

## Via 10 Protocol

The Via 10 communicates using a 10-BaseT connection with UDP type protocol. Since UDP is used, the programmer should take steps to ensure that the Via 10 actually received the transmitted data. This is accomplished by using the Callback data command or using the Get data command. AMX supports UDP control (See Rane AMX software *Rane VIA10 Netlinx Demo, Rev 1.AXS* at <http://www.rane.com/amxprogs.zip>). Crestron may support UDP in the future.

The Via 10 supports three type of commands:

1. Put data (Only writes to the transmit buffer)
2. Get data (Only writes to the receive buffer)
3. Callback (Writes data to transmit buffer and responds when data has arrived or there has been a timeout. Only the RS-232 and RS-485 ports are supported.)

There are two Ports on the Via 10:

1. Host Port (Relays and ADC)
2. RS-232/RS-485 Port

**Host Port** (Address: \$1024 + 111 or 4405)

This port is used to control the eight relay outputs and eight ADCs inputs. The relays are open collector logic drivers. The ADC are scaled 0 to 5 volts, with 5 volts = 255.

Packet: (All data except Values are word size)

ID                      Any word length ID you wish to assign

Command              1 = Send data to port  
                            2 = Get data from port

Destination          1 = Relay  
                            2 = ADC

Start                   Always set to 0

Length                  Always set to 8

Values                  Read or Write values for each Relay or ADC listed by Length

Examples are shown using AMX:

Turn on Relay 1

Via10Relays[1] = 1

// ID = 00,00 - Cmd = 00,01 - Dist = 00,01 - Start = 00,00 - Len = 00,08 - Data = Via10Relays  
SEND\_STRING VIA10\_HOST, "00,00,00,01,00,01,00,00,00,08,Via10Relays"

Turn off all Relays

Via10Relays = "00,00,00,00,00,00,00,00"

// ID = 00,00 - Cmd = 00,01 - Dist = 00,01 - Start = 00,00 - Len = 00,08 - Data = Eight 0s  
SEND\_STRING VIA10\_RELAYS, "00,00,00,01,00,01,00,00,00,08,Via10Relays"

Get Relays

// ID = 00,00 - Cmd = 00,02 - Dist = 00,01 - Start = 00,00 - Len = 00,08  
SEND\_STRING VIA10\_HOST, "00,00,00,02,00,01,00,00,00,08"

Returned Data

xx = Returned Data

Via10RxBuffer = "00,00,00,02,00,01,00,00,00,08,xx,xx,xx,xx,xx,xx,xx,xx"

## Via 10 Protocol

Get ADC Levels

// ID = 00,00 - Cmd = 00,02 - Dist = 00,02 - Start = 00,00 - Len = 00,08

SEND\_STRING VIA10\_RELAYS, "00,00,00,02,00,02,00,00,00,08"

Returned Data

xx = Returned Data

Via10RxBuffer = "00,00,00,02,00,02,00,00,00,08,xx,xx,xx,xx,xx,xx,xx,xx"

IF (DATA.TEXT[6] = 1 && DATA.TEXT[10]) // Test for Via10 Packet for Relays

```
{
  Via10Relays = "MID_STRING(DATA.TEXT,11,DATA.TEXT[10])"
  [TP,RELAY1_BTN] = Via10Relays[1]
  [TP,RELAY2_BTN] = Via10Relays[2]
  [TP,RELAY3_BTN] = Via10Relays[3]
  [TP,RELAY4_BTN] = Via10Relays[4]
  [TP,RELAY5_BTN] = Via10Relays[5]
  [TP,RELAY6_BTN] = Via10Relays[6]
  [TP,RELAY7_BTN] = Via10Relays[7]
  [TP,RELAY8_BTN] = Via10Relays[8]
}
```

IF (DATA.TEXT[6] = 2 && DATA.TEXT[10]) // Test for Via10 Packet for ADCs

```
{
  Via10ADClevels = "MID_STRING(DATA.TEXT,11,DATA.TEXT[10])"
  SEND_LEVEL TP,VIA10_ADC1_BAR,Via10ADClevels[1]
}
```

**RS-232/RS-485 Port** (RS-232 Address: \$1024 + 112 or 4406)

(RS-485 Address: \$1024 + 113 or 4407)

This port is used to transmit and receive data at the RS-232 or RS-485 ports. Before writing to these ports you must configure their protocol.

Packet: (All data except Values are word size)

ID                      Any word length ID you wish to assign

Command              0    = Get Status (Returns number of RX chars in Len field)  
                         1    = Send data to port  
                         2    = Get data from port  
                         3    = Callback ( Write data, Responds when data has arrived or time out)  
                         \$10 = Protocol Config  
                         \$FF = Reset Port (Clears all buffers)

Flags                      Set to 0 except for Command = 3 or \$10  
                         Command = 3 (Or'd values: \$01= Flush RX Buffer, \$02= Flush TX Buffer)  
                         Command = \$10 (Set to 2) Set Baud and Parity

Length/  
Timeout                      Command = 0 (Chars in RX buffer)  
                         Command = 1 (Chars to Write)  
                         Command = 2 (Max Chars to return)  
                         Command = 3 (Set Timeout)  
                         Command = \$10 (Set to 2)  
                         Command = \$FF (Set to 0)

MaxChars                Only used in Callback (Chars to receive before Callback)

TxLen                      Length of Chars in Values

## Via 10 Protocol

Values            Chars to Read or Write  
Command = \$10 (Baud, Parity)

### Baud Values:

0 = 300  
1 = 1200  
2 = 2400  
3 = 4800  
4 = 9600  
5 = 19200  
6 = 38400

### Parity Values:

0 = Odd  
1 = Even  
2 = None

### Examples:

Configure port for 19200, Even Parity

// ID = 00,00 - Cmd = 00,\$10 - Flags = 00,02 - txLen = 00,02 - Data = 05,01

SEND\_STRING VIA10\_RS232, "00,00,00,\$10,00,02,00,02,05,01"

Send data to port with no reply

SEND\_STRING VIA10\_RS232, "00,00,00,01,00,00,00,04,\$FB,01,\$FB,01"

Send data to port and request reply (Callback)

// ID = 00,00 - Cmd = 00,03 - Flags = 00,00 - Timeouts = 00,64, MaxChars = 00,xx - txLen = 00,xx - Data

SEND\_STRING Vport,

"00,00,00,03,00,00,00,64,00,Rw232LenRx+1,00,LENGTH\_STRING (RW232txBuffer),RW232txBuffer"

The Via 10 replies with a Data packet when MaxChars has been received or a Timeout has occurred. Timeout is the amount of time in milliseconds that the Via 10 will wait between receiving characters at the RS-232 or RS-485 ports before replying.

### Data Packet

ID                Same as what was sent

Command        Same as what was sent

Flags            Same as what was sent

Len              Length of the Array of Chars in Values or 0 is Timeout occurred

Values          Array of Chars