APPLICATION NOTE

YS170 Peer-to-Peer Communications

OVERVIEW

Peer-to-peer communications can be defined as the ability to digitally share data between two or more like devices, without the need of a master/slave relationship. Peer-to-peer allows selected data points to be shared continuously without the need to request and read data. YS170 (Style 3 or higher) has an option for peer-to-peer communications, model suffix /A33. YS-Net is the name for the communications network. Any address in a YS170 program can be transmitted via YS-Net to other YS170's. YS-Net peer-to-peer is pre-configured, allowing four (4) YS170 controllers to transmit up to four (4) analog values and sixteen (16) status points each to up to sixteen (16) controllers resident on the network. Refer to Figure 1.

YS-NET (PEER-TO-PEER)

This new network incorporates state-of-the-art communications using Echelon® *LONWorks*. In the pre-configured peer-to-peer mode, YS-Net operates at 78.125 Kbps over a single twisted pair of wires (nominal AWG 22) at a cable length up to 4000 feet. The Neuron® chip, the *brains* of *LONWorks*, can be configured to operate in many desired manners using a software development kit. The pre-configured peer-to-peer format described above is the standard offering, but alternative configurations may be designed for specialized applications.

WHY LONWorks?

LONWorks has been selected by more than 1000 companies to meet the communications needs in their specific equipment. The Neuron® chip is manufactured by both Motorola and Toshiba, guaranteeing availability. A PROM used with the Neuron® chip can be modified for unique requirements. And the chip is inexpensive to purchase and implement! Future considerations may be digital communications to external devices, e.g., relay or signal conditioning modules.

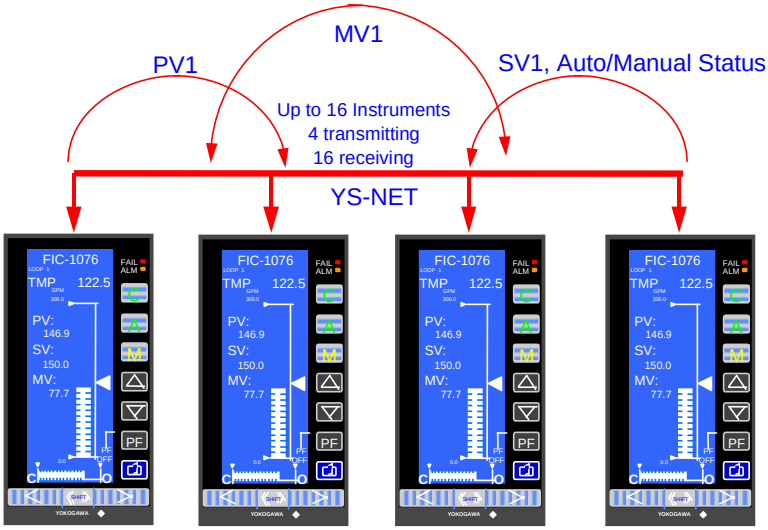

Figure 1

WHY USE PEER-TO-PEER?

Peer-to-peer communications allows analog and status values in up to four YS170's to be transmitted to others. This *cross-talk* allows more sophisticated control and interaction within a "system" of YS170 controllers. Analog or discrete inputs/outputs can be distributed, extending I/O capabilities. Installation and wiring costs are lowered.

IMPLEMENTING YS-NET

Style 3 and higher YS170 Dual Loop Programmable Controllers are offered with YS-Net peer-to-peer communications as an option (/A33). The actual communications instructions are written into the YS170 custom program. New addresses are available:
CY01-16: analog outputs
CX01-04: analog inputs
CO01-16: status inputs
CI01-64: status outputs
CF01-04: communications failure
No other configuration is required. Upon commissioning, the YS170 controller identifies the presence of a YS-Net card (/A33) and the communications configuration is automatically installed.

YS-NET ADDRESSES

Both analog and status (digital) data can be transmitted via peer-to-peer. YS170 controllers with addresses 1 to 4 serve as transmitters and receivers. Addresses 5 to 16 serve as receiving devices only. The communications address number must be set on the CONFIG 1

display of the YS170 controller. Each controller *must* have a different YS-Net address, 1-16. Refer to Figure 2.

COMMUNICATION ADDRESSES

Analog data from YS170 address 1 can be read by another device using communications input registers *CX01-04* in the receiving controller program. *CX05-08* are parameters sent from controller 2, *CX09-12* for 3, and *CX13-16* for 4. Remember, only the first four controllers on the YS-Net highway can transmit data, YS170 addresses 1-4.

EXAMPLE:

LD CX01 Analog data from CY01, controller #1
ST MV1 Data stored to receiving YS170 control output 1
LD CX06 Analog data from CY02, controller #2
ST SV1 Stored to set point 1

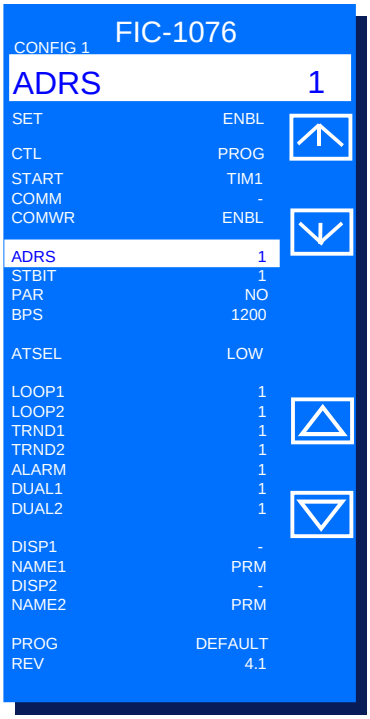

Figure 2

The transmitting YS170 analog data is written to output registers *CY01-04*. The registers are the same for all four sending devices. The receiving controllers can differentiate the data from each controller using *CX01-16*, as described above.

EXAMPLE:

LD SV2 Set point value
ST CY01 YS-Net send data
LD MV1 Control Output
ST CY02 YS-Net send data

Status inputs are identified as *CI01-CI64*. The same logic applies as described with analog values, i.e., *CI01-16* are status inputs from controller 1, *CI17-32* from controller 2, etc. Inversely, *CO01-16* are status outputs from each of the transmitting controller, addresses 1-4.

EXAMPLE:

LD CI01 YS-Net status input controller 1
ST CAMF1 Auto/manual status receiving YS170
LD CAMF1 Auto/manual status
ST CO01 YS-Net status output

Communications failure flags (*CF01-04*) are available. Each flag number designates the transmitting device address where communications has been interrupted. The flag is activated after 2 seconds and can be used in the receiving controller program to initiate a shutdown sequence or other fail-safe condition. The last transmitted data is retained or a *dummy* value can be installed to allow continued operation.

EXAMPLE:

LD CF01 Comm flag #1
ST DO1 Discrete Output 1
GIF @SHUTDOWN
Begin shutdown logic

Refer to Table 1 for quick reference.

CX01-04	Analog inputs from #1
CX05-08	Analog input from #2
CX09-12	Analog inputs from #3
CX12-16	Analog inputs from #4
CY01-04	Analog outputs from #1-#4
CI01-16	Status inputs from #1
CI17-32	Status inputs from #2
CI33-48	Status inputs from #3
CI49-64	Status inputs from #4
CO01-16	Status outputs from #1- #4
CF01-04	Comm. FAIL from #1- #4

Table 1

YS170 PROGRAMMING SOFTWARE

The YS170 programming software package, Model YSS10-210, used to write custom control programs, is required. The version level must be 3.03 or higher. This allows the communications registers *(CX, CY, CI, CO & CF)* to be included in the program. Earlier software versions do not allow use of these registers.

SUMMARY

YS-Net peer-to-peer communications provides the ability of interactive control and input/output expansion. It is easy to implement with no special configuration requirements. The YSS10-210 Version 3.03 or higher Programming Software Package is used to include the communications registers in the YS170 controller program. Peer-to-peer configuration is automatically installed. Simple to use and implement: YS-Net peer-to-peer communications.