## Serially Controlled Calibrations

**13.0  General**

The CPP supports serial control of instrument calibrations and calibration instruments, such as mass flow controllers. Calibrations are all tied into the digital output bits of the CPP.  As described in the CPP manual, digital I/O bit settings apprise the CPP when an instrument is measuring the sample input, the zero input or the span input.  These bit settings can be used to drive relays, which control solenoids, which switch the various gases into the instruments.

In addition, the serial control of calibrations can be tied to the digital output bit transitions.  Upon sensing the setting or resetting of digital output bits, the CPP can be set up to send command strings to instruments and to calibrators.  To set up serial control of calibrations, from the Main Menu select the calibration set up.  The CPP responds with:

Line 1          Number of Spans = 1**<cr>**
Line 2          Digital(D) or Serial(S) = **S<cr>**

As discussed above, the digital path allows bit settings to apprise the CPP of the type of input to each instrument.  As an example, digital output bit #30 being set true indicates that the SO2 analyzer (on CPP channel #02) is in the zero mode.  Digital output bit #31 being set true indicates that the SO2 analyzer is in the span mode.  These digital outputs can be used to directly drive solenoids, or to start sequences inside calibrators that control and conduct instrument calibrations.

In addition to using the digital outputs of the CPP to drive solenoids, the CPP can be configured to sense transitions of digital output bits, and to send serial commands to instruments to close solenoids and to start calibration sequences.  The calibration bit assignments in the digital path must still be set up as before.

The CPP can be set up to simultaneously control calibrations with relay closures, control calibrations with serial commands, or sense contact closures as calibration indicators. Column #1 below describes the setting up of a serially controlled zero calibration, and column #2 describes the setting up of a digitally controlled span calibration.  Prior to initializing the serial calibration features, the channels must be set up in the CPP.

At the bit number question, inputting a bit number followed by a C will clear all serial conditions set up for this particular bit transition.  For example, inputting **30TC<cr>** will clear all serial conditions for bit #30 going true.  The CPP responds with Bit #30 Cleared.

The commands presented with each question will vary depending on the manufacturer and the type of instrument connected to the channel selected.

**13.1  API Instrument - API Calibrator**

In the following example, an API SO2 analyzer is initialized in CPP channel #02, and an API model 700 mass flow controller is initialized in CPP channel #20.

| | Column #1 - Set Up Zero | Column #2 - Set Up Span |
|---|---|---|
| Line 3 | Bit # = **30T\<cr\>** | Bit # = **31T\<cr\>** |
| Line 4 | Cmd #1 | Cmd #1 |
| Line 5 | Chan # 00 **2\<cr\>** | Chan #  00 **2 \<cr\>** |
| Line 6 | API  100A   SO2 | API  100A   SO2 |
| Line 7 | Command | Command |
| | 01 =    Start Zero | 01 =    Start Zero |
| | 02 =    Stop Zero | 02 =    Stop Zero |
| | 03 =    Set Zero | 03 =    Set Zero |
| | 04 =    Start Span | 04 =    Start Span |
| | 05 =    Stop Span | 05 =    Stop Span |
| | 06 =    Set Slope | 06 =    Set Slope |
| | Selection = 00 **1\<cr\>** | Selection =00 **4\<cr\>** |
| Line 8 | Cmd #2 | Cmd #2 |
| Line 9 | Chan # 00 **20\<cr\>** | Chan # 00 **20\<cr\>** |
| Line 10 | API  700   MFC | API  700   MFC |
| Line 11 | Command | Command |
| | 01 =    Str SO2 | 01 =    Str SO2 |
| | 02 =    Str NOx | 02 =    Str NOx |
| | 03 =    Str CO | 03 =    Str CO |
| | 04 =    Str O3 | 04 =    Str O3 |
| | 05 =    Str HC | 05 =    Str HC |
| | 06 =    Str GPT | 06 =    Str GPT |
| | 07 =    SPC01 | 07 =    SPC01 |
| | 08 =    SPC02 | 08 =    SPC02 |
| | 09 =    SPC03 | 09 =    SPC03 |
| | 10 =    SPC04 | 10 =    SPC04 |
| | 11 =    SPC05 | 11 =    SPC05 |
| | 12 =    SPC06 | 12 =    SPC06 |
| | 13 =    Standby | 13 =    Standby |
| | Selection = 00 **1\<cr\>** | Selection = 00 **2\<cr\>** |
| Line 12 | Cmd #3 | Cmd #3 |
| Line 13 | Chan # = 00 **\<cr\>** | Chan # = 00 **00\<cr\>** |
| Line 14 | Bit # = **30F\<cr\>** | Bit # = **31F\<cr\>** |
| Line 15 | Cmd #1 | Cmd #1 |
| Line 16 | Chan # = 00 **2\<cr\>** | Chan # = 00 **2 \<cr\>** |

| | | |
|---|---|---|
| Line 17 | API  100A  SO2 | API  100A  SO2 |

| | | |
|---|---|---|
| Line 18 | Command | Command |
| | 01 =  Start Zero | 01 =  Start Zero |
| | 02 =  Stop Zero | 02 =  Stop Zero |
| | 03 =  Set Zero | 03 =  Set Zero |
| | 04 =  Start Span | 04 =  Start Span |
| | 05 =  Stop Span | 05 =  Stop Span |
| | 06 =  Set Slope | 06 =  Set Slope |
| | Selection = 1 **2<cr>** | Selection = 1 **5<cr>** |

| | | |
|---|---|---|
| Line 19 | Cmd #2 | Cmd #2 |
| Line 20 | Chan # = 00 **20<cr>** | Chan # 00 **20<cr>** |

| | | |
|---|---|---|
| Line 21 | API  700  MFC | API  700  MFC |

| | | |
|---|---|---|
| Line 22 | Command | Command |
| | 01 =  Str SO2 | 01 =  Str SO2 |
| | 02 =  Str NOx | 02 =  Str NOx |
| | 03 =  Str CO | 03 =  Str CO |
| | 04 =  Str O3 | 04 =  Str O3 |
| | 05 =  Str HC | 05 =  Str HC |
| | 06 =  Str GPT | 06 =  Str GPT |
| | 07 =  SPC01 | 07 =  SPC01 |
| | 08 =  SPC02 | 08 =  SPC02 |
| | 09 =  SPC03 | 09 =  SPC03 |
| | 10 =  SPC04 | 10 =  SPC04 |
| | 11 =  SPC05 | 11 =  SPC05 |
| | 12 =  SPC06 | 12 =  SPC06 |
| | 13 =  Standby | 13 =  Standby |
| | Selection = 1 **13<cr>** | Selection = 1 **13<cr>** |

| | | |
|---|---|---|
| Line 23 | Cmd #3 | Cmd #3 |
| Line 24 | Chan # = 00 **<cr>** | Chan # = 00 **<cr>** |

| | | |
|---|---|---|
| Line25 | Bit # = **<cr>** | Bit # = **<cr>** |
| | Return To Main Menu | Return to main Menu |

Referencing column #1 to begin with, in the above example, sending commands to an API model 100A SO2 instrument and an API model 700 calibrator have been set up. The commands start sequence #1 in the model 700 calibrator and at the same time switch the zero air input solenoid in the model 100 SO2 analyzer.  The timing control of the digital I/O bits are set up in the sequencers.  Tying the digital I/O bits to the calibration is set up in the Digital pass of the above command (line 2).

**Line 1** sets up the number of spans in the system.

**Line 2** allows selection of tying the digital I/O bits to the calibration channels or setting up the serial commands to send to instruments. In this example the serial set up is selected.

**Line 3** Asks which digital output bit is to be monitored. The input 30T indicates that when digital output bit # 30 goes true, the following commands are to be sent. Four commands are allowed for the setting or resetting of each bit. If more than four commands are needed then two bits can be set in the sequence set up.

---

**Note** - Only one command can be sent to any instrument on a bit change. For example, a command resetting the zero mode and setting the span mode in the same instrument **can not** be sent on the same bit transition. The sequencers must be set up to send the reset zero command, and then set another bit to send the set span command.

---

**Line 4** indicates that command #1 is being set up. There are four commands available with each bit transition.

**Line 5** asks which channel this command is to sent out. In the example, channel #2 was selected. The CPP then prints the manufacturer and model number of the instrument initialized in this CPP channel. The CPP prints that it found an API model 100A SO2 analyzer in CPP channel #2.

**Line 6** prints the manufacturer and model number found initialized in this CPP channel.

**Line 7** presents a list of commands that are supported for this instrument. This will vary depending on the manufacturer and the type of instrument. In the example, the command to close the zero air solenoid was selected.

**Line 8** indicates that command #2 is to be set up.

**Line 9** asks which CPP channel this command is to be sent out. Channel #20 was selected. The CPP print that an API model 700 mass flow controller was found in CPP channel #20.

**Line 10** prints the manufacturer and model number found initialized in this CPP channel.

**Line 11** asks for the command to send. In this example, a command to start sequence #01 in the model 700. This sequence should have been programmed to provide a zero air output.

**Line 12** indicates that command #3 is to be set up.

**Line 13** asks which CPP channel the commands are to be sent out. Selecting a channel # of zero indicates that there are no more commands associated with this bit transition. A carriage return is input as there are no more commands to set up for this bit transition

Thus far, when the CPP detects a true transition on bit #30, it will queue up and send out comm port #3-02 (which was initialized in CPP channel #2), a set zero solenoid command to an API model 100A SO2 analyzer, and queue up and send out comm port #3-08 (which was initialized in CPP channel #20), a start SO2 Sequence command to an

API model 700 calibrator.  In the digital portion set up, the true state of this bit (output bit #30) should indicate to the CPP that the SO2 analyzer is in the zero air mode.

**Line 14** asks for the next bit transition.  The input #30F indicates that when digital output bit #30 goes false, the following commands should be sent.

**Line 15** indicates that command #1 is to be set up,

**Line 16** asks for the CPP channel number the command is to be sent out.  02

**Line 17** prints the manufacturer and model number found on this channel. API  100A  SO2

**Line 18** asks for the command.  Reset zero air solenoid

**Line 19** indicates that command #2 is to be set up.

**Line 20** asks for the channel #. 20

**Line 21**  print the manufacturer and model number. API 700 MFC

**Line 22** asks for the command.  Standby

**Line 23** indicates that command #3 is to be set up.

**Line 24** asks for the Channel number.  00

**Line 25** asks for the next bit transition.  No More

The last section sends a reset zero air command out comm port #3-02 to an API model 100A SO2 analyzer.  It also sends a return to standby command out comm port #3-08 to an API model 700 calibrator.

The digital portion should have been set up so when detecting the false transition of digital output bit #30, the CPP takes the last reading as a zero value.

Column #2 sets up a span on the SO2 analyzer.  As an example, the sequencers could be set up so that at a particular time (12:00:15) the CPP sets bit #30. The CPP sends a START ZERO command to the SO2 analyzer, which closes the internal solenoid, and sends a START SEQUENCE #01 command to the M700 calibrator.  Fifteen minutes later (12:15:15) the CPP resets bit #30 and sets bit #31.  At 12:15:15 the CPP takes a zero reading from the SO2 analyzer and sends a STOP ZERO mode to the unit.  It also sends a RETURN TO STANDBY command to the M700 calibrator.

A flag is set for each comm port that is queued up to be sent a command.  The serial calibration firmware has to vie for the comm ports with the poller and possibly commands from an operator.  All commands are interleaved, so it normally appears instantaneous to the user.  However, some older API instrumentation can take several seconds to respond to an interrogation.  In such a case it could be several seconds before the serial controller firmware is given access to a given comm port.  The poller and operator initiated commands are given priority over the serial calibration control.

So given the response time of some instruments, and if they are being polled when the serial calibration is taking place, it could be 12:15:01, 02, 03 or later before the serial

calibration command can be sent.  It is a good practice to start and stop calibrations after the polling is completed, for example at 15 seconds, 12:15:15.

At 12:15:15 the CPP will send a START SPAN command to the SO2 analyzer, which closes the internal span solenoid, and also sends a START SEQUENCE #2 command to the M700 calibrator.   Fifteen minutes later (12:30:15) the CPP resets bit #31.  The CPP takes a span reading from the SO2 analyzer and sends a STOP SPAN command to the unit.  It also sends a STOP SEQUENCE command to the M700 calibrator.

The above has conducted a zero /span on the SO2 analyzer.

The L Keyin prints as follows.

| | | Serial Calibration Commands | | | | |
|---|---|---|---|---|---|---|
| Bit # | Cmd # | Chn#-CP# | | Manu | Model | Command |
| 30T | 01 | 02 | 02 | API | 100A | 01 Str Zero |
| 30T | 02 | 20 | 08 | API | 700 | 01 Str Seq #01 |
| | | | | | | |
| 31T | 01 | 02 | 02 | API | 100A | 04  Str Span |
| 31T | 02 | 20 | 08 | API | 700 | 03  Str Seq #02 |
| | | | | | | |
| 30F | 01 | 02 | 02 | API | 100A | 02  Stp Zero |
| 30F | 02 | 20 | 08 | API | 700 | 02  Standby |
| | | | | | | |
| 31F | 01 | 02 | 02 | API | 100A | 05  Stp Span |
| 31F | 02 | 20 | 08 | API | 700 | 04  Standby |

## 13.2 Environics Calibration Systems

The CPP firmware supports three types of interfaces with the Environics calibrators. One is a digital interface using the digital outputs of the CPP to drive the status input bits of the calibrators. The second uses the status output bits of the calibrator to drive the digital input bits of the CPP. The third type sends string commands over the RS-232 port to select concentrations.

## 13.2.1 CPP Digital Outputs Controlling Calibrator

To use the digital interfacing, sequencers should be set up in the calibrator that produce the desired concentration. These sequences can then be invoked by setting up the CPP digital outputs to drive the status input line of the calibrator. The duration of the sequence should be longer that required by the calibration set up, as the sequencers are stopped by the Stop status bit. The following example provides an indication of how this feature is set up.

**In Environics**
Sequence #1  set flows and concentrations for zero air
Sequence #2  set flows and concentrations for span value
Set status input #1 to Sequence #1
Set status input #2 to Sequence #2
Set Status #3 to Stop

**In CPP (bit numbers and channel numbers are arbitrary)**
Set digital output bit #1 true when channel #01 is in zero mode
Set digital output bit #2 true when channel #01 is in span mode
Use digital output bit #3 for Stop command
Set up a sequencer to output bit #1 - zero
After zero time, reset bit #1, set bit #3
After 5 seconds reset bit #3, set bit #2 - span
After span time, reset bit #2, set bit #3
After 5 seconds reset bit #3
set up the CPP so that the output bits flag zero/span values

**Wiring**
Connect CPP digital output #1 to Environics status input #1
Connect CPP digital output #2 to Environics status input #2
Connect CPP digital output #3 to Environics status input #3

## 13.2.2 Environics controlling calibration

This approach is similar to that presented above with the control of the calibration residing in the calibrator. The duration of the sequencers should be set to the desired calibration times.

**In Environics**
Set up sequencer #01 and #02 as above
Set up status output bit #1 to sequencer #01
Set up status output bit #2 to sequencer #02
Set up timer to start sequences

**In CPP**
Set up CPP to sense digital input #01 as zero mode for channel #01

Set up CPP to sense digital input #02 as span mode for channel #01

**Wiring**
Connect Environics status output #01 to CPP digital input #01
Connect Environics status output #02 to CPP digital input #02

These are simple examples, but they are indicative of how the digital interfacing operates, and can be readily expanded to more sophisticated applications. The digital output bits of the CPP can also be encoded to drive the "Sequence Group Binary Status Input Bit Sensing" of the Environics, as well.

### 13.2.3 RS-232 Controlled Calibrations
The CPP can be set up to send command to the Environics calibrators over the RS-232 port. This is accomplished by setting up and saving concentrations in the calibrator, and then having internal flags in the CPP queue up and send strings to the calibrator that invoke the stored concentrations. The internal flags in the CPP are derived from sequencers set up to provide the desired calibrations.

The standard CPP can be set up to select 10 concentration settings. Should more be needed, please contact H2NS, and this will be increased at no cost. An example is presented below.

**In Environics**
Set up a zero air concentration, save as concentration 01. Set up a span concentration, save as concentration 02.

**In CPP**
Set up digital output bit #30 to indicate that channel #01 is in a zero mode, and digital output bit #31 to indicate that channel #01 is in a span mode. Channel #1 would be the instrument channel number. In the below example, the Environics calibrator has been initialized into CPP channel number 20. Set up a sequence to provide the proper digital output bit settings to conduct the calibration. Going into a zero mode sets digital output bit #30. Exiting the zero mode resets bit #30. Going into the span mode sets digital output bit #31. Exiting the span mode resets bit #31.

| | **Column #1 - Set Up Zero** | **Column #2 - Set Up Span** |
|---|---|---|
| Line 3 | Bit # = **30T\<cr>** | Bit # = **31T\<cr>** |
| Line 4 | Cmd #1 | Cmd #1 |
| Line 5 | Chan # 00 **20\<cr>** | Chan # 00 **20 \<cr>** |
| Line 6 | Environics 9100 | Environics 9100 |
| Line 7 | Command | Command |
| | 01 = Set Conc 01 | 01 = Set Conc 01 |
| | 02 = Set Conc 02 | 02 = Set Conc 02 |
| | 03 = Set Conc 03 | 03 = Set Conc 03 |
| | 04 = Set Conc 04 | 04 = Set Conc 04 |
| | 05 = Set Conc 05 | 05 = Set Conc 05 |
| | 06 = Set Conc 06 | 06 = Set Conc 06 |
| | 07 = Set Conc 07 | 07 = Set Conc 07 |

|                    |                   |
|--------------------|-------------------|
| 08 =   Set Conc  08 | 08 =   Set Conc 08 |
| 09 =   Set Conc  09 | 09 =   Set Conc 09 |
| 10 =   Set Stop    | 10 =   Set Stop   |
| 11 =   Set Home    | 11 =   Set Home   |
| Selection = 00 **1\<cr>** | Selection =00 **2\<cr>** |

| | | |
|---|---|---|
| Line 8 | Cmd #2 | Cmd #2 |
| Line 9 | Chan # 00 **\<cr>** | Chan # 00 **\<cr>** |
| | Return To Main Menu | Return to main Menu |

**Line 3** Asks which digital output bit is to be monitored.  The input 30T indicates that when digital output bit # 30 goes true, the following commands are to be sent.  Four commands are allowed for the setting or resetting of each bit.  If more than four commands are needed then two bits can be set in the sequence set up.

---

  **Note** - Only one command can be sent to any instrument on a bit change.  For example, a command resetting the zero mode and setting the span mode in the same instrument **can not** be sent on the same bit transition.  The sequencers must be set up to send the reset zero command, and then set another bit to send the set span command.

---

**Line 4** indicates that command #1 is being set up.  There are four commands available with each bit transition.

**Line 5** asks which channel this command is to sent out.  In the example, channel #20 was selected.

**Line 6** The CPP then prints the manufacturer  and model number of the instrument initialized in this CPP channel.  The CPP prints that it found an Environics model 9100 calibrator in CPP channel #20.

**Line 7** presents a list of commands that are supported for this instrument.  This will vary depending on the manufacturer and the type of instrument.   In the example, the command to select concentration #01 was selected.

**Line 8** asks for the next command.   Here commands could be sent to other instrumentation.  The carriage return indicates that nothing else is to be send based on the digital output bit.

The above directed calibration gases to the instrumentation.  The following exits the calibration and shuts off the calibration gases.

**Column #1 - Exit Zero**                    **Column #2 - Exit Span**

| | | |
|---|---|---|
| Line 3 | Bit # = **30F\<cr\>** | Bit # = **31F\<cr\>** |
| Line 4 | Cmd #1 | Cmd #1 |
| Line 5 | Chan # 00 **20\<cr\>** | Chan #  00 **20 \<cr\>** |
| Line 6 | Environics 9100 | Environics 9100 |

Line 7

| Command | Command |
|---|---|
| 01 =    Set Conc 01 | 01 =    Set Conc 01 |
| 02 =    Set Conc 02 | 02 =    Set Conc 02 |
| 03 =    Set Conc 03 | 03 =    Set Conc 03 |
| 04 =    Set Conc 04 | 04 =    Set Conc 04 |
| 05 =    Set Conc 05 | 05 =    Set Conc 05 |
| 06 =    Set  Conc 06 | 06 =    Set Conc 06 |
| 07 =    Set Conc  07 | 07 =    Set Conc 07 |
| 08 =    Set Conc  08 | 08 =    Set Conc 08 |
| 09 =    Set Conc  09 | 09 =    Set Conc 09 |
| 10 =    Set Stop | 10 =    Set Stop |
| 11 =    Set Home | 11 =    Set Home |
| Selection = 00 **11\<cr\>** | Selection =00 **11\<cr\>** |

| | | |
|---|---|---|
| Line 8 | Cmd #2 | Cmd #2 |
| Line 9 | Chan # 00 **\<cr\>** | Chan # 00 **\<cr\>** |
| | Return To Main Menu | Return to main Menu |

tn13.apx