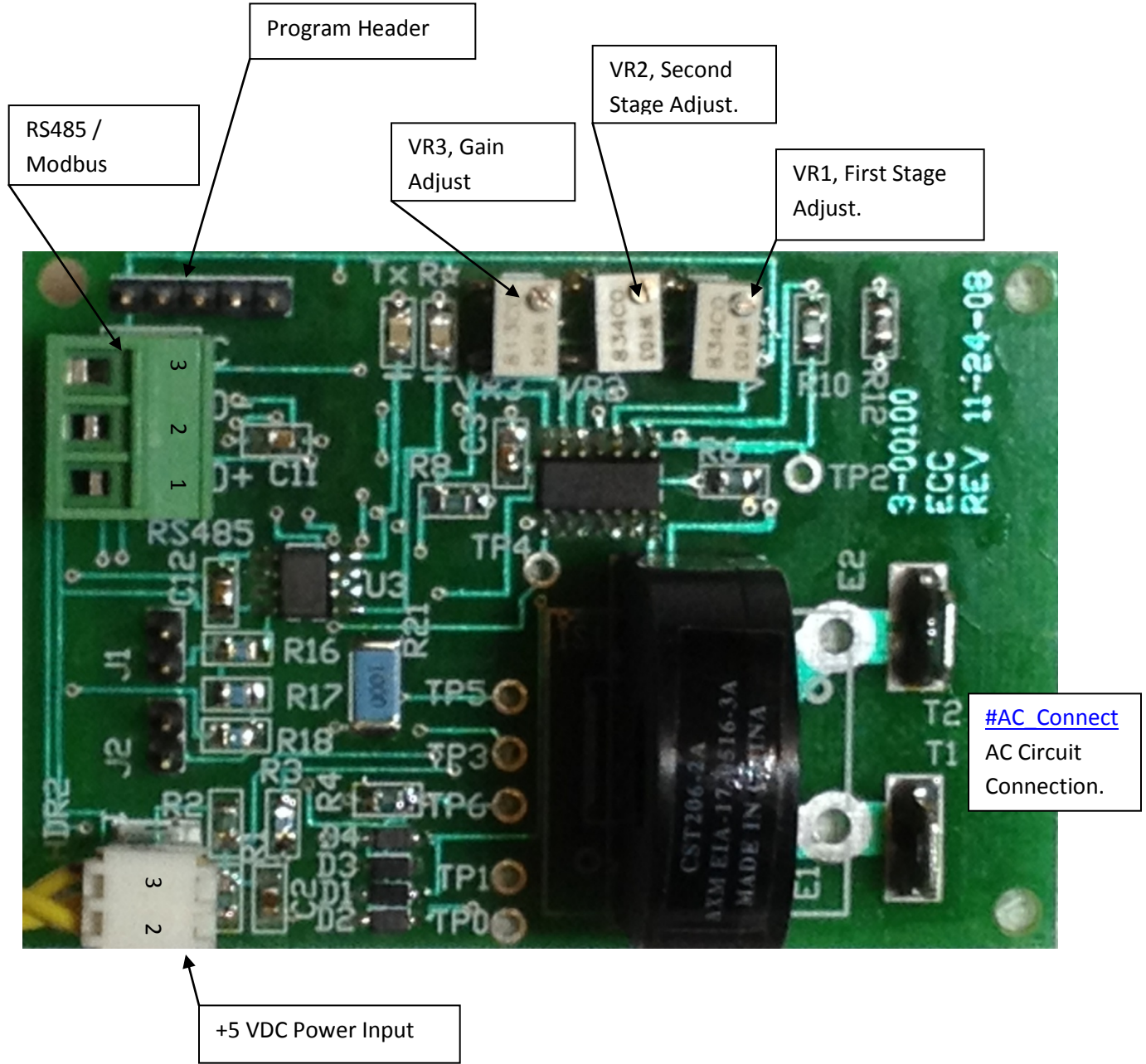


# Current Sensor with RS485/Modbus Communications

January 23, 2014



Model: AMPWTX-MOD.

## Specifications:

- Transformer Primary Current: 10 Amperes maximum.
- Transformer Primary Resistance: determined wire length; typically less than 1 ohm.
- Transformer Typical Output: 110 mV/AMP.
- Transformer Frequency: 10 KHZ to 200 KHZ. \*
- See TRIAD P/N CST206-2A for additional transformer specifications.
- [Input AC voltage](#): 0 to 480 VAC.
- Amplifier: Two stage Non-inverting; gain of 11 to 121 for milliamp to high current measurements.
- Potentiometers for each stage to set operation at one half of the supply voltage.
- Gain Adjustment: A 10 turn potentiometer for second stage – [schematic](#) below.
- Input Power: 5 VDC @ less than 100 MA.
- Output Range: 0 – 4.7 VDC.
- The AMPWTX-MOD can be panel mounted, three inch Snap Track mounted, or optional feet can be added to the Snap Track for DIN Rail mounting. The circuit board measures 2.0 x 2.95 inches.
- AC Input Connection: 0.250 Faston Tabs.
- Power/Output Connection: MTA100 connector.
- The microcontroller reads the voltage level VIA a 10 bit ADC and transmits the reading when polled.
- Protocol can be switched from Modbus to Simple ASCII Protocol.
- Operating temperature -40 to +85°C.

\*Good performance at 50/60 HZ.

The transformer output voltage is determined by number of turns of wire through transformer, and the current flow in wire. The AMPWTX-MOD is supplied with one turn of 18 AWG wire through the transformer. For very low milliamp applications, a user can replace the single turn with multiple turns as needed for the application. For current in excess of 10 amperes, the existing wire can be replaced with a suitable AWG passed straight through the transformer. This can handle up to 80 amperes.

## Application:

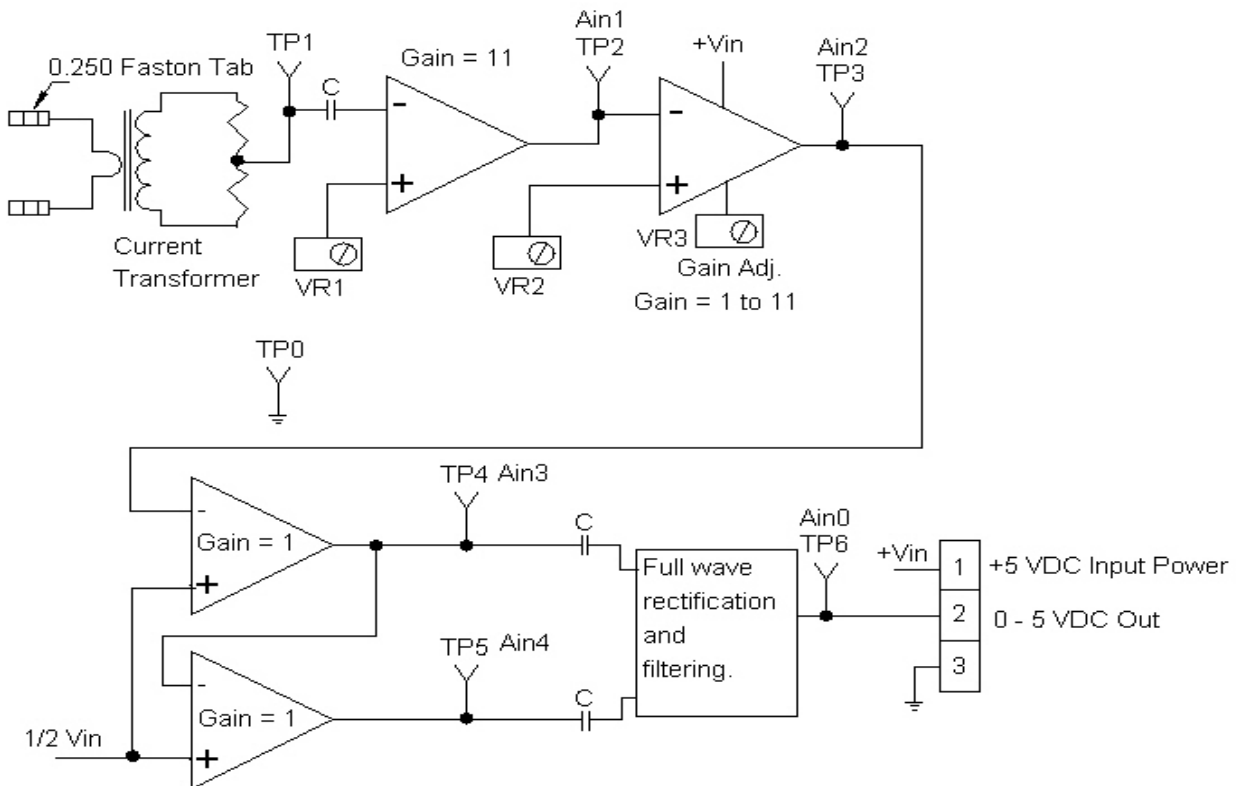
- Any circuit where it is desirable to monitor its operation VIA current flow – Lamps, motors, heaters, etc.
- Especially useful when the circuit to be monitored is not near the master controller. RS485 [twisted pair bus](#) can operate up to 4000 ft.
- Especially useful when no other analog to digital modules are available.
- Use when a PLC can't take the heat.

### Initial Circuit Calibration:

- Refer to the circuit diagram below. No AC current should be present for this procedure.
- Connect test points TP0 and TP1 together.
- Adjust potentiometer VR1 for one half of the supply voltage (2.5 VDC) measured at TP2.
- Adjust potentiometer VR2 for one half of the supply voltage (2.5 VDC) measured at TP3.
- The voltage at TP4 and TP5 will be the same as TP3. TP6 will be zero.

This calibration can also be performed by running a setup program, and reading the set points in terms of ADC counts. 2.5 volts will be 1024/2 counts. See [Modbus data format](#) below.

The circuit is now ready for the AC calibration described below.

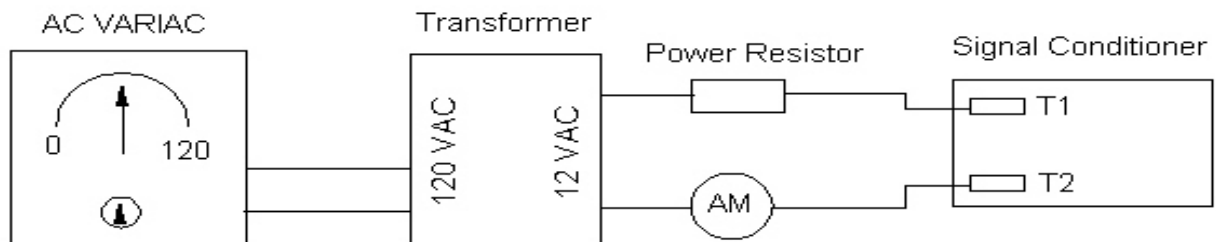


**AC Current Sensor with Modbus Communications**

## Calibration of AC Signal Conditioner:

**Caution: This should only be done by a person qualified to work with electrical circuits.**

An easy and safe method of calibrating a signal conditioner with a current transformer is to use a low voltage transformer with sufficient output current to handle the maximum the application requires. The primary of the transformer can be controlled with a variac, or the maximum current to be measured can be controlled with a power resistor. With the maximum current limited to the application maximum, connect the controlled current source to Faston tabs T1 and T2. Adjust VR3 to yield a 4.7 VDC output at TP6.



Select power resistor to limit current to a level that does not exceed ratings of devices.  
Adjust VARIAC for maximum needed current as indicated by ammeter AM.  
Using a DC voltmeter, check voltage at test point TP6.  
Adjust VR3 for a 4.7 VDC at TP6.

This procedure provides for current monitoring over a full range from 0 AMPS to full calibrated value.

If the application only requires knowledge of the presence or absence of current flow, connect the signal conditioner into a known good working circuit, and adjust for a 4.7 VDC output at TP6.

The ADC count for 4.7 volts is  $1024/5 \times 4.7 = 962$  out of 1024.

### Typical application:

A 0 – 15 PSI pressure transducer used to determine water level in a tank outputs 0 to 5 VDC. To monitor 0 to 2 PSI (4.62 ft.) and get a more accurate reading, use a signal conditioner to yield 0 to 4.7 volts over the 2 PSI range. The volts per PSI without a signal conditioner is  $5/15 = 0.3333$ . The 10 bit ADC count per PSI is  $1024/5 \times 0.3333 = 68.26$ . Using signal conditioner, volts per PSI =  $4.7/2 = 2.35$ . The 10 bit ADC count per PSI =  $1024/5 \times 2.35 = 481$ .

In terms of inches of water, without a signal conditioner the resolution is as follows:

$15 \text{ PSI} \times 2.31 \text{ ft/PSI} \times 12 \text{ inches/ft} = 415.8 \text{ inches}$

ADC count/inch = 1024/415.8 = 2.46 counts/inch.

With a signal conditioner the resolution is:

2 PSI \* 2.31 ft/PSI \* 12 inches/ft = 55.44 inches

ADC count/inch = 962/55.44 = 17.35 counts/inch.

### Modbus Communications:

The AMPWTX-MOD uses default values at power up/reset if not previously programmed.

To restore defaults to a programmed device, jumper pins 2 and 5 on program header and cycle power, or use the reset command.

### Defaults:

- Unit address: 0x01.
- MODBUS Protocol: RTU
- Baud Rate: 19.2K, 8, N, 2 - Compatible with Automation Direct PLCs and VFDs.

The AMPWTX-MOD was designed to operate as a slave device on a RS485, Modbus network. The master can be any controller such as a PC or PLC that communicates using a RS485 network and the Modbus protocol. It can be slaved to ECC's DCB2620 operating as a master network controller.

The AMPWTX-MOD has a PIC18F1320 microcontroller which can be re-programmed in circuit. Code developers can refer to the DCB2620 manual for instructions on programming PIC18Fxxxx units.

### MODBUS Read and Write Commands:

AMPWTX-MOD uses MODBUS read command (0x03) and write one word command (0x06). If a RS485 master such as the Automation Direct DL06 sends command 0x10 i.e. write multiple words command, the AMPWTX-MOD will process it as command 0x06 and write the first data word to memory.

### Command Table:

Hex Write Addr	( MODBUS)	CMD	Function
0x280	V1200	0x03	Master writes to slave – Slave returns ADC counts for TP2 – TP6.
0x281	V1201	0x06	Writes LS byte of data word to EEPROM unit address. *1, *2
0x282	V1202	0x06	Writes LS byte of data word to EEPROM baud rate address. *2
0x283	V1203	0x06	Writes LS byte of data word to EEPROM protocol address. *2
0x288	V1210	0x06	Writes LS byte of data word to EEPROM ADC hold until read configuration address.
0x299	V1231	0x06	Writes LS byte of data word to RESET unit. *3

**NOTE:** Configuration commands 0x281 – 0x283 do not take effect until the unit is reset. The changes are stored in non-volatile memory and become the operating parameters after a power cycle, or after a reset command.

## NOTES:

\*1 The default address is 0x01. If multiple units are on the same RS485 network, the addresses must be changed for all but one of the units. This is normally done using a free Configuration program that can be downloaded from ECC's Website.

The address can also be changed using an ASCII terminal program as follows:

- RESET AMPWTX-MOD
- Send keyboard <ESC> character (decimal 27, or hex 0x1B).
- Address MS character <MS> + <LS> +<E> Excluding 27 and 69, allowable addresses are 01 – 99 DEC. 27 (ESC) and 69 ('E') are reserved for the Simple Ascii Protocol.

Example: Enter address 16 (0x10).

RESET unit.

Enter keys: <ESC> <1> <6> <E>

No spaces between keys.

To enter addresses 1 – 9: <ESC> <0> <n> <E>

**Note:** This is the only place where decimal data are entered. RTU values are HEX.

If the address is set this way, the unit does a self reset. The programming device can be removed, and the network RS485 terminal block plugged in. The unit will start responding to its new address.

\*2. New values take effect after RESET (command or power cycle).

\*3. This command can be used to reset a unit instead of a power cycle.

## Broadcast Data:

If it is desirable to change a parameter in all networked devices, the broadcast slave address can be used. The slave address of zero (0x00) causes all units on the network to receive the command. For example, a broadcast command to change the baud rate, followed by a broadcast reset command would cause all units to switch baud rates at the same time.

## MODBUS write (one word) command 6, 0x06:

**Send:** |slave addr.|CMD 6|write Hi addr.|write Low addr.|data Hi|data low|crc Low|**crc Hi**|

Example: Address slave 5 and write 0x03 to 0x281 i.e. change slave address to 0x03 .

|**0x05**|0x06|0x02|0x81|0x00|0x03|crc Low|**crc Hi**|

## Response:

The response is the same as the data sent if the slave received valid data. If invalid data was received, the slave will not respond – master must time out and continue.

**MODBUS read command 3, 0x03:**

**Send:** |slave addr.|CMD 3|starting Hi addr.|starting Low addr.|#words Hi|#words low|crc Low|crc Hi|

**Response:**

|slave addr.|CMD 3|#bytes|data Hi|data Low|-----| data n Hi|data n Low| crc Low|crc Hi|

**Status Returned from AMPWTX-MOD Response to Command 0x03 Read 0x280:**

MODBUS RTU Serial Data Format:

|Unit Addr|CMD 0x03|#data bytes|data|CRC LO|CRC HI|

#data bytes = 8.

Data: |TP2 Hi|TP2 Low|TP3 Hi|TP3 Low|TP4 Hi|TP4 Low|TP5 Hi|TP5 Low|TP6 Hi|TP6 Low|

High data = ob000000nn where nn is bits 9 and 8 of the 10 bit ADC count.

Low data is the least significant byte of the ADC count.

**Write one word Command 0x06 – format above.**

**Unit Address 0x281:**

Write data word:

High byte = 0x00

Low byte = 0x01 – 0x63 set slave address (1 thru 99 decimal) excluding 27 and 69.

**Unit Baud Rate 0x282:**

Write data word:

High byte = 0x00

Low byte = 0x51 for 19.2k

Low byte = 0x52 for 9600

**Unit Protocol 0x283:**

Write data word:

High byte = 0x00

Low Byte	Meaning
0x51	MODBUS RTU, 8 data bits, no parity, 2 stop bits.
0x52	MODBUS RTU, 8 data bits, even parity, 1 stop bit.
0x53	MODBUS RTU, 8 data bits, odd parity, 1 stop bit.
0x54	Future

## Unit Analog Channel Configured to Hold Highest Reading Until it is Read by Master 0x288:

Write data word:

High byte = 0x00

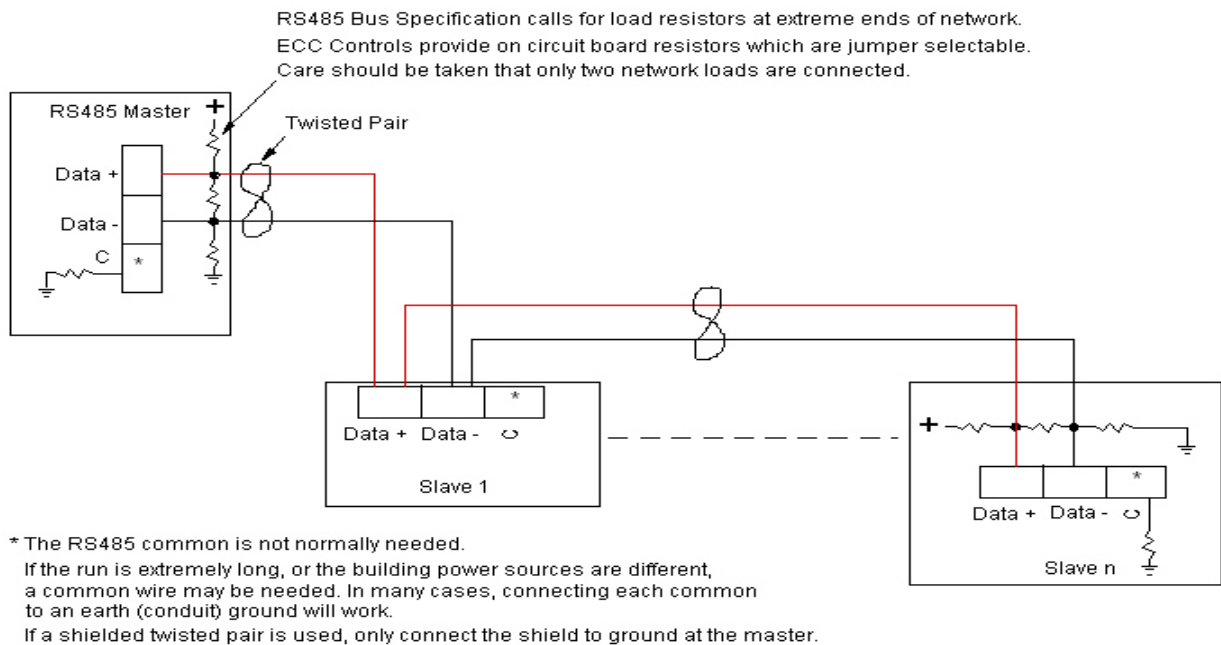
Low byte = 0x00 for no hold.

Low byte = 0x01 for hold.

The returned ADC value is the average of eight readings.

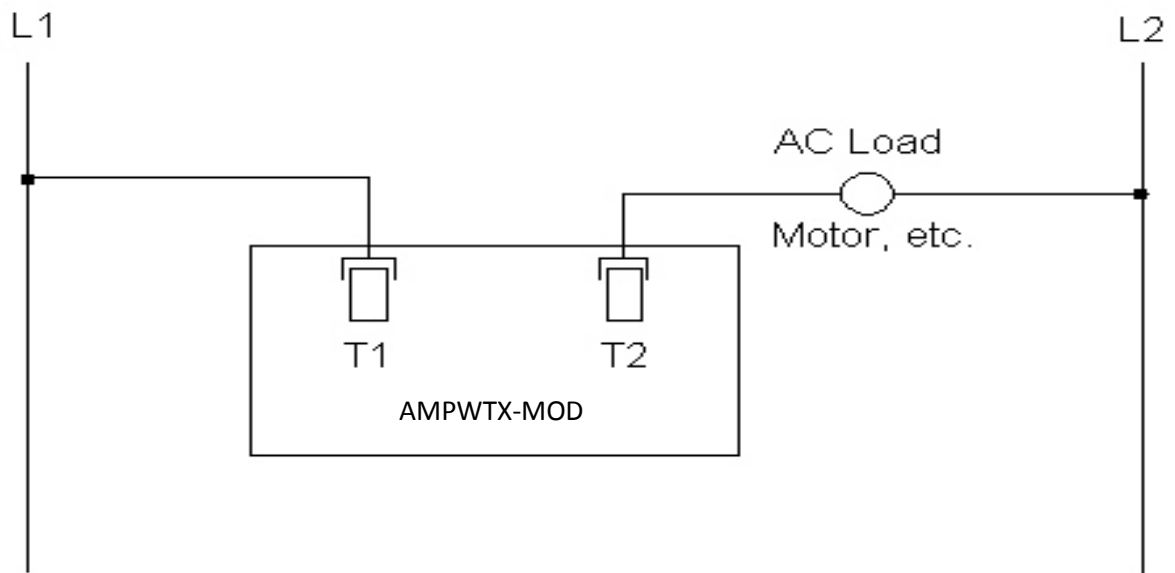
For additional Modbus protocol information, see the manual for DCB2620.

For a simple protocol, see Simple ASCII Protocol document.



Typical RS485 Bus Network.





Typical connection for the AMPWTX-MOD current sense module.