Controlling the SH ARC-GATEWAY by a PLC

IEC61131 sample project „GW-Client_IP_ARC_COM“, V1.0

SOHARD Embedded Systems GmbH

Project description

SOHARD‘s SH ARC-GATEWAY is a protocol converter between ARCNET and TCP/IP/Ethernet. To start the conversion the gateway has to be configured using its web interface and to be initialized by a couple of commands (packed in “OpCodes”) from the TCP/IP side.

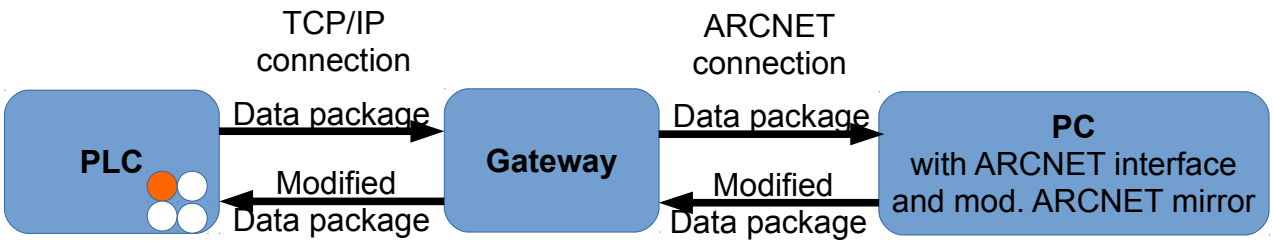
The configuration using the web interface is described in the user manual.

In the sample project “GW-Client_IP_ARC_COM” an SH ARC-GATEWAY is being initialized by a PLC via Ethernet (Profinet). By means of a PC with an ARCNET adapter and a server application, payload data is send to and fro afterwards.

The server application is a modifying ARCNET mirror (“Modifying_ArcMirror”) which modifies all bytes of a received ARCNET package in the following way and sends back (“mirrors”) the modified packages:

1 becomes 2, 2 becomes 8, 8 becomes 4, 4 becomes 1 again. All other value a byte can accept (except for 1, 2 and 8) become 1 likewise.

The reason for this is that the lower four bits of a returned byte may be connected to the digital outputs of a Phoenix PLC ILC 131. The byte modification causes a circulation of the control LEDs which grouped in a square.



Scope of delivery

- Phoenix PC WORX project “GW-Client_IP_ARC_COM”
- Source code of PLC software as PLCopen, PDF and ASCII
- “ARCNET-Mirror” and “ARCNET_ModifyingMirror” as EXE (Win7, 32bit) und C code (compilable by Gnu C Compiler, Windows library required)

Implementing a test set-up

The set-up is primarily described for ARCNET transmission via coaxial cables with BNC connectors. Of course you may use other transmission physics (RS232, optical fiber etc.). Please contact SOHARD for the materials required in this case.

- You need:
  - the software package mentioned above
  - a PLC with a Profinet/Ethernet interface (not available from SOHARD)
  - an SH ARC-GATEWAY-K from SOHARD
  - 1 coaxial cable, 2 T-connectors and 2 terminators from SOHARD
  - an ARCNET adapter (e.g. SH ARC-USB-K or PCIe-K from SOHARD) on or in a Windows PC
  - the current ARCNET RAW driver (for USB when required) from the SOHARD web page

    In case the PLC is being programmed via the Ethernet interface, you also need an Ethernet hub (not available from SOHARD)
- If you are applying a Phoenix PLC, you may load the PC WORX project and adapt the settings according to the PLC used and its IP address.
  When using another PLC, you have to
  - either import the PLCopen files or
  - to create a new project in the respecting IDE with the 3 functional blocks (FB_INIT_SEND_RECEIVE, FB_PACKAGE_USER_DATA, FB_EXTRACT_USER_DATA) and the program (myPGM_SAMPLE) by copy-and-paste from the ASCII files. The variable required are provided as pseudo code in the ASCII files.
  - myPGM_SAMPLE must be assigned either to a default or a cyclic task.
- If required the designators of the digital outputs need to be adapted in myPGM_SAMPLE.
- Connect the Profinet interface of the PLC to the Ethernet interface of SH ARC-GATEWAY using a CAT5 cable.
- Connect each end of the coaxial cable to a T-connector and join the terminators to the other side of each T-connector. You now have a terminated coaxial cable.
- Connect the BNC plugs (ARCNET interfaces) of SH ARC-GATEWAY and of the ARCNET adapter using the terminated coaxial cable.
- Install the ARCNET RAW driver to your PC and start the server application in a DOS shell (Modifying_ArcMirror <driver access name> <node ID>, e.g. „Modifying_ArcMirror farcusb 5“).

- Load the built PLC program into the CPU and start it.
- When the procedure has been successful the control LEDs of the digital outputs 0 to 3 will flash up consecutively.

Functional description

The PLC programming project “**GW-Client_IP_ARC_COM“** consists of the program myPGM_SAMPLE and the 3 functional blocks FB_INIT_SEND_RECEIVE, FB_PACKAGE_USER_DATA and FB_EXTRACT_USER_DATA. One instance (prefix my…) of each functional block is being used.

- FB_INIT_SEND_RECEIVE handles all communication with SH ARC-GATEWAY, i.e. the initialization of the gateway as well as sending and receiving payload data. It has to be called once in every cycle.
  When the gateway has been initialized, this block returns the flag init_ok zurück indicating that data may be transferred.
- FB_PACKAGE_USER_DATA packages the data to be sent as well as the IDs of the sender and destination node IDs into a send buffer in a structured way. This buffer is than transmitted by FB_INIT_SEND_RECEIVE into the ARCNET network. The sender node ID will automatically be set to the node ID of the gateway.
- FB_EXTRACT_USER_DATA extracts the payload data as well as the sender and destination node IDs out of the read buffer that has been “filled” by FB_INIT_SEND_RECEIVE. The destination node ID is the node ID of the gateway.
- myPGM_SAMPLE calls the instances of the functional blocks and increments the length of the payload data packages by 1 in each cycle. It will restart at length 1 when the maximum length of 508 bytes has been reached. The last four bits of the received bytes will be passed to the digital outputs.

The **server application** comes in the variants “ArcMirror“ and “Modifying_ArcMirror“ as C code and as Windows 32bit executible.

ArcMirror will return each ARCNET payload data package to the sender without modification. Modifying_ArcMirror modifies the data as described in the section “Project description” above.

Modifications of the project by the user

The PLC project as well as the server applications may be modified and subsequently be used. Modifying the functional blocks is discouraged unless you have a deeper knowledge of the gateway and the ways it has to be addressed. The sections commented as “Child‘s play” in the program myPGM_SAMPLE may be experimentally adapted.

Advanced users may expand the state machine in FB_INIT_SEND_RECEIVE by additional OpCodes when these are needed. OpCodes are being described in the programming manual of SH ARC-GATEWAY.

The ARCNET server applications ArcMirror.c and Modifying_ArcMirror.c may also be modified. Modifying_ArcMirror.c also contains a section commented as „Child‘s play“. ARCNET payload data is modified there before being returned. This section is a good starting point for modifications by the user.

Deeper modifications require a deeper knowledge of the API of SOHARD’s ARCNET RAW drivers.