



Using an AIMCO Gen IV Controller with Modbus TCP

Using the Touch Screen or System Port, navigate to Controller→I/O→Modbus TCP

Modbus TCP

The AcraDyne controller supports the Modbus/TCP Server protocol on the local Ethernet port. The controller can accept messages from a Modbus/TCP Client and return responses to the Client.

Supported Features:

The Modbus/TCP server supports the Modbus RTU protocol. Modbus RTU is a communication protocol that represents devices as "Registers" and "Coils". Modbus TCP defines multiple classes of devices based on functionality. The controller is a Class 1 device supporting all Class 0 and 1 functions.

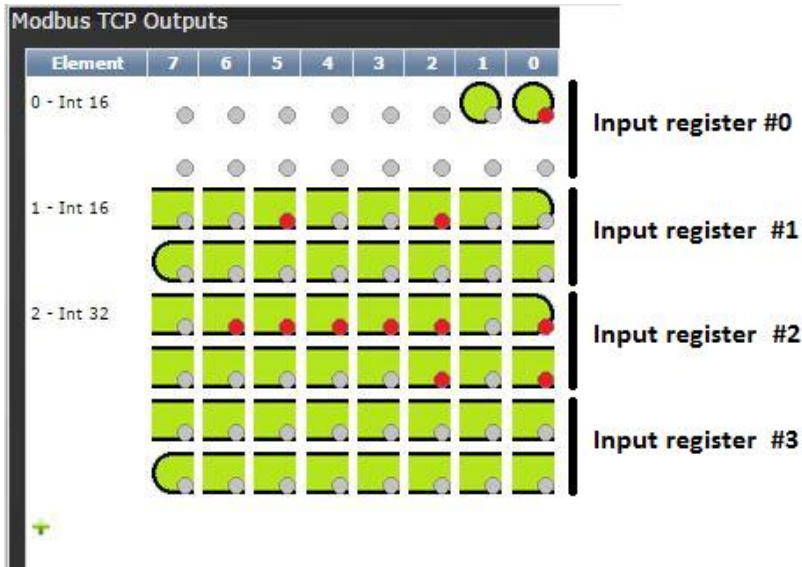
- Class 0 devices must support function codes 3 and 16.
- Class 1 devices must support function codes 1-7 and 16.

The supported functions are:

- Function Code 1 - Read Coil Status
- Function Code 2 - Read Input Status
- Function Code 3 - Read Holding Registers
- Function Code 4 - Read Input Registers
- Function Code 5 - Force Single Coil
- Function Code 6 - Write Single Holding Register
- Function Code 7 - Read Exception Status
- Function Code 16 - Write Multiple Holding Registers

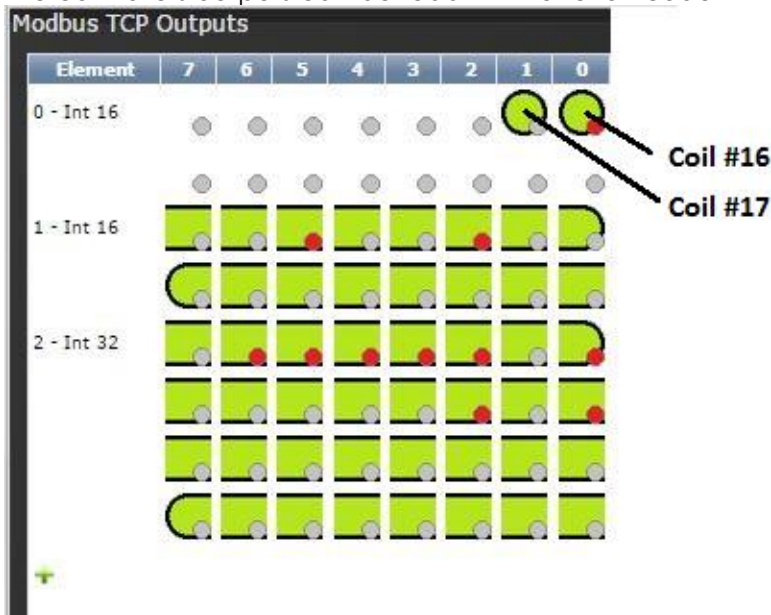
Controller Outputs Addressing

The controller's assignable outputs are mapped as Modbus TCP input registers. The first two assignable output bytes are register 0 followed by register 1 (bytes 2 & 3). Since Modbus TCP uses 16 bit registers it is helpful to create assignments with a size of INT16. The controller's outputs can read with function code 4 "Read Input registers".



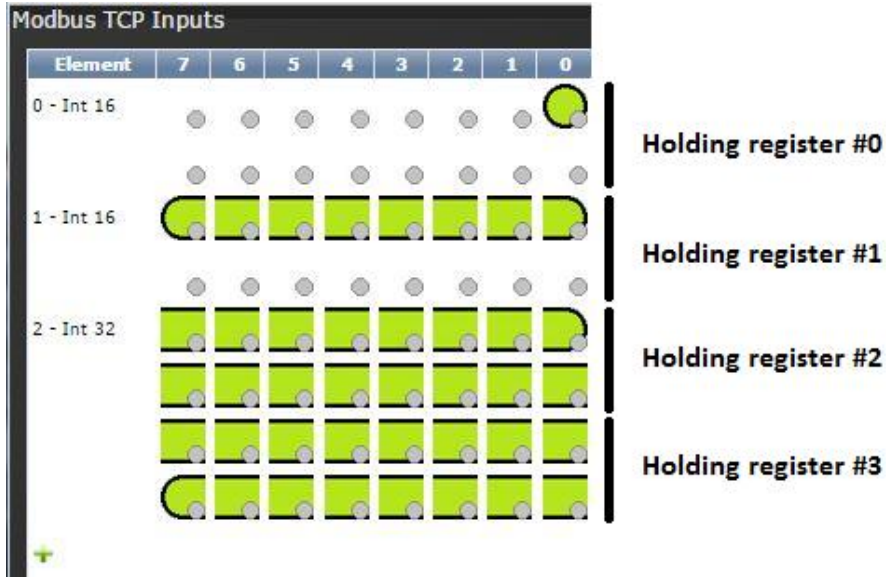
The controller's outputs can also be addressed as coils. The assigned outputs start at coil #16.

The controllers outputs can be read with function code 2 "Read Input Status".



Controller Inputs Addressing

The controller's assignable inputs are mapped as Modbus TCP holding registers. The first two assignable input bytes are register 1000 followed by register 1001 (bytes 2 & 3). Since Modbus TCP uses 16 bit registers it is helpful to create assignments with a size of INT16. The controller's inputs can be written with function code 6 "Write Single Holding register" and function Code 16 "Write Multiple Holding Registers". The controller's inputs can also be read with function Code 3 "Read Holding registers".



The controller's inputs can also be addressed as coils. The assigned inputs start at coil #1015.

The controller's inputs can be written with function code 5 "Force Single Coil". The controller's inputs can also be read with function code 1 "Read Coil Status".

