The Temperature 2 Pressure Probe (T2P): A User's Manual from Lab to Sea

Kristopher Darnell, Mark Andrews, Peter B. Flemings, and John T. Germaine

The University of Texas at Austin
Jackson School of Geosciences
Institute for Geophysics
Austin, TX
2010

TABLE OF CONTENTS

LIST OF FIGURES/TABLES

INTRODUCTION

The University of Texas (UT) has developed a pressure probe to be referred to as 'The Temperature Two Pressure Probe' (Figure 1) or T2P, for short (Flemings et al., 2006). The T2P is composed of interlocking steel cylinders accentuated by a needle shaft housing two porous pressure ports. Within the DAQ Housing of the probe is an IODP-USIO developed data acquisition system (CDAQ) (Meiring, 2008) adapted for the specific uses of the T2P.

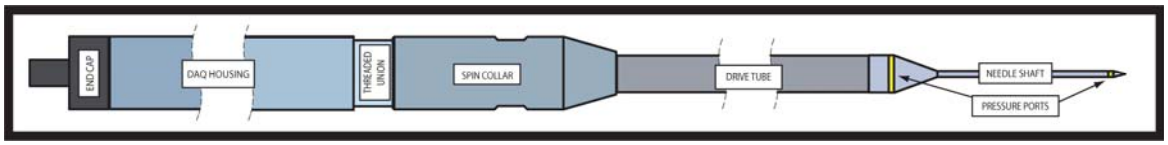


Figure 1: Diagram of Temperature Two Pressure Probe (T2P)

The CDAQ is an inclusive controller for the T2P. It operates an onboard accelerometer, onboard thermistor, and (1) thermistor and (2) analog pressure transducers that sample in-situ conditions at the tip of the probe. The program T2PLOGGR.run is housed on the CDAQ and is initiated by an external computer via serial cable. Once the program is initiated it follows the logic shown in Figure 2. It is terminated by user input. The subsequent data are analyzed by a separate post-processing program (T2PImport.xls).

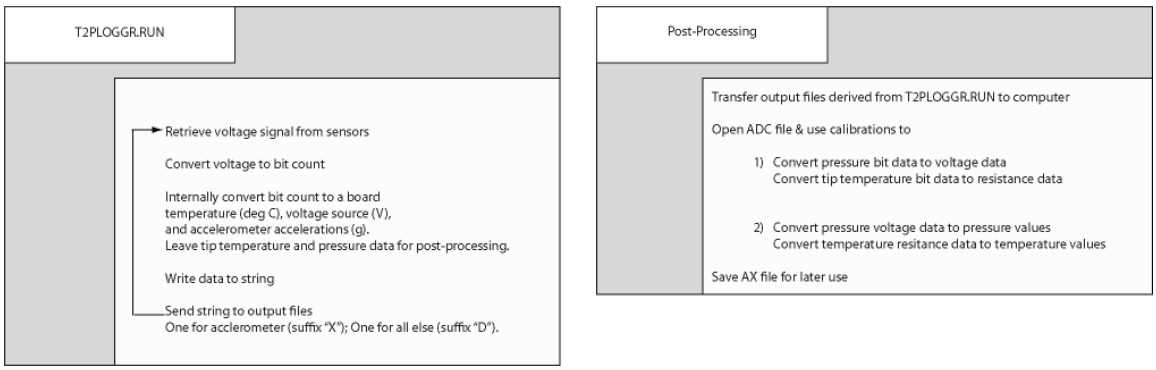


Figure 2: Flow Diagram for T2P Data Acquisition

This user manual is separated into three parts. Part 1 describes the various components of the CDAQ and provides instructions for the operation of the CDAQ, the transfer of data, and the calibrations required to interpret CDAQ data. Part 2 describes the connection configurations for operating the CDAQ. Part 3 describes the assembly and preparation of the probe for deployment. This manual provides the necessary information to utilize the T2P as a downhole tool.

PART 1:
CDAQ AND THE SOFTWARE

1.1. CDAQ & SOFTWARE OVERVIEW

The Integrated Ocean Drilling Program (IODP) built the CDAQ to be a flexible system for use with variable downhole tools and data types. For this reason, they included a generic code for a data logging program (IODP1D.run). That code was modified to interface with the two pressure trandsducers and one thermistor on the T2P. The CDAQ was also modified electrically. The user should become familiar with the physical CDAQ and the proper operation of the system as a data logging program (T2PLOGGR.run).

1.2. CDAQ DESCRIPTION

The CDAQ (Fig. 3) is composed of a Persistor, a voltage source, and a circuit board, which houses two analog-to-digital converters. The cable emanating from the 15-pin mdm connection can take many forms and serve different purposes. The cable is a necessary component for the utilization of the CDAQ, but is not truly a part of the system.

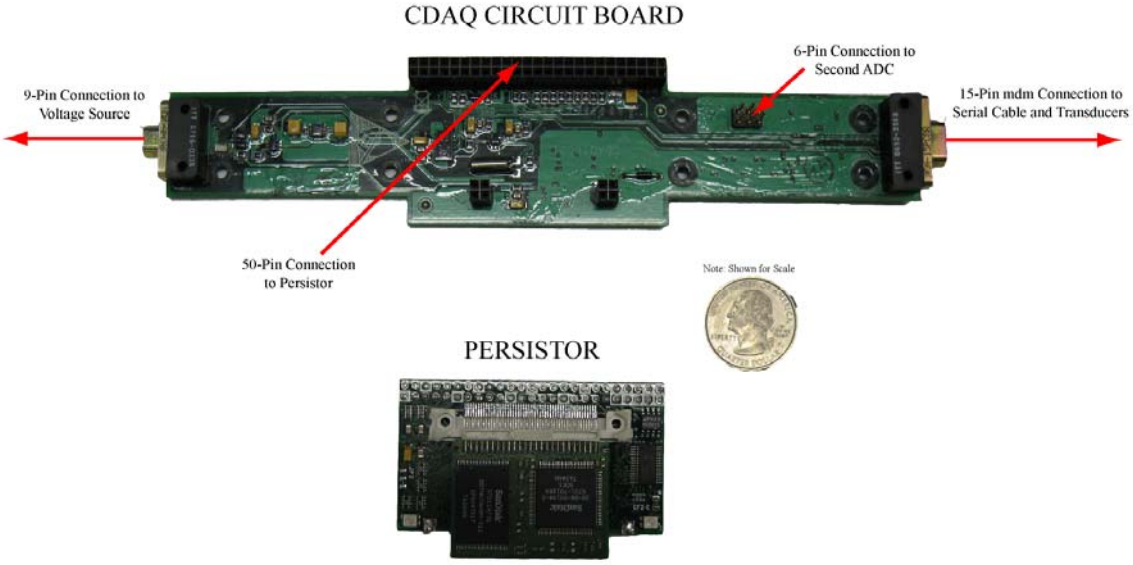


Figure 3: CDAQ System (voltage source excluded)

1.2.1. PERSISTOR

The Persistor is an open-market, single board computer (www.persistor.com). It contains a flash memory card for storage of executables and data files, a microprocessor for interface with peripherals, and the ability to execute programs. We use the CF-2 Persistor model with modifications made by the manufacturer. The flash cards were originally removable, but are now directly attached to the rest of the board because there are space constraints. The data logging program developed by UT, T2PLOGGR.run must be uploaded to any new Persistor purchased.

1.2.2. CIRCUIT BOARD

The circuit board for the CDAQ (Fig. 3) contains all the necessary wiring between peripheral components, the Persistor, and internal elements. The main components on this board are the ADC chips (see below), an accelerometer, and a low pass filter. The low pass filter is set to limit signal input in excess of ~52 kHz..

1.2.3. ANALOG-TO-DIGITAL CONVERTERS

The CDAQ contains two analog-to-digital converters labeled U5 and U7 (Appendix II). ADC-U5 is ratiometric by current for optimization of temperature data, while ADC-U7 is ratiometric by voltage. ADC-U7 is intended for use with analog transducers. Each chip has four channels. We use ADC-U5 for the tip thermistor, board thermistor, and battery voltage. We use ADC-U7 for the analog pressure transducers. During execution of the program T2PLOGGR.RUN, the ADC chips are cycled so that only the chip in use is powered in order to to limit noise effects.

1.2.4. RS-232 SERIAL CABLE

For connection to a PC, the user must attach an RS-232 connector to the cable emanating from the 15-pin mdm connection. This RS-232 serial connector or a USB connection with an RS-232 adapter is the only connection capable of transmitting or receiving data from the CDAQ. The RS-232 cable is to be used for sending files, receiving files, and executing the provided software. The serial cable is best equipped to operate through the PC program Hyperterminal.

Hyperterminal is a standard windows program found in the communications section of the accessories drop down menu from the start menu (Section 1.3.3). Hyperterminal provides a “DOS”-like interface where the user is capable of typing commands and viewing program output. The serial cable must be connected to initiate the start of the provided software and to end the program. The program will continually run until the serial cable is connected, Hyperterminal is running and the correct command is provided.

1.2.5. VOLTAGE SOURCE

The intended voltage source for the CDAQ system is a 7.2V, 12AHr Lithium battery. However, the CDAQ has been proven to operate from a 9V battery, or a 9V DC-Regulated voltage source. At the start of the program, the battery voltage is checked, and a warning will prompt if the source is below 7.2V. The battery voltage dramatically and unpredictably decays at voltages below 7.2V. Data can be recovered upon loss of voltage, but we do not advise operation of batteries below this voltage threshold.

1.2.6. TRANSDUCERS

The logging program (T2PLOGGR.run) samples voltage data from (1) analog thermistor and (2) analog pressure transducers. The logging program utilizes analog-to-digital converters to convert voltage data to bit counts. The bit count data is written to an output file (Example output in Section 1.4), and later converted to pressure and temperature values via a post-processing program, T2PImport.xls.

1.3. OPERATION OF CDAQ

1.3.1. KEY COMPONENTS

- Persistor
- 9-Pin connection to voltage source
- 15-pin connection to serial cable and transducers
- Cable from 15-pin connection
- Circuit Board

1.3.2. ASSEMBLY OF COMPONENTS

1. Connect Persistor to double row 50-Pin connection on circuit board

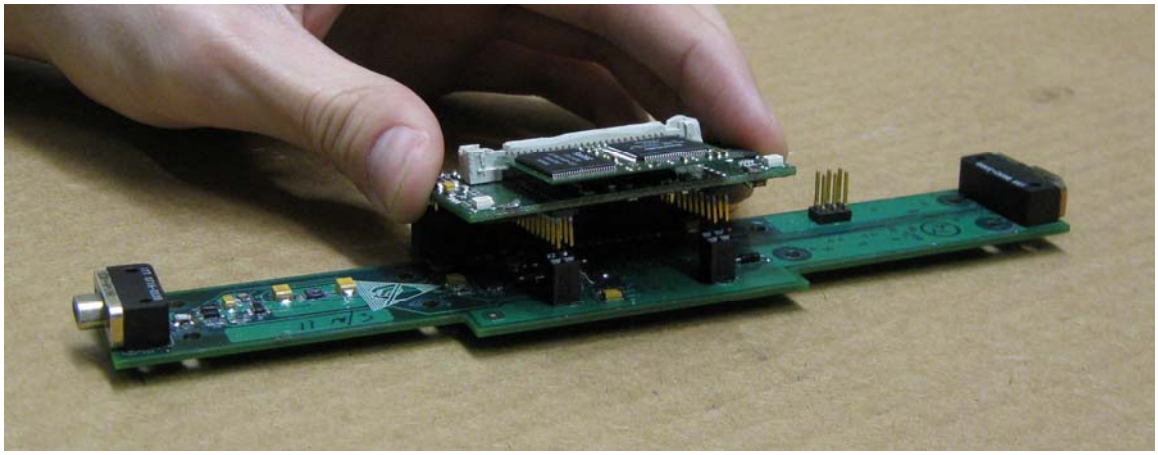


Figure 4: Persistor Insertion into CDAQ Circuit Board

2. Connect communication cable to 15-pin connection on circuit board
3. Connect RS-232 adapter on communication cable to computer
4. Connect transducers to desired sockets on cable
5. Connect voltage source to electrical outlet, if applicable
6. Connect 9-pin voltage source to circuit board, if applicable

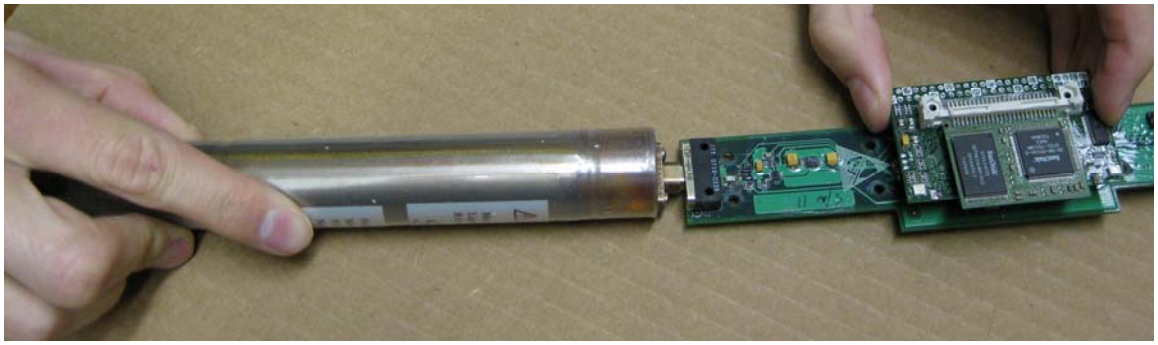


Figure 5: Battery Insertion into CDAQ Circuit Board

1.3.3. CONNECTION TO PC

(Please see Example in Section 1.5 for complete walk-through.)

1. Select Start->Accessories->Communications->Hyperterminal
    a. Provide connection name of your choice (this is not significant)
    b. Select - COM1
    c. Select Baud Rate – 9600
    d. Select Data – 8
    e. Select Parity – None
    f. Select Stop Bits – 1
    g. Select Flow Control – None
2. If screen appears as (1) or (2), program operation is normal.

    (1) C:\>

    (2)
    ```
    ----------------------------------------------------------------
     Persistor CF21M   SN 07458   PicoDOS V4.03r1     PBM V4.03
     (C) 1998-2007 Persistor Instruments Inc. - www.persistor.com
    ----------------------------------------------------------------
    C:\>
    ```
3. If program operation is not normal, start a new connection with a different COM selected.

1.3.4. HYPERTERMINAL COMMANDS

Hyperterminal is a command based interface mounted on all windows operating systems (with the exception of windows 7. See web tutorials for obtaining Hyperterminal). There are similar programs on linux and mac machines that can be downloaded. Additional information on the program can be found here: http://technet.microsoft.com/en-us/library/cc780754(WS.10).aspx. Hyperterminal when used with the Persistor utilizes the Pico-Dos command prompt (Fig. 6). For information on all the available commands, please reference a Persistor manual (www.persistor.com). The most important commands are listed below. The arrows "<>" contain user typed input. [ENTER] and [SPACE] represent hitting the ENTER and SPACE keys.

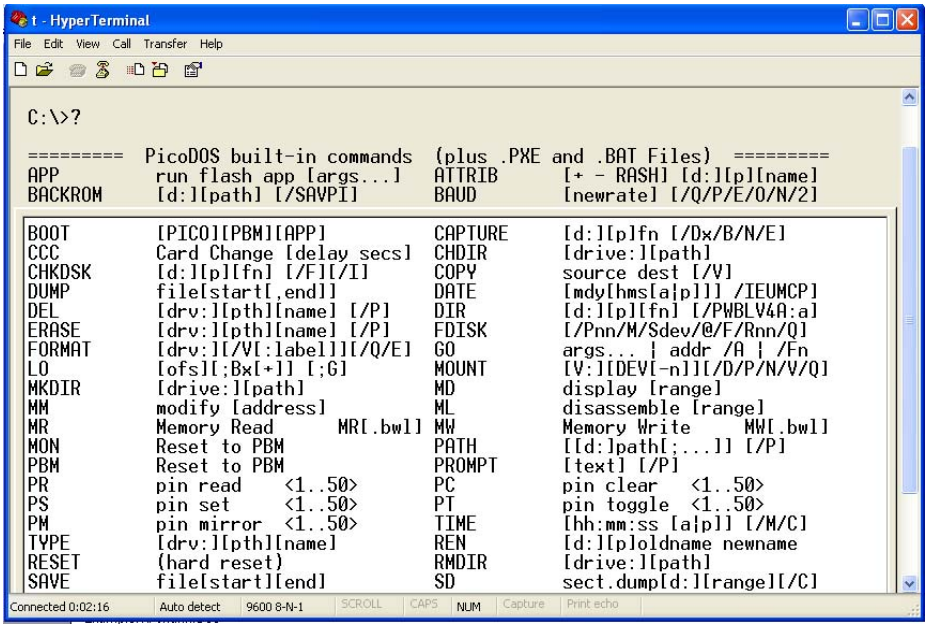


Figure 6: Command-Based PicoDos System

Command	Description
<DIR>[ENTER]	This command will bring up the directory of the flash memory on the Persistor. All transferred programs and data files should be visible with this command. The available and used memory should also be visible.
<DEL>[SPACE]<FILENAME>	This command will permanently delete the selected file from the Persistor memory.
<YS>[SPACE]<FILENAME>	This command will send a file on the Persistor to the PC. After typing this command, Select 'Transfer' then 'Receive File' in the top toolbar. From the dialog box, Select 'Browse' and choose where you want to place the received file. Select 'Ymodem' as receiving protocol, then select 'Receive'. A dialog box will prompt you for a filename then click OK. If you did not do this in the allocated time you will get a 'Failed' message and will have to repeat procedure, a little quicker next time. The file transfer dialog box will provide detail on file transfer and when successful, Persistor will send a 'Complete' message.
<YR>[ENTER]	This command will enable the Persistor to receive a file from the PC. This is the method for delivering executable files. After typing this command, Select 'Transfer' in the top toolbar, then select 'Send file'. In dialog box, select 'Browse', locate file on PC, select 'Ymodem' as protocol, and then click 'Send'. A Ymodem dialog box will open and display status of file transfer. A successful transfer will generate a 'Complete' response from Persistor. If you did not finish this process within the allocated time limit, then repeat the process, a little quicker this time.
<XXXX.RUN>[ENTER]	This command will execute a program present on the Persistor. Once an executable is initiated, the Persistor commands are no longer valid and the running program has all control.

1.4. LOGGING PROGRAM: T2PLOGGR.RUN

The logging program for the T2P described below retains much of the original IODP format. The changes made to the internal program are documented in Section 1.8, and a lengthy walk-through of the program is presented in Section 1.5. What follows below is a simple introduction to the program, and description of the available commands.

Once the program is initiated through Hyperterminal, a series of routines begin and the Persistor is capable of controlling all connected peripherals. The initial display is shown below.

1.4.1. PROGRAM START

```
Program: T2PLOGGR.c: Nov 13 2009 18:24:46
Persistor CF2 SN:7458   BIOS:4.3   PicoDOS:4.3
Enabling Accelerometers
The CF-2 date and time are: 01/15/21  01:11:42
Are these values correct  [Y] ? n
Enter date and time:  ? 11/13/2009 17:20:00
CF-2 time and date have been adjusted.
File Reference  ? Exp. 322
Output File ? Ex

CURRENT TIME AND DATE: 11/13/09  17:20:12
     PROGRAM VERSION: V1.0
LAST CALIBRATION DATE: 11/13/2009
      FILE REFERENCE: Exp. 322


Commands:
d :  Adjust date and time    e :  End Data Collection
s :  Start Data Collection   i :  Show system Information
p :  Play Back data          ? :  Show this command list
q :  Quit program
CMD:
```

1.4.2. USER COMMANDS

The program then allows for a set of user commands described below.

d: Adjust Date and Time.

You will be prompted for the new date and time setting for the CF-2 real time clock. You may press [Enter] if the displayed values are correct, or you may enter a new date and time in the following format. <mm/dd/yy>[SPACE]<HH:MM:SS>

s: Start Data Collection.

Acquisition will start and the program will display the file status and data in the following format:

```
Starting Data Collection.
Opened adtemp.dat
Opened axtemp.dat
CMD:

17:42:45.04, 1327432.000   25.8    9.7 236914.000 140570.000
17:42:46.04, 1327215.000   25.8    9.7 234061.000 134140.000
17:42:47.04, 1327179.000   25.8    9.7 231393.000 127028.000
```

During collection the program displays the time, tip bit count, board temperature, battery voltage, and bit counts on two channels of U7.

e: End Data Collection.

This command stops data collection and closes the temporary files. If this command is not properly performed, all data from logging session is unavailable.

i: Show System Information

This will show you some information about the system.

CURRENT TIME AND DATE: 11/16/07 16:30:08
PROGRAM VERSION: V1.00
FILE REFERENCE: Site 10TN-3 Hole 1275A

q: Quit Program

This command will cease program operation and return to Pico-Dos.

?: Show this command list

1.4.3. PROGRAM OUTPUT

Once a file has been sent to the PC, it is an ordinary text file capable of data manipulation or importation into various software. A typical output file looks like this:

```
ACQUISITION DATE:,,, 11/16/07
 CF-2 SERIAL NUMBER:,,, 7458
 PROGRAM VERSION:,,, V1.0
 LAST CALIBRATION DATE:,,, 11/16/2009
 FILE REFERENCE:,,, Site 10TN-3 Hole 1275A


 GMT TIME, Tip Value, Brd Temp, Batt Volts, U7-Ch. 1, U7-Ch. 2,
 HH:MM:SS  , Counts,  deg C , Volts ,  Counts,   Counts,
 17:42:45.04, 1327432.000   25.8    9.7 236914.000 140570.000
 17:42:46.04, 1327215.000   25.8    9.7 234061.000 134140.000
 17:42:47.04, 1327179.000   25.8    9.7 231393.000 127028.000
 17:42:48.04, 1327138.000   25.8    9.7 228105.000 119980.000
 17:42:49.04, 1327079.000   25.8    9.7 224933.000 113279.000
 17:42:50.04, 1326998.000   25.8    9.7 223267.000 106152.000
 17:42:51.04, 1326900.000   25.8    9.7 221015.000  98966.000
 17:42:52.04, 1327058.000   25.8    9.7 220025.000  91120.000
```

1.5. LOGGING PROGRAM: SLEEP.RUN

The logging program SLEEP.RUN functions identically to the previous program, T2PLOGGR.RUN with one significant difference. In this logging program, the user is given the option to initiate sampling from the ADC and also the accelerometer. If the user elects to sample from either of the two, they are then given the option to specify a sampling rate below 1 Hz (the standard rate). A sample section with the altered portions highlighted below.

If accelerometer is selected on, a series of similar statements will appear.

See 20.0 sec Sample Interval!

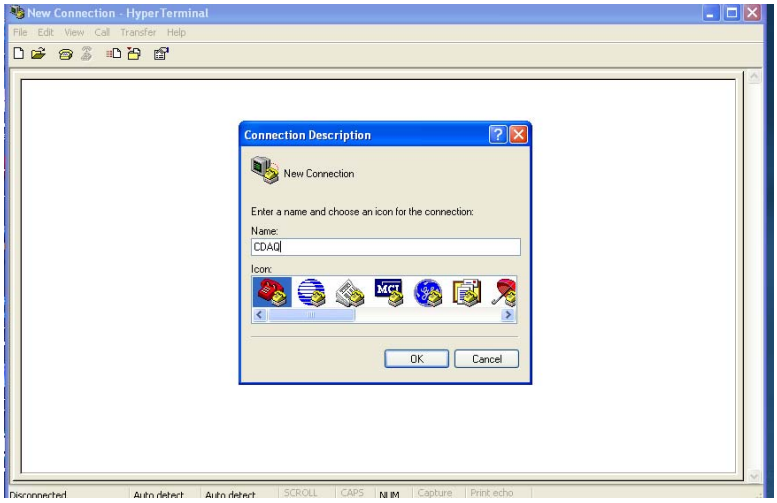
```
Program: SLEEP.c: Apr 30 2010 11:30:50
Persistor CF2 SN:7458   BIOS:4.3   PicoDOS:4.3
Enabling Accelerometers
The CF-2 date and time are: 04/30/10  11:32:06
Are these values correct  [Y] ? y
File Reference  ? none
Output File  ? test

Sample from Accelerometer  [Y] ? n

Sample from ADC  [Y] ? y
Analog to Digital converter regularly samples at 1Hz.
For longer sampling times, enter appropriate sampling rate.
What sampling rate ? .05

The accelerometer is not sampling.
The ADC interval period is  20.0 seconds.

There are 1021640704 bytes available on this device.
Therefore you have:
4077 days 10 hours 14 minutes and 24 seconds of logging time left.


CURRENT TIME AND DATE: 04/30/10  11:32:45
      PROGRAM VERSION: V1.0
       FILE REFERENCE: none


Commands:
d :  Adjust date and time    e :  End Data Collection
s :  Start Data Collection   i :  Show system Information
p :  Play Back data          ? :  Show this command list
q :  Quit program
CMD:
Starting Data Collection.

CMD: # 11:33:04.04     16777215   33.0 10.0    -258142    -248049
# 11:33 24 04     16777215   32.8 10.0    -234931    -224009
# 11:33 44 04     16777215   32.9 10.0    -211013    -201491
# 11:34 04 04     16777215   32.9 10.0    -190677    -184293
```

This program is intended to be run when the user is not interested in dynamic pressure effects and might be worried about memory usage. Such a situation might be a saturated probe sitting shipboard. The user might monitor the water pressure, looking for subtle changes, or perhaps battery decay.

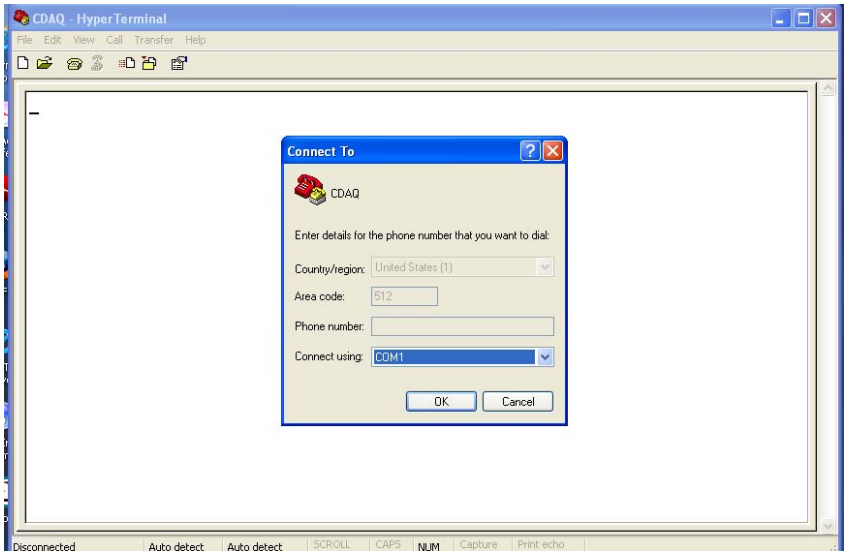
1.6. EXAMPLE OF LOGGING SESSION

The following set of steps is presented to provide a complete walkthrough of the data acquisition process. This example may be helpful for first time use, but does not provide any new information. All relevant information on the operation of the CDAQ system can be found elsewhere in this manual.

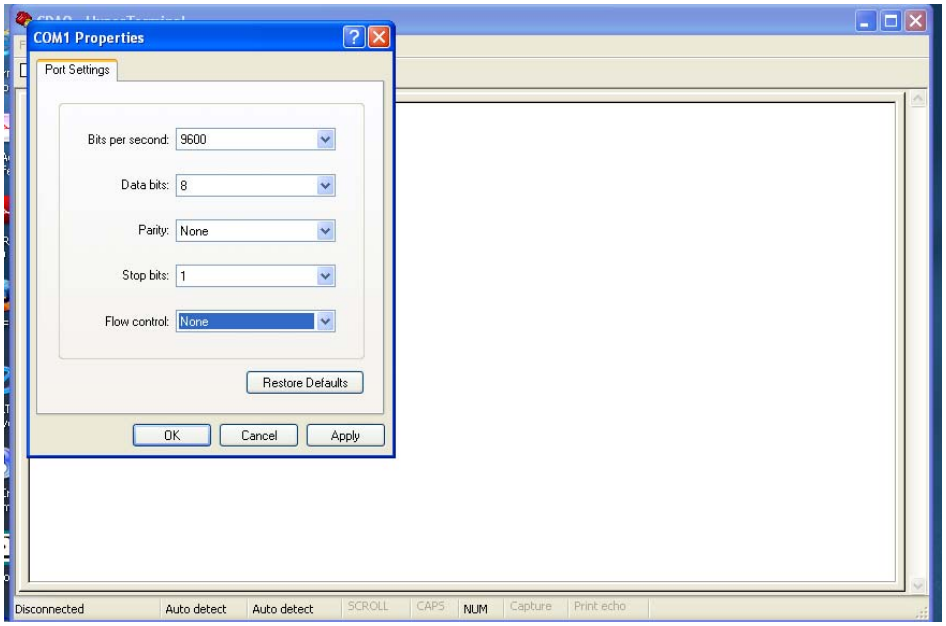
1. Open Hyperterminal from Start→All Programs→Communications→Hyperterminal. Then provide a name for the connection (this is completely arbitrary).



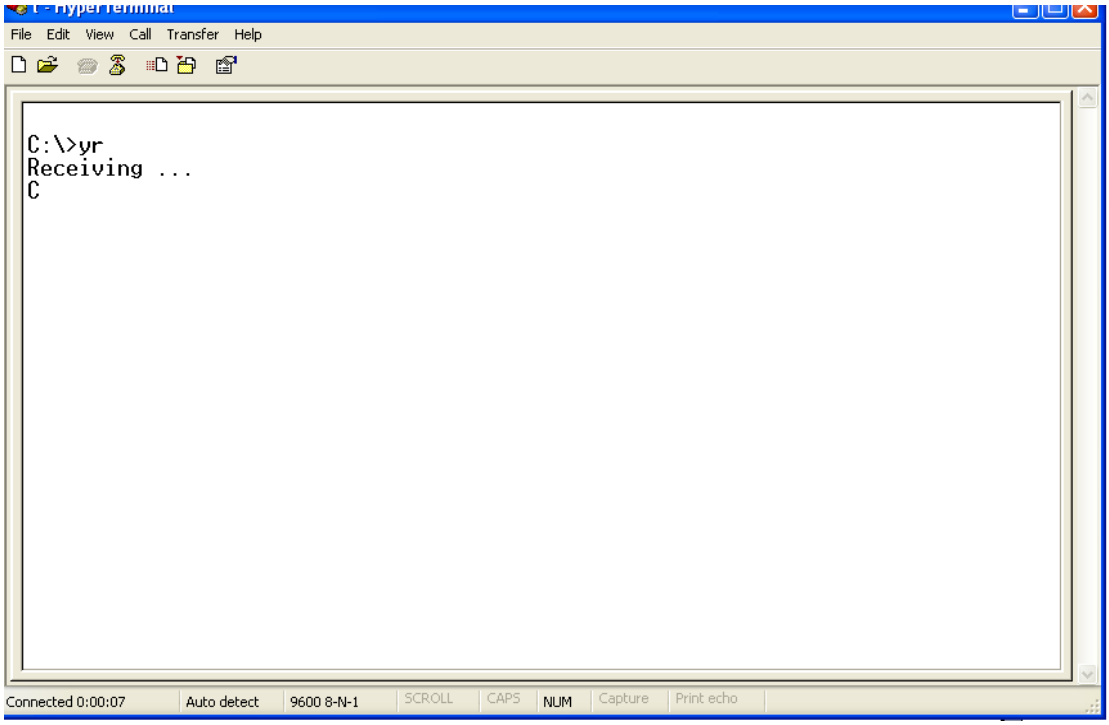
2. Select the port on the computer your CDAQ is attached to. This is typically COM1 by default, but may be different, especially if a serial to USB connector is being used.



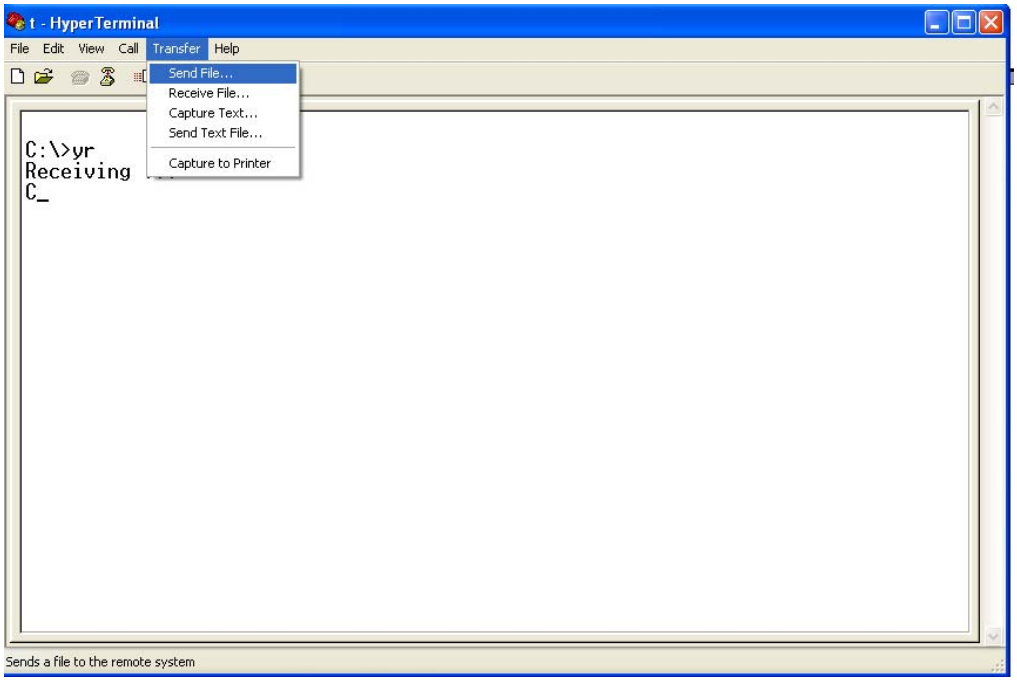
3. Apply the following connection settings.



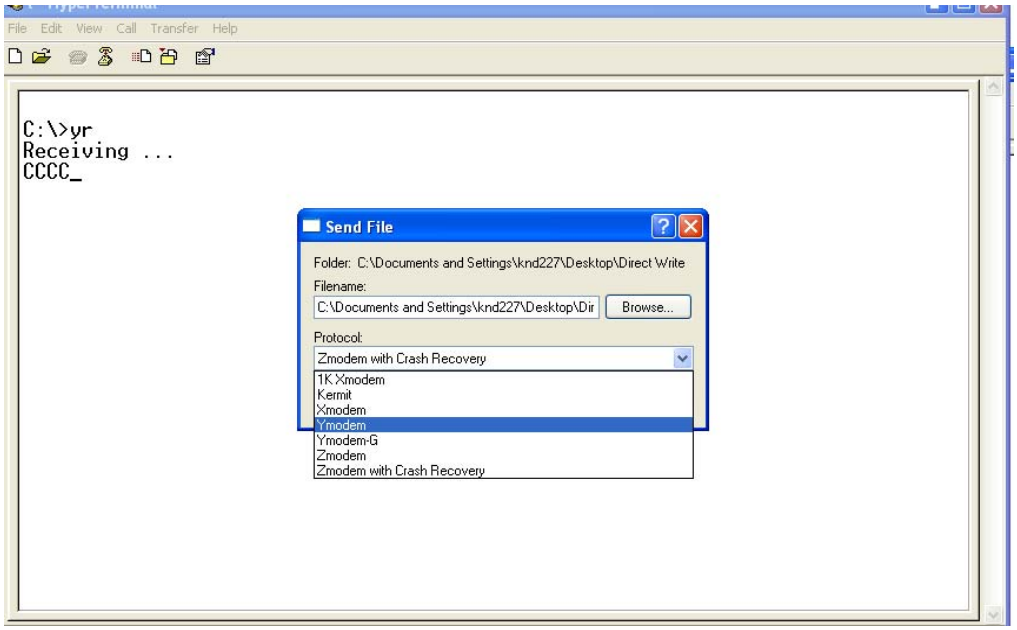
4. Persistor is operational when hitting [ENTER] returns the C:\> prompt.
   To upload logging program type <yr> to receive file on Persistor. Otherwise, proceed to step 7.



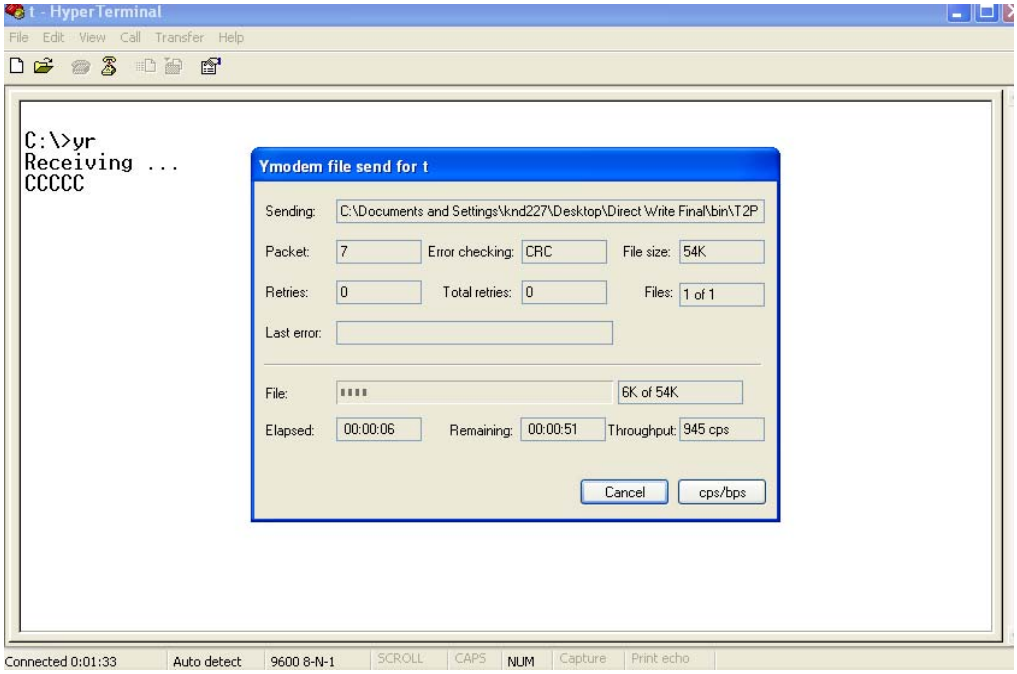
5. The file is ready to be received, but still requires action form the user. Once the transaction has started, click on the transfer toolbar, and then Send File from the pull down menu.



6. Browse to the desired destination for the executable and select “Ymodem” as the receiving protocol.

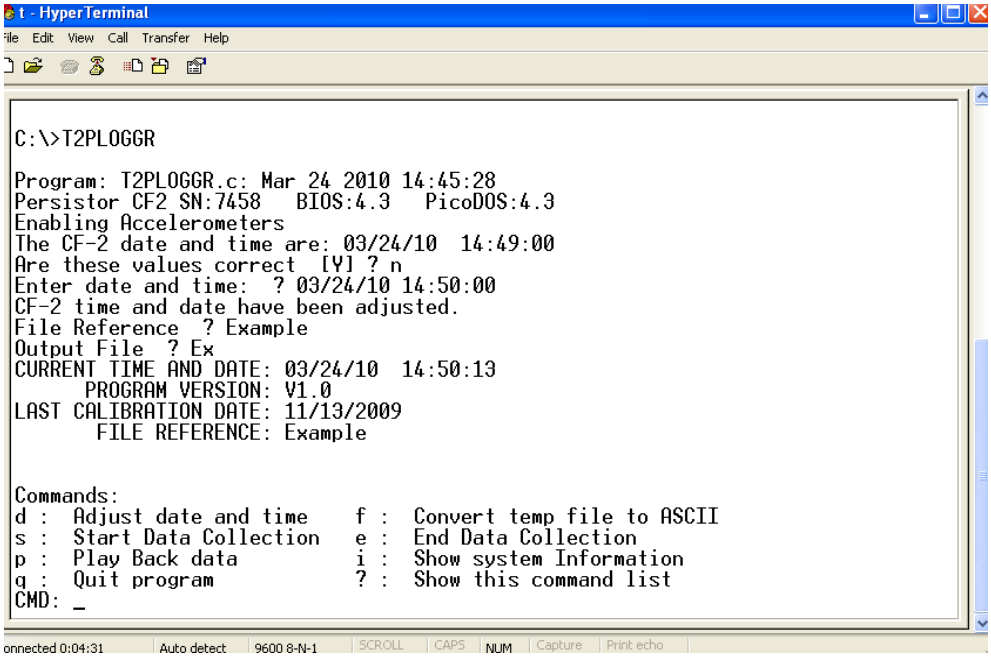


6. A dialog will pop-up displaying the progress of the transaction.

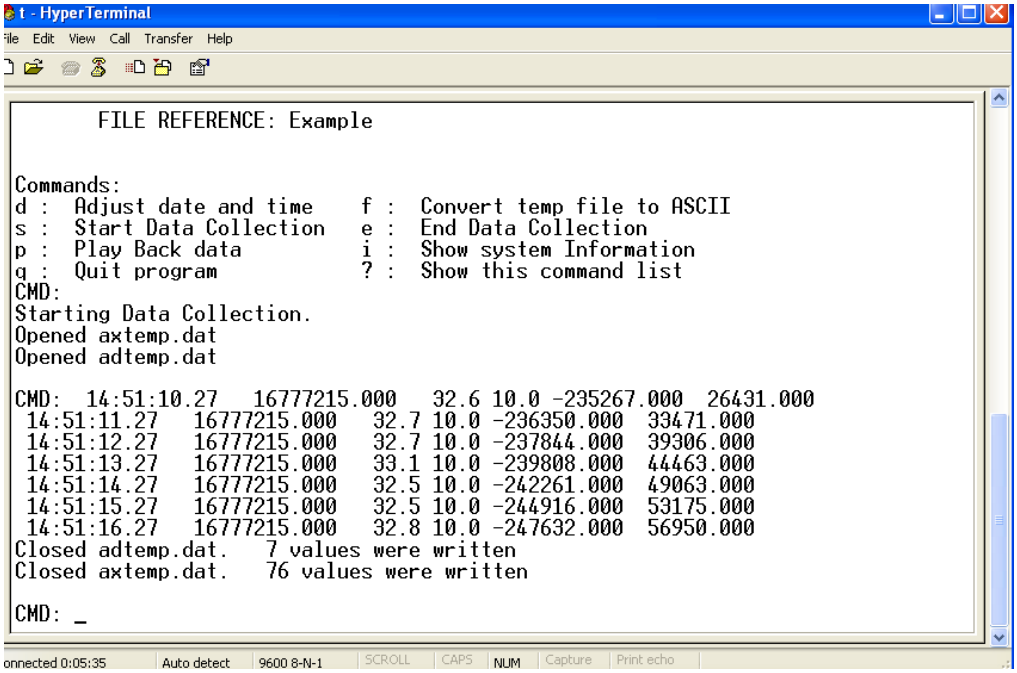


7. The program for logging data is then entered <T2PLOGGR> and the program is initiated.
   Type <N> if a new time must be established. Type the new time and date in this form: <mm/dd/yy>[SPACE]<HH:MM:SS>
   Type a File Reference of your choosing. This file reference will be displayed on the output file, and may be used for determining the contents of a file or specifying a description of the subsequent file.
   Type an output file name of your choosing (limit to 7 characters).

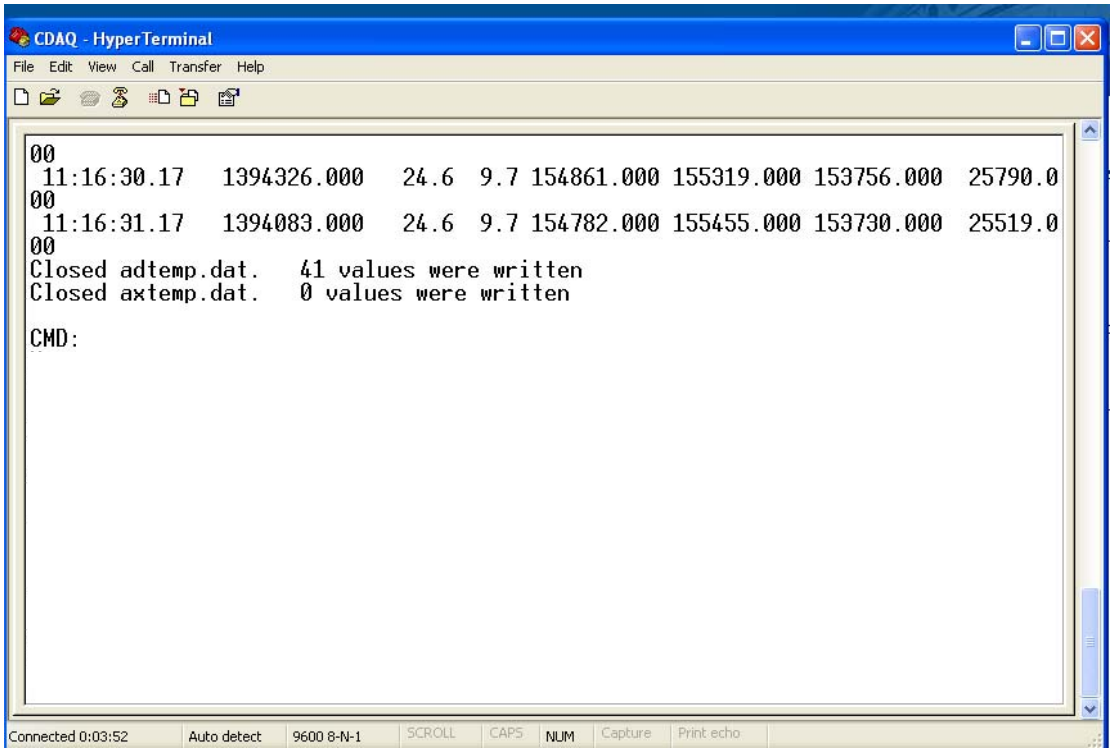
Warning: If you make a mistake, hit [BACKSPACE]. Use of the arrow key, will crash T2PLOGGR.run and return user to PICO-DOS. If T2PLOGGR.run crashes, simply restart it.



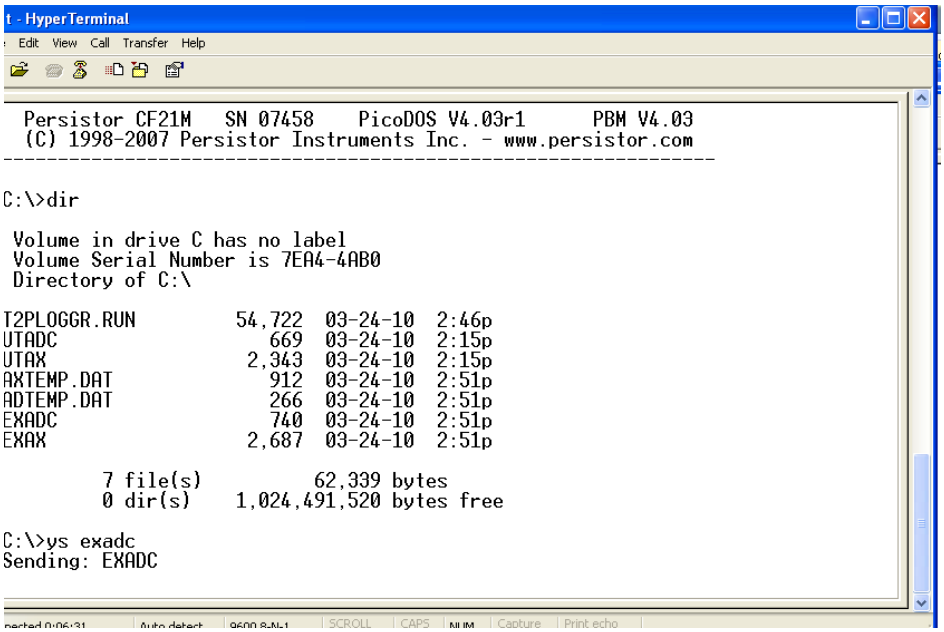
8. The <S> command starts data logging.
   The column headers are removed, but will appear in the data file.
   Once logging starts two files are created of the name given with suffixes x and d.
   The x file contains accelerometer data and the d file contains all else.



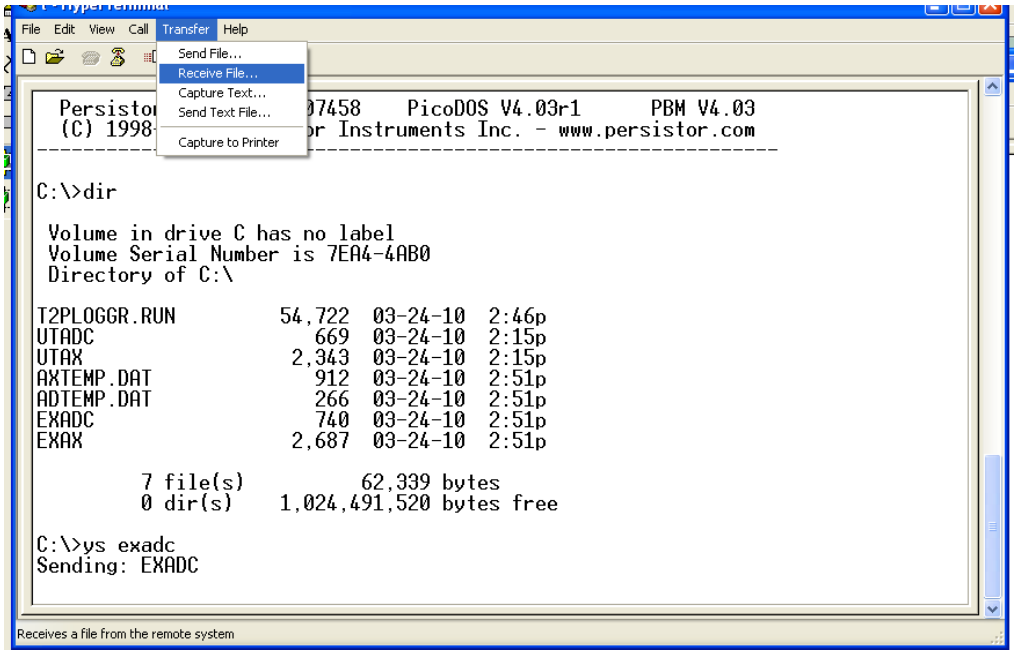
9. The <E> command will stop the logging session.



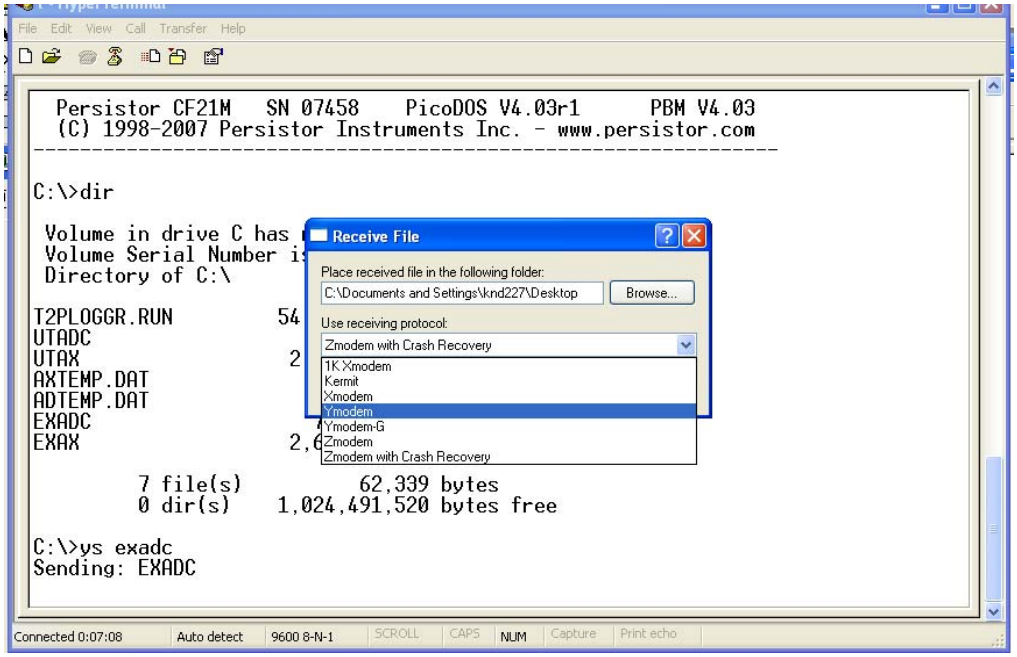
10. A <Q> command will end the logging program and return to the Persistor main screen.
From this point, the logging program can be restarted or the created files can be sent to the host computer.
To re-run T2PLOGGR, proceed to step 4.
To display all the files on the disk type <DIR>
To send the created file, type <YS>[SPACE]<Name of File>



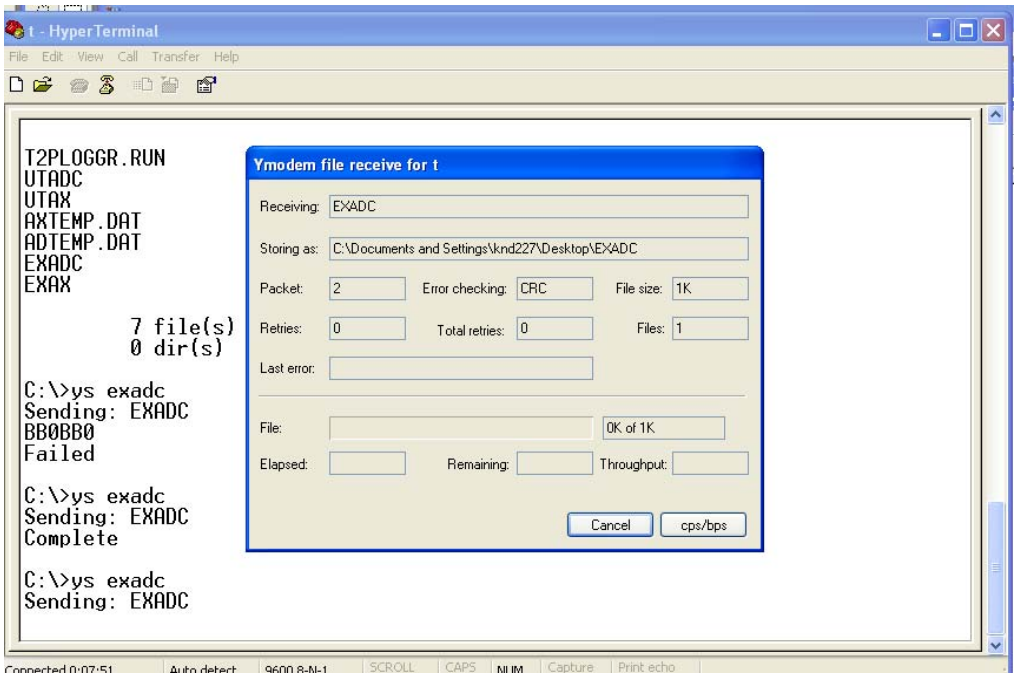
11. The file is ready to be sent, but still requires action form the user. Once the transaction has started, click on the transfer toolbar, and then Receive File from the pull down menu.



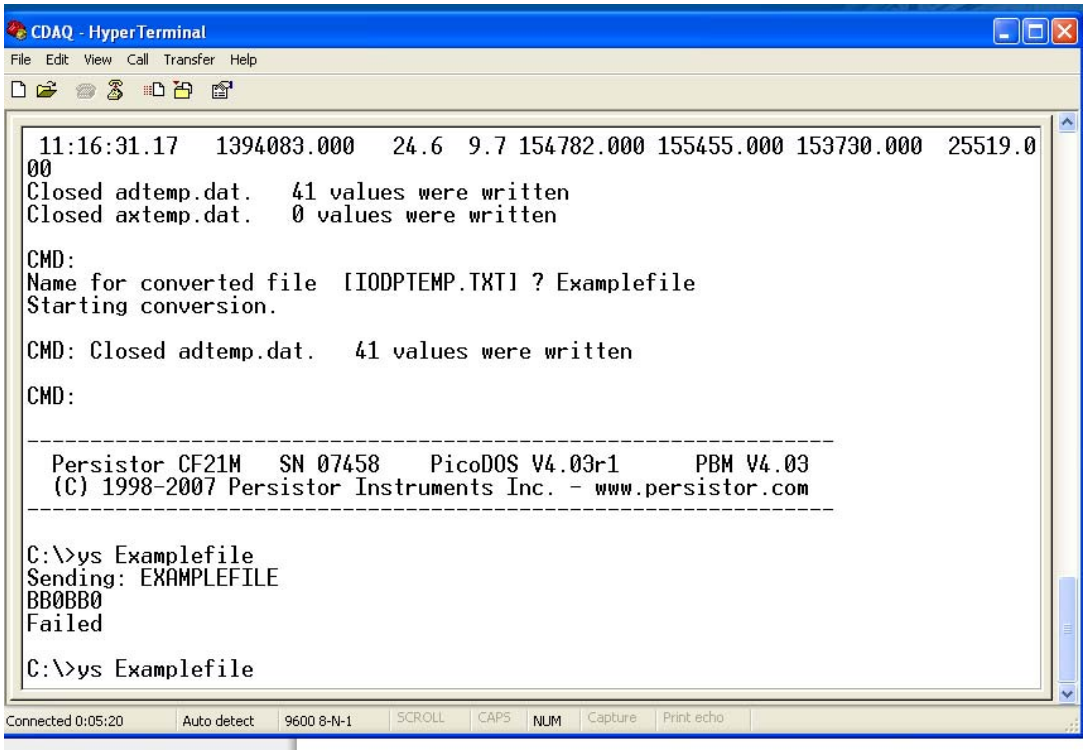
12. Browse to the desired destination for the sent file and select “Ymodem” as the receiving protocol.



13. A dialog will pop-up displaying the progress of the transaction.



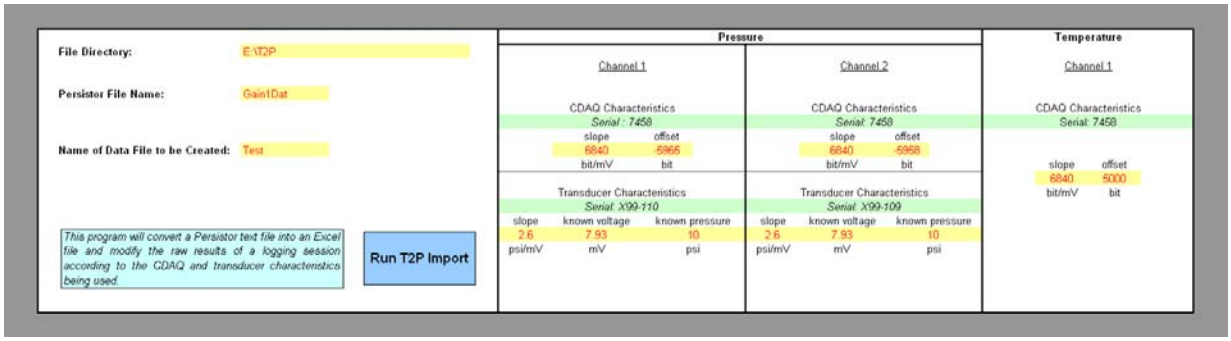
14. If the entire clicking process is not completed during the allotted time window, the transaction will cease and a failed message will be displayed. Please return to step 9 and perform the process quicker.



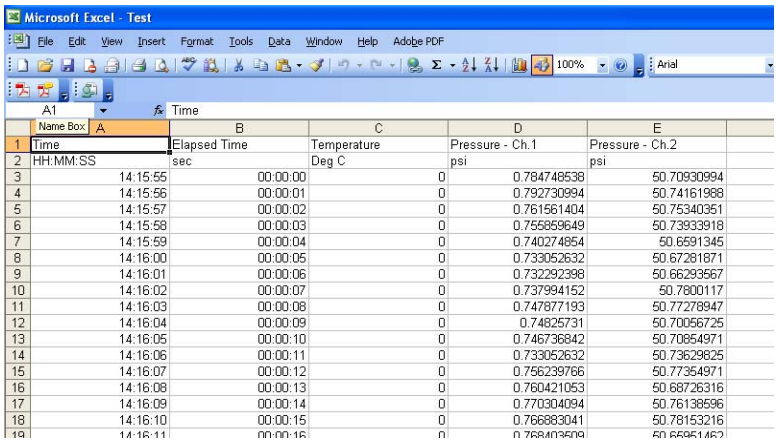
15. The CDAQ COM cable can be removed from the host computer or the CDAQ at any time. It will neither harm the logging session, nor terminate the session. The CDAQ will cease logging if the power is disconnected, but logging to this point will be retained. It is safest to disconnect power at the PICO-DOS screen or when the user sees: “C:\>”.

16. To view the file, first navigate to the directory specified during the hyperterminal transmission. Then open the files using a simple text editing program like Wordpad.

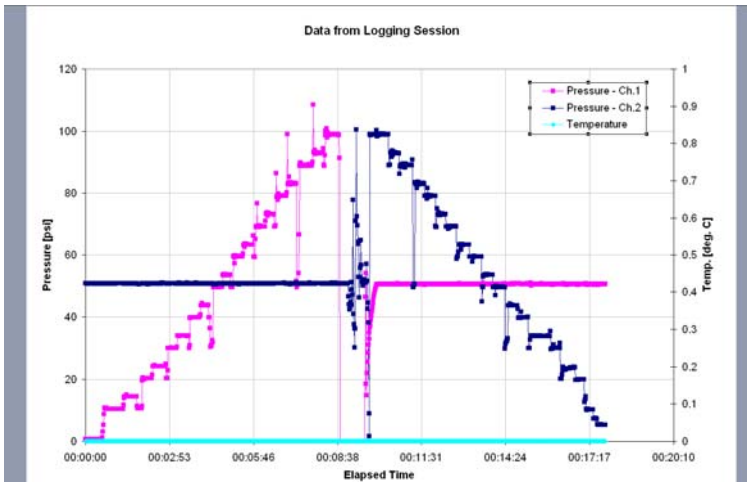
17. Once the file is on the host computer, it can be manipulated or processed like any standard data file. However, we have developed a simple script to convert the data into an excel file. The file is called T2PImport.xls, and is shown below. The script allows for adjustment of calibrations. The user must direct the script to the appropriate directory, provide the file name, and the name of the new excel file. The excel file will be created in the same directory.



18. Typical spreadsheet and graphic output is shown below.



	A	B	C	D	E
1	Time	Elapsed Time	Temperature	Pressure - Ch.1	Pressure - Ch.2
2	HH:MM:SS	sec	Deg C	psi	psi
3	14:15:55	00:00:00	0	0.784748538	50.70930994
4	14:15:56	00:00:01	0	0.792730994	50.74161988
5	14:15:57	00:00:02	0	0.761561404	50.75340351
6	14:15:58	00:00:03	0	0.755859649	50.73933918
7	14:15:59	00:00:04	0	0.740274854	50.6591345
8	14:16:00	00:00:05	0	0.733052632	50.67281871
9	14:16:01	00:00:06	0	0.732292398	50.66293567
10	14:16:02	00:00:07	0	0.737994152	50.7800117
11	14:16:03	00:00:08	0	0.747877193	50.77278947
12	14:16:04	00:00:09	0	0.74825731	50.70056725
13	14:16:05	00:00:10	0	0.746736842	50.70854971
14	14:16:06	00:00:11	0	0.733052632	50.73629825
15	14:16:07	00:00:12	0	0.756239766	50.77354971
16	14:16:08	00:00:13	0	0.760421053	50.68726316
17	14:16:09	00:00:14	0	0.770304094	50.76138596
18	14:16:10	00:00:15	0	0.766883041	50.78153216
19	14:16:11	00:00:16	0	0.768403509	50.66951462



1.7. CALIBRATION

Calibration of the CDAQ is composed of three parts. There is an internal calibration of the analog-to-digital converters, an internal calibration on the CDAQ bit count output, and an external calibration on the transducers used. Each type of calibration will be addressed separately.

Calibration of internal components is an isolated activity and can only be performed as a coding exercise within the built program (T2PLOGGR.run). However, the remaining two calibration activities could be combined into one activity. We choose to calibrate voltage to bit and then voltage to psi to maintain a general flexibility with CDAQ's, Channels, and transducers. As an alternative, one could calibrate psi to bit for a particular channel on a specific CDAQ. This combined method produces similar results as the direct method. The difference is primarily in the offset, with the example below being a 5 psi difference or 5% over the 100 psi range. This difference can be avoided with zeroing exercises before deployment like the response check.

The case is presented for the Transducer # 7648 below:

Calibration Equation with CDAQ connected (CDAQ#14 in Ch.1), converting Bit Count to PSI (Direct Method):

PSI = 3.5112e-3*Bit + 79.4

Calibration Equations as separate components (Combined Method):

PSI = 12.28*mV + 65.7 (Transducer) AND Bit = 3427.81*mV – 2699.87 (CDAQ Bit Ouput)

=> PSI = 3.5836e-3*Bit + 75.4

1.7.1. CALIBRATION OF INTERNAL COMPONENTS

The logging program, T2PLOGGR.run is devoid of gains and calibration constants. Optional gain constants represent the fractionation of a 2.5 V reference voltage (Vref) used explicitly as part of the analog to digital converters. Implementation of gain would force the Vref to approach the sampling voltage range of the transducer being used. This is typically performed to reduce noise effects. This gain is not implemented, and so the voltage difference on different channels is analyzed identically regardless of the transducer being used.

1.7.2. CALIBRATION OF CDAQ BIT OUTPUT

Pressure Channels

The CDAQ calibration was performed in the following manner. A particular voltage was passed through the channel using a known resistor, monitored on a high precision multimeter, and then, both, the resulting on-screen bit count and multimeter voltage was recorded. After maintaining the voltage for twenty seconds, the voltage was adjusted and the same steps were implemented. The twenty readings over the twenty second window were averaged as a single value. A total of ten values were obtained. We performed a linear regression on the data according to EQ-1.

Bit Count = Voltage [mV] * Slope + Offset [EQ – 1]

Temperature Channel

Calibration of the thermistor-dedicated channel differed in only one significant way. Instead of monitoring the voltage passed through the channel, the resistance was monitored. But the circuit for ADC-U5 is current ratiometric and thus depends on a constant current. For this reason, numerous electrical leads were used to ensure the circuit was always closed during a resistance change. A linear regression was again performed on the data.

Bit Count = Resistance [kOhms] * Slope + Offset [EQ – 2]

An alternative method of CDAQ calibration is used by IODP. The method involves a box and software interface for instituting strict linear models at the Analog to Digital converter level.

Table 1: CDAQ Calibrations

CDAQ Serial #	ADC Channel #	Slope		Offset		R^2
11	U7-1	3418.01	Bit/mV	-2876.29	Bit	0.99999995
11	U7-2	3418.54	Bit/mV	-2856.57	Bit	0.99999996
11	U5-2	122195.288	Bit/kOhms	-1722.17	Bit	0.999998
13	U7-1	3427.96	Bit/mV	-1755.59	Bit	0.999991
13	U7-2	3428.274	Bit/mV	-1730.4	Bit	0.99999
13	U5-2	121997.08	Bit/kOhms	-1474.9	Bit	0.999997
14	U7-1	3427.81	Bit/mV	-2699.87	Bit	0.99999998
14	U7-2	3428.35	Bit/mV	-2693.04	Bit	0.99999998
14	U5-2	122275.63	Bit/kOhms	-2543.69	Bit	0.999998

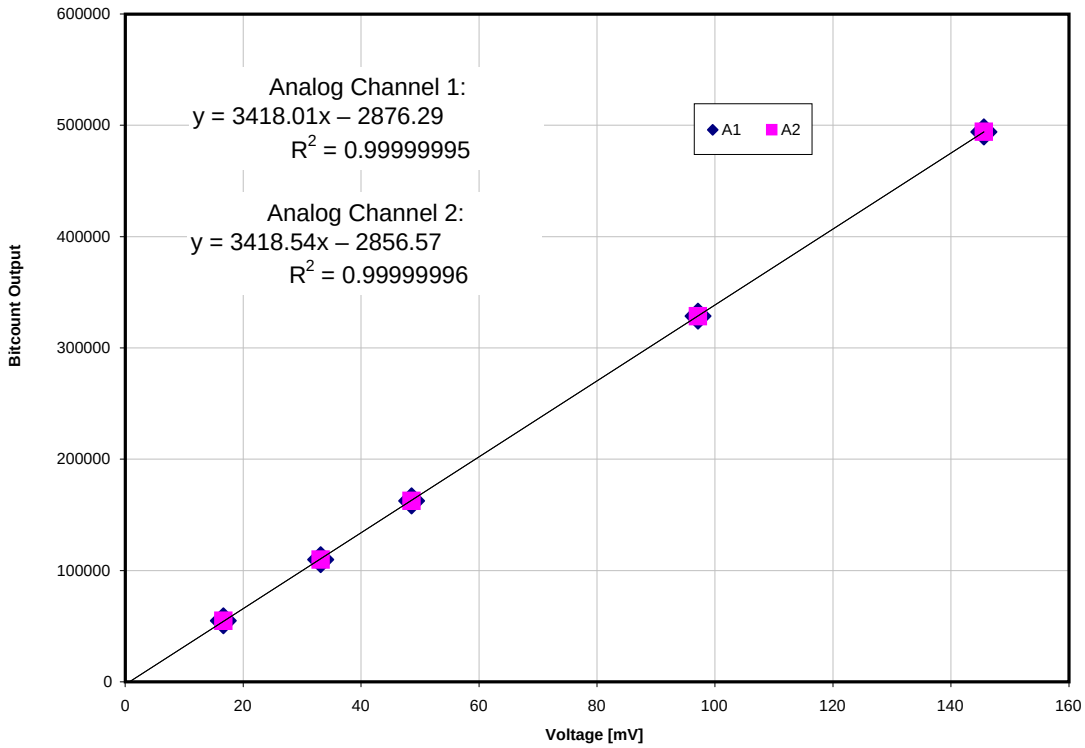


Figure 7: Voltage to Bit Count Mapping for ADC – U7 (S/N #11)
Channels are labeled A1-A2 to denote analog.

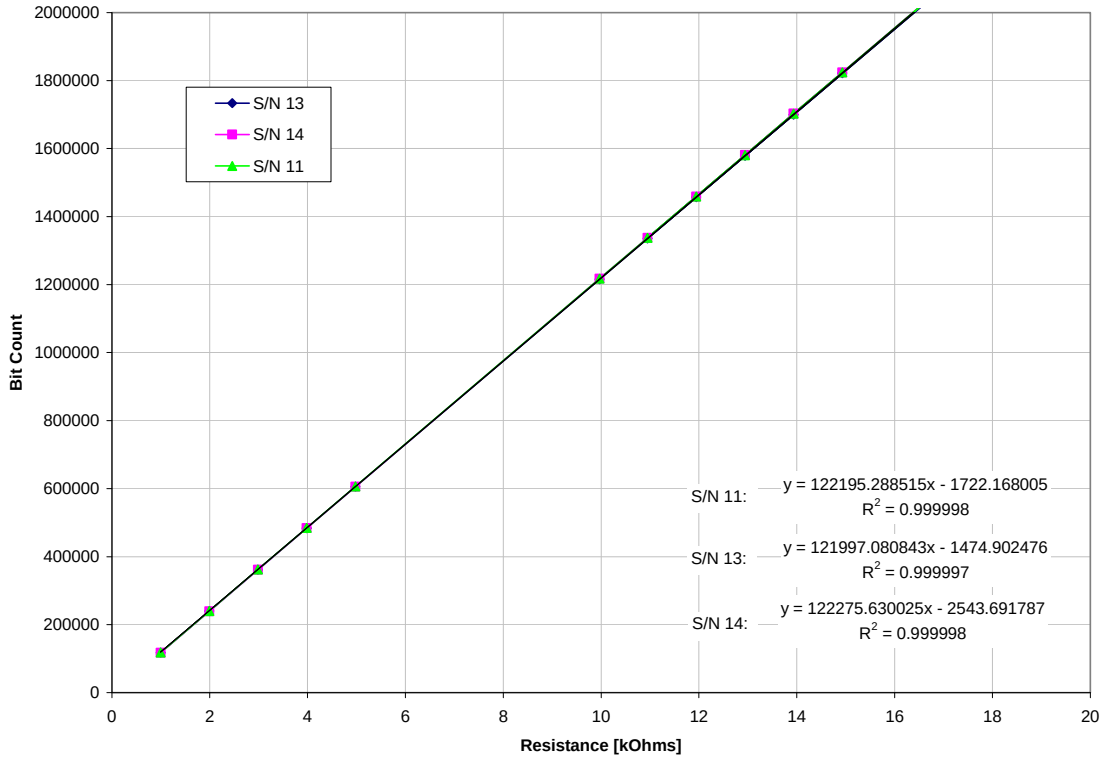


Figure 8: Resistance to Bit Count mapping for Channel 2 of ADC – U5

1.7.3. CALIBRATION OF TRANSDUCERS

The last calibration step in the data acquisition process is to develop a conversion from voltage/resistance to a scientific unit of measurement (PSI, Deg. C). This calibration step will be unique for every transducer used and the preference of the user. The user should consult the manufacturer of the transducer in operation, but as an example we present our pressure transducer calibration process.

For this task, a dead weight tester was used. A separate data acquisition system was (http://www.ametekcalibration.com/UK/Products/Pressure_calibrators.aspx?ProductID=PROD108&m=ecom_catalog) connected to the transducer in question and the transducer was physically connected to the dead weight tester. The dead weight tester had sets of weights that in turn produced a pressure on the system. This pressure was recorded and the voltage output from the data acquisition system was recorded. Then additional weight was applied to the system creating a new pressure and the resulting voltage was recorded. This process occurred from 0psi-100psi (for a 100 psi transducer) in increments of 10 psi ascending and descending. A linear regression analysis was performed on each transducer. It is recommended that this task be performed with any new transducer to be used with the CDAQ.

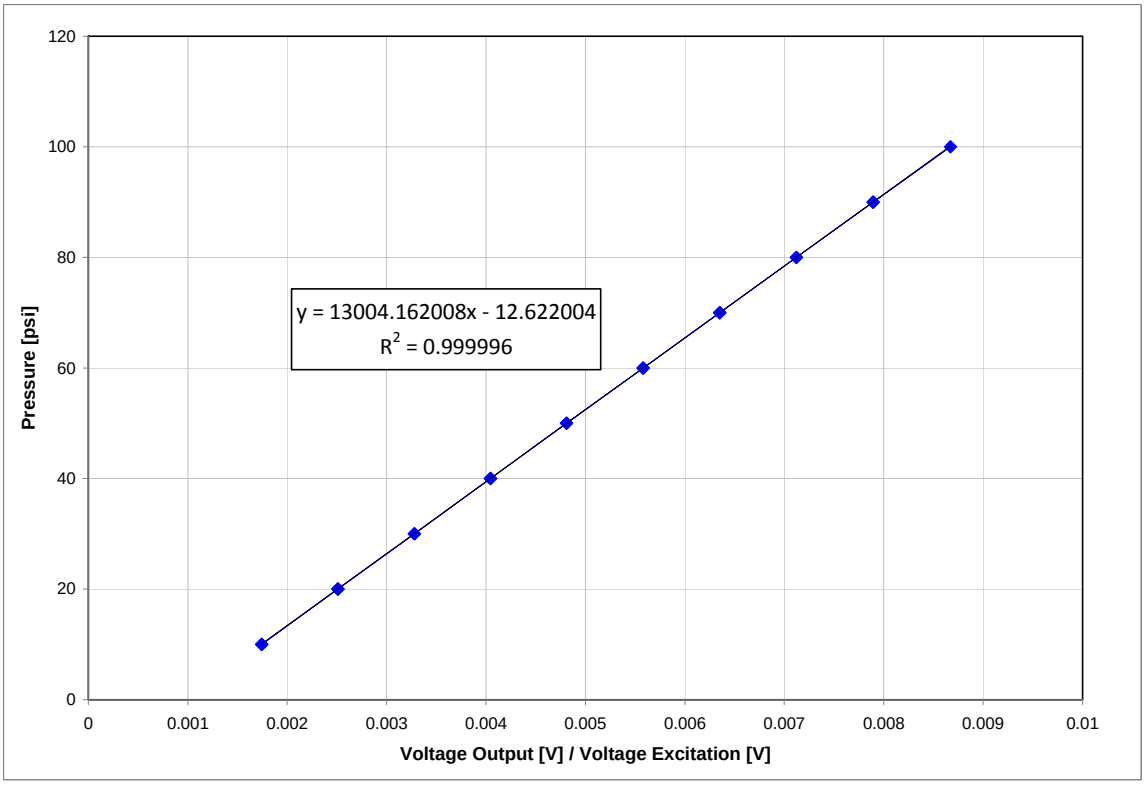
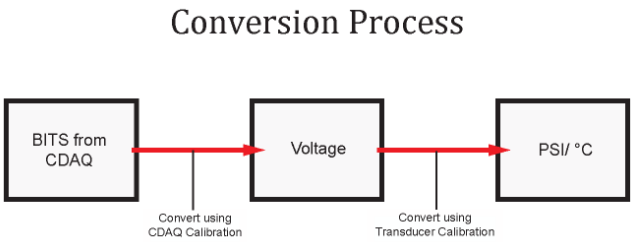


Figure 9: Example Calibration for Pressure Transducer X82-71

1.8. POST-PROCESSING OF DATA



To accomplish the final conversions of data to scientific units of measurement (PSI, Deg. C), an excel script was created (T2PImport.xls). The script takes a space and comma delimited text file produced from the logging program (T2PLOGGR.run), then uses the calibration factors determined in Sections 1.6 to output temperature and pressure data. The file version of the data is displayed in a tabular and graphical form as part of the excel file. Other forms of data processing could be performed on the data file; the user need only determine the calibration factors to perform a successful conversion.

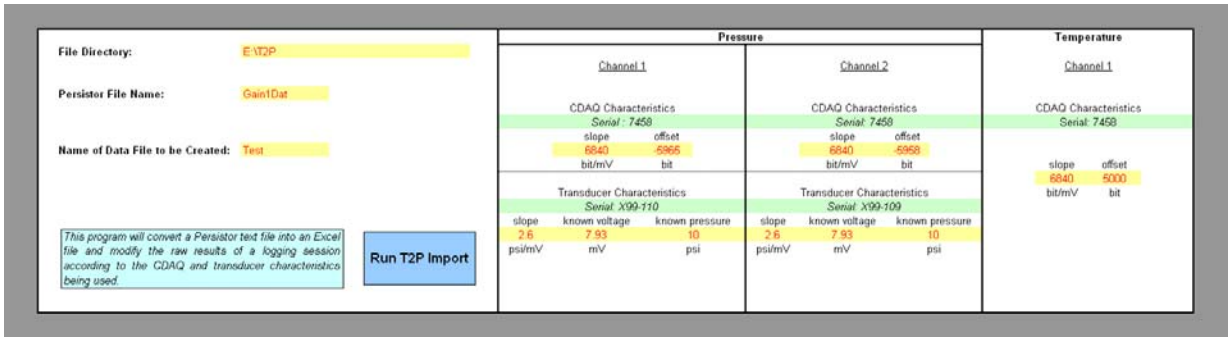


Figure 10: Screen Shot of T2P Post-Processing Script

1.9. ELECTRICAL MODIFICATIONS

According to the original IODP electrical schematic (Appendix II), the 15-pin mdm (Micro-miniature D Metal Sub connector) connection contained wiring for a dedicated thermistor on Ch.2-U5 and a standard channel from each of the ADCs (Ch.3-U5, Ch.1-U7). The remaining two channels on ADC U5 were unavailable for use, as they were already dedicated to the monitoring of the circuit board temperature and the input battery voltage. The remaining three channels on the ADC U7 could be made available through connection on the 6-pin header of the circuit board (Fig. 3).

Sampling of the analog pressure transducers is performed by ADC U7. Space constraints on the inside of the T2P dictated that the 6-pin header not be utilized. Instead, the circuit was rewired so that Ch.2-U7 was accessible from the 15-pin mdm connection instead of the original Ch.3-U5.

The rewiring was accomplished by removal of the resistors, R16 and R19 (Fig. 11). Then, a jumper wire was soldered as to connect pins from the 6-pin header to the circuit going to the 15-pin mdm to Ch.3-U5 (Fig. 11). The combination of these steps effectively stopped the circuit from Ch.3-U5 to the mdm cable, and then re-routed the circuit from Ch.2 of the 6-pin header to the mdm 15-pin connection. These modifications enabled sampling from channels (1) and (2) of the ADC U7 with a single connection from the 15-pin mdm.

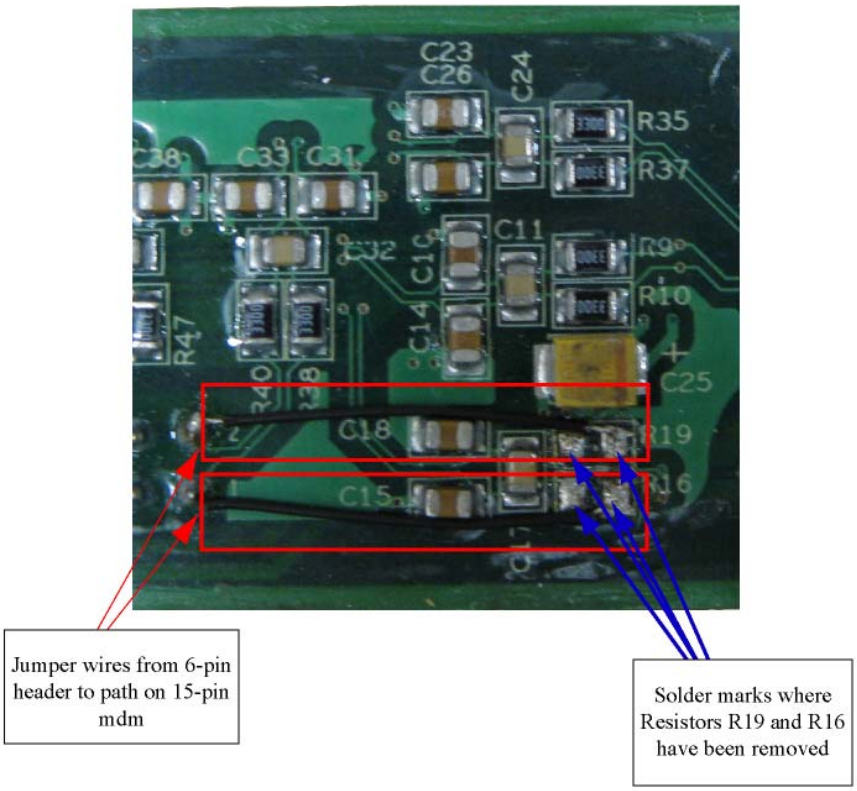


Figure 11: Backside of CDAQ Circuit Board showing Electrical Modifications

1.10. TEMPERATURE EFFECTS

In previous tools, temperature effects have played a role in the reliability of data. The lowering of temperature is believed to result in altered electrical characteristics and known to induce mechanical shrinking. As a means of understanding the specific temperature effects of the CDAQ a series of experiments were performed.

The experiment was designed to mimic the downhole environment of the tool. Therefore, a constant pressure was applied to a transducer by a long tubing system. The tubing, the transducer, and the CDAQ were all placed in a refrigerated room for a period of 15 minutes as the temperature approached 5 °C. The assembly was then removed from the refrigerator and allowed to reach room temperature. Several other experiments were performed, but this particular example characterized real conditions most approximately. Results of that experiment are presented below as Figure 12.

As determination of the temperature dependence, a linear regression was performed on the tip temperature versus the tip pressure. The tip pressure recorded a constant pressure, so any pressure changes should be a direct result of the temperature differences. The pressure dependence appears to very mild. It could easily be ignored and not significantly harm any data analysis.

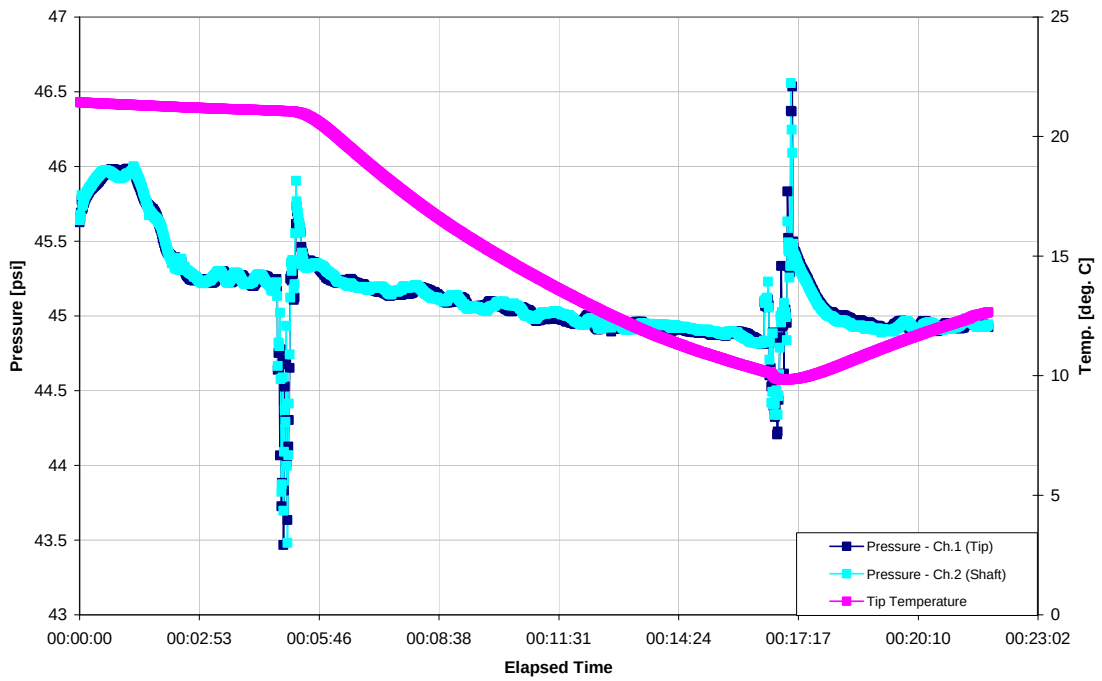


Figure 12: Data from Temperature Experiment

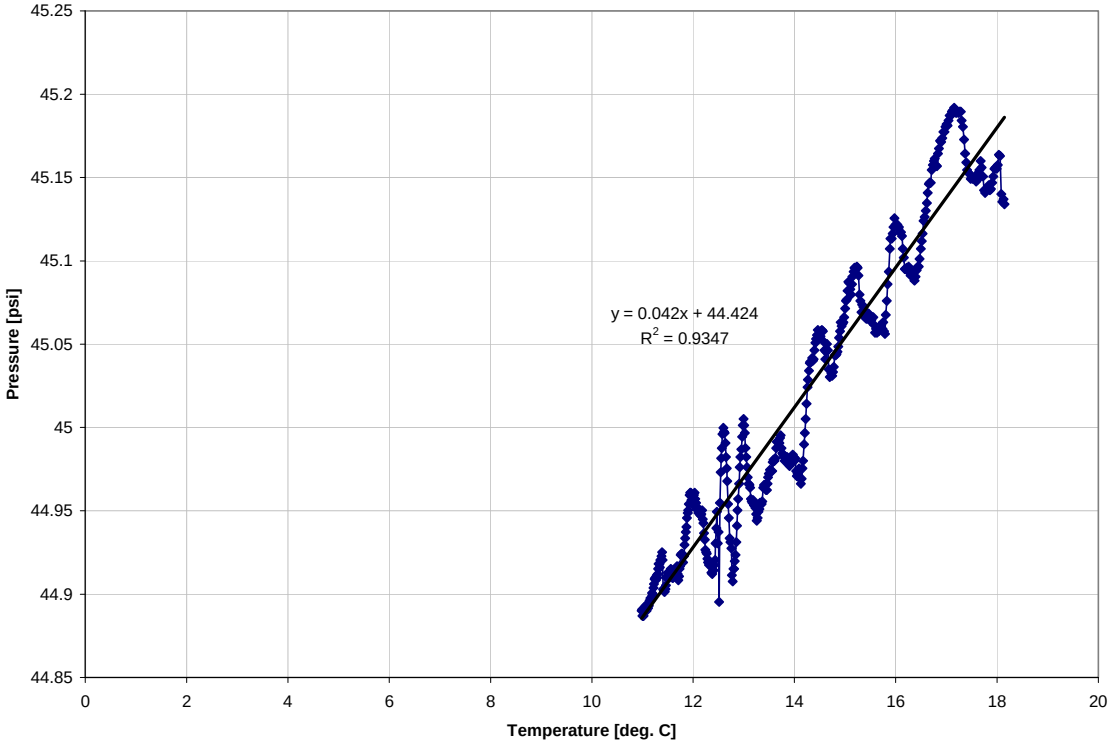


Figure 13: Temperature Dependence of CDAQ

1.11. SOFTWARE MODIFICATIONS TO IODP1D.RUN

The logging program, IODP1D.run, originally packaged with the CDAQ is available to users on our website http://www.ig.utexas.edu/resources/downhole/tech.htm. Program operation of T2PLOGGR.run differs from IODP by initiation of the second ADC (U7). During a data collection routine, data is transferred from ADC-U5, then ADC-U5 is powered down, ADC-U7 is powered up and data is transferred. ADC-U7 is powered down, ADC-U5 is powered up and the routine loops. This loop was instituted according to the guidance of the USIO engineering team. The team determined that if both ADCs were operational simultaneously, they would cause excessive noise disturbances.

In addition, the program now directly writes sampled data to ASCII text files. The IODP program wrote data to a binary file that was stored locally in a temporary file and converted to an ASCII file at the user's desire. This modification reduces the time required to retrieve data and circumvens data loss as a result of power outage.

It should be noted that a time stamp is applied at the beginning of the loop, but data is physically collected several milliseconds after this time stamp. Therefore, "instantaneous data" on different channels truly represents different data at different times.

The described changes and the resulting program presented in this manual, T2PLOGGR.run were products of the PC application MetroWerks Codewarrior (www.freescale.com/codewarrior). For user's interested in editing the software, please reference this application.

Comments have been included throughout the source code, and we attach the original IODP program description as Appendix III. To edit software, install the Codewarrior package and apply all the correct settings as described by the persistor company (www.persistor.com). Then, find the source code folder and open the Codewarrior project file (IODP_acel.mcp). This project file then allows access to all the components of the source code. The separate components are shown in Table 2 along with descriptions and primary changes.

If the user desires to make additional edits to any of the codes, the executable program needs to be re-built. All built programs will be saved in the bin folder. It is then the user's responsibility to upload the newly created executable to the Persistor according to the steps outlined in steps 1-6 of Section 1.6.

Table 2: Source Code Listing

Name of File/Folder	Description	Primary Changes
CS5534lib.c	Source code for the commands of the Analog to Digital Converter. The Analog to Digital Converter responds directly to 32-bit and 8-bit hexadecimal inputs, and thus this file appears primarily as machine language.	New Hexadecimals were added to power down the Analog to Digital Converter.
CS5534lib.h	Header file for sharing of variables.	None
IODP1D.APP	Built program from the original IODP source code. The APP extension cannot be used with PICO-DOS.	None
IODP1D.RUN	Built program from the original IODP source code. The RUN extension files are those to be used with PICO-DOS.	None
IODP_accel.c	Source code for the Accelerometer routines.	None
IODP_accel.h	Header file for sharing of variables	None
IODP_Accel.mcp	Project File for access to source codes through Codewarrior.	None
IODP_Accel.old.mcp	Relic project file. Upon codewarrior this was renamed to watch the differences in Codewarrior versions accessed by different users.	None
IODP_ADC.c	Source code for the Analog to Digital Routines.	The timing sequence for sampling was adjusted to sample on the channels stated in this manual. The output statement was adjusted.
IODP_ADC.h	Header file for sharing variables	Several new variables were added.
IODP_CalConstant.h	Header file with stored values for use if user desires to internally calibrate Analog to Digital Converters.	None
IODP_Paro.c	Source code for Digital Pressure Transducer routine.	No longer appears in the main source code or sampling routine.
IOD_Paro.h	Header file for sharing files	None
T2PLOGGR.c OR SLEEP.c	Main source code. All other arguments are called from this code and accessed in their respective codes.	Initial output to the screen upon program initiation was altered and an additional function for the determination of available memory was added. Two files are created to store Program Output.
bin	Folder for storing built programs.	None

PART 2:
CONNECTION OPTIONS

The T2P will have various connection configurations depending on the situation. The variable conditions are designed to allow the user freedom to operate the tool with limited adjustment.

2.1. PRE-ASSEMBLY

During normal operation of the CDAQ, before it is housed within the body of the T2P, the user will likely use a testing cabling, where one end terminates in a 15-pin mdm connection and the other terminates into a DB-9 serial connector. This connection configuration is ideal for checking the characteristics of transducers, altering programs, and transferring data. The user will have read and write access through hyperterminal and all that is contained within section 1.5.

2.2. POST-ASSEMBLY

After the tool is assembled, and the CDAQ is no longer readily accessible, the user is no longer capable of making modification to the transducers. However, with the use of the "Assembly Cable", the user can connect to the CDAQ via the port on the T2P endcap. This cable allows full two-way connection, which means, the programs can still be started, stopped and downloaded.

2.3. DEPLOYMENT

When the tool is set to be deployed, the "Tether" is connected to the T2P endcap and two-way communication ceases. The user is then only capable of monitoring the logged data with use of a program like T2PPlot.m. This also means that T2PLOGGR.RUN cannot be stopped. It will require the "Tether" to terminate.

These three connection configurations should give the user functionality of the tool in a variety of modes. The "Tether" is receive only for both space savings, and to prevent unintentional termination. The user should ensure adequate battery power is available before the "Tether" is utilized.

PART 3:
DEPLOYMENT OF THE T2P

3.1. INTRODUCTION

The IODP downhole tools are generally deployed using the drill string and/or coring line in several possible modes: (a) lowered on the coring line to either seat in the bit or run beyond the bit in open hole, (b) lowered to the bit as part of a coring assembly, (c) dropped down the drill pipe under "free-fall" conditions and later retrieved using the coring line, (d) built into the drill string, or (e) installed for long times in the seafloor. Often, the use of a particular downhole tool or technology (perhaps a combination of tools) requires careful advance planning in terms of the bottom-hole assembly (BHA) run at the end of the drill pipe.

3.2. DEPLOYMENT OF T2P:

3.2.1. OVERVIEW

The T2P delivery and drive assembly interfaces with the drill string and is integrated into the IODP operational protocols. We will use the motion de-coupled hydraulic delivery system (MDHDS) to deploy the T2P. The drill string is first raised several meters off the bottom of the borehole. The probe is then lowered down the drill string in the extended configuration (stroke = ~3.3 m) and engages the bottomhole assembly (BHA). The drill string is then lowered to insert the probe into the foundation. Once the force of penetration exceeds the weight of the telescoping section of the CDS, the section retracts with little change in force until it engages the collet. With the section fully retracted, the weight of the drill string is used to force the probe into the foundation. The CDS has 53 kN (12,000 lb) driving force capacity controlled by a safety mechanism. Following penetration (controlled by lowering the string a specified distance), the drill string is raised 2 m to extend the telescope and decouple the probe from the drill string. This prevents drill string movement (due to ship heave) from disturbing the probe during dissipation.

3.2.2. GREASE PROTOCOL:

1. Make sure the threads and o-ring groove are clean before greasing;
2. Grease the three o-rings with silicone grease;
3. Cover the large diameter thread with Mylar and put o-ring into groove;
4. Remove the Mylar and grease the thread with SS-30 copper anti-seize lubricant;
5. Be careful not to spread the grease for threads onto the o-ring and o-ring groove;

3.2.3. ASSEMBLY PROCEDURE FOR THE T2P TIP AND DRIVE TUBE:

6. Select two pressure transducers and one thermister with attached needle assembly and record ID numbers.

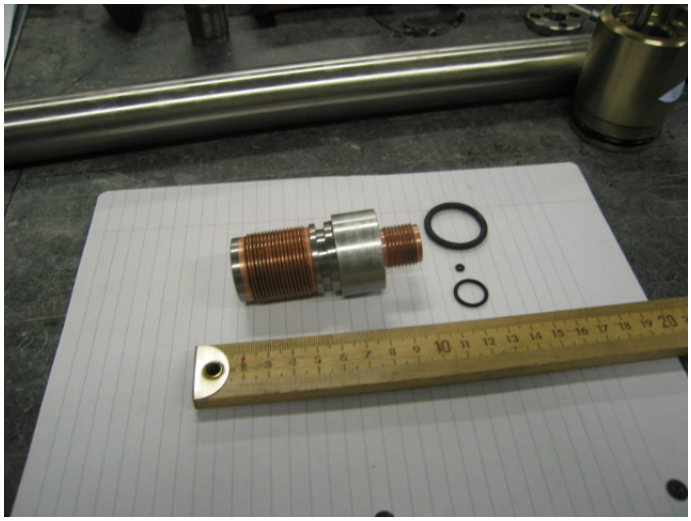


Figure 14: Transducer Block and O-Ring

7. Place transducer block, needle shaft, and 2 remaining transducer block o-rings in water.
8. Put a tooth brush into the water tank and deair the toothbrush;

9. Use squirt bottle to expel air from all conduits within the needle shaft and transducer block. Use brush to deair internal threads of needle and external threads of block.
10. Place small and medium o-rings in position at the base of the transducer block.
11. Place stone filters in a container of boiling water for twenty minutes.
12. Immediately after the twenty minute period, transfer stones to an ultrasound bath for 45 minutes.
13. Transfer filters to a sealed container completely filled with deaired and distilled water.
14. Open container with filter under water. Dump the saturated filter stones into water tank: never let the filter stones become exposed to air;
15. Put the big filter stone onto the shoulder of the needle shaft;

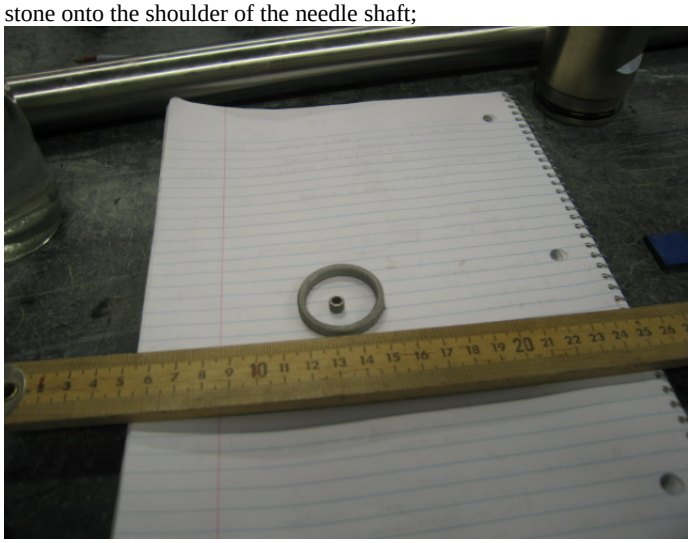


Figure 15: Filter Stones

16. Twist wire ends on thermister
17. Put the thermistor underwater. Deair the tip threads with toothbrush.

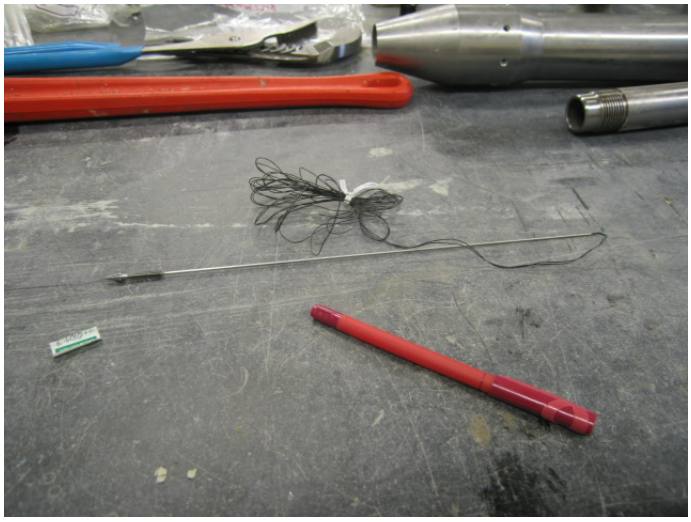


Figure 16: Thermistor Tip

18. Make sure thermistor wires are twisted together, then run the thermistor wires through the small filter. Slide small filter onto the base of the thermistor tip;
19. Put the thermistor wires through the hole at the bottom of the needle shaft under water, make sure the wires come out of the center hole at top of needle.

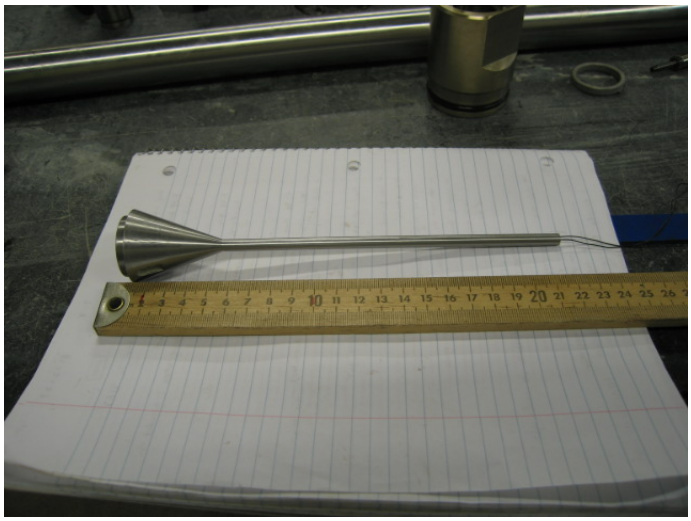


Figure 17: Needle Shaft

20. Screw the thermistor tip onto the needle by turning the needle shaft, tighten using the tip tool;

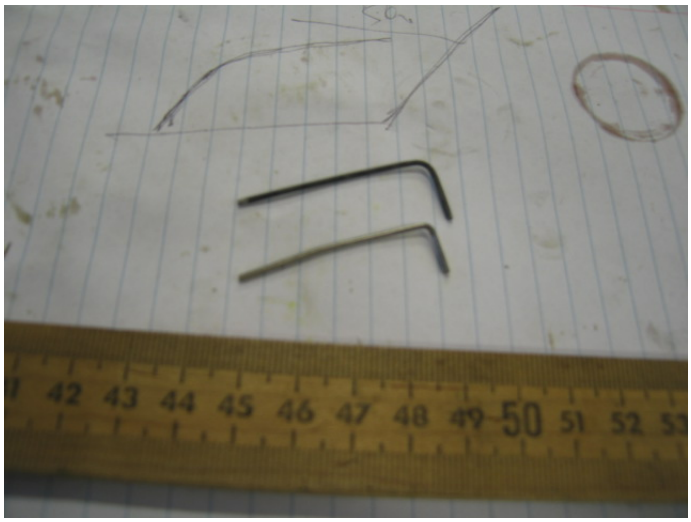


Figure 18: Tip Tool

21. Run thermistor wire and tube through center hole of transducer block;
22. Screw the transducer block into the needle shaft by rotating the block. Tighten connection using the spanner wrench and the needle shaft wrench;

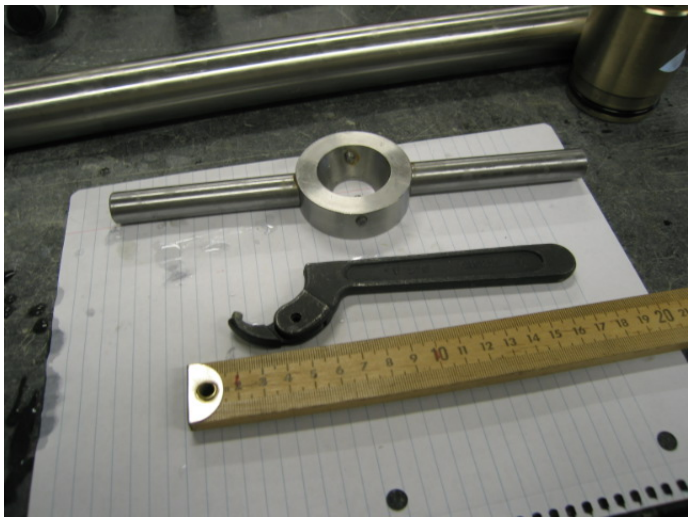


Figure 19: Spanner Wrench and Needle Shaft Wrench

23. Confirm that o-rings are on both transducers and always make sure one transducer has a positive zero (you really should deair the orings as for the transducer block, put the transducer in the water, deair the threads and then add the oring;
24. Deair the threads on the pressure transducers using brush (keep the fisher connector out of water).

Note: minimize the time the transducer back is exposed to water as this end is not sealed. Work quickly and keep the transducer as close to the surface as possible.

NOTE: To identify which transducer is connected to which filter: when the spanner wrench hole on the transducer block is facing you with the probe needle facing down, the pressure port on your right side is for the tip pressure transducer and the left pressure port is for the shaft pressure transducer;

25. Screw the pressure transducer onto the transducer block (be sure all the operations are done under water and not to twist the pressure transducer wires) Tighten transducers using a 3/8 inch box wrench:
26. Deair the seals for the filter stones and put them onto the tip and shaft filter stones underwater: rubber tube with plug for the tip, plastic sleeve for shaft;
27. Screw two hexagonal stand-offs to transducer block.
28. Screw T-shaped end bracket onto stand-offs using 8-32 screws.
29. Screw the tip labeled end of the crème-colored backing plate strip to the T-shaped end bracket using 4-40 socket cap screws.
30. Screw coiled cord bracket onto DAQ labeled end of the crème-colored backing plate strip using 4-40 socket cap screws.
31. Remove probe from water and solder each loose thermistor wire to one of the loose purple wires coming from the fisher plugs: first put on the right size heat shrink tubing, do the soldering work, then shrink the tubing to cover and protect the connections;
32. Slide needle probe needle through T-shaped bracket.
33. Slide the fisher sockets through the hole in the coiled cord bracket towards the probe tip.
34. Connect fisher plug + receptacles: Note the serial number of each transducer, and note which transducer is attached to channel A and channel B. Channel A corresponds to U7-A1, and channel B corresponds to U7-A2. Both fisher plugs are 5-pin, and thus interchangeable.

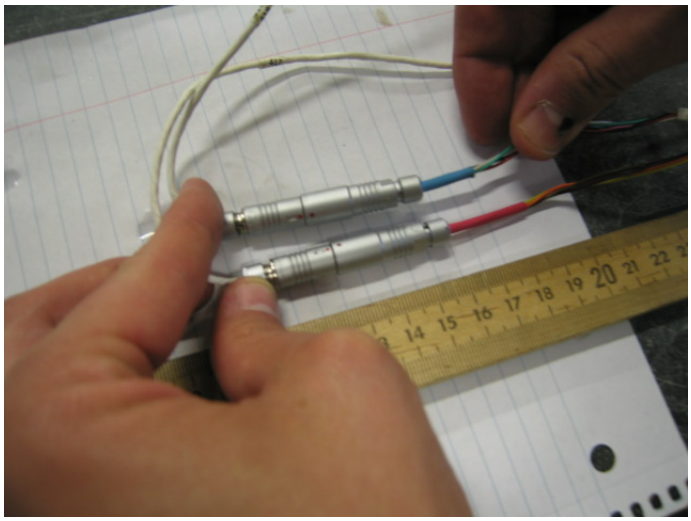


Figure 20: Fisher Connectors

35. Use electrical tape in conjunction with attached Velcro strips to mount the wire harness to the crème-colored backing plate strip.
36. Pass a draw string through the drive tube and tie it to the large (19-pin) fisher connector so that the fisher connector and coiled cord poke out.
37. Slide the drive tube over the crème-colored backing plate strip, while keeping the draw string in tension to stabilize the board and to pull the coiled cord from the end of drive tube.
38. Screw the drive tube to the transducer block by rotating drive tube (hand-tight) and maintaining tension on the string so that the fisher connector and coiled cord poke out.
39. Connect tubing to flow pump.
40. Insert tubing into assembly tank and deair tubing.
41. Place response chamber into assembly tank and deair response chamber. Connect tubing to response chamber. Then actuate valve so system is fully saturated and deaired.
42. Hold probe assembly vertical, insert probe into assembly tank, and remove sleeve and rubber seals from the porous discs. Then insert the probe into the chamber. Open the valve. (This may be accomplished with some ease if a support structure is implemented as shown below.)

Figure 21: Assembly tank with support structure and submerged probe

43. Insert the fisher retainer into the top of drive tube beneath the fisher connector to protect and position the wires;

Figure 22: Fisher Retainer

44. Properly clean and insert the two o-rings on the drive tube nut. Grease male threads on drive tube using copper anti-seize.

Figure 23: Drive Tube Nut with O-Ring

45. Screw the drive tube nut onto top of the drive tube and tighten it use square wrench on nut and needle shaft wrench on transducer block, while applying tension to coiled cord so that it sticks out of the drive tube nut

3.2.4. ASSEMBLY PROCEDURE FOR THE T2P THREADED UNION:

46. Grease and insert o-rings onto green tweed pass through.
47. Apply a light coating of copper anti-seize grease to the tip end of the threaded union
48. Screw ground nut to the fisher connector bulk head. Then slide fisher washer over bulk head, and secure fisher washer with the thin, flat nut.
49. Slide wire assembly through threaded union from tip end to DAQ end so that the 15-pin mdm connector and fisher connector poke out of the DAQ end of the threaded union.

50. Stand threaded union vertically on its DAQ end with clearance for 15-pin mdm connector so dangle. Then place green tweed insertion tool over green tweed connector. Gently tap polished end of tool until the body of the green tweed connector is completely situated in bore of union.

Figure 24: Threaded Union with dangling 15-pin mdm connection

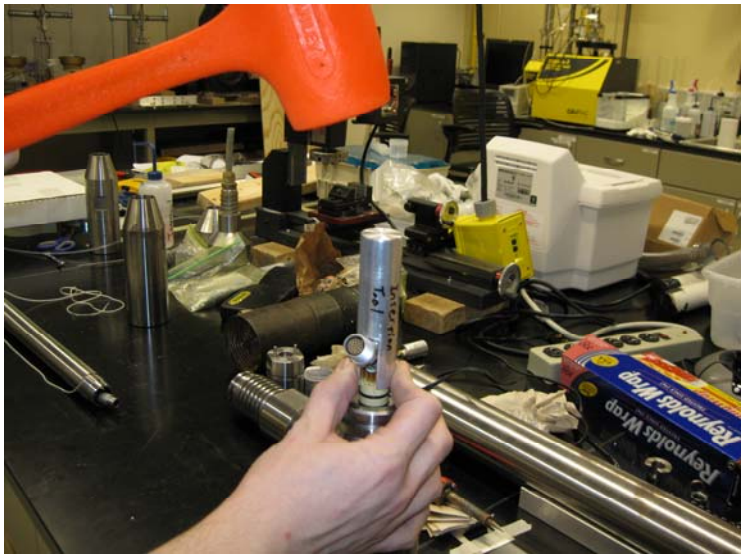


Figure 25: Green Tweed Connection Procedure

51. Place insertion tube and wires leading from green tweed to poke through to the fisher connector. Push the whole assembly into threaded union, until fisher washer is flush with the end of the threaded union.
52. Screw the fisher washer with 8-32 socket cap screws.
53. Screw the 2 ½” 8-32 long rods into the holes on the DAQ end of the threaded union.
54. Slide a 3/8” diameter standoff onto each threaded rod.
55. Slide 15-pin mdm bracket onto threaded rods. Set the 15-pin mdm connector into notch on the bracket. Clamp down on the 15-pin mdm connector by tightening the 4-40 socket cap screws.

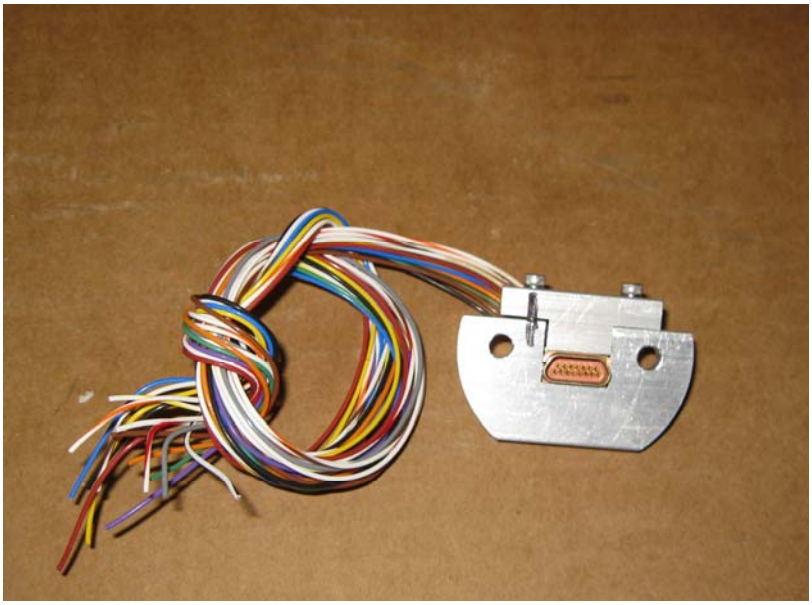


Figure 26: 15-Pin mdm Bracket

56. Secure bracket by screwing a pair of nuts onto threaded rods.
57. Grease o-ring grooves and threaded exterior of threaded union on both ends of the threaded union. Additionally, grease o-rings, o-ring grooves and exterior threads of end cap.

3.2.5. ATTACHING DRIVE TUBE TO THREADED UNION:

58. Screw locator pins into drive tube nut. Note: If tow sets of holes are present, use the set which is closest to the wrench flats
59. Carefully align drive tube nut and threaded union. Before joining the two, connect fisher plug from the drive tube nut to the fisher bulkhead connector on the threaded union.
60. Slide the spin collar over the drive tube and screw over the threaded union.
61. Rotate spin collar over the threaded union until snug tight.

3.2.6. ASSEMBLY PROCEDURES FOR THE DAQ HOUSING:

62. Place CDAQ clamshell housing onto rough surface with flat part of the aluminum backbone facing up.
63. Place CDAQ onto backbone with Persistor inserted into 50-pin header and 15-pin mdm connection facing away from the battery tube.
64. Align holes on CDAQ with holes in aluminum backbone. Place six standoffs over holes on CDAQ furthest from the 15-pin mdm connection. On the set of holes closest to the mdm connection, place aluminum header so that its holes align with those of the CDAQ
65. Place top half of the clamshell on top of standoffs with holes aligned and screw 1” 4-40 screws into each hole.
66. Slide 7.2 V Lithium Batttery into battery tube and ensure it is fully attached to the 9-pin connection on the CDAQ.
67. Position DAQ clamshell onto threaded union by sliding aluminum header through threaded rods on threaded union.
68. Secure aluminum header with one nut on each threaded rod.
69. Slide large spring over battery tube of clamshell.

70. Slide large white plastic shell over battery tube of clamshell.
71. Secure all wires with masking tape. Attempt to place all wires into the slit in the large white shell.
72. Tie a drawstring to end of wires headed out of probe.
73. Pass drawstring through DAQ housing, and then pull wires through.
74. Slide DAQ housing over CDAQ clamshell and screw onto threaded union.
75. Screw end cap onto DAQ housing, if ready for deployment.

3.2.7. RESPONSE CHECK PROCEDURES

1. Connect a witness transducer to the response chamber.
2. Connect witness transducer to a data acquisition system. Make note of the input voltage and the transducer characteristics.

Figure 27: Response chambers with Witness Transducer Port on End

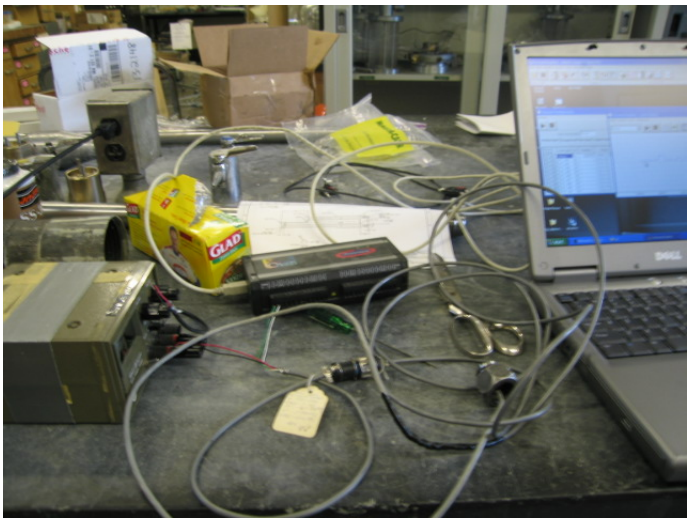


Figure 28: Data Acquisition system

3. The T2P should already be inserted into the response chamber. If not, insert the T2P partway into the chamber with the drainage valve open to allow water to vacate the chamber. Once the T2P is partly within the chamber, with clearance between the tip and the base of the tube, the drainage valve is then closed, thereby sealing the response chamber.
4. Start logging data from tip and shaft transducers at 1-second sampling rate.
5. Apply one or more 5-second pressure pulses to probe by pushing probe into chamber.
6. Stop data collection, save the response file (see file name convention).

3.2.8. DEPLOYMENT PROCEDURE:

1. specify the deployment depth, penetration rate, penetration distance and max load;
2. switch DC power to battery; Record battery number and time
3. Launch hyperterminal and start logging (see Part 1 of Manual);
4. assemble data acq housing as described above.
5. pressure response to make sure both transducers works;
6. Keep unit in AC as long as possible to reducer temperature variation
7. Driller lifts the CDS to the vertical position on the rig deck
8. Driller attach the T2P to the CDS using the quick connect;
9. take off the response chamber, lower tip into casing; record this time
10. Lower probe on wire line with slow fluid circulation (need to get stroke rate and wireline feed rate!!!!)
11. reference pressure reading at 500 meters water, stop circulation and hold the tool there for 30 seconds record time; (are we sure the pressure is Zero at the top of the drill string?)
12. reference pressure reading at 1000 meters water, stop circulation and hold the tool there for 30 seconds record time;
13. reference pressure reading at mudline, stop circulation and hold the tool there for 30 seconds record time;
14. Lower probe to the bottom of drill string and engage the BHA;
15. Mark drill string with reference intervals to track penetration.
16. record time and lower drill string required distance to retract the CDS at a reasonably fast rate (try 5 meters/minute)
17. record time and slow penetration rate to 2 meters/ minute to penetrate the probe to the required penetration. (Ideally I would not stop went the CDS is closed but use one continuous motion. Also try to pickup position on drill string vs time!!!)
18. Immediately lift drill string to activate the CDS
19. leave the probe there for 30 minutes (time to be decided on the spot);
20. pullout the probe;
21. Remove from the quick connect, clean the tip and attach the response chamber once the tool is out of casing;
22. have the drillers loosen the connections
23. pressure response check;
24. download data: the previous stability data file and the current deployment data file;
25. switch back to DC power; record time on battery log
26. start the next stability file (see file name convention);

3.2.9. NOTICE:

1. replace all the o-rings and grease all threads every time to re-assemble the tool;
2. do the above every time the vibration loosen the any threads;

3.3. TOOL BOX:

1. tip tool;
2. needle shaft wrench
3. spanner wrench;
4. square wrench ;
5. allen keys-hex key;
6. wire stripper;
7. screw driver;
8. lineman's pliers;
9. plastic hammer;
10. tooth brush;
11. round/triangle file;
12. tape: electronic and a duct tape;
13. knife;
14. scissors;
15. voltage meter;
16. tape measure;
17. heat shrink tubing
18. grease-lubraplate grease (peanut butter color grease, tons on ship) or copper coat grease (has copper flakes in it—for tool joint connection), they are good for stainless steel;
19. Desiccate Bags

3.4. T2P LOG SHEET FOR PROBE 1:

1. transducer A (tip): S50-73
2. transducer B (shaft): Z59-72
3. thermistor: 0509-3
4. board: Sn#4
5. Data acq electrical union (each has different resistor)

Log sheet:

Date & Time	Event	File name	Comments
06/01/05 18:10	Power up and start logging	Stab_1a	
06/02/05 22:57	Log stability data	Stab_1b	
06/03/05 7:21	Stop stab_1b		
06/03/05 7:26	Change time on CF2 to central time		
06/03/05 7:28	Log stability data	Stab_1c	Stopped by some unknown reason (power off?)
06/04/05 9:50	Power off		
06/04/05 11:15	Power on		
06/04/05 11:18	Log stability data	Stab_1d	
06/04/05 19:53	Log stability data	Stab_1e	
06/05/05 07:05	Power off then power up		
06/05/05 07:07	Log stability data	Stab_1f	
06/05/05 14:19	Power off and on		
06/05/05 14:24	Start stability log	Stab_1g	
06/05/05 23:30	Power off and on		
06/05/05 23:31	Start stability log	Stab_1h	
06/06/05 18:10	DC off		
06/06/05 18:28	Battery on		
06/07/05 01:07:50	Start log data	T2p_2	

T2P log sheet for Probe 2:

1. transducer A (tip): S50-75
2. transducer B (shaft): S50-74
3. thermistor: 0509-2
4. board: Sn#4
5. Data acq electrical union (each has different resistor)

Log sheet:

Date & Time	Event	File name	Comments
06/04/05 19:53	Start stability logging	Stab_2a	Probe2 has been fully assembled
06/05/05 07:05	Power off then power up		
06/05/05 07:10	Log stability data	Stab_2b	
06/05/05 14:19	Power off and on		
06/05/05 14:21	Start stability log	Stab_2c	
06/05/05 23:30	Power off and on		
06/05/05 23:32	Start stability log	Stab_2d	
06/06/05 18:10	DC off		
06/06/05 18:52	Battery on		
06/06/05 18:54	Start stability log	Stab_2e	
06/07/05 00:36:16	Battery off		
06/07/05 01:30:56	DC on		
06/07/05 01:33:10	Start stability log	Stab_2f	

3.5. T2P DEPLOYMENT DATA SHEET:

Hole:
Water depth:
Probe:
Filename:
Date:

Penetration rate:
Penetration distance:
Circulation rate:

Operations	Time/depth	comments
Get the depth after APC coring		
Switch DC power to battery	DC power off: Battery on:	
Start Logging		File name
Pressure response to make sure both transducers works		
Rig up probe/CDS		
Remove the chamber, lower tip into the casing		
Reference pressure reading At 500 meters		Stop circulation, hold the tool there for 30 seconds
Reference pressure reading At 1000 meters		Stop circulation, hold the tool there for 30 seconds
Reference pressure reading At mudline (meters)		Stop circulation, hold the tool there for 30 seconds
Tool at bottom of Borehole (locked in BHA)		Get the depth from driller
Start to retract CDS Start to penetrate formation (add lines for 10 cm drill string moverment		Specify the penetration displacement and penetration rate
End of push		
CDS activated		
Start to pullout		
Tool out of casing and put on the chamber		
Clean up the probe		

Pressure response test		
Download data		The previous stability data file & the deployment data
Switch back to DC power	Battery off: DC power on:	
Start the next stability file (2 minutes interval)		

3.6. SHIP LOGGING DATA

get the ship logging data from ship database (who exactly?)

3.7. DATA MANAGEMENT

7.1 File name convention for in-situ measurement
T2P_1, 2, 3, …
7.2 File name convention for stability data
Probe 1: stab_1a, 1b, 1c, ….
Probe 2: stab_2a, 2b, 2c, …
7.3 File name convention for ship logging data
T2P_1, 2, 3, …_ship

APPENDICES

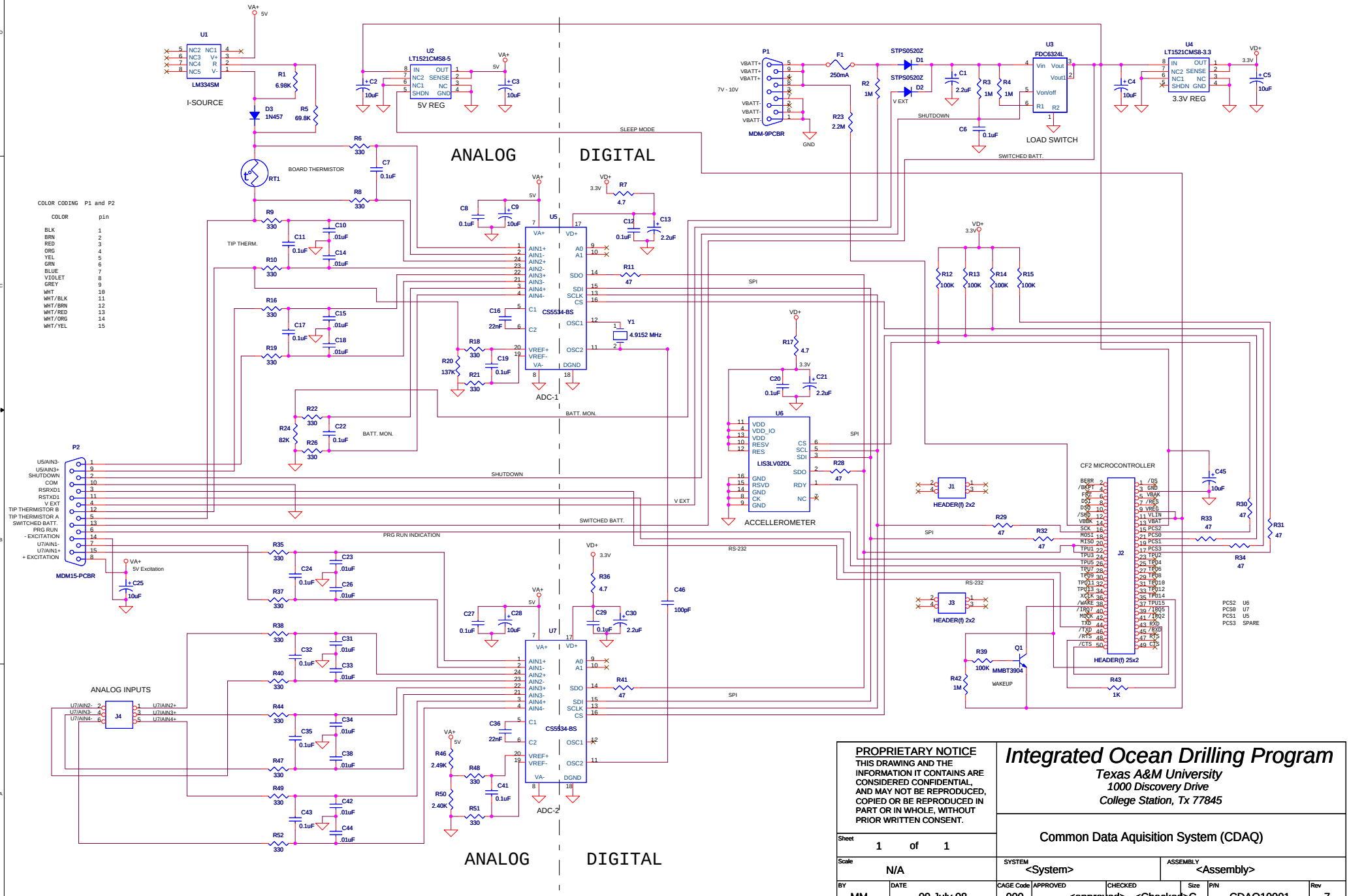
I. PARTS LIST

Part Name	Length	Diameter	Material	Number Needed	Number On Hand	Self Make	Send out	All Accounted For	
Complete Tip Assembly									
Threaded Tip Insert	1.25"	0.23"	Stainless Steel	6	6			x	
Small Porous Annulus	0.22"	0.24"	Inconel?	6?	4 used				
Thin Stainless Tube	11"	.06"	Stainless Steel	6	6			x	
Tip Thermistor			Misc.	6	12				
Outer Conic Tip Housing	8.375"	0.24 to 1.42"	Stainless Steel	6	6			x	
Large Porous Annulus	0.2"	1.42"	Inconel?	6?	6			x	
Threaded Tip Union	3.2"	1.42"	Stainless/Copper Coat	4	4 new 2 used			x	
Threaded Hex Standoff	1.875"	0.25"	Steel	4	8			x	
Pressure Transducer	1.75"	.375"	Stainless Housing	4?	3 used				
Body									
1.43" Drive Shaft	42.75"	1.43"	Stainless Steel	4	3		x		3 working and one machined out of spec
Wire Harness Strain Relief	30"	0.86"	G-10	2	1+1 prototype	x			
Wire Harness Fore Bracket	0.5"	0.93"	Aluminum?	2	1+1 prototype	x			
Wire Harness Aft Bracket	0.5"	0.93"	Aluminum?	2	1+1 prototype	x			
Skinny 5 Pin Fischer Locking Connector Male/Female Pair	1.41"	0.349	Aluminum etc	6?	2 pairs used		x		
Drive Shaft Wiring Harness 10 Conductors w/ wide fischer connector	35"	N/A	Copper Wire	2?	1+1 prototype	x			
Fischer Retainer	1.1"	1.1"	4140 Steel	2	2			x	
Drive Tube Nut	2.75"	2.0"	4140 Steel	2	2			x	
Cone Coupling	8.5"	2.75"	Steel	2	2			x	
Drive Tube Threaded Union	9.5"	2.75"	Steel	2	2			x	
Fischer Washer	0.2"	1.36"	?	2	0				
Wide Fischer Bulkhead Connector Female	1.06"	0.58"	Misc.	2	4			x	
Insertion Tube	1.2"	0.74"	Steel	2	2			x	
Greentweed Passthrough	1.56"	0.75"	Misc.	2	2			x	
Strain Relief Tube	1.2"	0.75"	Steel	2	2				
Wire harness from Greentweed Pass through to DAQ	~10"	N/A	Misc.	2	1+1 prototype	x			
DAQ Housing	30"	2.75"	Misc.	2	2			x	
Aft DAQ Housing End Cap	~8"	2.75"	Steel	2	2		x		Will need to be modified to fit new piston delivery system
DAQ Mount				2	1 prototype		x		
O-Rings	Thickness	Inner Diameter	Outer Diameter	Material	#				
Tip	1/16"		0.174"						
Tip	1/16"		0.635"						
Pressure Transducer	0.040"		0.26"						
Tip to Body Tube	0.149"	1.0"	1.22"						
Skinny to Fat Body Tube				(90) Nitrile	218				
Skinny to Fat Body Tube				(90) Nitrile	224				
Threaded Union to end of DAQ Tube	0.1875"	1.8"							
Greentweed Passthrough	0.1"	0.65"	0.75"						
	Note: We have a complete set of fresh O-rings.								
Screws/Misc.									
10-24 Screws									
Velcro									
8-32 Screws									
4-40 Screws									
Heat Shrink Tubing									

II. CDAQ & T2P WIRING SCHEMATICS

CHANGES Rev6 to Rev7

Replaced Buffer U8 and associated circuitry with Capacitor C 46
Add Resistor R23 from VBATT+ to J2 pin 14



PROPRIETARY NOTICE
THIS DRAWING AND THE INFORMATION IT CONTAINS ARE CONSIDERED CONFIDENTIAL, AND MAY NOT BE REPRODUCED, COPIED OR BE REPRODUCED IN PART OR IN WHOLE, WITHOUT PRIOR WRITTEN CONSENT.

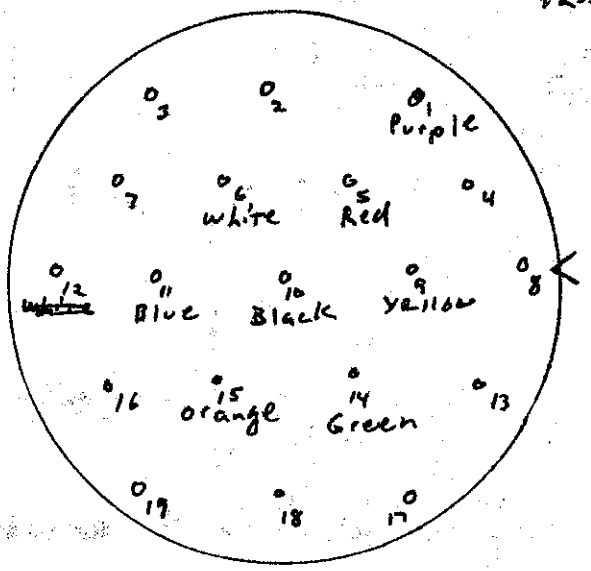
Integrated Ocean Drilling Program
Texas A&M University
1000 Discovery Drive
College Station, Tx 77845

Common Data Aquisition System (CDAQ)

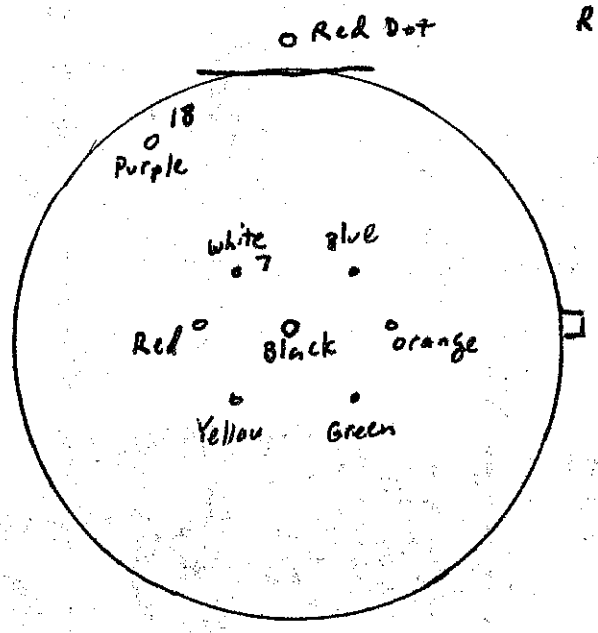
Sheet	Scale	SYSTEM	ASSEMBLY
1 of 1	N/A	<System>	<Assembly>

BY	DATE	CAGE Code	APPROVED	CHECKED	Size	P/N	Rev
MM	09 July 08	000	<approved>	<Checked>	C	CDAQ10001	7

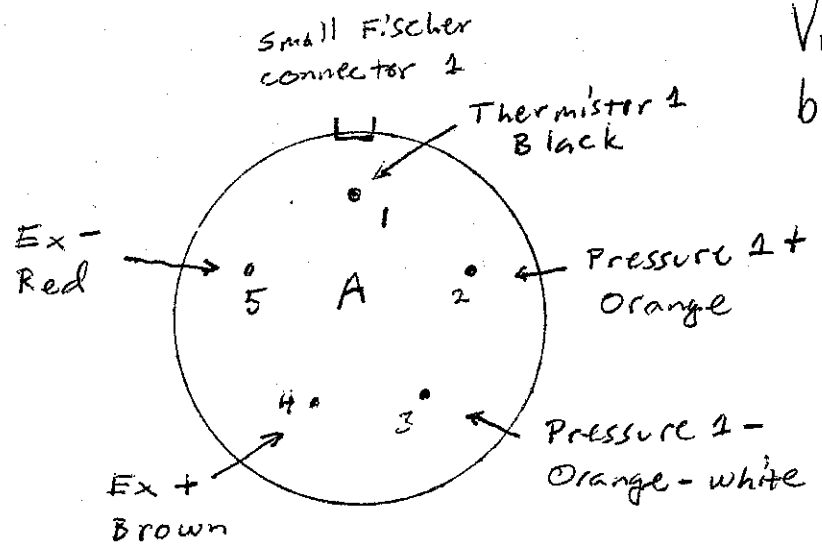
Tip Side Of Green Tweed
Pass Through

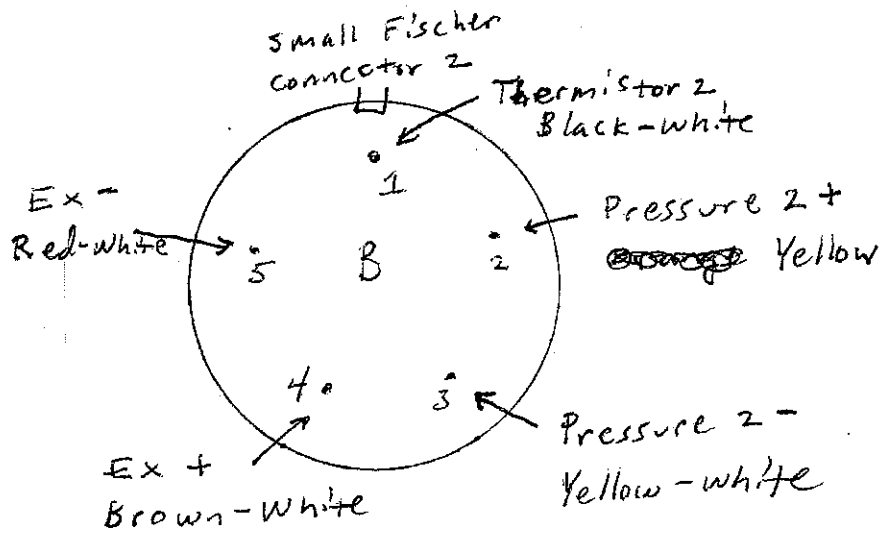


Back of 19 Pin fischer
Receptacle



Viewed from back of socket - Up Borehole





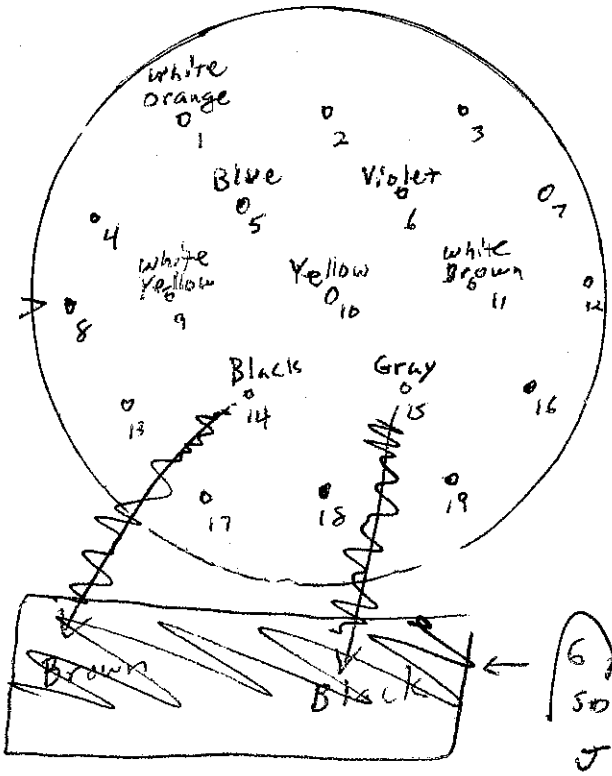
T2P Transducer Wiring Diagram
10-30-09 MJA

Greentweed DAQ side	Wire colors between 19 Pin Fischer connector and coiled cord (Fischer Pin)	Coiled cord wire colors	Small Fischer Connector
Purple 1	Purple (8)	Black	A1
Red 5	Red (6)	Black-White	B1
White 6	White (7)	orange	A2
Yellow 9	Yellow (5)	orange-white	A3
Black 10	Black (1)	Yellow	B2
Blue 11	Blue (2)	Yellow-White	B3
Green 14	Green (4)	Brown	A4
Orange 15	Orange (3)	Brown-White	B4
~~Brown 1~~		Red	A5
		Red-white	B5

	Wire colors between 19-Pin Fischer connector And coiled cord		Coiled cord Wire colors
1	Black	Thermistor 1	Black
2	Blue	Thermistor 2	Black-white
3	Orange	Pressure 1 +	Orange
4	Green	Pressure 1 −	orange-white
5	Yellow	Pressure 2 +	Yellow
6	Red	Pressure 2 −	Yellow-white
7	White	Excitation +	Brown
18	Purple	Excitation −	Brown white
			Red
			Red white

Coiled cord wire colors	Transducers
Black	
Black white	
Orange	
Orange-white	
Yellow	
Yellow-white	
Brown	
Brown white	
Red	
Red white	

Greenwood Passthrough
MDM 15 Pin CDAQ Connector
wires

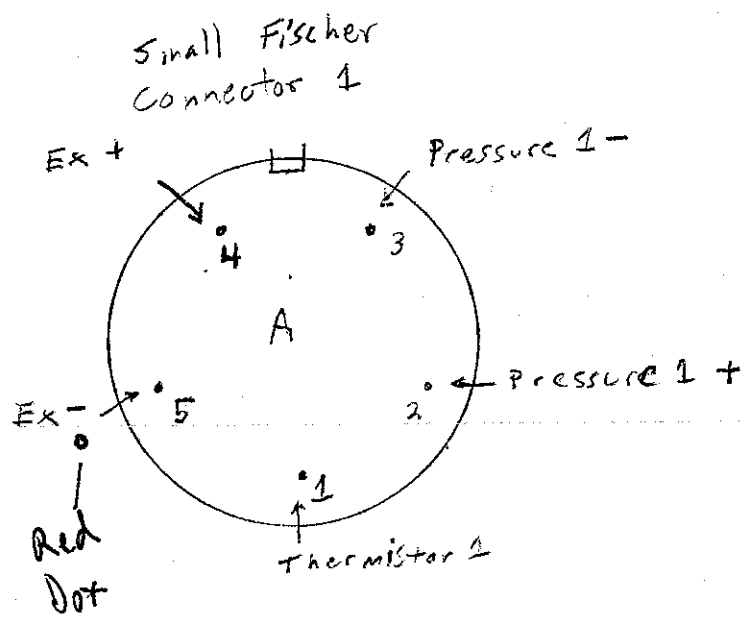


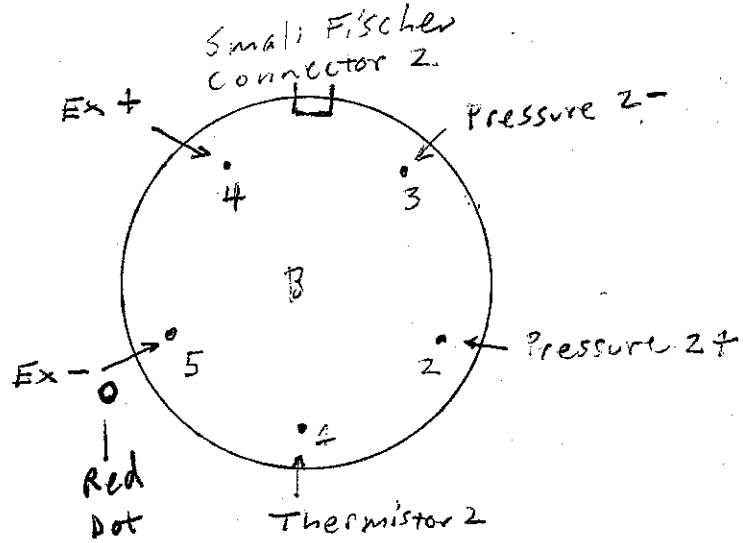
6 pin Double row header socket connects to J4 of CDAQ

Fischer Serial Connector → MDM

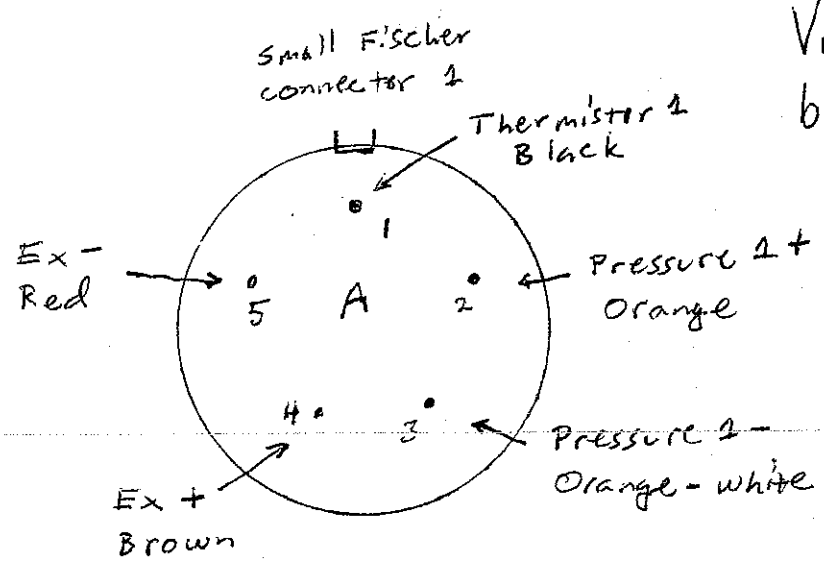
Red	Red
White	white
Black	White-Black

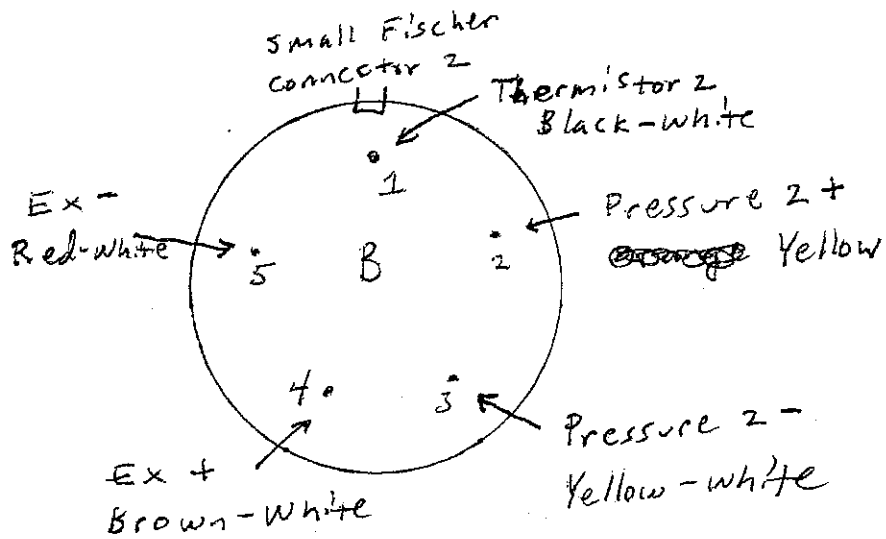
Small Fischer Connector
Viewed from back of plug
Down Borehole





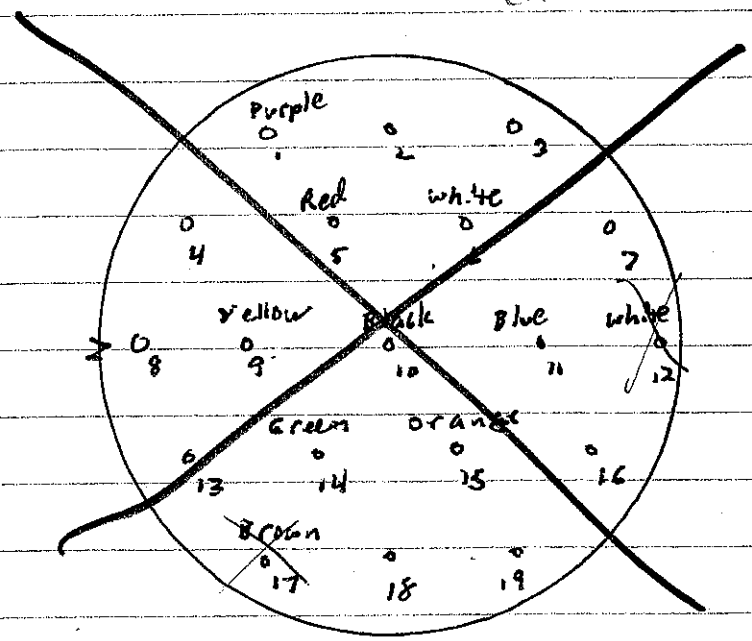
Viewed from back of socket



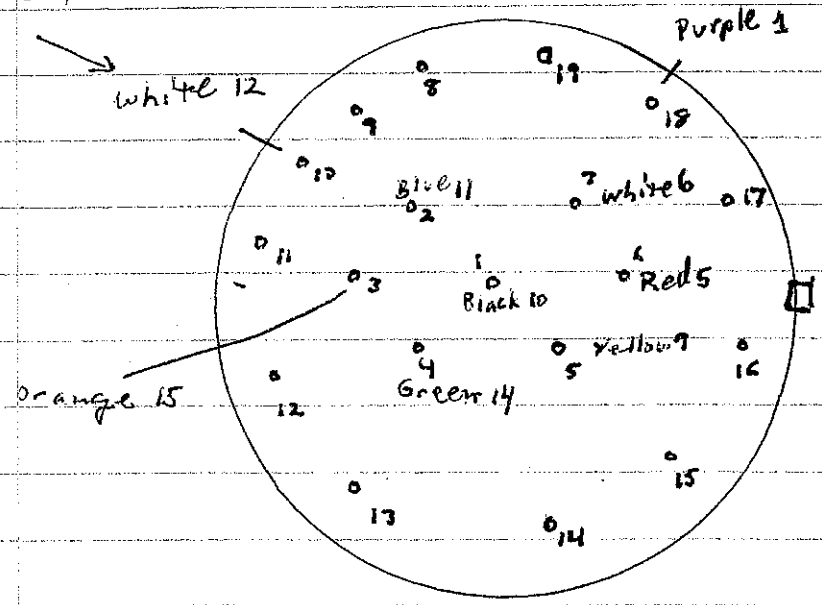


T2P Transducer Wiring Diagram
10-30-09 MJA

DAQ side of high pressure feedthrough Greenfield



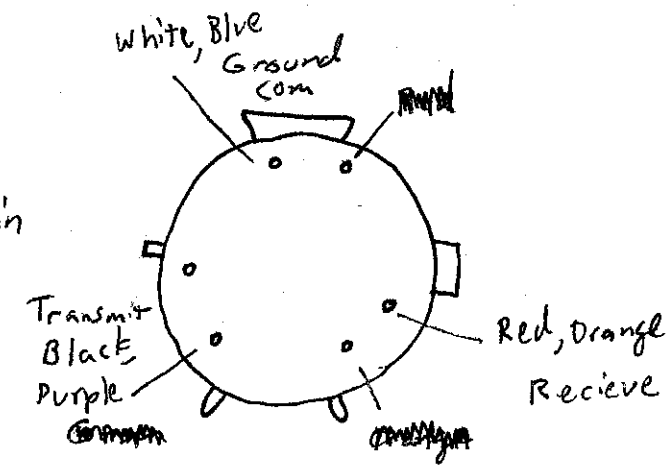
Unused



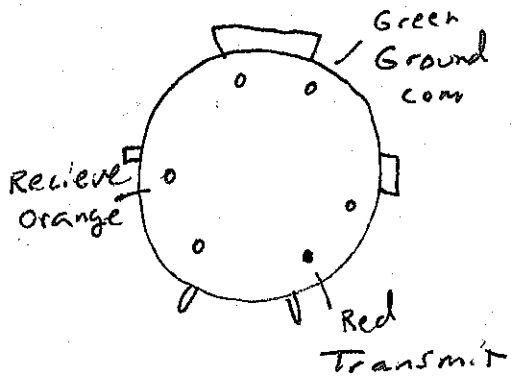
Serial Connection

Upstream of CDAQ

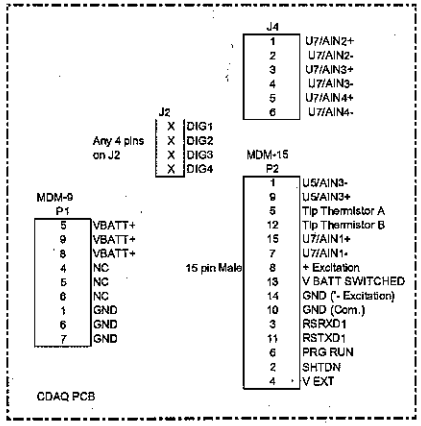
Back of Mini-Din Socket
Connected to 15 Pin MDM connector On CDAQ

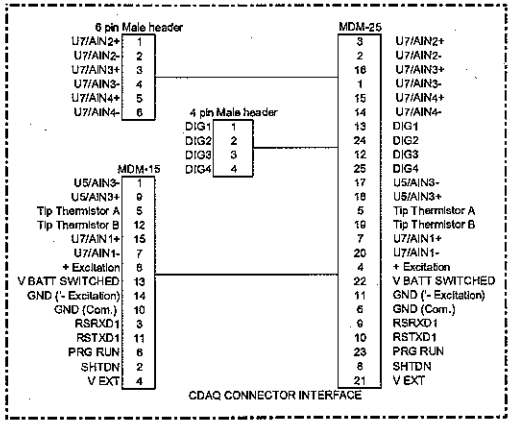


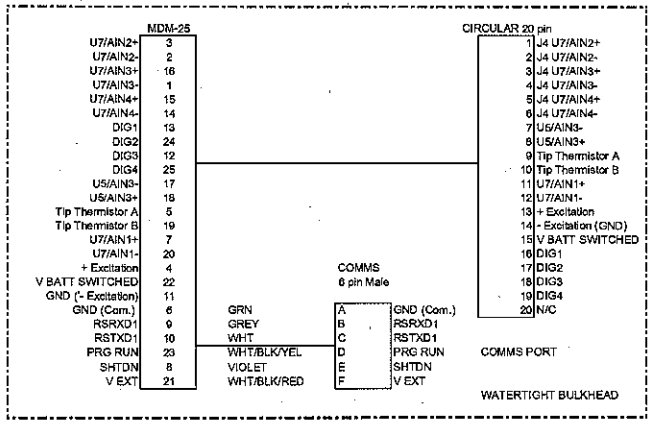
Back of Mini-Din Plug
Goes up and out of T2 & T6 Computer



Serial Connection	CDAQ
Green	Black - Purple (Transmit)
Red	Red - Orange (Recieve)
Orange	White - Blue (Ground, com)







MDM CONNECTOR

PIN	COLOR
1	BLK
2	BRN
3	RED
4	ORG
5	YEL
6	GRN
7	BLUE
8	VIOLET
9	GREY

MDM CONNECTOR

PIN	COLOR
1	BLK
2	BRN
3	RED
4	ORG
5	YEL
6	GRN
7	BLUE
8	VIOLET
9	GREY
10	WHT
11	WHT/BLK
12	WHT/BRN
13	WHT/RED
14	WHT/ORG
15	WHT/YEL

MDM CONNECTOR

PIN	COLOR
1	BLK
2	BRN
3	RED
4	ORG
5	YEL
6	GRN
7	BLUE
8	VIOLET
9	GREY
10	WHT
11	WHT/BLK
12	WHT/BRN
13	WHT/RED
14	WHT/ORG
15	WHT/YEL
16	WHT/GRN
17	WHT/BLUE
18	WHT/VIOLET
19	WHT/GREY
20	WHT/BLK/BRN
21	WHT/BLK/RED
22	WHT/BLK/ORG
23	WHT/BLK/YEL
24	WHT/BLK/GRN
25	WHT/BLK/BLUE

20 pin CICULAR (male)

1	WHT
2	BLK
3	RED
4	GRN
5	ORG
6	BLUE
7	VIOLET
8	YEL
9	BRN
10	GREY
11	WHT/BLK
12	WHT/RED
13	WHT/GRN
14	WHT/ORG
15	WHT/VIOLET
16	WHT/BLUE
17	WHT/GREY
18	BLK/RED
19	BLK/YEL
20	BLK/GRN

III. IODP1.D PROGRAM DESCRIPTION

IODP Borehole Logger

Oceanographic Embedded Systems

1260 NE Seavy Avenue

Corvallis, OR 97330

FIRST DRAFT

November 16, 2007

Introduction

This document describes the program that collects borehole temperature data using the CDAQ interface board and a Persistor CF-2 microcontroller system. The program manages the collection of borehole temperatures from a tool tip thermistor, and the collection of board temperature, ambient pressure and acceleration data. The primary variable of interest is the tip temperature. Ambient pressure aids in the determination of the tool position within the borehole. Board temperature and acceleration data are collected for diagnostic purposes. Tip temperature, board temperature, and pressure are collected at 1-second intervals. Acceleration data is collected at 100-millisecond intervals. The results are stored on the compact flash card attached to the CF-2, then played back through the serial port of the logger after the conclusion of the data acquisition.

Section 1 Program Structure

The data collection program is written in the C language and uses the PicoDOS operating system provided as part of the CF-2 development system from Persistor Instruments. The code was compiled using the MetroWerks Codewarrior development system for the PalmOS, version 8.0. The program is divided into four main sections:

1. System Initialization and main user input and control loop
2. Interrupt-driven data collection functions
3. Data retrieval and formatting functions
4. Peripheral drivers for the attached hardware

Section 1.1 Initialization and Main Loop

When the program starts up, it calls functions that initialize the peripheral drivers, then displays a prompt for the operator. It then enters a program loop that monitors the serial port and collects input characters used as commands to control the program operation. These commands are described in Section 2 of this document. After initialization, the program asks the operator to verify the date and time being used by the CF-2. If the time is incorrect, the operator is asked to enter the correct date and time.

The main program loop runs continuously until the operator requests the termination of the program or the CDAQ power is shut down. While this loop is executing, it examines internal variables to determine whether data collection is in progress. It data is being acquired, the main loop displays a subset of the collected data. Only the 1Hz data is displayed. If data is being acquired, the main loop also executes code to retrieve the data from First-In, First-Out (FIFO) queues and write it to the compact flash card.

Section 1.2 Interrupt-driven Data Collection

When the operator requests the start of data acquisition, the program starts collecting the data as part of an interrupt routine that runs asynchronously to the main processor loop. This data acquisition is timed and sequenced by an interrupt handler that is called in response to a 50Hz interrupt from the Programmable Interrupt Timer (PIT).

The PIT handler chore increments an internal variable, PITCount, at each interrupt. When PITCount is greater than 49, it is reset to zero. The PITCount value is used to control state machine code that sequences the data collection.

At each interval where PITCount modulus 5 is zero, the program calls a function to collect data from the accelerometers. The raw binary values from the accelerometer and the time of collection are stored in a FIFO queue in RAM memory. The main program loop will pull data from this FIFO and write it to a temporary file on the CF card.

The PIT interrupt chore also calls a function that implements a state machine to control collection of data from the A/D converter and pressure sensor. This state machine sequences operations to allow for the time required to digitize data from the different sensors. The state machine carries out operations with the timing and sequence shown in Table 1.

Table 1. 1Hz Collection Queue State Machine

PIT Count	Nominal Time	Operation
1	20	Request start of data collection from ADC Store time in new ADC queue element.
2	40	Request a pressure reading from the ParoScientific sensor
36	730	Read ADC result from tip temperature channel. Store in queue element
37	740	Request board temperature counts from ADC channel 1
41	820	Collect board temperature counts from ADC channel 1 and place in queue element
42	840	Request battery voltage from ADC channel 4
46	920	Collect battery voltage counts and place in queue element
47	940	Collect the output of the pressure sensor and store in queue element
48	960	Place the element in the ADC queue

The nominal time in the table is the number of milliseconds after the start of a second as determined by the CF-2 real-time clock. Since the collection is not exactly synchronized with the RTC, the times stored in the queue elements may be slightly offset from the nominal times.

There are several key restrictions on the coding of the interrupt-driven data collection:

- No function called during the PIT chore can take longer to execute than the time between interrupts (20milliseconds). Because of this restriction, the state machine is required to break up the collection sequence into discrete steps because the ADC and pressure sensor both require more than 20msec to return data (more than 530msec in the case of the tip temperature channel) In this implementation, the longest execution time for a PIT chore state is approximately 200 microseconds.

- The time expended by the PIT chore is not available to the CPU foreground code that stores data in the CF card files, so the total time must not significantly affect data throughput. The foreground routine must be able to fetch data from the queues and store it as fast as the PIT chore is collecting.

- The PIT chore and the foreground routines must communicate using variables that are atomic (that are written or read in a single CPU cycle). This is necessary so that changing a variable, such as queue length, in the interrupt routine, cannot result in an invalid value being read by the foreground routine.

You should note that, while the pressure and tip temperature data are requested within 20msec of each other, they represent data collected at different times. The tip temperature value is sampled and held shortly after the request is issued, then converted during the ensuing 500 milliseconds. The pressure sampling is started at the time requested, but the signal is integrated throughout the sampling interval. Therefore, the tip temperature value represents the temperature at the start of the interval, while the pressure represents the average over the whole integration interval (about 700msec for the setup in the sensor). More precise synchronization of this sampling is beyond the scope of this project.

Section 1.3 Data Output and Formatting

The data collected in the acquisition phase is stored in two temporary files; one for the 1Hz ADC and pressure data, and the other for the 10Hz accelerometer data. The data is written in binary format as copies of the queue elements in the FIFO queues. The data is written to temporary files, which are reused with each data collection session. Since the data is not in human readable format, the program contains commands to convert the data to formatted ASCII output and transfer that output either to the serial port or to a new file on the CF-2 CF card.

The output format is as specified in the original Statement of Work from IODP. The output data stream contains the merged output from both the 1Hz sensor file and the 10Hz accelerometer file. The rate of transfer to the serial port is limited by three factors: The rate at which the CF-2 can read data from the CF card, the time required to do the conversion from binary counts to ASCII output format, and the baud rate of the serial

port. When the baud rate is greater than 57600 baud, the limiting factor appears to be the time needed to do the output calculations.

The format chosen for the standard output is to have the data aligned in columns with spaces as separators. There is a compile-time option to have the data columns separated with tab characters. If you wish to use this format, you need to uncomment the define statement for USETABS in the CalConstants.h header file.

Section 2 Program Operation

When the program starts up, the user is presented with a menu of single-character commands that are used to control the program operation. Unless you have your terminal set up to echo input, you will not see the commands that you enter. At startup, you will see a display like this:

```
Program: IODP1.c: Nov 15 2007 16:55:12
Persistor CF2 SN:7107   BIOS:4.2   PicoDOS:4.2
Enabling Accelerometers
The CF-2 date and time are: 10/31/91  14:16:03
Are these values correct  [Y] ? n
Enter date and time:  ? 11/16/07 14:57:06
CF-2 time and date have been adjusted.

CURRENT TIME AND DATE: 11/16/07  14:57:06
      PROGRAM VERSION: V1.00B
LAST CALIBRATION DATE: 11/12/2007
             TOOL S/N: 444


Commands:
d :  Adjust date and time    f :  Convert temp file to ASCII
s :  Start Data Collection   e :  End Data Collection
p :  Play Back data          i :  Show system Information
q :  Quit program            ? :  Show this command list

CMD:
```

The user is first prompted to verify the date and time. In the example, they are not correct and new values are entered.

Section 2.1 User Commands

The single character commands behave as follows:

d: Adjust Date and Time. You will be prompted for the new date and time setting for the CF-2 real time clock. You may press ‘Enter’ if the displayed values are correct, or you may enter a new date and time in the format shown in the prompt.

s: Start Data Collection. You will be prompted for a tool serial number. You may enter a string up to 12 characters long. This string will be saved and displayed as part of the output header. After you enter the tool serial number, acquisition will start and the program will display the file status and 1Hz data in the following format:

```
Tool Serial Number  ? 431A5
Starting Data Collection.
Opened adtemp.dat
Opened axtemp.dat

CMD:  15:05:51.26    19.052  13018.85     24.8             6.6
 15:06:07.26    19.076  13005.11     24.8             6.6
 15:06:08.26    19.077  13004.59     24.8             6.6
 15:06:09.26    19.077  13004.09     24.8             6.6
 15:06:10.26    19.078  13003.59     24.8             6.6
 15:06:11.26    19.079  13003.15     24.8             6.6
```

During collection the program displays the time, tip temperature, tip resistance, board temperature pressure, and battery voltage. In the example above, the pressure data is blank because no pressure sensor was connected.

p: Playback Data The binary data collected will be converted, formatted and sent to the serial port. You may play back the data as many times as you wish. However, if you start another data collection session, the old binary data is overwritten. When you play back the data, you shoud record the output in your PC terminal program. The output will look like this:

```
       ACQUISITION DATE: 11/15/07
        PROGRAM VERSION: V1.00B
  LAST CALIBRATION DATE: 11/12/2007
               TOOL S/N: 431A5
PRESSURE TRANSDUCER S/N: 333

 GMT Time   Tip Temp Tip R     Brd Temp Battery  Pressure  GMT Time     X    Y    Z
 HH:MM:SS      deg C  Ohms      deg C   Volts        PSI   HH:MM:SS     g    g    g
 16:55:43.04  19.066 13011.10   25.8     6.6       14.36  17:00:44.62 0.13 0.01 1.01
 17:00:45.04  19.103 12989.16   26.2     6.6       14.36  17:00:44.72 0.13 0.01 1.01
 17:00:46.04  19.103 12989.14   26.2     6.6       14.36  17:00:44.82 0.13 0.01 1.01
 17:00:47.04  19.103 12989.11   26.2     6.6       14.36  17:00:44.92 0.13 0.01 1.01
 17:00:48.04  19.103 12989.06   26.2     6.6       14.36  17:00:45.02 0.13 0.01 1.01
 17:00:49.04  19.103 12989.08   26.2     6.6       14.36  17:00:45.12 0.13 0.01 1.01
 17:00:50.04  19.103 12989.15   26.2     6.6       14.36  17:00:45.22 0.13 0.01 1.01
 17:00:51.04  19.103 12989.14   26.2     6.6       14.36  17:00:45.32 0.13 0.01 1.01
 17:00:52.04  19.103 12989.12   26.2     6.6       14.36  17:00:45.42 0.13 0.01 1.01
 17:00:53.04  19.103 12989.01   26.2     6.6       14.36  17:00:45.52 0.13 0.01 1.01
 17:00:54.04  19.103 12988.94   26.2     6.6       14.36  17:00:45.62 0.13 0.01 1.01
 17:00:55.04  19.103 12988.90   26.2     6.6       14.36  17:00:45.72 0.13 0.01 1.01
 17:00:56.04  19.103 12988.90   26.2     6.6       14.36  17:00:45.82 0.13 0.01 1.01
 17:00:57.04  19.103 12988.88   26.2     6.6       14.36  17:00:45.92 0.13 0.01 1.01
                                                         17:00:46.02 0.13 0.01 1.01
                                                         17:00:46.12 0.13 0.01 1.01
                                                         17:00:46.22 0.13 0.01 1.01
                                                         17:00:46.32 0.13 0.01 1.01
                                                         17:00:46.42 0.13 0.01 1.01
```

```
17:00:46.52 0.13 0.01 1.01
17:00:46.62 0.13 0.01 1.01
17:00:46.72 0.13 0.01 1.01
```

NOTE: The merged file output format results in 9 times as many lines of 10-Hz data with only accelerometer values as there are lines of combined 1Hz ADC and accelerometer data.

You may also see some differences in column alignment if your program changes runs of space characters into tabs or if the terminal window is not wide enough to display all the data. These are inevitable side effects of the merged ASCII output format.

f: Convert temp file to ASCII This command prompts you for a file name for the ASCII file to be written to the CF card. After you enter the file name, the program will do the same conversion as the playback command, but the output is directed to a file on the CF card. The program will keep you informed of the progress by putting a dot (period character) on the screen for each 60 lines of data that is converted. For a long file, you will note that the dots come more quickly when the program is converting only the accelerometer values at the end of the file. The output from the conversion of a short file looks like this:

```
Name for converted file  [IODPTEMP.TXT] ?
Starting conversion.

.................................................
```

The default file name of IODPTEMP.TXT was used.

e: End Data Collection. This command stops data collection and closes the temporary files. There is no user feedback except the cessation of the 1Hz data output.

i: Show System Information. This will show you some information about the system. Much of the data comes from the constants defined in CalConstants.h.

```
CURRENT TIME AND DATE: 11/16/07  16:30:08
      PROGRAM VERSION: V1.00B
LAST CALIBRATION DATE: 11/12/2007

             TOOL S/N: 345a
```

The tool serial number will the value entered during the last data collection session.

q: Quit Program. Self-explanatory.

?: Show this command list. Self-explanatory.

Section 3 Peripheral Driver Code

The code that allows the main program to collect and display the desired data calls upon functions in peripheral driver files. These files, IODP_Accel.c, IODP_Paro.C, and IODP_ADC.c, contain functions that collect store, and process the data from the sensors. The IODP_ADC.c driver calls functions in CS5534Lib.c to manage interactions with the A/Dconverters. The driver code in these files is sufficiently modular that it should be straightforward to adapt these drivers for other uses.

Section 3.1 CS5534 ADC Driver

The A/D driver communicates with the CS-5534 converters using the SPI port of the CF-2. This driver receives setup and control commands for the 8 available channels on the CDAQ board and returns data from the converters. The IODP program does not use all the available functions in this library, and only those that are used are included in the compiled program.

It is important to note that the A/D driver does not use or monitor the busy status bit from the converters. There are two reasons for this: First, the SPI port to which that bit is connected is also used for communications with the accelerometers while A/D conversions are in progress. Secondly, the A/D conversions occur in the background, and the program cannot wait in the interrupt service routine for a conversion to complete. Instead, the driver sets up a conversion for a particular channel, then collects the output at a later time in another call to the interrupt handler. The data collection state machine is set up so that the A/D has time enough to convert the channel between the request and the data readout. The time required depends on the conversion rate for the particular channel. For the tip temperature 7.5Hz conversions, the time between request and readout must be more than 530msec. For the other two channels a delay of 80msec is adequate.

The 24-bit results from the converter are saved in a FIFO queue in the interrupt routine and read out by the main program for storage. Conversion of the results is done according to an algorithm provided by IODP.

The A/D does not use its own internally calculated scale and offset calibration values. Instead, it is given the constants defined in the IODP_CalConstant.h file for each channel. These constants are separately determined using procedures and equipment provided by IODP.

Section 3.2 SPI Accelerometer Driver

The accelerometer driver uses the CF-2 SPI port to set up the accelerometers and to read the data ten times per second. The data is read in the PIT interrupt service routine. This

driver also contains code to store the accelerometer data in a FIFO queue, and to fetch data from the FIFO and write it to a temporary file.

The driver does nothing complex to set up the accelerometers. It uses the data rate and format settings that are the defaults for the accelerometer. Conversion of the accelerometer outputs to engineering units is done using scale and offset constants in the IODP_CalConstant.h file.

Section 3.3 Paroscientific Pressure Sensor Driver

The driver for the ParoScientific pressure sensor is the least complex of the drivers in this program. It simply sends a command to the sensor to return a conversion result, and, later returns to collect the data string from the sensor. The program communicates with the sensor using a TPU UART port which is connected to spare RS-232 drivers on the CF-2.

The program does not attempt to convert the string returned by the pressure sensor to a floating point value. It simply records the output string and saves it in the 1Hz FIFO queue. This avoids the requirement to have a time-consuming ASCII to floating point conversion in the interrupt service routine, and it avoids any rounding errors that might occur during format conversions. The program simply strips excess characters from the sensor output and saves the resulting string. At playback what you see on the screen is the same string that was sent by the sensor. This storage format requires that you set up your desired output format for the sensor and write that format to the sensor EEPROM so that it will become the default when the sensor is next turned on. The program also assumes that the sensor is communicating at 9600 baud and has been programmed for the correct signal integration time (I have used 700msec on the sensor I was loaned.)

IV. ACKNOWLEDGEMENTS

The authors wish to acknowledge the support and guidance from the IODP engineering team at Texas A&M University. We owe many thanks to Mike Meiring, Dean Ferrell, Liping Chen, and Kevin Grigar. We must also acknowledge the financial support received from the National Science Foundation under grant 6284184 and the Geofluids Consortium, University of Texas at Austin.

V. REFERENCES

Flemings, P.B., Germaine, J., Long, H., Dugan, B., Sawyer, D., Behrmann, J.H., John, C., and 308, S.S.P.I.E., 2006, Measuring Temperature and Pressure With the Temperature Two Pressure (T2P) Probe in the Ursa Basin, Gulf of Mexico: Development of a New Pressure and Temperature Probe for the IODP, Offshore Technology Conference: Houston, Texas, p. 14.

Meiring, M., 2008, Common Data Acquisition (CDAQ) Logger Engineering Manual, Integrated Ocean Drilling Program United States Implementing Organization, p. 56.

VI. SUPPLEMENTAL REFERENCES

Chartier, M.G., 2005, The Design and Field Evaluation of a New Dual Pressure and Temperature Tapered Probe: Boston, Massachusetts Institute of Technology.

Flemings, P.B., Long, H., Dugan, B., Germaine, J., John, C.M., Behrmann, J.H., Sawyer, D., and Scientists, I.E., 2008, Pore pressure penetrometers document high overpressure near the seafloor where multiple submarine landslides have occurred on the continental slope, offshore Louisiana, Gulf of Mexico (vol 269, pg 309, 2008): Earth and Planetary Science Letters, v. 274, p. 269-283.

Long, H., 2007, Interpreting Pore Pressure in Marine Mudstones with PorePressure Penetrometers, In Situ Data and Laboratory Measurements, The Pennsylvania State University.

Long, H., Flemings, P.B., Dugan, B., Germaine, J.T., and Ferrell, D., 2008, Data report: penetrometer measurements of in situ temperature and pressure, IODP Expedition 308, Proceedings of the Integrated Ocean Drilling Program, Volume 308.

Long, H., Flemings, P.B., and Germaine, J.T., 2007, Interpreting in situ pressure and hydraulic properties with borehole penetrometers in ocean drilling: DVTPP and Piezoprobe deployments at southern Hydrate Ridge, offshore Oregon: Journal of Geophysical Research-Solid Earth, v. 112.

Whittle, A.J., Sutabutr, T., Germaine, J.T., and Varney, A., 2001, Prediction and interpretation of pore pressure dissipation for a tapered piezoprobe: Geotechnique, v. 51, p. 601-617.