1. PLC\_CommunicationTo2400S Specifications And Details

This paper describes the actual measurement system created and the recommended method of constructing the system. Project files are located in the following directories.

・“C:\....\PLC\_CommunicationTo2400S\ PLC\_CommunicationTo2400S.kpr”

* 1. Structure of the created system

Tab Connection Wire Color of KV-XL202



[Equipment List]

・PC

・CPU unit: Keyence KV-8000

・Communication unit: Keyence KV-XL202

・Power unit: Keyence CA-U4

・Source meter: Keithley 2400s

・RS-232c cable (Need to cutting)

・USB cable

・Ethernet cable

・3-terminals power cable

The following Figures of equipment connections (Fig 1) and system structure (Fig 2) are shown below. See Tab 1 for wire color when connecting RS-232c cable to KV-XL202. It is recommended that a continuity check be performed when changing RS-232c cables.

This measurement system consists of a PC, a PLCs CPU unit and Communication unit, and a source meter device. The PC is the user interface, the PLC is the processing unit, and the source meter is controlled by the PLC.

ダイアグラム

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Equipment connection

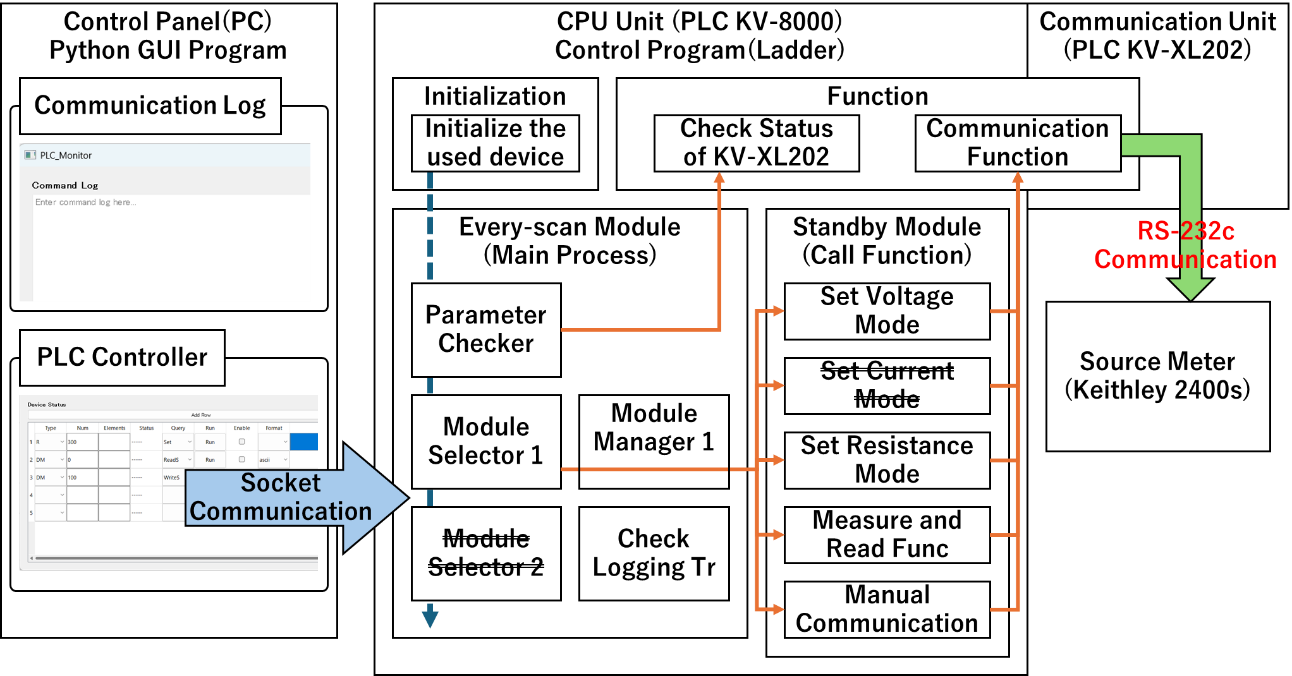


Fig System Structure

[Process Flow]

1. PLC: Initialization. Initialize process flags, Data Memory, and device values of KV-XL202.
2. PLC: Every-scan module is started. And wait for orders from the user. It also monitors device status for error handling.
3. GUI: Send relay rise command to flag the module startup. (\*Host-Link communication can be used to change device values for most PLCs. In other words, a similar system can be built by controlling the ladder program with a Python Program without transferring it to a PLC. However, that would lose the advantage of PLCs. Therefore, it is necessary to properly restrict the devices to which GUI programs have access. This system assumes access only to the start flag and DM for manual communication.)
4. PLC: When the module start flag is raised, check if the module is executable in Module Manager and call the Standby Module.
5. PLC: Each Standby Module calls a “Communication Function” to send commands to the 2400s for RS-232c communication.
6. PLC: The “Communication Function” handles the sending and receiving of RS-232c communications, and controls the KV-XL202 appropriately. (\*Controls the physical devices of the KV-XL202. Therefore, it must be single-threaded and properly processed.)
7. 2400s: Processing is performed according to the received commands.
8. PLC: Check triggers and perform logging. This system records sending command and receiving command. (\*To use the logging function, PLC settings must be made in addition to ladder program.)

PLCs have a very low logic level compared to C++, Python, etc. Therefore, careful specification is necessary. Also, the execution method is not the sequential execution method of C++, Python, etc., but the every scan method, so it is necessary to create programs that take device state transitions into consideration.

A state transition refers to a change in output values depending on various input conditions. In a ladder program, set modules continue to be scanned. And there is no concept of waiting; it is asynchronous. Therefore, process flags must be used to properly process the program.

* 1. Module execution sequence

The order in which modules are scanned can be set. As the system becomes more complex, this setting becomes more important. The setup method and the Execute sequence of modules of this system (Fig 3) are shown below.

[Setup Method]

1. Click on [Program] -> [Setup module execute sequence].
2. CPU system setting: Select on [Program setting] -> [Execute sequence of modules].
3. CPU system setting: The order of scanning modules can be set.

グラフィカル ユーザー インターフェイス

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Module execution sequence

* 1. Definition of PLC variables

Similar to C++ and Python, PLC can define global and local variables. This can be used to properly manage process flags and status. As a point of reference, global variables are defined statically, while local variables have the characteristic of being managed dynamically.

[Setup Method]

1. Click on [View] -> [Variable edit window].
2. Variable edit: Select the Global or Local tab. (Fig 4)
3. Variable edit: For Global variables, enter the Variable name, Data type, Assigned device.
4. Variable edit: For Local variables, select the program from the downlist and enter the Local Variable name, Data type, Value (initial value).

テーブル

AI によって生成されたコンテンツは間違っている可能性があります。

Fig PLCs Global Variables

* 1. PC network settings

The GUI program requires PC network configuration for network communication.

[Setup Method]

1. Launch the PC settings and open the Ethernet settings in the Network & internet section.
2. Click on the Edit button of IP Assignment to open it and select Manual.
3. Set IPv4 to On and enter the following settings. (Fig 5)

・IP address: 192.168.0.100 (Match the network settings of the KV-8000. 192.168.0.n)

・Subnet mask: 255.255.255.0 (Match the network settings of the KV-8000.)

・Gateway: 192.168.0.1 (Match the network settings of the KV-8000. 192.168.0.n)

・Preferred DNS: 192.168.0.1(Match the network settings of the KV-8000. 192.168.0.n)

・DNS over HTTPS: Off (default)

・Alternate DNS: \*Do not fill in (default)

・DNS over HTTPS: Off (default)

1. Click on Save. And close Settings.

グラフィカル ユーザー インターフェイス, アプリケーション

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Network Configuration

* 1. Host-Link Communication by Python script

To build a user interface for the PLC system, a library for Host-Link communication and library for GUI creation were imported in Python, and a control monitor was created. The KV-8000 can perform Host-Link communication as an EtherNet/IP function. This function is used to override and control device values provided in the KV-8000.

The main Python program is located in the following directory.

“C:\....\PLC\_CommunicationTo2400S\PLC\_ControlMonitor\PLC\_MonitorMain.py”

[How to use]

1. Launch a terminal and execute the following command.

“\PLC\_CommunicationTo2400S\PLC\_ControlMonitor> python .\PLC\_MonitorMain.py”

1. Fill in the command settings in the table at the bottom of the window, as shown in Fig 6.

(\*To control a Relay, enter R in the Type column, the number of the Relay to be controlled in the Num column, and Set (Rise) or Reset (Fall) in the Query column, and click Run.)

1. As shown in Tab 2, each device corresponds to a global variable in the PLC. Therefore, by clicking on Run and setting the start flag, the Standby Module can be executed.
2. To use the manual communication function, set the command settings as shown in rows 6~7 of the table in Fig 6, and set the command and command length in DM. Then, the “ManualComRS232cP1Flag” flag is set and executed.

テキスト が含まれている画像

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Control Monitor Window

Tab Module flags and control monitor settings for PLCs



[Note.]

・When using “plural form” instruction, the array length must be entered in the Elements column. For example, Sets, Resets, Writes, Reads, etc.

・The Host-Link communication functions of the Python program are described in the following file. Socket library is required.

“\PLC\_CommunicationTo2400S\PLC\_ControlMonitor> python .\PLC\_CommandClass.py”

・For information on creating “PLC\_CommandClass.py”, see the manual below.

“C:\....\PLC\_CommunicationTo2400S\Manual\English\EtherNetsIP\_Function\_KV-EP21V\_KV-8000(A)etc\_User's\_Manual.pdf”

-> Section 8 HOST-LINK COMMUNICATION FUNCTION

・GUI library used PyQt5 version 5.15.11.

* 1. FTP server function

Set up an FTP server function to access the PLC’s internal storage and the SD card inserted in the KV-8000. Using this function, log data and other data can be comfortably obtained.

[How to use]

1. As shown in Fig 7, select KV-8000 in the Unit Editor and set the “FTP server” to “Used”.
2. After setting up the FTP server, transfer the settings and program to the PLC.
3. After confirming that the PC and KV-8000 are connected via EtherNet cable, launch Explorer of PC as shown in Fig 8 and enter the following command to access the system.

“ftp://192.168.0.10”, (\*Enter the IP address of the KV-8000 in Fig 7)

1. When the logon dialog appears, enter KV for the username and logon without a password.
2. Access to PLC and SD card storage to view files.

グラフィカル ユーザー インターフェイス

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Enable FTP Server Settings

グラフィカル ユーザー インターフェイス, アプリケーション

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Logon to KV-8000

* 1. Logging function

Set up a logging function to obtain experimental data and operation records. Operation records can be used to debug the program.

[Setup Method]

1. Click on [Tool] -> [Setup logging/trace].
2. Logging/Trace setting list: Click on Set. This system was set up as follows. (Fig 9)

・Function select: Logging

・Saving location: Memory card

・File comment: SendCommandLog

・File name, File No.: Auto numbering

・File name, No. upper limit: 99

・File name, Start operation: New a file

・File name, When no empty No.: Overwrite by the oldest files

・File name, String: SendCommandLog

・File name, Date/time: Year, Month, Day

・Device/variable, Device/variable: RS232cP1CommandLog

・Device/variable, Word Pts: 255

・Device/variable, Data format: ASCII16BIT

・Trigger, Type: Bit device

・Trigger, Device: RS232cP1SendStatus

・Option, Insert file comments into the leading of CSV file: Enable

・Option, Add comments row of device: Enable

・Option, Attach time stamp to each row: Enable

・Option, Add data No. to each row: Enable

・Option, Add data obtain span to each row: Enable

・Option, Add space to secure the character count: Disable

・Option, Perform CSV file save for each trigger: Enable

・Option, Save the file to non-volatile memory: Enable

・Option, Auto-restart logging by inserting memory card: Enable

・Option, Set upper limit of file capacity: Enable

・Option, Upper limit: 1000, Row

1. The logging function for reception was also set up with reference to ②
2. In this setup, logging continues as long as the Bit device (Relay) is up, so a “Check Logging Trigger” module was add for the logging function.

グラフィカル ユーザー インターフェイス, テキスト, アプリケーション

AI によって生成されたコンテンツは間違っている可能性があります。

Fig Loggint/Trace setting

* 1. Program module type

This section describes the “Initialize module”, “Every-scan module”, “Standby module”, and “Function/Function Block” used to build this system. For specifications of each module, refer to the following manual.

“C:\....\PLC\_CommunicationTo2400S\Manual\English\Programmable\_Controller\_KV-8000\_User's\_Manual.pdf”

-> Section 4-9 Module Type

* 1. Description of each module
     1. Initialize the used device

[Process Flow]

1. Execute “Initialization of KV-XL202” Function.
   * 1. Parameter Checker

[Process Flow]

1. Execute “Check Status of KV-XL202” Function.
   * 1. Module Selector

[Process Flow]

1. Monitor process startup flags.
2. Once the process flag is raised, check to see if the module can be started.
3. If the module can be activated, activate the corresponding module.
   * 1. Module Manager

[Process Flow]

1. Monitor process startup flags.
2. If the module flag memory is not Busy, enter the corresponding Busy code.
3. When the process flag goes down, check to see if the busy code corresponds and clear the busy status.
   * 1. Check Logging Trigger

[Process Flow]

1. Monitor the corresponding relay to see if it comes up.
2. Copy the command to the DM for logging at the timing when the corresponding relay comes up, and start up the trigger for logging.
   * 1. Set Resistance Mode

[Process Flow]

1. Initialize the process flags when called by the Module selector in the Main program.
2. Invoke and execute “Communication Function” Function with the following command length and command order.

・4, “\*RST”

・16, “:SENS:FUNC “”RES”””

・18, “:SENS:RES:RANG 2E8”

・23, “:SENS:RES:RANG:AUTO OFF”

・14, “:SYST:RSEN OFF”

1. Reset process flags.
2. Start flag is lowered.
   * 1. Set Voltage Mode

[Process Flow]

1. Initialize the process flags when called by the Module selector in the Main program.
2. Invoke and execute “Communication Function” Function with the following command length and command order.

・4, “\*RST”

・38, “:SOUR:FUNC CURR; :SOUR:CURR:MODE FIXED” (\*Multiple commands can be sent like this.)

・58, “:SENS:FUNC “”VOLT””; :SENS:VOLT:PROT 30; :SENS:VOLT:RANG 200”

・37, “:SOUR:CURR:RANG MIN; :SOUR:CURR:LEV 0”

1. Reset process flags.
2. Start flag is lowered.

\*This program sends commands together, but it is not necessary to send them all at once. Rather, it is easier to debug if the commands are sent one at a time.

* + 1. Measure and Read Func

[Process Flow]

1. Initialize the process flags when called by the Module selector in the Main program.
2. Execute “Communication Function” Function with a command length and command (8, “:OUTP ON”).
3. Set the “ResRequest” flag and execute “Communication Function” Function with a command length and command (8, “:OUTP ON”).
4. When the receive flag is raised, the received command is copy to the DM.
5. If the “MeasReadP1Flag” flag is ON, processes ③ and ④ are repeated.
6. Execute “Communication Function” Function with a command length and command (9, “:OUTP OFF”).
7. Reset process flags.
8. Start flag is lowered.
   * 1. Manual Communication

This module uses the Python scripts from Section 2.5. See there for command length and how to set up and run commands.

[Process Flow]

1. Initialize the process flags when called by the Module selector in the Main program.
2. Execute “Communication Function” Function using any command length and command set in the global variables.
3. Reset process flags.
4. Start flag is lowered.
   * 1. Initialization of KV-XL202

[Process Flow]

1. When this function is called, it initializes Relay, DM and process flags used in the communication process.
   * 1. Check Status of KV-XL202

This ladder program is used to monitor the KV-XL202 communication processing device in KV-Studio. It has absolutely no impact on the system with respect to basic processing.

* + 1. Communication Function

This program is the core function in communication. In order to control physical devices, it is recommended that modules be created in “Function” rather than “Function Block”. The process flow of this system program is described below.

[Process Flow]

1. When this function is called, make sure that the busy flag is not raised.
2. Set the busy flag and initialize the process flags.
3. Initialize Relay and DM occupied by the physical device of KV-XL202.
4. Scan the KV-XL202 read-only Relay to confirm that initialization is complete.
5. Set “R34200” to activate the communication device.
6. Check “R34300” to confirm that the communication device is up.
7. Once the device is up and running, set the DM command length and command as follows, start “R34201” and send the command.

・Command Length: DM10310

・Command: DM10310~DM10566 (512 Byte)

1. When the transmission completion relay “R34301” rises, the transmission relay “R34201” is turned down.
2. When a receive request is made, the receive flag “R34302” rises, the command length and command are copied from the DM as follows, and the receive completion relay “R34202” rises.

・Command Length: DM10567

・Command: DM10568~DM10823

1. Reset process flags. And Reset busy flags. Also If “R34302” is low, “R34202” is lowered.
   1. References

[Project Repository]

・https://github.com/shimasou/PLC\_CommunicationTo2400S

[PLC Manuals]

・