Aug 03

Recording Digital I/O

General

After entering the name, units and scale factors, the CPP then asks for the data source for this channel. If other is selected (0), the CPP presents the following table.

A = Analog to Digital
D = Digital I/O
C = Computed
F = Counter
R = Rolling

The CPP has provisions for collecting and storing digital I/O information in time synchronization with the data values. Selecting a D from the above table sets this channel up to collect and store digital data. The CPP responds with the following:

1.) Digital Input(I), Output(O), Inst(M) or Clr(C) = **I<cr>**
2.) Strt Bit 1, 9, 17, 25, 33 = 00 **9<cr>**
3.) Channel # =

Line 1 asks if the digital data value to be collected and stored is from the digital inputs, the digital output bits or from an instrument. This question also allows the digital set up for this channel to be cleared. In the above, the user selected the digital inputs.

Line 2 asks which block of 8 bits is to be recorded in this channel. In the above example, the user selected recording bits 9-16.

Digital Data From Instrument

To set the CPP up to collect and store digital data from an instrument requires two passes through the channel set up. The first pass should be the digital pass and an **M** entered as the digital input selection. When the CPP again asks for the channel # from the Main Menu, enter the same channel number and set up the desired instrument. This **MUST** follow the digital set up for this channel.

Upon exiting, the CPP will set this channel's units to **DgM-xx**, where xx is the bit starting number. The bit starting number is not used when getting digital information from an instrument. The CPP also sets the interim and final number of samples required to one.

The CPP will also set the full scale value to 1000 and the zero value to 0. The CPP changes the units, full scale and zero values, and the number of samples criteria in the same fashion for a strictly digital input channel.

The values inserted by the CPP into the units position can not be changed. The CPP uses these values to detect the set up of a digital recording channel.

Digital Weighting
For clarity the full scale value of this channel should be set at 1000 and the zero value set to 0 with no decimal points. The CPP automatically changes the full scale and zero settings to these values. The CPP stores the digital data as the binary weighted sum of the incoming bits as given in the table below.

Binary Bit #	Weighting Weight
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128

Example
An example of digital weighting is given below.

Bit #	Setting	Weight
1	1	1
2	0	0
3	0	0
4	1	8
5	1	16
6	0	0
7	1	64
8	1	128
		217 = weighted sum

The CPP will store this combination of bit settings as a value of 217. This is the only combination of these bits that will yield a sum of 217.

Operation
The CPP samples all digital I/O every second and keeps a logical OR of all bit settings. At the minute boundary the CPP uses these setting in the channel validation, and stores the relevant weighted value for channels set up as digital channels. The I/O arrays are then cleared for the start of the next minute. The interim and final values are formed from the minute values. Normal channels are averaged to produce a value. Digital interim and final values are produced by logically OR'ing the minute binary values to form the value to be stored. If any bit is set during a one minute period it will show up in all subsequent numbers using this minute value.