Interfacing with the Andersen FH 62 C14

The CPP offers an interface to the Andersen FH 62 C14 dust sampler.

1.0 Initialization

In the channel initialization, instead of asking for a model number, the CPP asks for a one or two character command. The CPP will accept any two character input. Any command input that the Andersen does not respond to will result in an error flag on the data points.

Any command sent to the Andersen from which a proper response is received results in the CPP sending a second command to the instrument requesting the composite instrument status. This status response is checked, and if the instrument has any global or operating status conditions set flags the data point with an I, and with an A in the case of warnings. If no status conditions are set, the data point does not have a flag. The status response is checked in the order given above, global, operating, and then warnings.

The CPP configures the Andersen comm port for seven data bits, two stop bits and even parity.

2.0 Composite Status

The composite status command, #, results in eleven pairs of status bytes being returned by the Andersen unit. If a channel is initialized to poll for the composite status, #x command, then the CPP converts the two status bits, identified by the x, into a value between 0 and 255. This can be reviewed by looking at the minute values with a PM keyin. The x can have a value between 1_H and B_H, which translates into 1_{10} -11_{10} in decimal numbering.

The last page of this application note provides a table breakdown of the status string sent by the Andersen unit.

3.0 Connections

The interface connector to the Andersen FH-62 instrument is a DB-25M connector. Presented below is the wiring for connecting the instrument to a CPP.

Signal	RJ-11	DB-25M	Color
TxD from CPP	2	3	Yel
RxD into CPP	5	2	Blk
Gnd	3,4	7	Red/Grn

4.0 Detailed Initialization

The CPP is initialized via the operator port, comm port #4. Entering an **I<cr>** results in the CPP bringing up a Main Menu selection. Selecting the channel initialization results in the CPP asking a series of questions as presented below. Operator responses are in **bold**, and **<cr>** means a carriage return. The numbers 1), 2) and so forth are used for discussion purposes only and are not printed in an actual system initialization.

1.) Channel # = **1<cr>**
2.) Name = NAME **Ands<cr>**
3.) Units = UNITS **PPB<cr>**
4.) Full Scale = 100.0 **500.0<cr>**
5.) Zero = 0.0 **0.0<cr>**

6.) Instrument Manufacturer
00 = Other
01 = API
02 = Ander
03 = ML
04 = Siemens
05 = Thermo

7.) Setting = 00 **2<cr>**

8.) Andersen – Mdl#
01 = FH-62
9.) Setting = 01<cr>
10.) Enter 2 Char Command = C<cr>

11.) Comm Port (3-00 Thru 3-08) = 3-00 3- **02<cr>**
12.) Baud Rate
1 = 300, 2 = 1200, 3 = 2400, 4 = 4800, 5 = 9600, 6 = 19.2K, 7 = 38.4K **= 5<cr>**

13.) Instrument ID = 0000 **<cr>**
14.) IMD Installed = 00 **<cr>**
15.) # Points For Interim Avg = 01 **04<cr>**
16.) # Points For Final Avg = 01 **9<cr>**
17.) Unip(U)/Bip(B) = U **<cr>**
18.) Unip(U)/Bip(B) Cals = B **<cr>**
19.) Channel # = **<cr>**

Line one simply asks which channel is to be initialized. A carriage return here exits back to the Main Menu. In other steps, a carriage return input leaves the setting unchanged and the CPP goes to the next question. In cases where the operator must provide an input, the CPP asks the same question again.

Lines 2 through 5 request information that is not required for proper operation when interfaced to an instrument over the serial port. It is, however, good operating procedures to set these inputs to match those in the instrument. The correct full scale and zero values are required if data stored in the CPP is being output to some other device (for example, over a Modus connection) in a binary format. This is also true of channels set up as ADC input channels.

Channel names and the units can be up to six characters in length. The full scale and zero values can be four digits in length and the number of digits after the decimal point (if a decimal point is used) must be the same in the full scale and zero inputs. The inputs can also have a minus sign.

Line 6 lists all of the manufacturer drivers that are resident in this software version. In the example, only five manufacturers are presented. As the manufacturers are alphabetized, the numbers delineating the various manufacturers will change depending on how many drivers are installed.

In Line 7 of our example, the operator selected Andersen as the manufacturer. The CPP then asks which model number is to be associated with this channel.

In Lines 8 & 9, for the Andersen, the CPP defaults to model #01, the FH-62.
In Line 10 the operator is asked to enter a one or two digit command code associated with this channel. In the above example, the operator entered the C command which requests mass concentration.

In Line 11 the CPP asks to which comm port this analyzer is going to be connected. This is the connector on the CPP into which the cable from the instrument is plugged. The allowable inputs are 00-08, which totals nine comm ports.

The comm port nomenclature should not be made overly complicated or confusing. Although comm Ports 3-00 through 3-08 are independent serial ports, internally the comm port 3 designation references interfacing to the instrumentation. Comm port 3-00 addresses comm port #3 physically located on the CPP printed circuit board and labeled comm #3 on the CPP back panel. Comm ports 3-01 through 3-08 address comm ports physically located on the expansion printed circuit board and are labeled comm 3-01 through comm 3-08 on the CPP back panel.

It should be noted that comm port #2 can be used to interface to some instrumentation. The Andersen line of instrumentation is **not** included in that group, and should always be assigned to a comm port number 3 designation. Comm port #2 is usually used with instruments that broadcast, which is normally meteorological instrumentation.

In line 12, the CPP is asking what baud rate should be used to communicate with this instrument. This setting should be 9600 Baud.

Line 13 asks if an instrument ID is required. An ID is not required with the Andersen instruments. It is recommended that an ID not be used. If an ID is set up, Input the appropriate digits ID. In the example, the user did not input an ID.

Line 14 is asking if an IMD (Intelligent Multidrop Device) is installed in this comm port cabling. The IMD, manufactured by H2NS, allows a number of different instruments to be connected to a single comm cable. The instruments can have different protocols and even different baud rates. This is very useful when a collection of instrumentation is located remotely from the CPP. The IMD allows only one cable to be run between the CPP and the instruments, with the IMD providing the instrument multiplexing locally at the instruments. The IMD supports addresses from 01 through 60. An address of 00 indicates that an IMD is not installed in this comm cabling.

Lines 15 & 16 are questions regarding data validity qualifiers. The first (line 15) asks how many valid one minute samples the CPP must have to qualify the interim average as valid. Question 16 asks the same in regards to the final average.

Lines 17 & 18 allow negative data values to either be accepted or set to zero. Usually, negative values are set to zero, except during calibrations.

5.0 Communicating directly with the FH-62

The CPP supports two modes in which an operator can communicate directly with the connected instruments. One is an on line mode, in which the CPP interleaves operator and polling commands, and the second is an off line, unabated pass through mode.

5.1 On Line Communications

Via comm port #4, the CPP has a menu that defines the appropriate commands associated with the instrument selected. Via comm port #1, enter SETQ to switch comm port #1 into comm port #4. Input **D43<cr>** and the CPP responds with a list of instrumentation that is connected to the CPP, as shown below;

```
Chn#    Manu        Mdl#          Type
01 =   Ander       FH-62        - Dust
02 =   THERMO      SO2-43C      - SO2
03 =   API         NOX-200A     - NOX
04 =   API         NO2-200A     - NOX
05 =   API         NO-  200A    - NO

Selection = 1<cr>
CMD> ?<cr>

        C – Mass Conc mg/m3            H – ½ Hr Conc
        J2 – Air Flow                 H1 – Last Hr Conc
        J3 – Low Press P2             H3 – 3 Hr Conc
        J5 – T3 Temp                  HT – 24 Hr Conc
        J8 – Mass                     m1 – Actual Mass
       JB – T1 Temp                    JE – T4 Temp
        JI – Air Flow Samp              JJ – Normal Air Flow


CMD> %#<cr>          [Operator entered string]

%#                   [String sent to instrument- % Not sent]

 0000000000000000000000          [Instrument response]
```

```
[No response from instrument – if instrument does not respond]

CMD> ^<cr>            [Exit]

>
```

In the pass through mode, strings can then be sent to the instrument with the **%[string]<cr>** inputs. The brackets are not included in the string, and are presented here for clarity only. At the command prompt, entering a question mark results in the CPP returning a menu of some commands used be the instrument.

The CPP remains in the pass through mode with this channel until it is exited as discussed below, or the pass through times out from inactivity. To select another channel, the pass through mode can be exited and reentered with another **D43xx** input, or at the CMD> prompt, enter an **M<cr>**. The CPP will present a listing of all instruments connected and ask for a selection. Select the channel desired.

To terminate the pass through mode input [**^<cr>**] at the CMD> prompt. If a character is not input for two minutes, the CPP times out and the pass through mode is exited automatically.

The CPP remains on line collecting data even in the pass through mode. Operator commands are interlaced with the CPP normal polling or interrogation commands.

5.2 Off Line Pass Through

The CPP offers an off line, unabated pass through mode to connected instruments. Over comm port #1, a command is entered commanding the CPP to connect one of the comm ports to comm port #1 and back off. Information can be passed via the instrument and the user without adhering to any CPP protocol. Any manufacturer supplied diagnostics can be conducted remotely. Only the channel being passed through is off line. Data is being collected from all other channels.

Using this feature may require that the user change the communications protocol to communicate with the Fh-62. The CPP is normally set up for 9600 Baud, 8 data bits, no parity, and one stop bit. The Andersen uses 9600 Baud, 7 data bits, no parity, and two stop bits.

More discussion of this is provided in Tech Note 36.

6.0 Error Log

The CPP maintains a running error log that lets the user determine if communications are occurring properly. After connecting instrumentation clear the error log by inputting an **EC<cr>**. After some period of time (two to three minutes) check the error log (**E<cr>**) for errors. In proper operation, there should be no errors. The Error Log lists the following.

No Response - No response from the instrument to an interrogation
Response Error - CPP received response but detected an error in the data field
Unknown Resp - CPP received a response but could not decode the response
Model # Error - CPP received a response but not from the type of instrument it is configured to interrogate on this channel

The three most common causes of errors are, the cable does or does not need a null modem, or the cable is connected to the wrong comm port on the CPP, or the Baud rate set in the CPP does not match that set in the instrument.

7.0 H2NS

A complete list of manufacturers supported by the CPP, and a list of all Tech Notes and Application Notes offered by H2NS can be found at www.h2ns.com. The telephone number at H2NS is (512) 918-8035.

This Page Left Blank.

Accessing the 11 values from the status string

# return string			DETAILED ERROR STATUS								GLOBAL ERROR	WARNING	OPERATING
			not used	black smoke measure	temperature measure	external suction tube heater	air flow regulation	air flow rate measure	dust sampling & measure	Data & program memory			
		n=	1	2	3	4	5	6	7	8	9	10	11
		start at character at (n*2)-1 position from the left	1	3	5	7	9	11	13	15	17	19	21
8040201008040201EEWWOO			80	40	20	10	08	04	02	01	EE	WW	OO
0000000000000000000000		normal operation	00	00	00	00	00	00	00	00	00	00	00

Examples of channel configurations on the CPP for extracting data from the status string

0000000001000000080000	channel with n=9 get global error status	00	00	00	00	01	00	00	00	08	00	00

08 Hex stored in channel as 08 Decimal
This would tell us there was a status set
for the air flow regulation area

0000000001000000080000	channel with n=5 get detailed air flow status	00	00	00	00	01	00	00	00	08	00	00

01 Hex stored in channel as 01 Decimal
This would tell us there 'regulation > 5%'
error in the air from regulation area