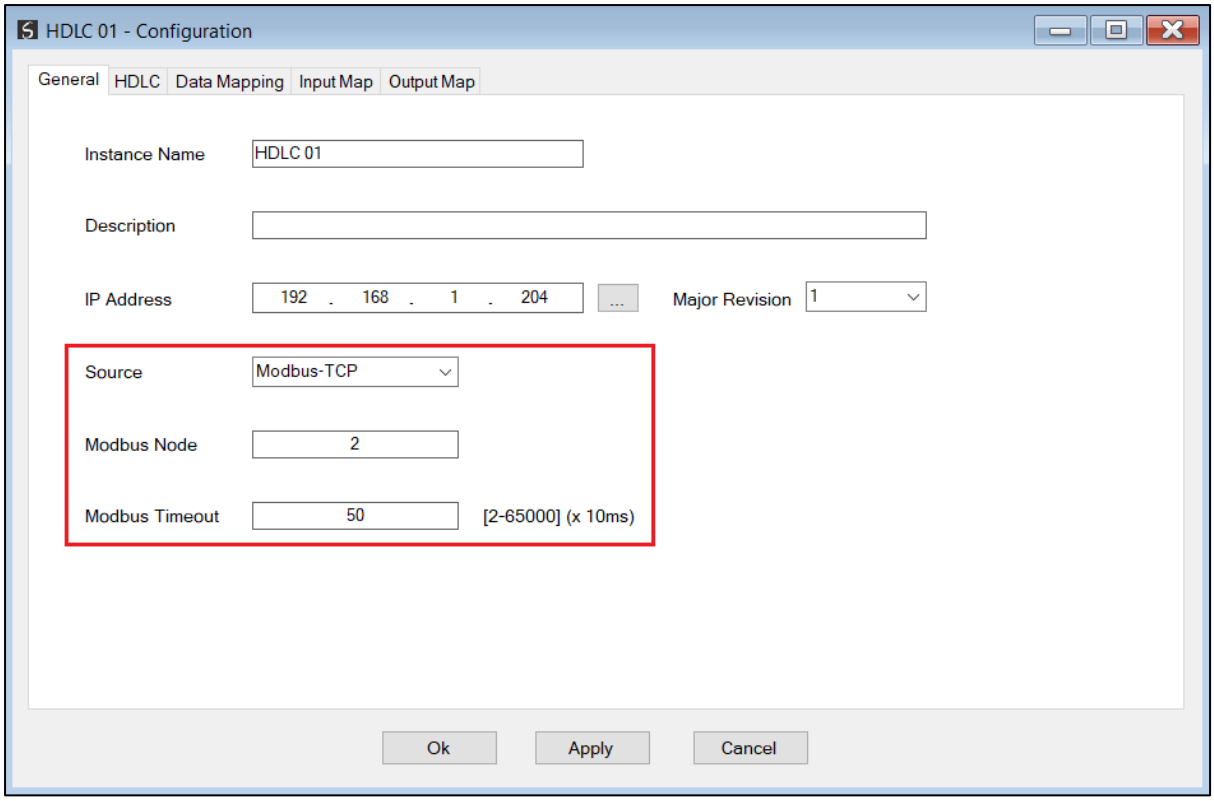



Figure 3.33. – Modbus TCP source

# 4. OPERATION

When the module is running it will exchange data between the remote HDLC node/s and the Logix controller. The operation of the module depends on the mode:

## 4.1. MASTER MODE

When the HDLC Router is in the Master mode, it will exchange data with the configured nodes in the mapping table one at a time. In this mode, the HDLC Router initiates the communication. The rate at which the remote HDLC nodes are polled is controlled by the configured ***Master Rate*** parameter.

## 4.2. SLAVE MODE

When in the Slave mode the HDLC Router responds only to the requests from the remote HDLC Master where the node number matches that of one of the entries in the mapping configuration. This allows the module to masquerade as more than one slave node on the HDLC network.

## 4.3. LOGIX ASSEMBLIES

The Logix controller will establish a class 1 cyclic communication connection with the HDLC Router and the input and output assemblies are exchanged at the RPI interval. The UDTs provided, will convert the input and output arrays into tag based assemblies. Refer to the additional information section in this document for the input and output UDTs.

| | | |
|---|---|---|
| HDLCRouter01Input | | AparianHDLCRouterInput |
| HDLCRouter01Input.InstanceName | | AparianString16 |
| HDLCRouter01Input.Status | Decimal | DINT |
| HDLCRouter01Input.Temperature | Float | REAL |
| HDLCRouter01Input.LastControlWord | Decimal | DINT |
| HDLCRouter01Input.MessageTxCount | Decimal | DINT |
| HDLCRouter01Input.MessageRxCount | Decimal | DINT |
| HDLCRouter01Input.SequenceErrors | Decimal | DINT |
| HDLCRouter01Input.Timeouts | Decimal | DINT |
| HDLCRouter01Input.CRCErrors | Decimal | DINT |
| HDLCRouter01Input.NodeDeactiveCount | Decimal | DINT |
| HDLCRouter01Input.ForeignNodeReceived | Decimal | DINT |
| HDLCRouter01Input.TransactionRate | Decimal | DINT |
| HDLCRouter01Input.Reserved | Decimal | DINT[8] |
| HDLCRouter01Input.DataInputMap | Decimal | INT[192] |
| HDLCRouter01Input.DataInputMap[0] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[1] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[2] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[3] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[4] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[5] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[6] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[7] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[8] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[9] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[10] | Decimal | INT |
| ● ● ● | | |
| HDLCRouter01Input.DataInputMap[190] | Decimal | INT |
| HDLCRouter01Input.DataInputMap[191] | Decimal | INT |

Figure 4.1. - Input assembly UDT structure

### 4.3.1. INPUT ASSEMBLY

The following parameters are used in the input assembly of the module.

| Parameter | Datatype | Description |
|---|---|---|
| InstanceName | STRING | This parameter is the instance name of the module that was configured under the general HDLC configuration in Slate. |
| Status.00 | BOOL | Configuration Valid. |
| Status.01 - .15 | BOOL | Reserved |
| Temperature | REAL | The internal temperature of the module. |

| | | |
|---|---|---|
| LastControlWord | DINT | The value of the control word in the last receive HDLC packet. |
| MessageTxCount | DINT | The total number of HDLC packets sent by the module. |
| MessageRxCount | DINT | The number of HDLC packets received by the module. |
| Sequence Errors | DINT | The number of HDLC packet sequence errors detected. |
| TimeOuts | DINT | The number of HDLC requests that did not receive a valid reply within the timeout period. |
| CRCErrors | DINT | The number of corrupted HDLC packets received by the module. |
| NodeDeactiveCount | DINT | The number of times a node was flagged as deactive. |
| ForeignNodeReceived | DINT | The number of HDLC requests received for a node number not configured in the module.<br>Note that this does not necessarily represent an error should other slave nodes be present on the network. |
| TransactionRate | DINT | The transaction rate is the number of HDLC messages per second that the module is currently processing. |
| Reserved[8] | DINT | Reserved |
| DataInputMap[192] | INT | The Data Input map. An array of 16 bit integers containing the input data received from HDLC devices as mapped in the configuration. |

Table 4.1 - RSLogix 5000 input assembly parameters

### 4.3.2. OUTPUT ASSEMBLY

The following parameters are used in the output assembly of the module.

| | | |
|---|---|---|
| HDLCRouter01Output | | AparianHDLCRouterOutput |
| HDLCRouter01Output.GeneralControl | Decimal | DINT |
| HDLCRouter01Output.Reserved | Decimal | DINT[2] |
| HDLCRouter01Output.DataOutputMap | Decimal | INT[192] |
| HDLCRouter01Output.DataOutputMap[0] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[1] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[2] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[3] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[4] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[5] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[6] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[7] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[8] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[9] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[10] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[11] | Decimal | INT |
| ● ● ● | | |
| HDLCRouter01Output.DataOutputMap[190] | Decimal | INT |
| HDLCRouter01Output.DataOutputMap[191] | Decimal | INT |

Figure 4.2. - Output assembly UDT structure

| Parameter | Datatype | Description |
|---|---|---|
| GeneralControl | DINT | Reserved for future use. |
| Reserved[2] | DINT | Reserved |
| DataOutputMap[192] | INT | The Data Output map. An array of 16 bit integers containing the output data to be sent to the HDLC devices as mapped in the configuration. |

Table 4.2 - RSLogix 5000 output assembly parameters

## 4.4. STATUS WORD

Each mapped item contains the offset for the Status word. This word provides diagnostic information about that specific node.

| Bit | Parameter | Description |
|---|---|---|
| 0-7 | Transaction Counter | Increments each time a successful exchange with the node takes place. |
| 8 | Reserved | |
| 9 | Reserved | |
| 10 | Timeout | Flag indicating that the node has timed-out. |
| 11 | CRC Error | Flag indicating that the last packet received from the node was corrupted, that is, the CRC failed. |
| 12 | Data Size Mismatch | The data received from a specific node does not match the configured Input Size value in Slate. |
| 13 | Reserved | |
| 14 | Active | Flag indicating that the node is active and responding normally.<br>Note: This flag should be used in the Logix application code in conjunction with the HDLC Router connection status, so as to interlock or disable certain control functions when the HDLC communication has failed. |
| 15 | Reserved | |

Table 4.3. – Node Status Word structure

## 4.5. MODBUS REGISTERS

The Modbus TCP operation is enabled when the mode is set to either Modbus TCP. The HDLC Router module will then operate as a Modbus TCP Slave supporting the following Modbus registers.

| Register Type: | **Holding Registers** | | | |
|---|---|---|---|---|
| **Input Assembly (read-only)** | **Byte Length** | **Date Type** | **Register** | **Description** |
| Instance Name Length | 4 | DINT | 0 | This parameter is the instance name of the module that was configured under the general Time Sync configuration in Slate. |
| Instance Name | 16 | SINT[16] | 2 | |
| Status | 4 | DINT | 10 | |
| Status.00 | | BOOL | | Configuration Valid. |

| | | | | |
|---|---|---|---|---|
| Status.01 - .15 | | BOOL | | Reserved |
| Temperature | 4 | REAL | 12 | The internal temperature of the module. |
| LastControlWord | 4 | DINT | 14 | The value of the control word in the last receive HDLC packet. |
| MessageTxCount | 4 | DINT | 16 | The total number of HDLC packets sent by the module. |
| MessageRxCount | 4 | DINT | 18 | The number of HDLC packets received by the module. |
| Sequence Errors | 4 | DINT | 20 | The number of HDLC packet sequence errors detected. |
| TimeOuts | 4 | DINT | 22 | The number of HDLC requests that did not receive a valid reply within the timeout period. |
| CRCErrors | 4 | DINT | 24 | The number of corrupted HDLC packets received by the module. |
| NodeDeactiveCount | 4 | DINT | 26 | The number of times a node was flagged as deactive. |
| ForeignNodeReceived | 4 | DINT | 28 | The number of HDLC requests received for a node number not configured in the module.<br>Note that this does not necessarily represent an error should other slave nodes be present on the network. |
| TransactionRate | 4 | DINT | 30 | The transaction rate is the number of HDLC messages per second that the module is currently processing. |
| Reserved[8] | 32 | DINT[8] | 32 | |
| DataInputMap[192] | 384 | INT[192] | 48 | The Data Input map. An array of 16 bit integers containing the input data received from HDLC devices as mapped in the configuration. |
| | | | | |
| **Output Assembly (write-only)** | **Byte Length** | **Date Type** | **Register** | **Description** |
| GeneralControl | 4 | DINT | 500 | Reserved for future use. |
| Reserved | 8 | DINT[2] | 502 | Reserved |
| DataOutputMap[192] | 384 | INT[192] | 506 | The Data Output map. An array of 16 bit integers containing the output data to be sent to the HDLC devices as mapped in the configuration. |
| | | | | |

Table 4.4 - Modbus Register Map

# 5. DIAGNOSTICS

## 5.1. LEDS

The module provides four LEDs for diagnostics purposes as shown in the front view figure below. A description of each LED is given in the table below.

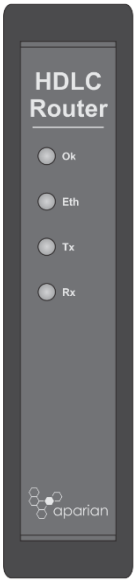



Figure 5.1 - HDLC Router front view

| LED | Description |
|---|---|
| Ok | The module's Ok LED will provide information regarding the system-level operation of the module. If the LED is red then the module is not operating correctly. For example, if the module application firmware has been corrupted or if there is a hardware fault the module will have a red Module LED.<br>If the LED is green then the module has booted and is running correctly. |
| Eth | The Ethernet LED will light up when an Ethernet link has been detected (by plugging in a connected Ethernet cable). The LED will flash every time traffic was detected. |
| Tx | The Transmit LED will flash green each time an HDLC frame is sent. |
| Rx | The Receive LED will flash green each time a valid HDLC frame is received. If the frame is not valid, then the LED will flash red. |

Table 5.1 - Module LED operation

## 5.2. MODULE STATUS MONITORING IN SLATE

The HDLC Router can provide a number of statistics which can assist with module operation, maintenance, and fault finding. The statistics can be accessed in full by Slate or using the web server in the module.

To view the module's status in the Aparian-Slate environment, the module must be online. If the module is not already Online (following a recent configuration download), then right-click on the module and select the ***Go Online*** option.

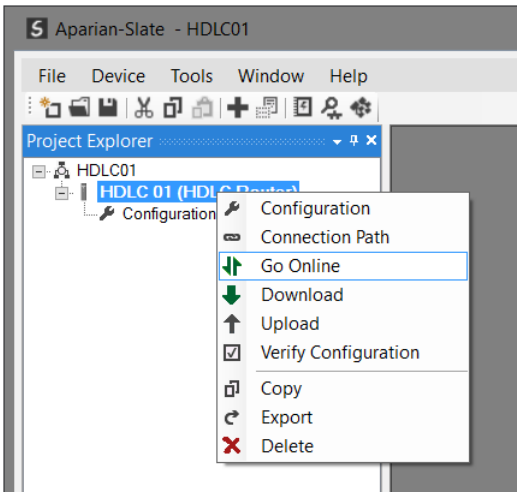



Figure 5.2 - Selecting to Go Online

The Online mode is indicated by the green circle behind the module in the Project Explorer tree.

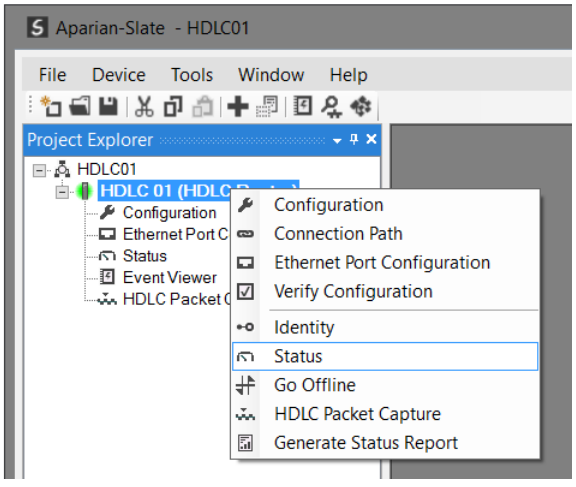



Figure 5.3 - Selecting Online Status

The Status monitoring window can be opened by either double-clicking on the ***Status*** item in the Project Explorer tree, or by right-clicking on the module and selecting ***Status***.

The status window contains multiple tabs to display the current status of the module.

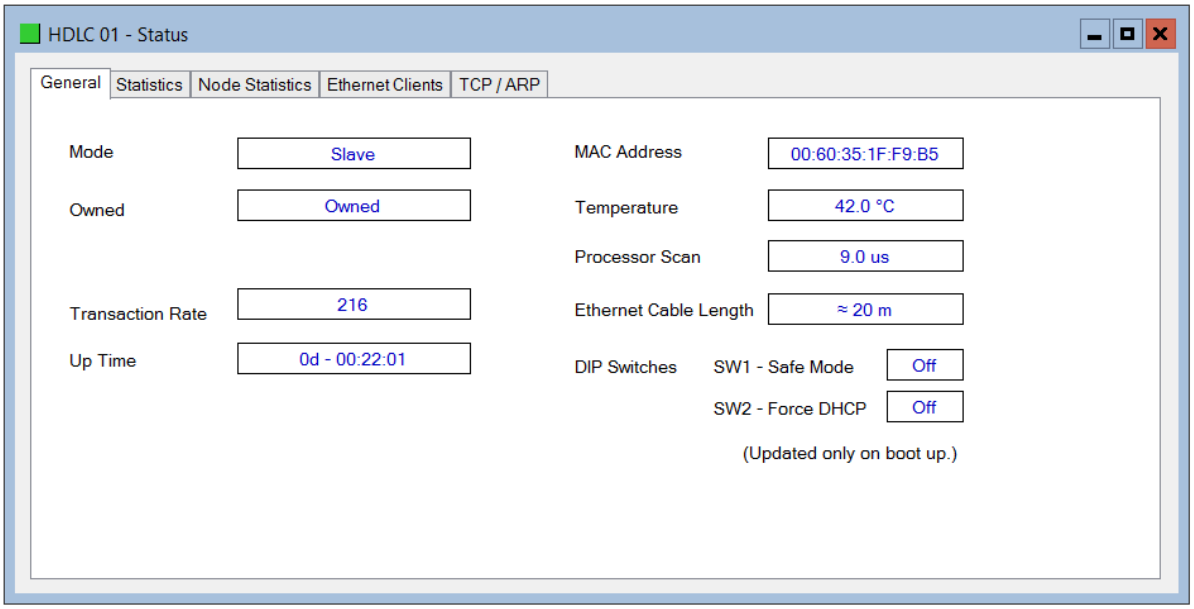



Figure 5.4 - Status Monitoring - General

The General tab displays the following general parameters and can also be used to set the module time to the PC time:

| Parameter | Description |
|---|---|
| Mode | Indicates the current operating mode, either Master or Slave. |
| Owned | Indicates whether or not the module is currently owned (Class 1) by a Logix controller. |
| Transaction Rate | The transaction rate is the number of HDLC messages per second that the module is currently processing. |
| Up Time | Indicates the elapsed time since the module was powered-up. |
| MAC Address | Displays the module's unique Ethernet MAC address. |
| Temperature | The internal temperature of the module. |
| Processor Scan | The amount of time (microseconds) taken by the module's processor in the last scan. |
| Ethernet Cable Length | Indicates the estimated length of the Ethernet cable attached to the module. (Accuracy of 5m) |
| DIP Switch Position | The status of the DIP switches when the module booted.<br>Note that this status will not change if the DIP switches are altered when the module is running. |

Table 5.2 - Parameters displayed in the Status Monitoring – General Tab

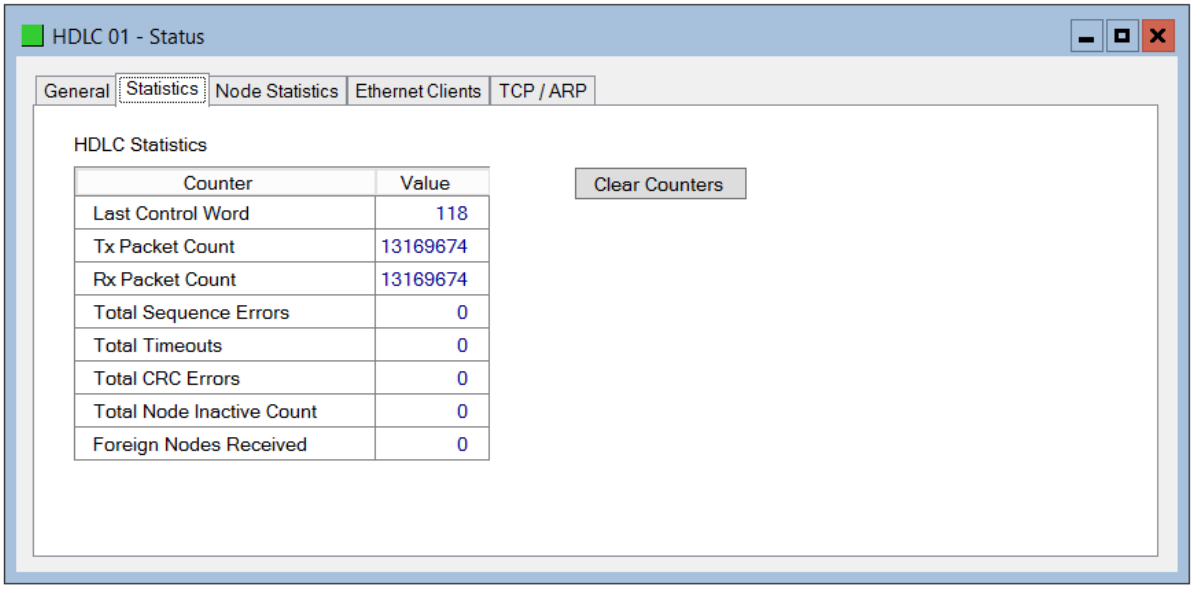



Figure 5.5 - Status monitoring - HDLC Statistics

The HDLC Statistics tab displays the statistics associated with the HDLC communication and mapping.

| Statistic | Description |
|---|---|
| Last Control Word | The value of the control word in the last receive HDLC packet. |
| Tx Packet Count | The total number of HDLC packets sent by the module. |
| Rx Packet Count | The number of HDLC packets received by the module. |
| Total Sequence Errors | The number of HDLC packet sequence errors detected. |
| Total Timeouts | The number of HDLC requests that did not receive a valid reply within the timeout period. |
| Total CRC Errors | The number of corrupted HDLC packets received by the module. |
| Total Node Inactive Count | The number of times any node was flagged as inactive. |
| Foreign Nodes Received | The number of HDLC requests received for a node number not configured in the module. |

Table 5.3 – HDLC statistics

The Map Items tab will display the successful packet counts processed by each mapping item.

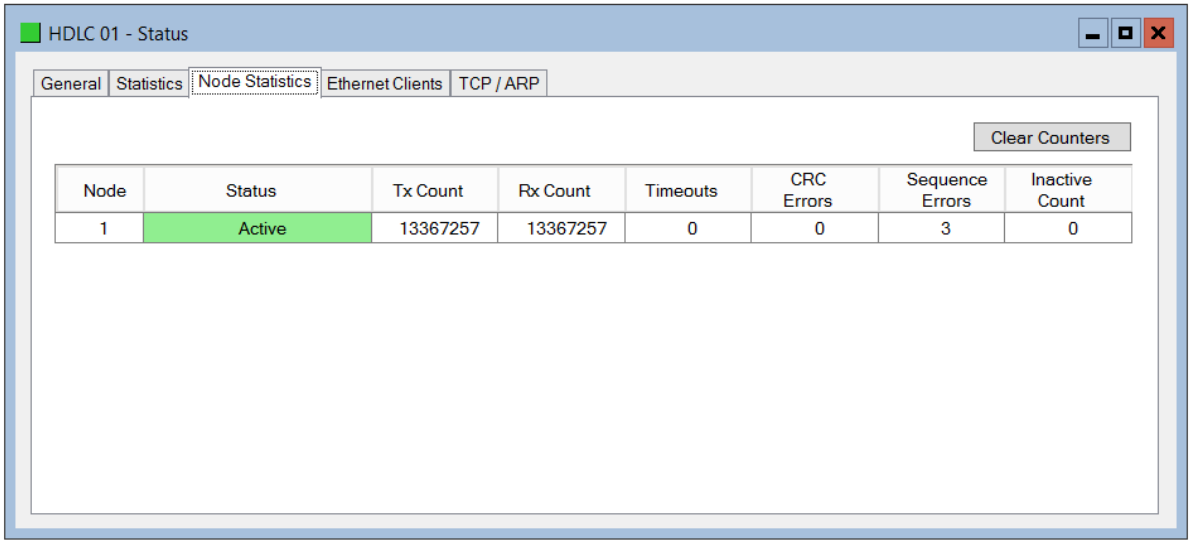



Figure 5.7. - Map Item status

The HDLC Node Statistics tab displays the statistics associated with each individual HDLC node.

| Statistic | Description |
|---|---|
| Node | The HDLC node address. |
| Status | Indicates the current status of the node. |
| Tx Packet Count | The number of HDLC packets sent by the module for that node. |
| Rx Packet Count | The number of HDLC packets received by the module for that node. |
| TimeOuts | The number of HDLC requests for that node that did not receive a valid reply within the timeout period. |
| CRC Errors | The number of corrupted HDLC packets received by the module for that node. |
| Sequence Errors | The number of HDLC packet sequence errors detected for that node. |
| Inactive Count | The number of times the node was flagged as inactive. |

Table 5.4 – HDLC Node Statistics

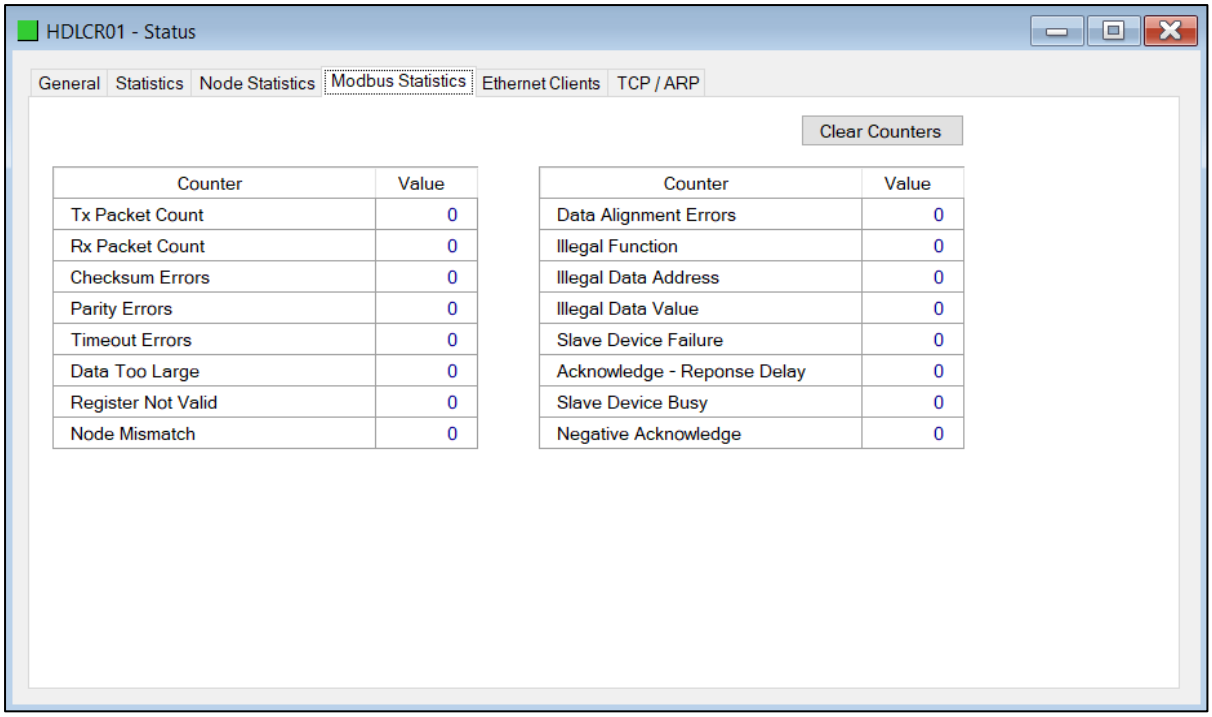



Figure 5.8. - Status monitoring – Modbus Statistics

The Modbus Statistics tab displays the following Modbus Statistics:

| Statistic | Description |
|---|---|
| Tx Packet Count | The number of Modbus packets sent by the module. |
| Rx Packet Count | The number of Modbus packets received by the module. |
| Checksum errors | The number of corrupted Modbus packets received by the module. |
| Parity errors | The number of bytes with parity errors received by the module. |
| Timeout Errors | The number of message response timeouts the module has encountered. |
| Data Too Large | The number of Modbus requests or responses where the data was too large to process. |
| Map Item Not Found | The number of Modbus requests did not match any mapped items. |
| Node Mismatch | The received Modbus request did not match the module's Modbus node address. |
| Data Alignment Errors | The Modbus request and associated mapped item is not byte aligned with the destination Logix tag. |
| Illegal Function | The number of times the Modbus device responded with an Illegal Function exception. |
| Illegal Data Address | The number of times the Modbus device responded with an Illegal Data Address exception. |
| Illegal Data Value | The number of times the Modbus device responded with an Illegal Data Value exception. |

| | |
|---|---|
| Slave Device Failure | The number of times the Modbus device responded with a Device Failure exception. |
| Acknowledge –Response Delay | The number of times the Modbus device responded with an Acknowledge exception. |
| Slave Device Busy | The number of times the Modbus device responded with a Slave Busy exception. |
| Negative Acknowledge | The number of times the Modbus device responded with a Negative Acknowledge exception. |

Table 5.5 - Parameters displayed in the Status Monitoring – General Tab

## 5.3. HDLC PACKET CAPTURE

The module provides the capability to capture the HDLC traffic for analysis. The will allow the user and the support team to resolve any possible issue on site.

To invoke the capture of the module, double-click on the ***HDLC Packet Capture*** item in the Project Explorer tree.

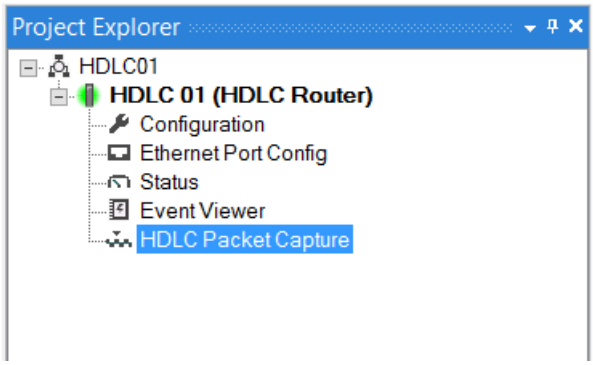



Figure 5.9. - Selecting HDLC Packet Capture

The HDLC Packet Capture window will open and automatically start capturing all HDLC packets.

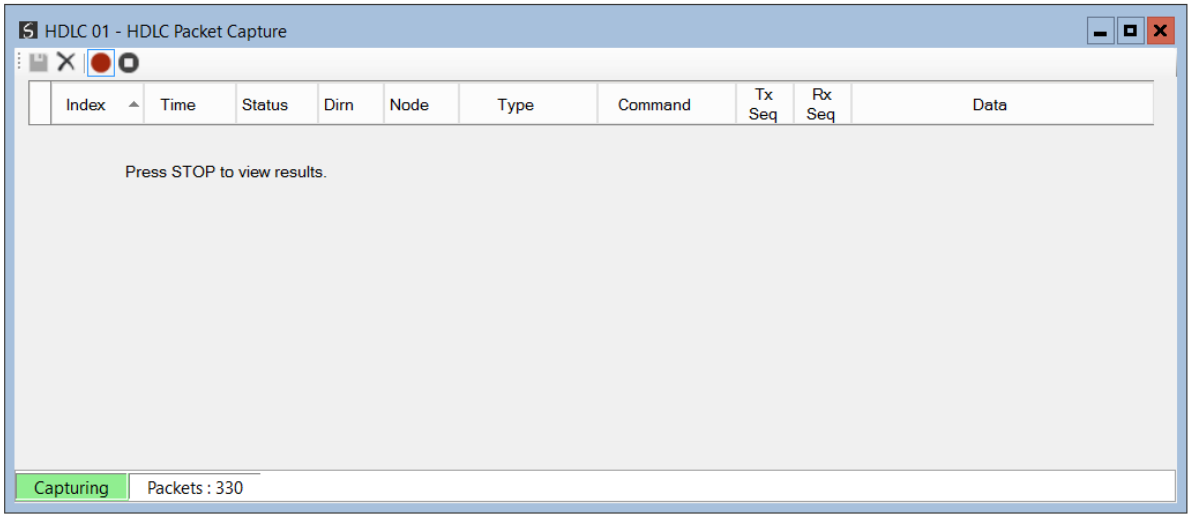



Figure 5.10. – HDLC packet capture

To display the captured HDLC packets, the capture process must first be stopped, by pressing the Stop button.

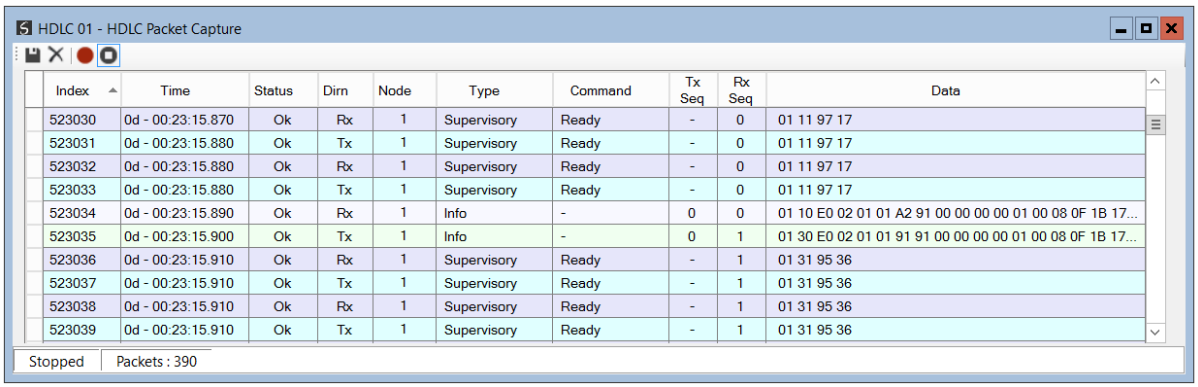



Figure 5.11. – HDLC Packet Capture complete

The captured HDLC packets are tabulated as follows:

| Statistic | Description |
|---|---|
| Index | The packet index, incremented for each packet sent or received. |
| Time | The elapsed time since the module powered up. |
| Status | The status of the packet. Received packets are checked for valid HDLC constructs and valid checksums. |
| Dirn | The direction of the packet, either transmitted (Tx) or received (Rx). |
| Node | HDLC node address of the message destination. |

| | |
|---|---|
| Type | The HDLC packet type:<br>• Info<br>• Supervisory<br>• Unnumbered |
| Command | The Supervisory Command:<br>• Ready<br>• Not Ready<br>• Reject<br>• Selective Reject |
| Tx Seq | The current transmit sequence number |
| Rx Seq | The current receive sequence number |
| Data | The raw packet data. |

Table 5.6. – HDLC Packet Capture fields

The packet capture can be saved to a file for further analysis, by selecting the Save button on the toolbar. Previously saved HDLC Packet Capture files can be viewed by selecting the HDLC Packet Capture Viewer option in the tools menu.

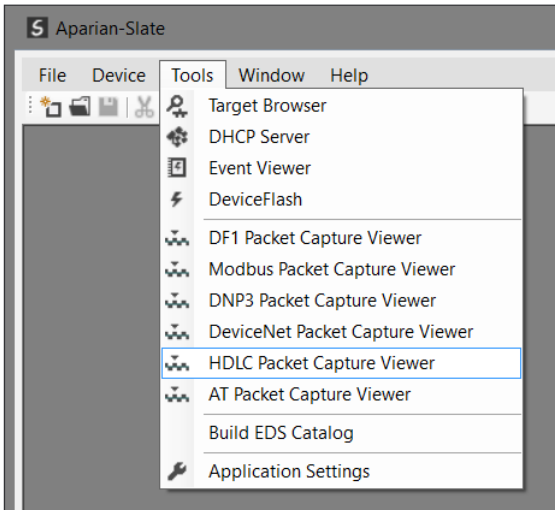



Figure 5.12. - Selecting the HDLC Packet Capture Viewer

## 5.4. MODULE EVENT LOG

The HDLC Router module logs various diagnostic records to an internal event log. These logs are stored in non-volatile memory and can be displayed using Slate or via the web interface. To vie them in Slate, select the ***Event Viewer*** option in the Project Explorer tree.

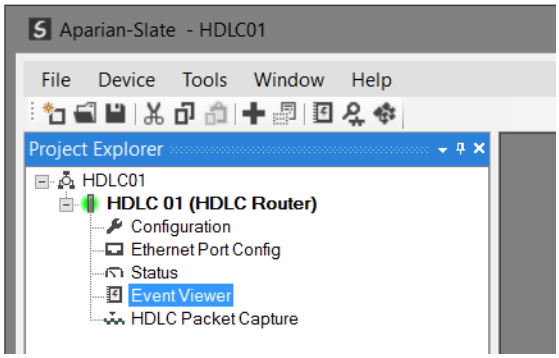
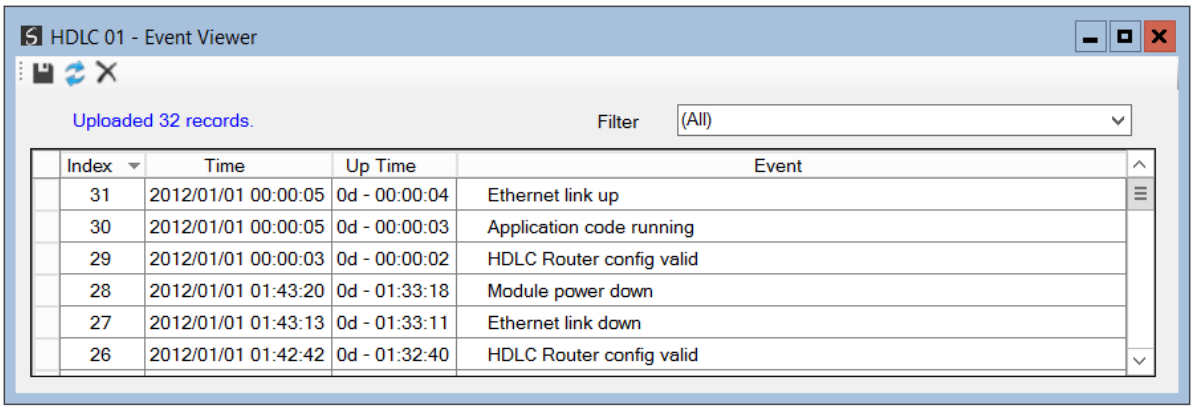



Figure 5.13. - Selecting the module Event Log

The Event Log window will open and automatically read all the events from the module. The log entries are sorted so as to have the latest record at the top. Custom sorting is achieved by double-clicking on the column headings.




| Index | Time | Up Time | Event |
|---|---|---|---|
| 31 | 2012/01/01 00:00:05 | 0d - 00:00:04 | Ethernet link up |
| 30 | 2012/01/01 00:00:05 | 0d - 00:00:03 | Application code running |
| 29 | 2012/01/01 00:00:03 | 0d - 00:00:02 | HDLC Router config valid |
| 28 | 2012/01/01 01:43:20 | 0d - 01:33:18 | Module power down |
| 27 | 2012/01/01 01:43:13 | 0d - 01:33:11 | Ethernet link down |
| 26 | 2012/01/01 01:42:42 | 0d - 01:32:40 | HDLC Router config valid |

Figure 5.14. – Module Event Log

The log can also be stored to a file for future analysis, by selecting the ***Save*** button in the tool menu.

To view previously saved files, use the ***Event Log Viewer*** option under the tools menu.

## 5.5. STATUS REPORT

A Status Report can be generated which describes the configuration and current status of the module in a rich-text formatted document. Generating this document and emailing it to Aparian support can drastically speed-up the remote support process.

To generate the Status Report, the module must first be online. Right-click on the HDLC Router and select the ***Generate Status Report*** option.

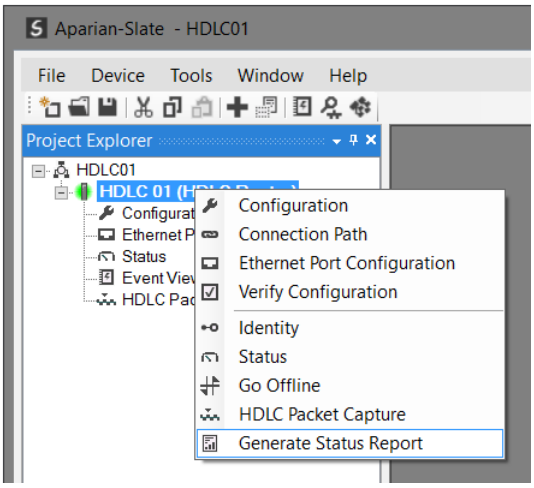



Figure 5.15 - Generate Status Report option

Then, using the file save dialog, select a suitable path for the resulting RTF file.

The Status Report (RTF) file can be opened in Microsoft Word, WordPad or any other suitable rich-text editor.

***Status Report for HDLC 01 at Thursday, 08 December 2016 10:14:04 AM***

**Device**

| | |
|---|---|
| *Device Name* | HDLC 01 |
| *Description* | |
| *Firmware Revision* | 1.001 |
| *Serial Number* | 351F-F9B5 |
| *IP Address* | 192.168.1.202 |
| *Connection Path* | 192.168.1.202 |

**Hardware**

| | |
|---|---|
| *MAC Address* | 00:60:35:1F:F9:B5 |
| *Owned* | Yes |
| *Up Time* | 0d - 18:39:56 |
| *Process Scan* | 9.0 us |
| *Process Cycles* | 110648 |
| *DIP Switches* | 1100 |
| *Temperature* | 42.9 |
| *Ethernet Cable Length* | ~ 20 m |

**HDLC Router**

| | |
|---|---|
| *Config Valid* | Yes |
| *Config Signature* | CB59 |
| *Mode* | Slave |
| *Transaction Rate* | 214 |

**General HDLC Statistics**

| | |
|---|---|
| *Last Control Word* | 113 |
| *Tx Packet Count* | 13133996 |
| *Rx Packet Count* | 13133997 |
| *Total Sequence Errors* | 0 |

Figure 5.16 - Status Report example

## 5.6. WEB SERVER

The HDLC Router provides a web server allowing a user without Slate or RSLogix 5000 to view various diagnostics of the module. This includes Ethernet parameters, system event log, advanced diagnostics, and application diagnostics (HDLC diagnostics).

**NOTE:** The web server is view **only** and thus no parameters or configuration can be altered from the web interface.

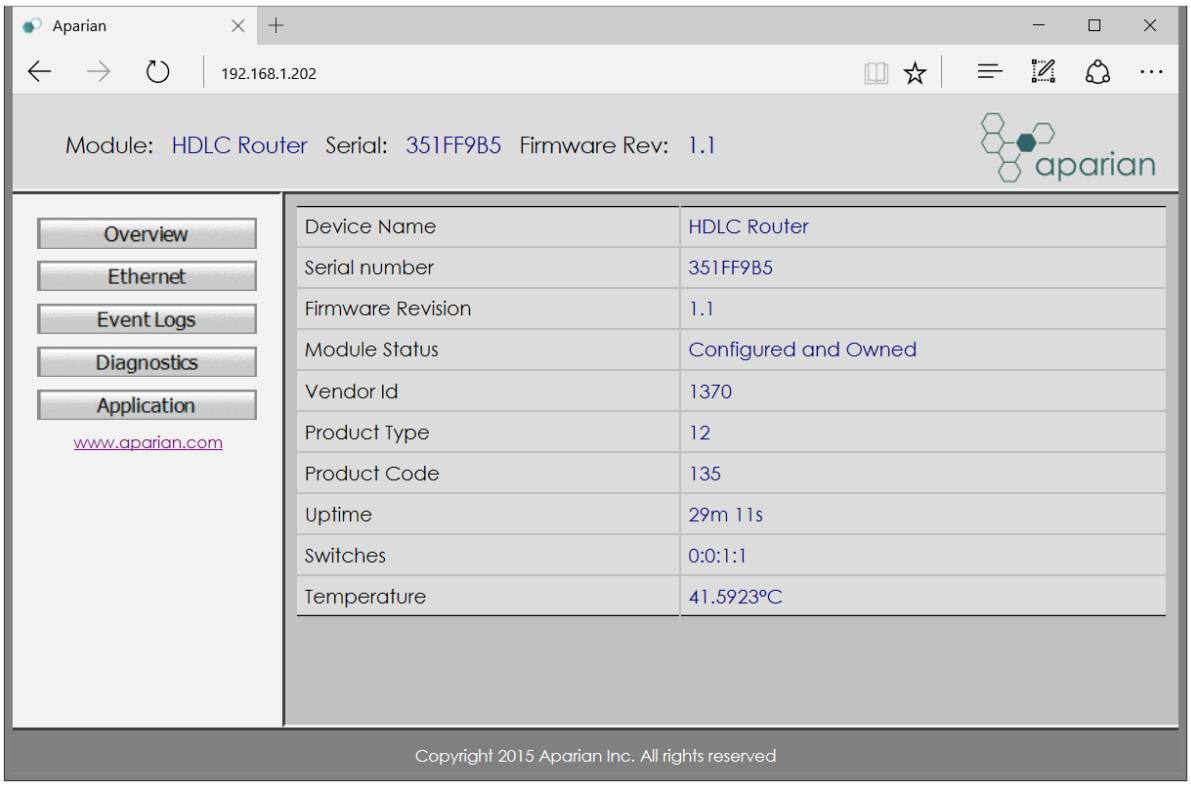



Figure 5.17. - Web interface

# 6. TECHNICAL SPECIFICATIONS

## 6.1. DIMENSIONS

Below are the enclosure dimensions as well as the required DIN rail dimensions. All dimensions are in millimetres.

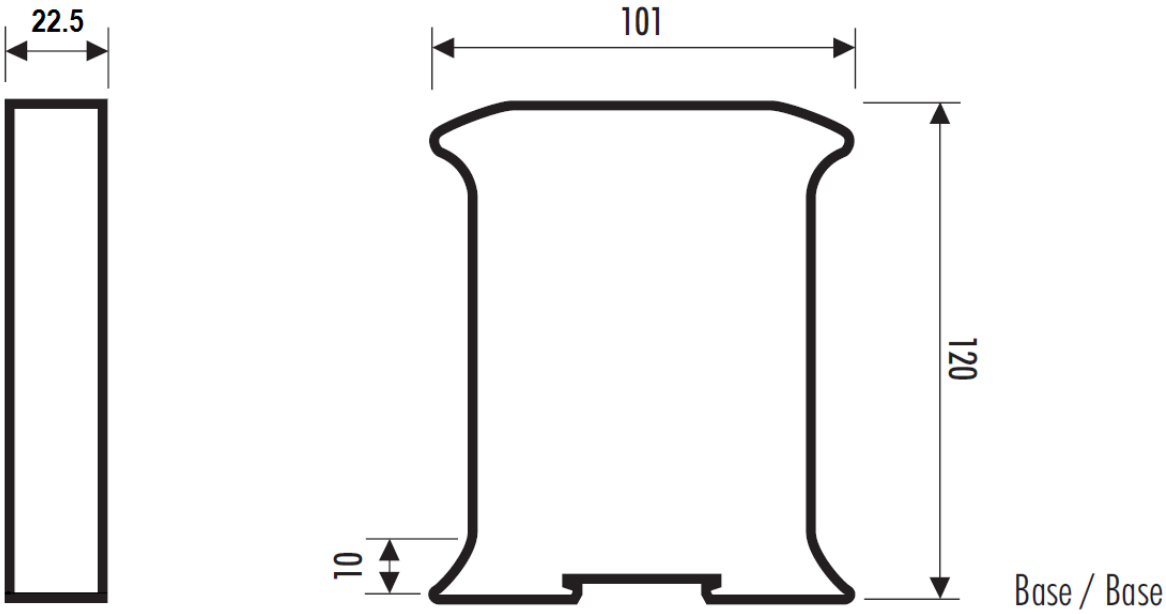



Figure 6.1 - HDLC Router enclosure dimensions

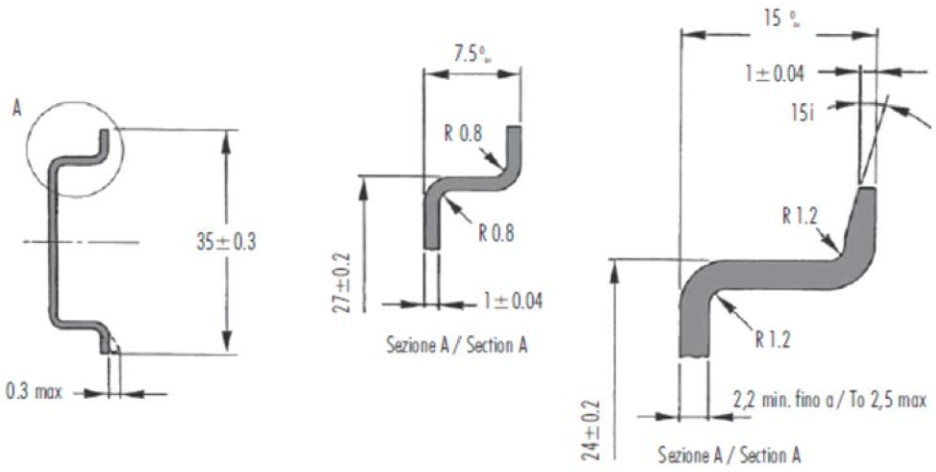



Figure 6.2 - Required DIN rail dimensions

## 6.2. ELECTRICAL

| Specification | Rating |
|---|---|
| Power requirements | Input: 10 – 28V DC, (70 mA @ 24 VDC) |
| Power consumption | 1.7 W |
| Connector | 3-way terminal |
| Conductors | 24 – 18 AWG |
| Enclosure rating | IP20, NEMA/UL Open Type |
| Temperature | -20 – 70 °C |
| Earth connection | Yes, terminal based |

Table 6.1 - Electrical specification

## 6.3. ETHERNET

| Specification | Rating |
|---|---|
| Connector | RJ45 |
| Conductors | CAT5 STP/UTP |
| ARP connections | Max 20 |
| TCP connections | Max 20 |
| CIP connections | Max 10 |
| Communication rate | 10/100Mbps |
| Duplex mode | Full/Half |
| Auto-MDIX support | Yes |
| Protocol Support | EtherNet/IP<br>Modbus TCP |

Table 6.2 - Ethernet specification

## 6.4. HDLC (RS485) PORT

| Specification | Rating |
|---|---|
| RS485 Connector | 5-way terminal |
| RS485 Conductor | 24 – 18 AWG |
| BAUD | 48000 b/s<br>180000 b/s |
| Encoding | NRZI<br>FM0<br>FM1 |
| Interface Modes | Table Exchange Mode<br>Broadcast with response<br>Broadcast with no response |

Table 6.3 - HDLC Port specifications

## 6.5. CERTIFICATIONS

| Certification | Mark |
|---|---|
| RoHS2 Compliant | RoHS2 |
| CE Mark | CE |
| ODVA Conformance | EtherNet/IP™<br>* F/W 1.004 |
| RCM | |

Table 6.4 - Certifications

# INDEX


## A

assembly instance, 30

## C

class 1, 30, 31, 35
CompactLogix, 5
Connection path, 27
Contact Us, 7
ControlLogix, 5

## D

DC power, 8
DHCP, 9, 14, 15, 16
dimensions, 53
DIN rail, 10, 53
DIP, 9

## E

Ethernet Bridge, 29
Ethernet connector, 13
Ethernet TCP, 5
EtherNet/IP, 5, 17

## F

firmware upgrade, 21

## H

HDLC, 26, 36, 37, 40, 43, 44, 45, 47, 48, 49
HDLC general configuration, 21
HDLC parameters, 21
HDLC Router, 5, 6, 7, 19, 20, 26, 30, 32, 35, 42, 49, 52

## I

input assembly, 36
input voltage, 11

## L

LED, 41
Logix controller, 31, 35, 43
Logix platform, 5

## M

MODBUS, 46

## O

output assembly, 35, 37, 38

## P

Protocol, 14

## R

requested packet interval (RPI), 31
Rockwell Automation, 18
RS232, 8, 11
RSLinx, 18
RSLogix 5000, 29, 31, 33, 37, 38, 52

## S

Safe Mode, 9
Slate, 6, 7, 14, 16, 19, 20, 21, 36, 39, 42, 49, 52
statistics, 42
Support email, 7

## T

Target Browser, 16, 17, 27

## U

UDTs, 7, 31, 32, 33, 35
User Defined Types (UDTs), 31

## W

web server, 42, 52