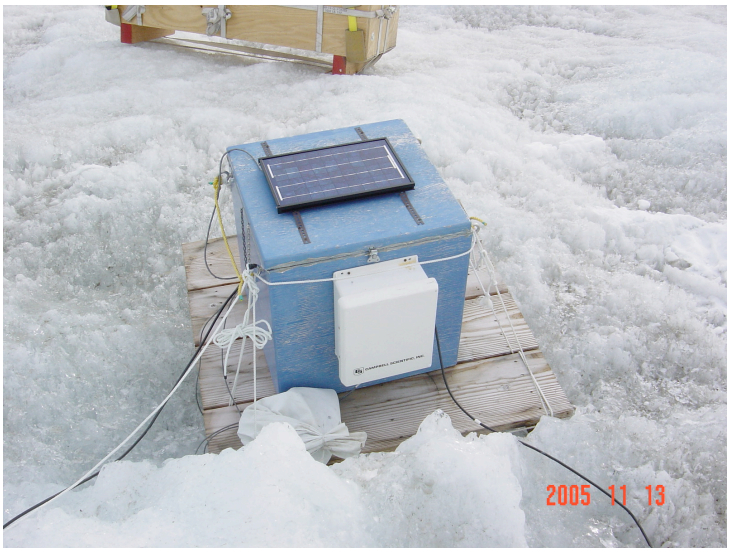
Documentation of The Taylor Valley Blue Boxes for season 2008/09



Maciej Obryk
Department of Earth and Environmental Sciences
University of Illinois at Chicago
845 W. Taylor Street
mobryk2@uic.edu

Season: 2008-09
Version: 1
Last change: 12/16/2008

TABLE OF CONTENT:

1. Sensors

1.1 Available Sensors

Blue Box Sensor List								
SENSORS	SENSOR TYPE	Aplications	Company	Units	Deployment Date			
					Fryxell	Hoare	ELB	WLB
Surface PAR	LI190 SB Quantum	Measures incedent Photosynthetically Active Radiation	LI-COR Biosciences 4421 Superior St. Lincoln, NE 68504 Phone: 800-447-3576 Fax: 402-467-2819	micromoles of quanta per second per square meter (mmol s-1 m-2)	11/03/08	11/01/08	12/??/05 10:02	11/26/05 15:12
Underwater PAR	LI-193 Spherical Quantum Sensor	Measures PAR coming from all directions.	LI-COR Biosciences 4421 Superior St. Lincoln, NE 68504 Phone: 800-447-3576 Fax: 402-467-2819		11/08/08	11/03/08	2nd Dec. 2007 12:40	1st Dec. 2007 18:20
Ablation Transducer (pressure transducer)	Druck Pressure Transducer PDCR 1830 or 1230; Keller Pressure Transducer Series 169/173 only at Lake Hoare.	Continuous lake ice ablation measurement	Druck Incorporated (203) 746-0400 Keller ?	cm	11/19//05	?	19/11/04	17/11/04
Lake Lavel (Stage) Transducer		Continuous lake level measurements	Druck Incorporated (203) 746-0400		21/11/04	17/11/04	19/11/04	N/A
Underwater Altimeter/Sonar	PSA-916	Distance measurements	Teledyne Benthos, Inc. 49 Edgerton Drive North Falmouth, MA 02556 Tel: 508 563-1000 Fax: 508 563-6444 http://www.benthos.com	cm	2007/08 Installed longer cable	2007/08 Installed longer cable	11/04/08 Short cable	2007/08 Left shorter cable

1.2 Sensors, Data Logger and Programming Information and Manuals at the Internet

(1) ***LI190 SB Quantum***

Sensor details:
http://www.licor.com/env/Products/Sensors/190/li190_description.jsp

Instruction Manual from Campbell Scientific:
http://www.campbellsci.com/documents/manuals/li190sb.pdf

(2) ***LI-193 Spherical Quantum Sensor***

Sensor details:
http://www.licor.com/env/Products/Sensors/193UW/li193_description.jsp
http://www.licor.com/env/PDF_Files/193SA.pdf

(3) ***Druck's Pressure Transducer*** **(Lake Fry, ELB and WLB)**

Sensor details:
http://www.gesensing.com/products/resources/datasheets/PDSA065june02.pdf

Instruction Manual from Campbell Scientific
http://www.campbellsci.com/documents/manuals/cs420-l.pdf

Note: Ablation Transducer (frequently called Pressure Transducer) and Lake Level Transducer (also known as Stage Transducer) are both the same type of transducers. The only difference between two is that they are differently programmed – one for measuring the Ice ablation and the other for measuring the lake level.

(4) *Keller Pressure Transducer 173* (only at Lake Hoare)

Sensor details:
N/A

Instruction Manual from Campbell Scientific
www.campbellsci.ca/Catalogue/KELLER169_173_Man.pdf

(5) CR10X Data Logger

Measurements and Control Module Operator's Manual
http://www.campbellsci.com/documents/manuals/cr10x-ov.pdf

CR10X Specifications
http://www.campbellsci.com/documents/lit/s_cr10x.pdf

CR10X Brochure
http://www.campbellsci.com/documents/lit/b_cr10x.pdf

(6) PSA-916 Underwater Altimeter

Information
http://www.benthos.com/undersea-acoustic-altimeters-pingers.asp

Details
http://www.benthos.com/pdf/psa900_916.pdf

(7) LoggerNet Instructions

Logger Net Datalogger Support Software
http://www.campbellsci.com/loggernet3x

LoggerNet Users Manual 3.3
http://www.campbellsci.com/documents/manuals/loggernet.pdf

LoggerNet Users Manual 2.1b
http://www.campbellsci.com.au/documents/manuals/loggernet2-1.pdf

1.3 Diagram

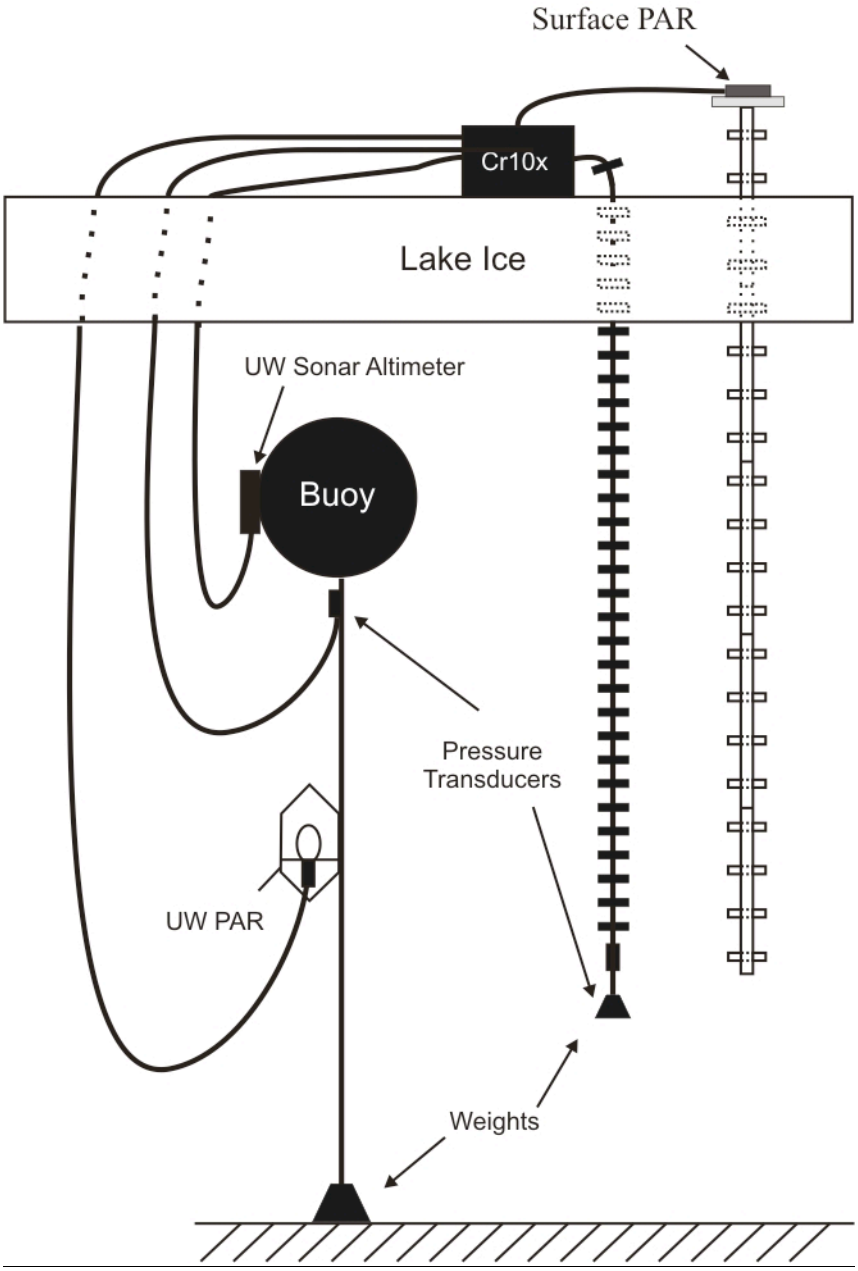


Fig.1: Lake cross-section showing buoy, data logger, position of deployed sensors and ablation stake.

UW par deployment depths <u>measured from the lake/water level</u> (see images below):

Lake Fryxel: 9. 78 m
Lake Hoare: 10 m
ELB: 10 m
WLB: 10m

2. BlueBox Filed Setup

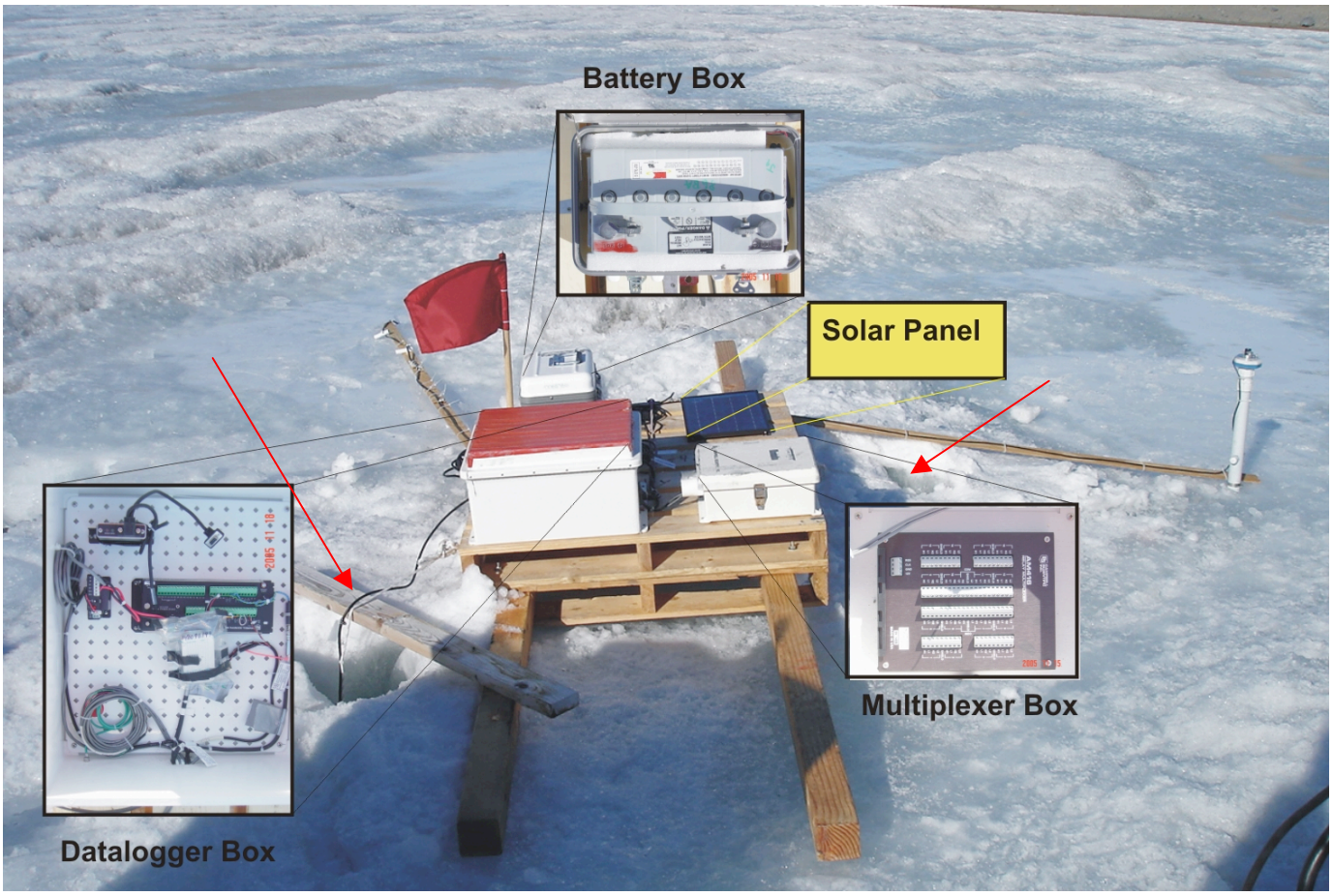


Fig. 2: Field setup and components of a Blue Box system. Outriggers are facing main wind direction and with two anchors (marked by red arrows) to prevent box from flipping.

3. Notes on changes made in November 2006

3.1 Hardware Changes

Sensor replacement/installation:

- UW PAR were replaced at Lake Hoare and Fryxell according to two-year schedule established by John Priscue.
- UW altimeter/sonar bathos PSA916 was installed at ELB.

3.1.1 Lake Fryxell

- Replaced UW PAR
- Replaced surface PAR
- Installed longer (15m) cable for the PSA-916 sonar

- New sensor information (2008/09):

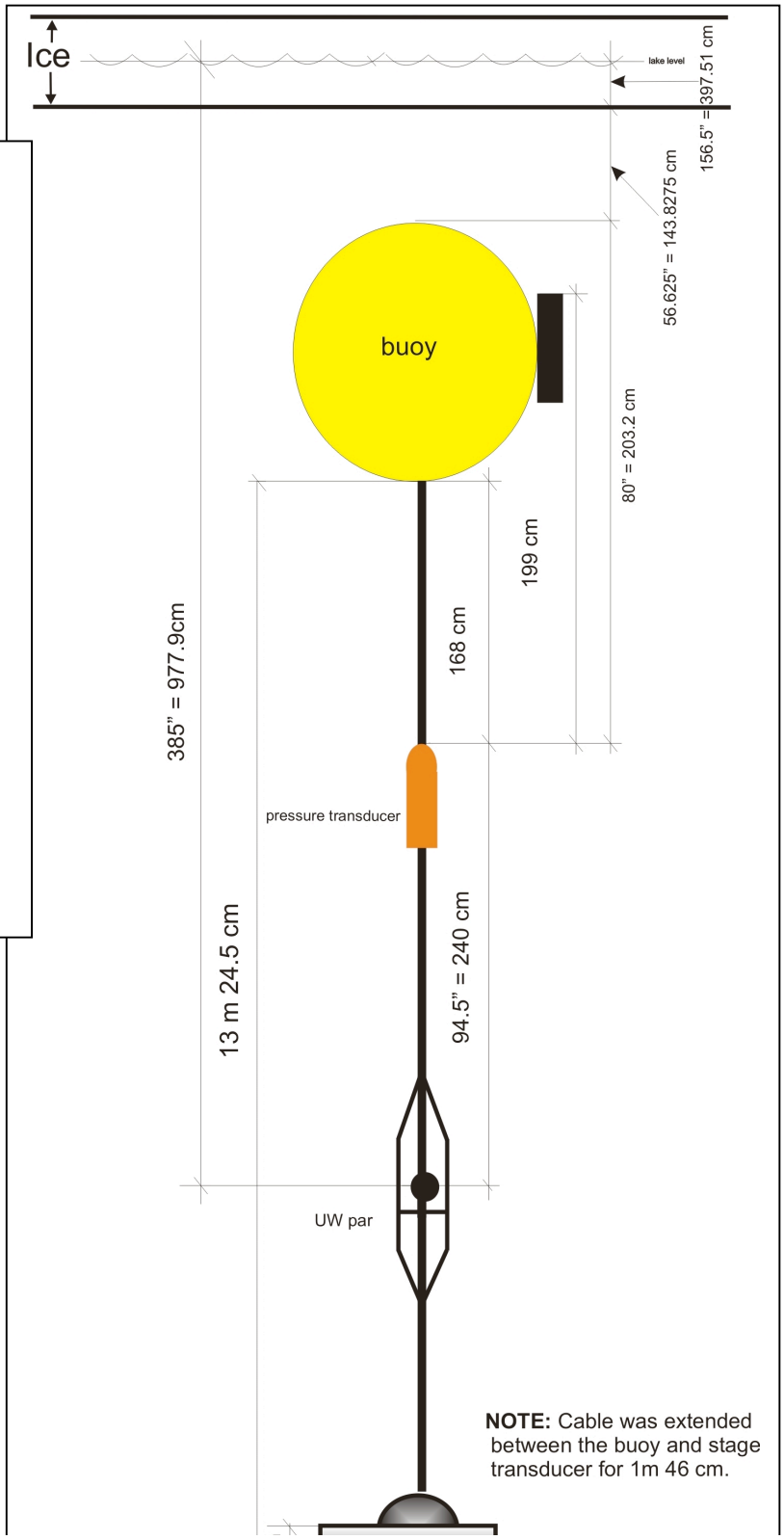
UW PAR

MULT –132.98 (in air)
-215.43 (in water)
tcoff: 0.0036
Units: µmol $s^{-1}m^{-2}$
Serial#: SPQA4140
Cal. Date:17 Jul. 08 LiCor

Back of laminated card:
µmol $s^{-1}m^{-2}$ (instantaneous)
µmol m^{-2} (integrated)
µmol $s^{-1}m^{2}$ (average)

Surface PAR

Serial: Q29773
Calibration: April 4, 2008
Output: 3.90 milivolts/1000µmol s-1 m-2
604.0 Ω resistor installed



3.1.2 Lake Hoare

- Replaced UW PAR
- Surface PAR
- Installed longer (15m) cable for the PSA-916 sonar

● New sensor information (2008/09):

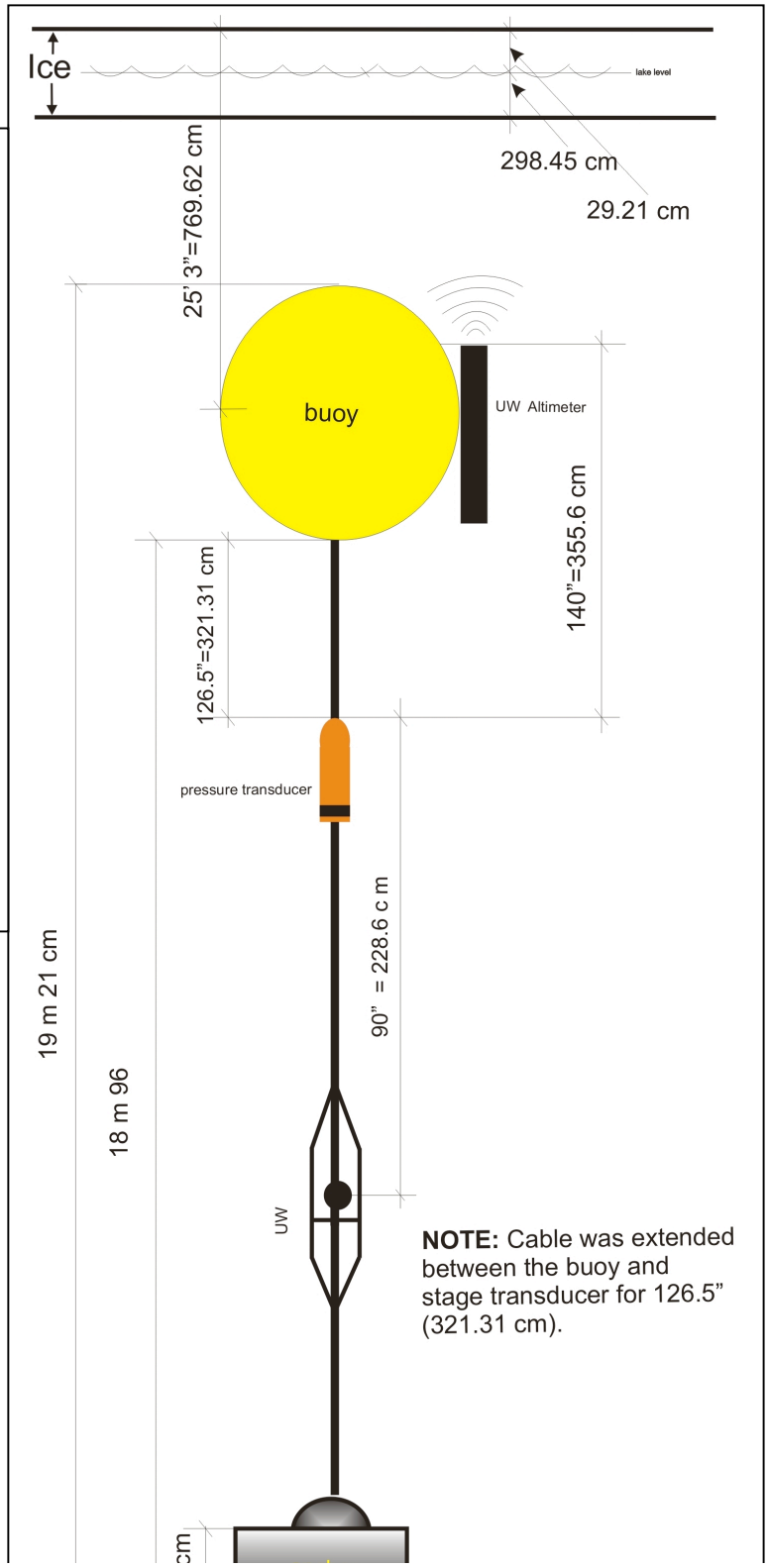
UW PAR

MULT ? (in air)
-239.65 (in water)
tcoff: 0.0036
Units: µmol $s^{-1}m^{-2}$
Serial#: SPQA2868
Cal. Date: LiCor

Back of laminated card:
µmol $s^{-1}m^{-2}$ (instantaneous)
µmol m^{-2} (integrated)
µmol $s^{-1}m^{2}$ (average)

Surface PAR

Serial: Q99253
Calibration: April 4, 2008
Output: 4.06 milivolts/1000µmol s-1 m-2
603.0 Ω resistor installed



3.1.3 East Lake of Lake Bonney

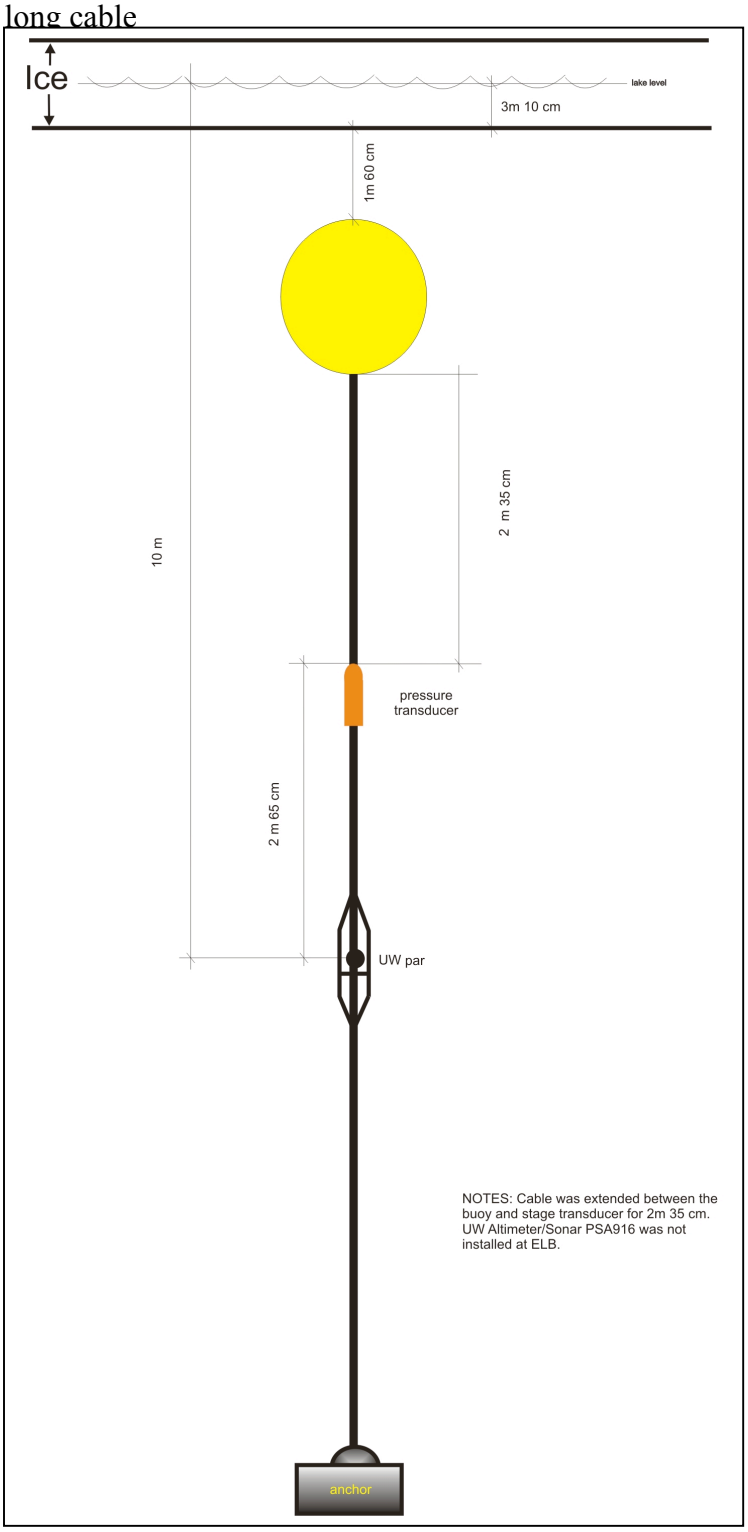
- Installed a PSA-916 sonar with 10m long cable
- Replaced surface PAR

● New sensor information (2008/09):

Surface PAR
Serial: Q30803
Calibration: April 4, 2008
Output: 4.48 milivolts/1000µmol s-1 m-2
604.0 Ω resistor installed

PSA-916 sonar
Serial: 40266

Surface PAR
Serial: Q30803
Calibration: April 4, 2008
Output: 4.48 milivolts/1000µmol s-1 m-2
604.0 Ω resistor installed

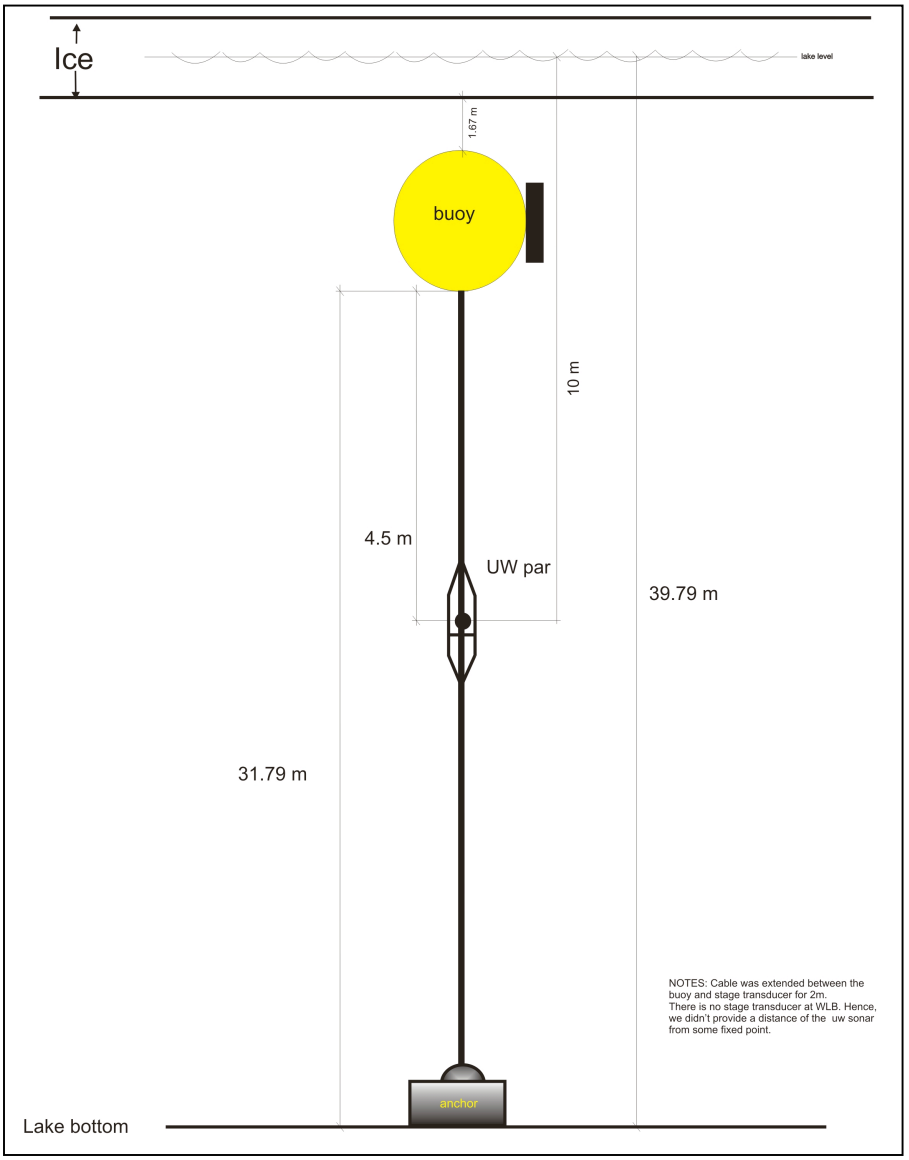


3.1.4 West Lobe of Lake Bonney

- Replaced surface PAR

- New sensor information (2008/09):

Surface PAR
Serial: Q22174
Calibration: April 4, 2008
Output: 2.93 milivolts/1000µmol s-1 m-2
603.0 Ω resistor installed



3.2 Software Changes

● All lakes were uploaded with a new program that included PSA-916 sonar.

3.2.1 Note on Multipliers

In the program running at Blue Boxes, we always have multiplier value of:
● 100 for UW par (or "–100" - depending on how we wired when we installed sensor).
● 200 for Surface PAR.

4. Programs and CR10x Data logger Wiring

4.1 Lake Fryxell

3.1.1 Wiring

; Control Ports

CP1 – not used
CP2 – not used
CP3 - jump wire to SW 12V CTRL (Fryxel, ELB & WLB)
CP4 - white (input)
CP5 - blue (output)

sw 12V CTRL - jump wire to CP3
sw 12V – Red wire from PSA-916
G - black wire from PSA-916

Stage transducer (instrument with desiccant-filled vent tube)

Red	E1
Orange	H5
Black	L5
Yellow	H6
Blue	L6
White	AG (any one)
Clear	G (any one)

Ablation transducer (instrument with desiccant-filled vent tube)

Red	E1
Orange	H1
Black	L1
Yellow	H2
Blue	L2
White	AG (any one)
Clear	G (any one)

Underwater PAR

Green	3L
Blue	3H

Surface PAR

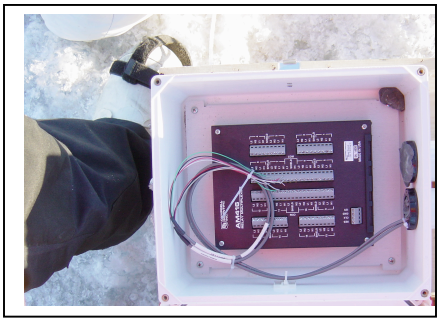
Black	4L
Red	4H
Clear	G

Jumper Wire from Black 4L to AG

Underwater Altimeter

Green Not used - (analog output ground)
Orange Not used - (analog output)
Red 12V - (external power 6-12V) - it goes to SW 12 V
Black G - (Ground)
White C4 - (RS-232 External Key Input)
Blue C5 - (RS-232 Output/Error Output)

AM416 Relay Multiplexer:
Two gray wires inside the multiplexer box are not connected:



4.1.2 Program

```
;{CR10X}
*Table 1 Program
  01: 60        Execution Interval (seconds)


; MEASURE ABLATION SENSOR
1:  Full Bridge w/mv Excit (P9)
 1: 1         Reps
 2: 5         2500 mV Slow Ex Range
 3: 3         25 mV Slow Br Range
 4: 1         DIFF Channel
 5: 1         Excite all reps w/Exchan 1
 6: 2500      mV Excitation
 7: 1         Loc [ ablat_cm  ]
 8: 102.22    Mult
 9: 0.0       Offset
;NOTE ABLATE SENSOR PROGRAMMED BUT NOT INSTALLED WHEN
PROGRAM STARTED IN NOV 2004. NEED TO PUT IN CORRECT MULTIPLIER
;
;MEASURE UNDERWATER LIGHT
2:  Volt (Diff) (P2)
 1: 1         Reps
 2: 2         7.5 mV Slow Range
 3: 3         DIFF Channel
 4: 2         Loc [ uwlight   ]
 5: -100      Mult
 6: 0.0       Offset

; MEASURE  SURFACE LIGHT SENSOR (QUANTUM)
3:  Volt (Diff) (P2)
 1: 1         Reps
 2: 3         25 mV Slow Range
 3: 4         DIFF Channel
 4: 3         Loc [ par       ]
 5: 200       Mult
 6: 0.0       Offset

; MEASURE BATTERY VOLTAGE
4:  Batt Voltage (P10)
 1: 4         Loc [ battvolts ]

; MEASURE STAGE TRANSDUCER SENSOR
5:  Full Bridge w/mv Excit (P9)
 1: 1         Reps
```

```
 2: 5        2500 mV Slow Ex Range
 3: 3        25 mV Slow Br Range
 4: 5        DIFF Channel
 5: 1        Excite all reps w/Exchan 1
 6: 2500     mV Excitation
 7: 5        Loc [ stage_cm  ]
 8: 102.12   Mult
 9: 0.0      Offset

;TIME INTERVAL SETUP FOR 20 MINUTES
6:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 20       Interval (same units as above)
 3: 10       Set Output Flag High

; SETUP STORAGE AREA & ARRY ID TO 1
7:  Set Active Storage Area (P80)^11360
 1: 1        Final Storage Area 1
 2: 1        Array ID

; SETUP TIME
8:  Real Time (P77)^19867
 1: 1220     Year,Day,Hour/Minute (midnight = 2400)

9:  Resolution (P78)
 1: 1        High Resolution

;CALCULATE AVERAGE FOR ABLAT_CM
10:  Average (P71)^1315
 1: 1        Reps
 2: 1        Loc [ ablat_cm  ]

;CALCULATE AVERAGE FOR STAGE_CM
11:  Average (P71)^20053
 1: 1        Reps
 2: 5        Loc [ stage_cm  ]


12:  Resolution (P78)
 1: 0        Low Resolution

;CALCULATE AVERAGE FOR, UWLIGHTS, PAR, AND BATTERY VOLTS
13:  Average (P71)^23697
 1: 3        Reps
 2: 2        Loc [ uwlight   ]
```

```
; COMUNICATE WITH STORAGE AREA
14:  Serial Out (P96)
 1: 71       Storage Module



;*************************************************************************
;***********PROGRAM FOR PSA-916 AT LAKE
FRYXELL**********************
;*************************************************************************



     ;*********Internal Swiches in PSA-916*****
          ;1-ON
          ;2-OFF
          ;3-ON
          ;4-ON
          ;5-ON
          ;6-ON
          ;7-OFF
          ;8-OFF



   ;*****************************************
   ; *** PSA-916 & CR10X WIRING AND CONTROLS ***
   ;*****************************************

;Green  Not used - (analog output ground)
;Orange Not used - (analog output)
;Red    12V      - (external power 6-12V) - it goes to SW 12 V
;Black  G        - (Ground)
;White  C4       - (RS-232 External Key Input)
;Blue   C5       - (RS-232 Output/Error Output)

; Control Ports

     ;   CP1 -

     ;   CP2 -

     ;   CP3 - jump wire to SW 12V CTRL (Fryxel, ELB & WLB)

     ;   CP4 - white (input)
```

```
      ;   CP5 - blue (output)

;SW
      ;sw 12V CTRL - jump wire to CP3

      ;sw 12V - Red
;12V
      ; red wire from PSA-916
      ; G - black wire from PSA-916




;-----------------------------------------------------------------

;  Start program every 2 hours

*Table 2 Program
  02: 7200      Execution Interval (seconds)

; If is midnight, exicute program

1:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 1440     Interval (same units as above)
 3: 30       Then Do


      ;*********************************************
      ;************ Turn Port 3 High to ***********
      ;************ turn ON Switched 12 V***********
      ;*********************************************

2:  Do (P86)
 1: 43       Set Port 3 High


3:  Excitation with Delay (P22)
 1: 2        Ex Channel
 2: 0000     Delay W/Ex (0.01 sec units)
 3: 100      Delay After Ex (0.01 sec units)
 4: 0000     mV Excitation


      ;*********************************************
      ;****Turn Port 4 High to set trigger high*****
```

```
;**********  (pause sounder)  ***************
;*******************************************
```

4: Do (P86)
 1: 44   Set Port 4 High

```
;*********************************
; Reset read ascii values to 00000
;*********************************
```

;---

; Decimal 48= 0 ASCII

5: Z=F x 10^n (P30)
 1: 48   F
 2: 00   n, Exponent of 10
 3: 6   Z Loc [Uascii_0]

;---

; Decimal 48= 0 ASCII

6: Z=F x 10^n (P30)
 1: 48   F
 2: 00   n, Exponent of 10
 3: 7   Z Loc [Uascii_1]

;---

; Decimal 48= 0 ASCII

7: Z=F x 10^n (P30)
 1: 48   F
 2: 00   n, Exponent of 10
 3: 8   Z Loc [Uascii_2]

;---

; Decimal 48= 0 ASCII

```
8:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 9        Z Loc [ Uascii_3  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

9:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 10       Z Loc [ Uascii_4  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

10:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 11       Z Loc [ Uascii_5  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

11:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 12       Z Loc [ Uascii_6  ]


    ;*************************************************************
    ; Set port 4 low to trigger measurement with Benthos PSA-916
    ;*************************************************************

; * set port 4 low to click under ice sounder

   12:  Do (P86)
    1: 54       Set Port 4 Low


    ;*************************************************************
    ;**********Read Benthos PSA-916 RS232 output******************
    ;*************************************************************
```

; Instruction P15 uses CP4 as DTR, CP5 as RX.

; Receives ascii string as binary RS232, 1200 baud (set inside sounder

; with dip switches, see the above). Term char 13 = <cr>. Output is"Rxx.xx(E)<cr><lf>"

; i.e. 8(9) characters (the E and 9 if no echo rec'd).

; P63 identifies 82 (Ascii R) as leading

; tag, and P15 identifies <cr> as end tag. Five values xx.xx (Ascii . = 46)

; read into 5 input locations from Uascii_1

; DTR - Data Transmit?
; RTS - Receive to Send
; CTS - Clear To Send
; RX - Receive ?
; TX - Transmit ?
; <cr> - carrage return
; <lf> - line feed

13: Port Serial I/O (P15)
1: 1 Reps
2: 21 8-Bit, RS-232 Binary, 1200 Baud
3: 1 Delay (0.01 sec units) before TX
4: 4 First (RTS/DTR) of Control Ports Used
5: 13 Start Loc for TX [junk]
6: 0 Number of Locs to TX
7: 13 Termination Character for RX
8: 9 Max Characters to RX
9: 200 Time Out for CTS (TX) and/or RX (0.01 sec units)
10: 7 Start Loc for RX [Uascii_1]
11: 1.0 Mult for RX
12: 0.0 Offset for RX

;* * * filter to synchronize the output stream Data output starts with an R.
; 82 = R in ASCII table for alphabethic characters

14: Extended Parameters (P63)
1: 82 Option
2: 0 Option
3: 0 Option

```
   4: 0       Option
   5: 0       Option
   6: 0       Option
   7: 0       Option
   8: 0       Option


   ;*****************************************
   ; Set port 4 high to end measure mode PSA-916
   ;*****************************************


  15:  Do (P86)
   1: 44      Set Port 4 High


   ;*****************************************
   ;****Conversion from ASCII to Decimal*******
   ;*****************************************

; Conversion from ascii (Uascii_1 .. _5) to decimal.

; Using subroutine 1 for conversion digit by digit

; First put all readings in same format: xx.xx
; If format is x.xx insert a leading zero after
; bumping digits along one place.


;------------------------ set if format 0X.XXX-------------------------
;If second location is "." than move digit one place to the right


  16:  If (X<=>F) (P89)
   1: 8       X Loc [ Uascii_2  ]
   2: 1       =
   3: 46      F
   4: 30      Then Do


;---------move value from location 4 to location 5-------------

    17:  Z=X (P31)
     1: 10      X Loc [ Uascii_4  ]
     2: 11      Z Loc [ Uascii_5  ]
```

```
;---------move value from location 3 to location 4-------------

        18:  Z=X (P31)
         1: 9        X Loc [ Uascii_3  ]
         2: 10       Z Loc [ Uascii_4  ]


;----------move value from location 2 to location 3------------

        19:  Z=X (P31)
         1: 8        X Loc [ Uascii_2  ]
         2: 9        Z Loc [ Uascii_3  ]

;----------move value from location 1 to location 2-------------

        20:  Z=X (P31)
         1: 7        X Loc [ Uascii_1  ]
         2: 8        Z Loc [ Uascii_2  ]


;-----Insert "0" at the first location (decimal 48=0 ascii)-----


        21:  Z=F x 10^n (P30)
         1: 48       F
         2: 00       n, Exponent of 10
         3: 7        Z Loc [ Uascii_1  ]




    22:  End (P95)




    ;************************************
    ;***Now do digit by digit conversion***
    ;************************************

;-------------------first digit-------------------------------
; Take value from location Uascii_1 and put it in location num
;(num location is further use in a subroutine).
; Make location multiple to be 10. Reduce this value for 0.1 for
; eqach incoming digit conversion (see end of soubroutine).
```

```
; Value in location Convert returned from the end of soubroutine
; and store in location Under.
; After first digit is saved, next digits
; added in next decimal place via decrease in location 'multipl'
; by factor of 10 (see end of soubroutine).

    23:  Z=X (P31)
     1: 7       X Loc [ Uascii_1  ]
     2: 14      Z Loc [ num       ]

    24:  Z=F x 10^n (P30)
     1: 1       F
     2: 1       n, Exponent of 10
     3: 15      Z Loc [ mult      ]

    25:  Do (P86)
     1: 1       Call Subroutine 1

    26:  Z=X (P31)
     1: 16      X Loc [ Convert   ]
     2: 17      Z Loc [ Under     ]

;--------------------Now second digit---------------------------------


    27:  Z=X (P31)
     1: 8       X Loc [ Uascii_2  ]
     2: 14      Z Loc [ num       ]

    28:  Do (P86)
     1: 1       Call Subroutine 1

    29:  Z=X+Y (P33)
     1: 17      X Loc [ Under     ]
     2: 16      Y Loc [ Convert   ]
     3: 17      Z Loc [ Under     ]

;--------------------Now third digit---------------------------------
;skipping decimal point at third location (Uascii_3)


    30:  Z=X (P31)
     1: 10      X Loc [ Uascii_4  ]
     2: 14      Z Loc [ num       ]

    31:  Do (P86)
```

```
   1: 1       Call Subroutine 1

  32:  Z=X+Y (P33)
   1: 17      X Loc [ Under     ]
   2: 16      Y Loc [ Convert   ]
   3: 17      Z Loc [ Under     ]

;-------------------Now fourth digit--------------------------------

  33:  Z=X (P31)
   1: 11      X Loc [ Uascii_5  ]
   2: 14      Z Loc [ num       ]

  34:  Do (P86)
   1: 1       Call Subroutine 1

  35:  Z=X+Y (P33)
   1: 17      X Loc [ Under     ]
   2: 16      Y Loc [ Convert   ]
   3: 17      Z Loc [ Under     ]

;---------------------------------------------------------------------
; Uascii_6 skipped. This Location is only for "E" or 9 as an error message
; - no signal recived.



  ;************************************************
  ; ************** OUTPUT ************************
  ;************************************************



  36:  Do (P86)
   1: 10      Set Output Flag High (Flag 0)



;Setup Storage Area & Arrey ID to 12 for Fryxell, 22 Hoare
;32 ELB and 42 WLB

  37:  Set Active Storage Area (P80)^6101
   1: 1       Final Storage Area 1
   2: 12      Array ID
```

```
; Setup Time

    38:  Real Time (P77)^1075
     1: 1220     Year,Day,Hour/Minute (midnight = 2400)

;Setup Resolution

    39:  Resolution (P78)
     1: 0        Low Resolution


    40:  Sample (P70)^9038
     1: 1        Reps
     2: 17       Loc [ Under     ]


; Activate Serial Output: Comunicate with storage area - send
; data to SM4M for storage.

    41:  Serial Out (P96)
     1: 71       Storage Module




    ;*********************************************
    ;************* Turn Port 3 Low to ************
    ;************* turn OFF Swiched 12 V**********
    ;*********************************************


    42:  Do (P86)
     1: 53       Set Port 3 Low


; Closing If statement/command/P92

43:  End (P95)

;*********************************************************
*Table 3 Subroutines
;*********************************************************
```

```
; ------ Subroutine for  ascii to decimal conversion -----

; Benthos sounder return depth in ascii format. Convert digit-by-digit

; to decimal here, then save as one number [Convert] for output.


1:  Beginning of Subroutine (P85)
 1: 1        Subroutine 1

;--------------***Decimal 48=0 ASCII ****----------------------
; Take value from location num and if this value is equal 48
; than put it in location Convert .


    2:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 48       F
     4: 30       Then Do


        3:  Z=F x 10^n (P30)
         1: 0        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    4:  End (P95)

;--------------***Decimal 49=1 ASCII ****----------------------

    5:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 49       F
     4: 30       Then Do


        6:  Z=F x 10^n (P30)
         1: 1        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    7:  End (P95)
```

```
;--------------***Decimal 50=2 ASCII ****----------------------

    8:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 50       F
     4: 30       Then Do


        9:  Z=F x 10^n (P30)
         1: 2        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    10:  End (P95)

;--------------***Decimal 51=3 ASCII ****----------------------

    11:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 51       F
     4: 30       Then Do


        12:  Z=F x 10^n (P30)
         1: 3        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    13:  End (P95)

;--------------***Decimal 52=4 ASCII ****----------------------

    14:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 52       F
     4: 30       Then Do


        15:  Z=F x 10^n (P30)
         1: 4        F
```

```
        2: 00       n, Exponent of 10
        3: 16       Z Loc [ Convert   ]


    16:  End (P95)

;--------------***Decimal 53=5 ASCII ****----------------------

    17:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 53       F
     4: 30       Then Do


        18:  Z=F x 10^n (P30)
         1: 5        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    19:  End (P95)

;--------------***Decimal 54=6 ASCII ****----------------------

    20:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 54       F
     4: 30       Then Do


        21:  Z=F x 10^n (P30)
         1: 6        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    22:  End (P95)

;--------------***Decimal 55=7 ASCII ****----------------------


    23:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
```

```
 3: 55       F
 4: 30       Then Do


    24:  Z=F x 10^n (P30)
     1: 7       F
     2: 00      n, Exponent of 10
     3: 16      Z Loc [ Convert   ]


  25:  End (P95)

;--------------***Decimal 56=8 ASCII ****----------------------


  26:  If (X<=>F) (P89)
   1: 14      X Loc [ num       ]
   2: 1       =
   3: 56      F
   4: 30      Then Do


    27:  Z=F x 10^n (P30)
     1: 8       F
     2: 00      n, Exponent of 10
     3: 16      Z Loc [ Convert   ]


  28:  End (P95)

;--------------***Decimal 57=9 ASCII ****----------------------

  29:  If (X<=>F) (P89)
   1: 14      X Loc [ num       ]
   2: 1       =
   3: 57      F
   4: 30      Then Do


    30:  Z=F x 10^n (P30)
     1: 9       F
     2: 00      n, Exponent of 10
     3: 16      Z Loc [ Convert   ]


  31:  End (P95)
```

```
;*****************************************************
;***save decimal as one number [Cononvert] for output***
;*****************************************************

; take value from Convert, multiple and put back in location Convert
; (rewrite location Convert)
; This value in location Convert returne back in Table 1
; and add it in location Under.

    32:  Z=X*Y (P36)
     1: 16      X Loc [ Convert   ]
     2: 15      Y Loc [ mult      ]
     3: 16      Z Loc [ Convert   ]


    33:  Z=X*F (P37)
     1: 15      X Loc [ mult      ]
     2: .1      F
     3: 15      Z Loc [ mult      ]



34:  End (P95)

End Program
```

4.1.3 Input Locations-

-Input Locations-
1 ablat_cm 5 1 1
2 uwlight 1 1 1
3 par 1 1 1
4 battvolts 1 1 1
5 stage_cm 1 1 1
6 Uascii_0 1 0 1
7 Uascii_1 1 2 3
8 Uascii_2 1 3 2
9 Uascii_3 1 1 2
10 Uascii_4 1 2 2
11 Uascii_5 1 1 2
12 Uascii_6 1 0 1
13 junk 1 1 0
14 num 1 10 4
15 mult 1 2 2
16 Convert 1 5 11
17 Under 1 4 4
18 _____ 0 0 0
19 _____ 0 0 0
20 _____ 0 0 0
21 _____ 0 0 0
22 _____ 0 0 0
23 _____ 0 0 0
24 _____ 0 0 0
25 _____ 0 0 0
26 _____ 0 0 0
27 _____ 0 0 0
28 _____ 0 0 0

-Program Security-
0000
0000
0000
-Mode 4-
-Final Storage Area 2-
0
-CR10X ID-
0
-CR10X Power Up-
3
-CR10X Compile Setting-
3
-CR10X RS-232 Setting-
-1
-DLD File Labels-
0
-Final Storage Labels-
0,1,18689
1,Year_RTM,19867
1,Day_RTM
1,Hour_Minute_RTM
2,ablat_cm_AVG~1,1315
3,uwlight_AVG~2,23697
3,par_AVG~3

3,battvolts_AVG~4
4,stage_cm_AVG~5,20053
5,12,32576
6,Year_RTM,1075
6,Day_RTM
6,Hour_Minute_RTM
7,Under~17,9038

4.1.4 Final storage array definition (*.FSL file)

Final Storage Label File for: FRL PROGRAM 2007 UPDATED FOR PSA 916.csi
Date: 7/2/2008
Time: 15:46:32

1 Output_Table 20.00 Min
1 1 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 ablat_cm_AVG H
6 stage_cm_AVG H
7 uwlight_AVG L
8 par_AVG L
9 battvolts_AVG L

12 Output_Table 7200.00 Sec
1 12 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 Under L

Estimated Total Final Storage Locations used per day 852

4.2 Lake Hoare

4.2.1 Wiring

Control Ports:
CP1 - jump wire to SW 12V CTRL
CP2 - used for Green for Mux (see wiring descritpion in Table 1)
CP3 - used for White for Mux (see wiring descritpion in Table 1)
CP4 - white (input)
CP5 - blue (output)

sw 12V CTRL - jump wire to CP1
sw 12V – Red wire from PSA-916
G - black wire from PSA-916

Stage transducer jumper wire (3 wires with black heat shrunk resistor)

Yellow	1H
Black	1L (the black stage transducer wire also goes in here, see below)
Purple	E1 (together with purple from other jumper)

Stage transducer (instrument with dessicant filled vent tube, this one will have more pink dessicant than the other)

Black	1L (the Black stage transducer jumper wire also goes here, see above)
Red	2H
Green	2L
White	AG (any one)
Blue	G (any one)

Ablation transducer jumper wire (3 wires with black heat shrunk resistor)

Yellow	3H
Black	3L (the black ablation transducer wire also goes in here, see below)
Purple	E1 (together with purple from other jumper)

Ablation transducer (instrument with dessicant filled vent tube, this one will have less pink dessicant than the other)

Black	3L (the Black ablation transducer jumper wire also goes here, see above)	
Red	4H	
Green	4L	
White	AG (any one)	
Blue	G (any one)	this one was disconected

Mux signal cable

Red	5H
White	5L
Black	E2
Green	AG (any one)

Note: Green should be connected to AG according to BBB 04/05. However, at CR10 was connected at G (right to AG port). Because everything works properly, it was left at G when Medved rebuild blue boxes in 2006.

Clear G (any one)

Mux power/reset cable
Red 12V (any one)
White C3
Green C2
Black G (any one)
Clear G (any one)

Disconnected at CR10x in order to connect PSA-916

Surface PAR
Red 6H
Black 6L
Clear G
Jumper Wire from Black 6L to AG

Underwater Altimeter/Sonar
Green Not used - (analog output ground)
Orange Not used - (analog output)
Red 12V - (external power 6-12V) - it goes to SW 12 V
Black G - (Ground)
White C4 - (RS-232 External Key Input)
Blue C5 - (RS-232 Output/Error Output)

Multiplexer Wiring

Underwater light (PAR) sensor (twisted blue and green wires)
Blue Set 1, H1
Green Set 1, L1

Mux Common wires

Mux signal cable
Red Com, H1
White Com, L1
Black Com, L2
Green Com, H2
Clear Com, Shield

Mux power/reset cable
Red 12V
White CLK
Green RES
Black GND
Clear Com, Shield

4.2.2 Program

```
;*********************PROGRAM*******************************

*Table 1 Program
  01: 60.0000   Execution Interval (seconds)

1:  Batt Voltage (P10)
 1: 7        Loc [ volts     ]

2:  Do (P86)
 1: 1        Call Subroutine 1

3:  Do (P86)
 1: 2        Call Subroutine 2

4:  Volt (Diff) (P2)
 1: 1        Reps
 2: 3        25 mV Slow Range
 3: 6        DIFF Channel
 4: 27       Loc [ surflight ]
 5: 200      Mult
 6: 0.0      Offset

5:  Do (P86)
 1: 42       Set Port 2 High

6:  Do (P86)
 1: 73       Pulse Port 3

7:  Volt (Diff) (P2)
 1: 1        Reps
 2: 1        2.5 mV Slow Range
 3: 5        DIFF Channel
 4: 26       Loc [ uwlight   ]
 5: -100     Mult
 6: 0        Offset

8:  Do (P86)
 1: 52       Set Port 2 Low

9:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 20       Interval (same units as above)
 3: 10       Set Output Flag High (Flag 0)
```

10: Set Active Storage Area (P80)^17914
1: 1 Final Storage Area 1
2: 2 Array ID

11: Real Time (P77)^2775
1: 1220 Year,Day,Hour/Minute (midnight = 2400)

12: Average (P71)^14666
1: 1 Reps
2: 4 Loc [stage_psi]

13: Standard Deviation (P82)^24383
1: 1 Reps
2: 4 Sample Loc [stage_psi]

14: Resolution (P78)
1: 1 High Resolution

15: Average (P71)^5591
1: 1 Reps
2: 5 Loc [stage_cm]

16: Resolution (P78)
1: 0 Low Resolution

17: Average (P71)^19363
1: 2 Reps
2: 26 Loc [uwlight]

18: Average (P71)^7360
1: 1 Reps
2: 14 Loc [ablat_psi]

19: Standard Deviation (P82)^25199
1: 1 Reps
2: 14 Sample Loc [ablat_psi]

20: Resolution (P78)
1: 1 High Resolution

21: Average (P71)^6446
1: 1 Reps
2: 15 Loc [ablat_cm]

22: Resolution (P78)
1: 0 Low Resolution

```
23:  Average (P71)^7857
 1: 1        Reps
 2: 7        Loc [ volts     ]

24:  Serial Out (P96)
 1: 71       Storage Module

;*************************************************************************
;***********PROGRAM FOR PSA-916 AT LAKE
HOARE************************
;*************************************************************************


    ;*********Internal Swiches in PSA-916*****
            ;1-ON
            ;2-OFF
            ;3-ON
            ;4-ON
            ;5-ON
            ;6-ON
            ;7-OFF
            ;8-OFF



  ;*****************************************
  ; *** PSA-916 & CR10X WIRING AND CONTROLS ***
  ;*****************************************

;Green  Not used - (analog output ground)
;Orange Not used - (analog output)
;Red    12V      - (external power 6-12V) - it goes to SW 12 V
;Black  G        - (Ground)
;White  C4       - (RS-232 External Key Input)
;Blue   C5       - (RS-232 Output/Error Output)

; Control Ports

    ;   CP1 - jump wire to SW 12V CTRL

    ;   CP2 - used for Green for Mux (see wiring descritpion in Table 1)

    ;   CP3 - used for White for Mux (see wiring descritpion in Table 1)

    ;   CP4 - white (input)
```

```
      ;   CP5 - blue (output)

;SW
      ;sw 12V CTRL - jump wire to CP1

      ;sw 12V - Red
;12V
      ; red wire from PSA-916
      ;G - black wire from PSA-916




;---------------------------------------------------------------------

;  Start program every 2 hours
*Table 2 Program
  01: 7200      Execution Interval (seconds)

; If is midnight, exicute program

1:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 1440     Interval (same units as above)
 3: 30       Then Do


      ;*********************************************
      ;************ Turn Port 1 High to ***********
      ;************ turn ON Swiched 12 V***********
      ;*********************************************

2:  Do (P86)
 1: 41       Set Port 1 High

;This instruction is necessary to get a reading for PSA916.
;Without this instruction you'll get zero reading.
;This command will delayed program for amount of time to
;allow voltage to rise to certain level necessary for
;the sensor to operate. I experimented in Polar Haven at the Lake Fryxell
;using a Limno team hole with 500,400,300,200 and 100 at line 3.
;They all produced a good result and I left 1 second delay
;to save on power budget.
```

3: Excitation with Delay (P22)
 1: 2   Ex Channel
 2: 0000   Delay W/Ex (0.01 sec units)
 3: 200   Delay After Ex (0.01 sec units)
 4: 0000   mV Excitation

```
;*****************************************
;****Turn Port 4 High to set trigger high*****
;**********  (pause sounder)  ***************
;*****************************************
```

4: Do (P86)
 1: 44   Set Port 4 High

```
;*********************************
; Reset read ascii values to 00000
;*********************************
```

;---

; Decimal 48= 0 ASCII

5: Z=F x 10^n (P30)
 1: 48   F
 2: 00   n, Exponent of 10
 3: 28   Z Loc [Uascii_0]

;---

; Decimal 48= 0 ASCII

6: Z=F x 10^n (P30)
 1: 48   F
 2: 00   n, Exponent of 10
 3: 29   Z Loc [Uascii_1]

;---

; Decimal 48= 0 ASCII

7: Z=F x 10^n (P30)

```
 1: 48       F
 2: 00       n, Exponent of 10
 3: 30       Z Loc [ Uascii_2  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

8:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 31       Z Loc [ Uascii_3  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

9:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 32       Z Loc [ Uascii_4  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

10:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 33       Z Loc [ Uascii_5  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

11:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 34       Z Loc [ Uascii_6  ]


    ;*************************************************************
    ; Set port 4 low to trigger measurement with Benthos PSA-916
    ;*************************************************************

; * set port 4 low to click under ice sounder
```

12: Do (P86)
1: 54 Set Port 4 Low

;***
;**********Read Benthos PSA-916 RS232 output*****************
;***

; Instruction P15 uses CP4 as DTR, CP5 as RX.

; Receives ascii string as binary RS232, 1200 baud (set inside sounder

; with dip switches, see the above). Term char 13 = <cr>. Output is"Rxx.xx(E)<cr><lf>"

; i.e. 8(9) characters (the E and 9 if no echo receceived).

; P63 identifies 82 (Ascii R) as leading tag,

; and P15 identifies <cr> as end tag. Five values xx.xx (Ascii . = 46)

; read into 5 input locations from Uascii_1

; DTR - Data Transmit?
; RTS - Receive to Send
; CTS - Clear To Send
; RX - Receive
; TX - Transmit
; <cr> - carrage return
; <lf> - line feed

13: Port Serial I/O (P15)
1: 1 Reps
2: 21 8-Bit, RS-232 Binary, 1200 Baud
3: 1 Delay (0.01 sec units) before TX
4: 4 First (RTS/DTR) of Control Ports Used
5: 35 Start Loc for TX [junk]
6: 0 Number of Locs to TX
7: 13 Termination Character for RX
8: 9 Max Characters to RX
9: 200 Time Out for CTS (TX) and/or RX (0.01 sec units)
10: 29 Start Loc for RX [Uascii_1]
11: 1.0 Mult for RX
12: 0.0 Offset for RX

```
;* * * filter to synchronize the output stream Data output starts with an R.
; 82 = R in ASCII table for alphabethic characters

    14:  Extended Parameters (P63)
     1: 82       Option
     2: 0        Option
     3: 0        Option
     4: 0        Option
     5: 0        Option
     6: 0        Option
     7: 0        Option
     8: 0        Option


     ;*****************************************
     ; Set port 4 high to end measure mode PSA-916
     ;*****************************************


    15:  Do (P86)
     1: 44       Set Port 4 High


     ;*****************************************
     ;****Conversion from ASCII to Decimal*******
     ;*****************************************

; Conversion from ascii (Uascii_1 .. _5) to decimal.

; Using subroutine # 3 for conversion digit by digit

; First put all readings in same format: xx.xx
; If format is x.xx insert a leading zero after
; moving/bumping digits along one place.


;------------------------ set if format 0X.XXX-------------------------
;If second location is "." than move digit one place to the right


    16:  If (X<=>F) (P89)
     1: 30       X Loc [ Uascii_2  ]
     2: 1        =
     3: 46       F
     4: 30       Then Do
```

```
;---------move value from location 4 to location 5-------------

        17:  Z=X (P31)
         1: 32       X Loc [ Uascii_4  ]
         2: 33       Z Loc [ Uascii_5  ]

;---------move value from location 3 to location 4-------------

        18:  Z=X (P31)
         1: 31       X Loc [ Uascii_3  ]
         2: 32       Z Loc [ Uascii_4  ]


;----------move value from location 2 to location 3------------

        19:  Z=X (P31)
         1: 30       X Loc [ Uascii_2  ]
         2: 31       Z Loc [ Uascii_3  ]

;----------move value from location 1 to location 2-------------

        20:  Z=X (P31)
         1: 29       X Loc [ Uascii_1  ]
         2: 30       Z Loc [ Uascii_2  ]


;-----Insert "0" at the first location (decimal 48=0 ascii)-----


        21:  Z=F x 10^n (P30)
         1: 48       F
         2: 00       n, Exponent of 10
         3: 29       Z Loc [ Uascii_1  ]



    22:  End (P95)



    ;*************************************
```

```
      ;***Now do digit by digit conversion***
      ;************************************

;------------------first digit-------------------------------
; Take value from location Uascii_1 and put it in location num
;(num location is further use in a subroutine).
; Make location multiple to be 10. Reduce this value for 0.1 for
; eqach incoming digit conversion (see end of soubroutine).
; Value in location Convert returned from the end of soubroutine
; and store in location Under.
; After first digit is saved, next digits
; added in next decimal place via decrease in location 'multipl'
; by factor of 10 (see end of soubroutine).

    23:  Z=X (P31)
     1: 29       X Loc [ Uascii_1  ]
     2: 36       Z Loc [ num       ]

    24:  Z=F x 10^n (P30)
     1: 1        F
     2: 1        n, Exponent of 10
     3: 37       Z Loc [ mult      ]

    25:  Do (P86)
     1: 3        Call Subroutine 3

    26:  Z=X (P31)
     1: 38       X Loc [ Convert   ]
     2: 39       Z Loc [ Under     ]

;------------------Now second digit--------------------------------


    27:  Z=X (P31)
     1: 30       X Loc [ Uascii_2  ]
     2: 36       Z Loc [ num       ]

    28:  Do (P86)
     1: 3        Call Subroutine 3

    29:  Z=X+Y (P33)
     1: 39       X Loc [ Under     ]
     2: 38       Y Loc [ Convert   ]
     3: 39       Z Loc [ Under     ]

;------------------Now third digit---------------------------------
```

```
;skipping decimal point at third location (Uascii_3)


    30:  Z=X (P31)
     1: 32       X Loc [ Uascii_4  ]
     2: 36       Z Loc [ num       ]

    31:  Do (P86)
     1: 3        Call Subroutine 3

    32:  Z=X+Y (P33)
     1: 39       X Loc [ Under     ]
     2: 38       Y Loc [ Convert   ]
     3: 39       Z Loc [ Under     ]

;-------------------Now fourth digit---------------------------------

    33:  Z=X (P31)
     1: 33       X Loc [ Uascii_5  ]
     2: 36       Z Loc [ num       ]

    34:  Do (P86)
     1: 3        Call Subroutine 3

    35:  Z=X+Y (P33)
     1: 39       X Loc [ Under     ]
     2: 38       Y Loc [ Convert   ]
     3: 39       Z Loc [ Under     ]

;---------------------------------------------------------------------
; Uascii_6 skipped. This Location is only for "E" or 9 as an error message
; - no signal recived.



    ;***********************************************
    ; ************** OUTPUT FOR PSA_916**************
    ;***********************************************



    36:  Do (P86)
     1: 10       Set Output Flag High (Flag 0)
```

```
;Setup Storage Area & Arrey ID to 12 for Fryxell, 22 Hoare
;32 ELB and 42 WLB

    37:  Set Active Storage Area (P80)^4782
     1: 1       Final Storage Area 1
     2: 22       Array ID


; Setup Time

    38:  Real Time (P77)^3949
     1: 1220     Year,Day,Hour/Minute (midnight = 2400)

;Setup Resolution

    39:  Resolution (P78)
     1: 0       Low Resolution


    40:  Sample (P70)^15324
     1: 1       Reps
     2: 39       Loc [ Under     ]


; Activate Serial Output: Comunicate with storage area - send
; data to SM4M for storage.

    41:  Serial Out (P96)
     1: 71       Storage Module




    ;*****************************************
    ;************ Turn Port 1 Low to ***********
    ;************ turn OFF Swiched 12 V**********
    ;*****************************************


    42:  Do (P86)
     1: 51       Set Port 1 Low
```

; Closing If statement/command/P92

43: End (P95)

;**********
;**********SUBROUTINES**********
;**********

; SUBROUTINES RUN THE TWO PRESSURE TRANSDUCERS (DIFFERENT MODELS THAN AT OTHER LAKES)
; Third Soubroutine is for PSA-916 from Table 2

*Table 3 Subroutines

1: Beginning of Subroutine (P85)
1: 1 Subroutine 1

2: Full Bridge (P6)
1: 1 Reps
2: 3 25 mV Slow Range
3: 1 DIFF Channel
4: 1 Excite all reps w/Exchan 1
5: 770 mV Excitation
6: 1 Loc [_____]
7: .01 Mult
8: 0 Offset

3: Full Bridge (P6)
1: 1 Reps
2: 3 25 mV Slow Range
3: 2 DIFF Channel
4: 1 Excite all reps w/Exchan 1
5: 770 mV Excitation
6: 2 Loc [_____]
7: .5 Mult
8: 0 Offset

4: Z=X/Y (P38)
1: 2 X Loc [_____]
2: 1 Y Loc [_____]
3: 3 Z Loc [_____]

5: Z=X*F (P37)

1: 3 X Loc [_____]
2: .52 F
3: 4 Z Loc [stage_psi]

6: Z=X*F (P37)
1: 3 X Loc [_____]
2: 36.5 F
3: 5 Z Loc [stage_cm]

7: End (P95)

8: Beginning of Subroutine (P85)
1: 2 Subroutine 2

9: Full Bridge (P6)
1: 1 Reps
2: 3 25 mV Slow Range
3: 3 DIFF Channel
4: 1 Excite all reps w/Exchan 1
5: 770 mV Excitation
6: 11 Loc [_____]
7: .01 Mult
8: 0 Offset

10: Full Bridge (P6)
1: 1 Reps
2: 3 25 mV Slow Range
3: 4 DIFF Channel
4: 1 Excite all reps w/Exchan 1
5: 770 mV Excitation
6: 12 Loc [_____]
7: .5 Mult
8: 0 Offset

11: Z=X/Y (P38)
1: 12 X Loc [_____]
2: 11 Y Loc [_____]
3: 13 Z Loc [_____]

12: Z=X*F (P37)
1: 13 X Loc [_____]
2: .52 F
3: 14 Z Loc [ablat_psi]

13: Z=X*F (P37)
1: 13 X Loc [_____]

```
 2: 36.5     F
 3: 15       Z Loc [ ablat_cm  ]


14:  End (P95)


; -**********Subroutine for  ascii to decimal conversion**********

; Benthos sounder return depth in ascii format. Convert digit-by-digit

; to decimal here, then save as one number [Convert] for output.


15:  Beginning of Subroutine (P85)
 1: 3        Subroutine 3

;--------------***Decimal 48=0 ASCII ****----------------------
; Take value from location num and if this value is equal 48
; than put it in location Convert .


    16:  If (X<=>F) (P89)
     1: 36       X Loc [ num       ]
     2: 1        =
     3: 48       F
     4: 30       Then Do


        17:  Z=F x 10^n (P30)
         1: 0        F
         2: 00       n, Exponent of 10
         3: 38       Z Loc [ Convert   ]


    18:  End (P95)

;--------------***Decimal 49=1 ASCII ****----------------------

    19:  If (X<=>F) (P89)
     1: 36       X Loc [ num       ]
     2: 1        =
     3: 49       F
     4: 30       Then Do
```

```
        20:  Z=F x 10^n (P30)
         1: 1       F
         2: 00      n, Exponent of 10
         3: 38      Z Loc [ Convert   ]


    21:  End (P95)

;--------------***Decimal 50=2 ASCII ****----------------------

    22:  If (X<=>F) (P89)
     1: 36      X Loc [ num       ]
     2: 1       =
     3: 50      F
     4: 30      Then Do


        23:  Z=F x 10^n (P30)
         1: 2       F
         2: 00      n, Exponent of 10
         3: 38      Z Loc [ Convert   ]


    24:  End (P95)

;--------------***Decimal 51=3 ASCII ****----------------------

    25:  If (X<=>F) (P89)
     1: 36      X Loc [ num       ]
     2: 1       =
     3: 51      F
     4: 30      Then Do


        26:  Z=F x 10^n (P30)
         1: 3       F
         2: 00      n, Exponent of 10
         3: 38      Z Loc [ Convert   ]


    27:  End (P95)

;--------------***Decimal 52=4 ASCII ****----------------------

    28:  If (X<=>F) (P89)
     1: 36      X Loc [ num       ]
```

```
 2: 1        =
 3: 52       F
 4: 30       Then Do


    29:  Z=F x 10^n (P30)
     1: 4        F
     2: 00       n, Exponent of 10
     3: 38       Z Loc [ Convert   ]


30:  End (P95)

;--------------***Decimal 53=5 ASCII ****----------------------

31:  If (X<=>F) (P89)
 1: 36       X Loc [ num       ]
 2: 1        =
 3: 53       F
 4: 30       Then Do


    32:  Z=F x 10^n (P30)
     1: 5        F
     2: 00       n, Exponent of 10
     3: 38       Z Loc [ Convert   ]


33:  End (P95)

;--------------***Decimal 54=6 ASCII ****----------------------

34:  If (X<=>F) (P89)
 1: 36       X Loc [ num       ]
 2: 1        =
 3: 54       F
 4: 30       Then Do


    35:  Z=F x 10^n (P30)
     1: 6        F
     2: 00       n, Exponent of 10
     3: 38       Z Loc [ Convert   ]


36:  End (P95)
```

```
;--------------***Decimal 55=7 ASCII ****----------------------


    37:  If (X<=>F) (P89)
     1: 36       X Loc [ num       ]
     2: 1        =
     3: 55       F
     4: 30       Then Do


        38:  Z=F x 10^n (P30)
         1: 7        F
         2: 00       n, Exponent of 10
         3: 38       Z Loc [ Convert   ]


    39:  End (P95)

;--------------***Decimal 56=8 ASCII ****----------------------


    40:  If (X<=>F) (P89)
     1: 36       X Loc [ num       ]
     2: 1        =
     3: 56       F
     4: 30       Then Do


        41:  Z=F x 10^n (P30)
         1: 8        F
         2: 00       n, Exponent of 10
         3: 38       Z Loc [ Convert   ]


    42:  End (P95)

;--------------***Decimal 57=9 ASCII ****----------------------


    43:  If (X<=>F) (P89)
     1: 36       X Loc [ num       ]
     2: 1        =
     3: 57       F
     4: 30       Then Do
```

```
    44:  Z=F x 10^n (P30)
     1: 9       F
     2: 00      n, Exponent of 10
     3: 38      Z Loc [ Convert   ]


   45:  End (P95)

;*****************************************************
;***save decimal as one number [Cononvert] for output***
;*****************************************************

; take value from Convert, multiple and put back in location Convert
; (rewrite location convert)
; This value in location Convert returne back in Table 1
; and add it in location Under.

   46:  Z=X*Y (P36)
    1: 38      X Loc [ Convert   ]
    2: 37      Y Loc [ mult      ]
    3: 38      Z Loc [ Convert   ]


   47:  Z=X*F (P37)
    1: 37      X Loc [ mult      ]
    2: .1      F
    3: 37      Z Loc [ mult      ]



48:  End (P95)

End Program
```

4.2.3 Input storage (*6) locations

-Input Locations-
1 _____ 1 1 1
2 _____ 1 1 1
3 _____ 1 2 1
4 stage_psi 1 2 1
5 stage_cm 1 1 1
6 _____ 0 0 0
7 volts 1 1 1
8 _____ 0 0 0
9 _____ 0 0 0
10 _____ 0 0 0
11 _____ 1 1 1
12 _____ 1 1 1
13 _____ 1 2 1
14 ablat_psi 1 2 1
15 ablat_cm 1 1 1
16 _____ 0 0 0
17 _____ 0 0 0
18 _____ 0 0 0
19 _____ 0 0 0
20 _____ 0 0 0
21 _____ 0 0 0
22 _____ 0 0 0
23 _____ 0 0 0
24 _____ 0 0 0
25 _____ 0 0 0
26 uwlight 5 1 1
27 surflight 1 0 1
28 Uascii_0 1 0 1
29 Uascii_1 1 2 3
30 Uascii_2 1 3 2
31 Uascii_3 1 1 2
32 Uascii_4 1 2 2
33 Uascii_5 1 1 2
34 Uascii_6 1 0 1
35 junk 1 1 0
36 num 1 10 4
37 mult 1 2 2
38 Convert 1 5 11
39 Under 1 4 4

-Program Security-
0
0
0
-Mode 4-
-Final Storage Area 2-
0
-CR10X ID-
0
-CR10X Power Up-
0
-CR10X Compile Setting-
0
-CR10X RS-232 Setting-
-1
-DLD File Labels-
0
-Final Storage Labels-
0,2,9651
1,Year_RTM,2775
1,Day_RTM

1,Hour_Minute_RTM
2,stage_psi_AVG~4,14666
3,stage_psi_STD~4,24383
4,stage_cm_AVG~5,5591
5,uwlight_AVG~26,19363
5,surflight_AVG~27
6,ablat_psi_AVG~14,7360
7,ablat_psi_STD~14,25199
8,ablat_cm_AVG~15,6446
9,volts_AVG~7,7857
10,22,31386
11,Year_RTM,3949
11,Day_RTM
11,Hour_Minute_RTM
12,Under~39,15324

4.2.4 Final storage array definition

Final Storage Label File for: LAKE HOARE PROGRAM 2007 UPDATED FOR PSA-916 .csi
Date: 11/8/2008
Time: 23:17:33

2 Output_Table 20.00 Min
1 2 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 stage_psi_AVG L
6 stage_psi_STD L
7 stage_cm_AVG H
8 uwlight_AVG L
9 surflight_AVG L
10 ablat_psi_AVG L
11 ablat_psi_STD L
12 ablat_cm_AVG H
13 volts_AVG L

22 Output_Table 7200.00 Sec
1 22 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 Under L

Estimated Total Final Storage Locations used per day 1140

4.3 Lake Bonney East Lobe

4.3.1 Wiring

;Stage transducer (instrument with desiccant-filled vent tube)
;Red E1
;Orange H5
;Black L5
;Yellow H6
;Blue L6
;White AG (any one)
;Clear G (any one)
;
;Ablation transducer (instrument with desiccant-filled vent tube)
;Red E1
;Orange H1
;Black L1
;Yellow H2
;Blue L2
;White AG (any one)
;Clear G (any one)
;
;Underwater PAR
;Green 3L
;Blue 3H

;Surface PAR
;Black 4L
;Red 4H
;Clear G
;Jumper Wire from Black 4L to AG

;AM416 Relay Multiplexer: Present but not connected

;PSA-916 Wiring

;*********Internal Swiches in PSA-916*****
;1-ON
;2-OFF
;3-ON
;4-ON
;5-ON
;6-ON
;7-OFF
;8-OFF

;Green Not used - (analog output ground)
;Orange Not used - (analog output)
;Red 12V - (external power 6-12V) - it goes to SW 12 V
;Black G - (Ground)
;White C4 - (RS-232 External Key Input)
;Blue C5 - (RS-232 Output/Error Output)

; Control Ports

; CP1 -
; CP2 -
; CP3 - jump wire to SW 12V CTRL (Fryxel, ELB & WLB)
; CP4 - white (input)
; CP5 - blue (output)

;SW 12V
;sw 12V CTRL - jump wire to CP3

;sw 12V - Red

; Black - Ground

4.3.2 Program

```
;{CR10X}
;ELB PROGRAM 2007 UPDATED FOR PSA916- ALL IN ONE TABLE

*Table 1 Program
  01: 60.0000   Execution Interval (seconds)

; MEASURE ABLATION SENSOR
1:  Full Bridge w/mv Excit (P9)
 1: 1        Reps
 2: 5        2500 mV Slow Ex Range
 3: 3        25 mV Slow Br Range
 4: 1        DIFF Channel
 5: 1        Excite all reps w/Exchan 1
 6: 2500     mV Excitation
 7: 1        Loc [ ablat_cm  ]
 8: 102.21   Mult
 9: 0.0      Offset

; MEASURE UNDERVATER LIGHT  (UW PAR)
2:  Volt (Diff) (P2)
 1: 1        Reps
 2: 2        7.5 mV Slow Range
 3: 3        DIFF Channel
 4: 2        Loc [ uwlight   ]
 5: -100     Mult
 6: 0.0      Offset

;MEASURE SURFACE LIGHT (PAR)
3:  Volt (Diff) (P2)
 1: 1        Reps
 2: 3        25 mV Slow Range
 3: 4        DIFF Channel
 4: 3        Loc [ par       ]
 5: 200      Mult
 6: 0.0      Offset

;MEASURE BATTERY VOLTAGE
4:  Batt Voltage (P10)
 1: 4        Loc [ battvolts ]

; MEASURE STAGE TRANSDUCER VOLTAGE
5:  Full Bridge w/mv Excit (P9)
 1: 1        Reps
```

2: 5 2500 mV Slow Ex Range
3: 3 25 mV Slow Br Range
4: 5 DIFF Channel
5: 1 Excite all reps w/Exchan 1
6: 2500 mV Excitation
7: 5 Loc [stage_cm]
8: 101.53 Mult
9: 0.0 Offset

```
;*****************************************************************
;**************PSA-916 MEASURMENT*********************************
;*****************************************************************
```

; If is midnight, exicute program

6: If time is (P92)
1: 0 Minutes (Seconds --) into a
2: 1440 Interval (same units as above)
3: 30 Then Do

```
    ;*******************************************
    ;************ Turn Port 3 High to ***********
    ;************ turn ON Switched 12 V***********
    ;*******************************************
```

7: Do (P86)
1: 43 Set Port 3 High

8: Excitation with Delay (P22)
1: 2 Ex Channel
2: 0000 Delay W/Ex (0.01 sec units)
3: 100 Delay After Ex (0.01 sec units)
4: 0000 mV Excitation

```
    ;*******************************************
    ;****Turn Port 4 High to set trigger high*****
    ;**********  (pause sounder)  ***************
    ;*******************************************
```

9: Do (P86)

```
 1: 44       Set Port 4 High




     ;*********************************
     ; Reset read ascii values to 00000
     ;*********************************

;-----------------------------------------------------------------

; Decimal 48= 0 ASCII

10:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 6        Z Loc [ Uascii_0  ]

;-----------------------------------------------------------------

; Decimal 48= 0 ASCII

11:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 7        Z Loc [ Uascii_1  ]

;-----------------------------------------------------------------

; Decimal 48= 0 ASCII

12:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 8        Z Loc [ Uascii_2  ]

;-----------------------------------------------------------------

; Decimal 48= 0 ASCII

13:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 9        Z Loc [ Uascii_3  ]
```

```
;-------------------------------------------------------------

; Decimal 48= 0 ASCII

14:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 10       Z Loc [ Uascii_4  ]

;-------------------------------------------------------------

; Decimal 48= 0 ASCII

15:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 11       Z Loc [ Uascii_5  ]

;-------------------------------------------------------------

; Decimal 48= 0 ASCII

16:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 12       Z Loc [ Uascii_6  ]


    ;*************************************************************
    ; Set port 4 low to trigger measurement with Benthos PSA-916
    ;*************************************************************

; * set port 4 low to click under ice sounder

   17:  Do (P86)
    1: 54       Set Port 4 Low


    ;*************************************************************
    ;**********Read Benthos PSA-916 RS232 output******************
    ;*************************************************************

; Instruction P15 uses CP4 as DTR, CP5 as RX.

; Receives ascii string as binary RS232, 1200 baud (set inside sounder
```

; with dip switches, see the above). Term char 13 = <cr>. Output is"Rxx.xx(E)<cr><lf>"

; i.e. 8(9) characters (the E and 9 if no echo rec'd).

; P63 identifies 82 (Ascii R) as leading

; tag, and P15 identifies <cr> as end tag. Five values xx.xx (Ascii . = 46)

; read into 5 input locations from Uascii_1

; DTR - Data Transmit?
; RTS - Receive to Send
; CTS - Clear To Send
; RX - Receive ?
; TX - Transmit ?
; <cr> - carrage return
; <lf> - line feed

18: Port Serial I/O (P15)
1: 1 Reps
2: 21 8-Bit, RS-232 Binary, 1200 Baud
3: 1 Delay (0.01 sec units) before TX
4: 4 First (RTS/DTR) of Control Ports Used
5: 13 Start Loc for TX [junk]
6: 0 Number of Locs to TX
7: 13 Termination Character for RX
8: 9 Max Characters to RX
9: 200 Time Out for CTS (TX) and/or RX (0.01 sec units)
10: 7 Start Loc for RX [Uascii_1]
11: 1.0 Mult for RX
12: 0.0 Offset for RX

;* * * filter to synchronize the output stream Data output starts with an R.
; 82 = R in ASCII table for alphabethic characters

19: Extended Parameters (P63)
1: 82 Option
2: 0 Option
3: 0 Option
4: 0 Option
5: 0 Option
6: 0 Option
7: 0 Option
8: 0 Option

```
;*****************************************
; Set port 4 high to end measure mode PSA-916
;*****************************************


 20:  Do (P86)
  1: 44       Set Port 4 High


;*****************************************
;****Conversion from ASCII to Decimal*******
;*****************************************

; Conversion from ascii (Uascii_1 .. _5) to decimal.

; Using subroutine 1 for conversion digit by digit

; First put all readings in same format: xx.xx
; If format is x.xx insert a leading zero after
; bumping digits along one place.


;------------------------ set if format 0X.XXX-------------------------
;If second location is "." than move digit one place to the right


 21:  If (X<=>F) (P89)
  1: 8        X Loc [ Uascii_2  ]
  2: 1        =
  3: 46       F
  4: 30       Then Do


;---------move value from location 4 to location 5-------------

 22:  Z=X (P31)
  1: 10       X Loc [ Uascii_4  ]
  2: 11       Z Loc [ Uascii_5  ]

;---------move value from location 3 to location 4-------------

 23:  Z=X (P31)
  1: 9        X Loc [ Uascii_3  ]
  2: 10       Z Loc [ Uascii_4  ]
```

```
;----------move value from location 2 to location 3------------

        24:  Z=X (P31)
         1: 8       X Loc [ Uascii_2  ]
         2: 9       Z Loc [ Uascii_3  ]

;----------move value from location 1 to location 2-------------

        25:  Z=X (P31)
         1: 7       X Loc [ Uascii_1  ]
         2: 8       Z Loc [ Uascii_2  ]


;-----Insert "0" at the first location (decimal 48=0 ascii)-----


        26:  Z=F x 10^n (P30)
         1: 48      F
         2: 00      n, Exponent of 10
         3: 7       Z Loc [ Uascii_1  ]




    27:  End (P95)




    ;************************************
    ;***Now do digit by digit conversion***
    ;************************************

;------------------first digit-------------------------------
; Take value from location Uascii_1 and put it in location num
;(num location is further use in a subroutine).
; Make location multiple to be 10. Reduce this value for 0.1 for
; eqach incoming digit conversion (see end of soubroutine).
; Value in location Convert returned from the end of soubroutine
; and store in location Under.
; After first digit is saved, next digits
; added in next decimal place via decrease in location 'multipl'
; by factor of 10 (see end of soubroutine).
```

```
    28:  Z=X (P31)
     1: 7       X Loc [ Uascii_1  ]
     2: 14      Z Loc [ num       ]

    29:  Z=F x 10^n (P30)
     1: 1       F
     2: 1       n, Exponent of 10
     3: 15      Z Loc [ mult      ]

    30:  Do (P86)
     1: 1       Call Subroutine 1

    31:  Z=X (P31)
     1: 16      X Loc [ Convert   ]
     2: 17      Z Loc [ Under     ]

;-------------------Now second digit--------------------------------


    32:  Z=X (P31)
     1: 8       X Loc [ Uascii_2  ]
     2: 14      Z Loc [ num       ]

    33:  Do (P86)
     1: 1       Call Subroutine 1

    34:  Z=X+Y (P33)
     1: 17      X Loc [ Under     ]
     2: 16      Y Loc [ Convert   ]
     3: 17      Z Loc [ Under     ]

;-------------------Now third digit---------------------------------
;skipping decimal point at third location (Uascii_3)


    35:  Z=X (P31)
     1: 10      X Loc [ Uascii_4  ]
     2: 14      Z Loc [ num       ]

    36:  Do (P86)
     1: 1       Call Subroutine 1

    37:  Z=X+Y (P33)
     1: 17      X Loc [ Under     ]
     2: 16      Y Loc [ Convert   ]
```

```
     3: 17      Z Loc [ Under     ]

;-------------------Now fourth digit-------------------------------

    38:  Z=X (P31)
     1: 11      X Loc [ Uascii_5  ]
     2: 14      Z Loc [ num       ]

    39:  Do (P86)
     1: 1       Call Subroutine 1

    40:  Z=X+Y (P33)
     1: 17      X Loc [ Under     ]
     2: 16      Y Loc [ Convert   ]
     3: 17      Z Loc [ Under     ]

;---------------------------------------------------------------------
; Uascii_6 skipped. This Location is only for "E" or 9 as an error message
; - no signal recived.


     ;*********************************************
     ;************* Turn Port 3 Low to ************
     ;************* turn OFF Swiched 12 V**********
     ;*********************************************


    41:  Do (P86)
     1: 53      Set Port 3 Low


; Closing If statement/command/P92

42:  End (P95)




     ;*********************************************
     ; ************** OUTPUT **********************
     ;*********************************************


43:  Do (P86)
 1: 10      Set Output Flag High (Flag 0)
```

```
; SETUP 20 MINUTES INTERVAL TIME
44:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 20       Interval (same units as above)
 3: 10       Set Output Flag High

; SETUP STORAGE AREA AND ARRAY ID
45:  Set Active Storage Area (P80)^29610
 1: 1        Final Storage Area 1
 2: 3        Array ID

46:  Real Time (P77)^25760
 1: 1220     Year,Day,Hour/Minute (midnight = 2400)

47:  Resolution (P78)
 1: 1        High Resolution

;MEASURE AVERAGE ABLATION SENSOR
48:  Average (P71)^25466
 1: 1        Reps
 2: 1        Loc [ ablat_cm  ]

;MEASURE AVERAGE  STAGE TRANSDUCER
49:  Average (P71)^32453
 1: 1        Reps
 2: 5        Loc [ stage_cm  ]

50:  Resolution (P78)
 1: 0        Low Resolution

;MEASURE AVERAGE FRO UNDERVATER LIGHT, SURFACE LIGHT AND
BATTERY VOLTAGE (REPS 3)
51:  Average (P71)^23865
 1: 3        Reps
 2: 2        Loc [ uwlight   ]

;COMMUNICATE WITH STORAGE MODULE (7)
52:  Serial Out (P96)
 1: 71       Storage Module

;*********now for PSA916*****************************


;TIME INTERVAL ONCE A DAY (AT NOON)
```

```
53:  If time is (P92)
 1: 720      Minutes (Seconds --) into a
 2: 1440     Interval (same units as above)
 3: 10       Set Output Flag High

; SETUP STORAGE AREA & ARRY ID TO 1
54:  Set Active Storage Area (P80)^1813
 1: 1        Final Storage Area 1
 2: 32       Array ID

; SETUP TIME
55:  Real Time (P77)^28407
 1: 1220     Year,Day,Hour/Minute (midnight = 2400)


56:  Resolution (P78)
 1: 0        Low Resolution


; GET A DISTANCE TO THE ICE FROM PSA-916
57:  Sample (P70)^7309
 1: 1        Reps
 2: 17       Loc [ Under     ]


; Activate Serial Output: Comunicate with storage area - send
; data to SM4M for storage.

58:  Serial Out (P96)
 1: 71       Storage Module

*Table 2 Program
  01: 0.0000    Execution Interval (seconds)

;**********************************************************
*Table 3 Subroutines
;**********************************************************

; ------ Subroutine for  ascii to decimal conversion -----

; Benthos sounder return depth in ascii format. Convert digit-by-digit

; to decimal here, then save as one number [Convert] for output.


1:  Beginning of Subroutine (P85)
```

```
 1: 1       Subroutine 1

;--------------***Decimal 48=0 ASCII ****----------------------
; Take value from location num and if this value is equal 48
; than put it in location Convert .


    2:  If (X<=>F) (P89)
     1: 14      X Loc [ num      ]
     2: 1       =
     3: 48      F
     4: 30      Then Do


        3:  Z=F x 10^n (P30)
         1: 0       F
         2: 00      n, Exponent of 10
         3: 16      Z Loc [ Convert   ]


    4:  End (P95)

;--------------***Decimal 49=1 ASCII ****----------------------

    5:  If (X<=>F) (P89)
     1: 14      X Loc [ num      ]
     2: 1       =
     3: 49      F
     4: 30      Then Do


        6:  Z=F x 10^n (P30)
         1: 1       F
         2: 00      n, Exponent of 10
         3: 16      Z Loc [ Convert   ]


    7:  End (P95)

;--------------***Decimal 50=2 ASCII ****----------------------

    8:  If (X<=>F) (P89)
     1: 14      X Loc [ num      ]
     2: 1       =
     3: 50      F
     4: 30      Then Do
```

```
        9:  Z=F x 10^n (P30)
         1: 2       F
         2: 00      n, Exponent of 10
         3: 16      Z Loc [ Convert   ]


    10:  End (P95)

;--------------***Decimal 51=3 ASCII ****----------------------

    11:  If (X<=>F) (P89)
     1: 14      X Loc [ num       ]
     2: 1       =
     3: 51      F
     4: 30      Then Do


        12:  Z=F x 10^n (P30)
         1: 3       F
         2: 00      n, Exponent of 10
         3: 16      Z Loc [ Convert   ]


    13:  End (P95)

;--------------***Decimal 52=4 ASCII ****----------------------

    14:  If (X<=>F) (P89)
     1: 14      X Loc [ num       ]
     2: 1       =
     3: 52      F
     4: 30      Then Do


        15:  Z=F x 10^n (P30)
         1: 4       F
         2: 00      n, Exponent of 10
         3: 16      Z Loc [ Convert   ]


    16:  End (P95)

;--------------***Decimal 53=5 ASCII ****----------------------
```

```
17:  If (X<=>F) (P89)
 1: 14       X Loc [ num       ]
 2: 1        =
 3: 53       F
 4: 30       Then Do


    18:  Z=F x 10^n (P30)
     1: 5        F
     2: 00       n, Exponent of 10
     3: 16       Z Loc [ Convert   ]


19:  End (P95)

;--------------***Decimal 54=6 ASCII ****----------------------

20:  If (X<=>F) (P89)
 1: 14       X Loc [ num       ]
 2: 1        =
 3: 54       F
 4: 30       Then Do


    21:  Z=F x 10^n (P30)
     1: 6        F
     2: 00       n, Exponent of 10
     3: 16       Z Loc [ Convert   ]


22:  End (P95)

;--------------***Decimal 55=7 ASCII ****----------------------


23:  If (X<=>F) (P89)
 1: 14       X Loc [ num       ]
 2: 1        =
 3: 55       F
 4: 30       Then Do


    24:  Z=F x 10^n (P30)
     1: 7        F
     2: 00       n, Exponent of 10
     3: 16       Z Loc [ Convert   ]
```

```
    25:  End (P95)

;--------------***Decimal 56=8 ASCII ****----------------------


    26:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 56       F
     4: 30       Then Do


        27:  Z=F x 10^n (P30)
         1: 8        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    28:  End (P95)

;--------------***Decimal 57=9 ASCII ****----------------------

    29:  If (X<=>F) (P89)
     1: 14       X Loc [ num       ]
     2: 1        =
     3: 57       F
     4: 30       Then Do


        30:  Z=F x 10^n (P30)
         1: 9        F
         2: 00       n, Exponent of 10
         3: 16       Z Loc [ Convert   ]


    31:  End (P95)

;*****************************************************
;***save decimal as one number [Cononvert] for output***
;*****************************************************

; take value from Convert, multiple and put back in location Convert
; (rewrite location Convert)
; This value in location Convert returne back in Table 1
```

; and add it in location Under.

```
32:  Z=X*Y (P36)
 1: 16      X Loc [ Convert   ]
 2: 15      Y Loc [ mult      ]
 3: 16      Z Loc [ Convert   ]


33:  Z=X*F (P37)
 1: 15      X Loc [ mult      ]
 2: .1      F
 3: 15      Z Loc [ mult      ]


34:  End (P95)
End Program
```

4.3.3 Input storage (*6) locations

-Input Locations-
1 ablat_cm 1 1 1
2 uwlight 1 1 1
3 par 1 1 1
4 battvolts 1 1 1
5 stage_cm 1 1 1
6 Uascii_0 1 0 1
7 Uascii_1 1 2 3
8 Uascii_2 1 3 2
9 Uascii_3 1 1 2
10 Uascii_4 1 2 2
11 Uascii_5 1 1 2
12 Uascii_6 1 0 1
13 junk 1 1 0
14 num 1 10 4
15 mult 1 2 2
16 Convert 1 5 11
17 Under 1 4 4
18 _____ 0 0 0
19 _____ 0 0 0
20 _____ 0 0 0
21 _____ 0 0 0
22 _____ 0 0 0
23 _____ 0 0 0
24 _____ 0 0 0
25 _____ 0 0 0
26 _____ 0 0 0
27 _____ 0 0 0
28 _____ 0 0 0

-Program Security-
0000
0000
0000
-Mode 4-
-Final Storage Area 2-
0
-CR10X ID-
0
-CR10X Power Up-
3
-CR10X Compile Setting-
3
-CR10X RS-232 Setting-
-1
-DLD File Labels-
0
-Final Storage Labels-
0,3,29610
1,Year_RTM,25760
1,Day_RTM
1,Hour_Minute_RTM
2,ablat_cm_AVG~1,25466
3,stage_cm_AVG~5,32453
4,uwlight_AVG~2,23865
4,par_AVG~3
4,battvolts_AVG~4

5,32,1813
6,Year_RTM,28407
6,Day_RTM
6,Hour_Minute_RTM
7,Under~17,7309

4.3.4 Final storage array definition (*.FSL)

Final Storage Label File for: ELB PROGRAM 2007 UPDATED FPR PSA916-ALL IN ONE TABLE.csi
Date: 12/4/2008
Time: 11:52:00

143 Output_Table 60.00 Sec
1 143 L

3 Output_Table 20.00 Min
1 3 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 ablat_cm_AVG H
6 stage_cm_AVG H
7 uwlight_AVG L
8 par_AVG L
9 battvolts_AVG L

32 Output_Table 1440.00 Min
1 32 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 Under L

Estimated Total Final Storage Locations used per day 797

4.4 Lake Bonney West Lobe

4.4.1 Wiring

Control Ports:
CP1 – not used
CP2 – not used
CP3 - jump wire to SW 12V CTRL (Fryxel, ELB & WLB)
CP4 - white (input)
CP5 - blue (output)

sw 12V CTRL - jump wire to CP3
sw 12V - Red
red wire from PSA-916
G - black wire from PSA-916

Ablation transducer (instrument with dessicant filled vent tube)

Red	E1
Orange	H1
Black	L1
Yellow	H2
Blue	L2
White	AG (any one)
Clear	G (any one)

Underwater PAR

Green	3L
Blue	3H

Note: uw par has jumper wire at this CR10X. Probably this PAR sensor were purchased from LiCor directly; LiCor PAR sensor purchased from Campbell have them build into the lines.

Surface PAR

Black	4L
Red	4H

Jump wire from Black 4L to AG

PSA-916
Green Not used - (analog output ground)
Orange Not used - (analog output)
Red 12V - (external power 6-12V) - it goes to SW 12 V
Black G - (Ground)
White C4 - (RS-232 External Key Input)
Blue C5 - (RS-232 Output/Error Output)

Note: Mux cable wiring description wasn't presented in BBB 04/05. Meda just transfer wiring from old Blue Box to the new Blue Box. Wiring description below was the same we found it at the field. This wiring needs to be double-checked.

Multiplexer Box
Present, but not connected to CR10X.

Mux Cable # 1:

Connection at CR10X:		Connection at AM 416	
White	C3	Clear	shield COM (b/w 10-11)
Green	C2	Green	res
Black	G	White	clk
Clear	G (b/w Com 11-12)	Black	gnd
Red	12V	Red	12V

Mux Cable # 2

Connection at CR10X:		Connection at AM 416	
White	L5	Clear	shield
Green	disconnected	Green	COM H2
Black	disconnected	White	COM L1 (b/w 7-8)
Clear	G	Black	COM L2 (b/w 10-11)
Red	H5	Red	disconnected

4.4.2 Program

```
;{CR10X}
;BONNEY WEST PROGRAM 2007 UPDATED FOR PSA196
; M. Medved, UIC 12/16/07

;{CR10X}  Change existing program on 11/26/05
;

; See the blue box 2005,2006 and 2007 for wiring
; PSA916 wiring is below


*Table 1 Program
  01: 60.0000   Execution Interval (seconds)

; MEASURE ABLATION SENSOR
1:  Full Bridge w/mv Excit (P9)
 1: 1        Reps
 2: 5        2500 mV Slow Ex Range
 3: 3        25 mV Slow Br Range
 4: 1        DIFF Channel
 5: 1        Excite all reps w/Exchan 1
 6: 2500     mV Excitation
 7: 1        Loc [ ablat_cm  ]
 8: 101.86   Mult
 9: 0.0      Offset

;MEASURE UNDERWATER LIGHT
2:  Volt (Diff) (P2)
 1: 1        Reps
 2: 2        7.5 mV Slow Range
 3: 3        DIFF Channel
 4: 2        Loc [ uwlight   ]
 5: -100     Mult
 6: 0.0      Offset

;MEASURE SURFACE LIGHT SENSOR (QUANTUM) - SURFACE PAR
3:  Volt (Diff) (P2)
 1: 1        Reps
 2: 3        25 mV Slow Range
 3: 4        DIFF Channel
 4: 3        Loc [ par       ]
 5: 200      Mult
 6: 0.0      Offset
```

```
; MEASURE BATTERY WOLTAGE
4:  Batt Voltage (P10)
 1: 4        Loc [ battvolts ]

; TIME INTERVAL SETUP FOR 20 KMINUTES
5:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 20       Interval (same units as above)
 3: 10       Set Output Flag High

;SETUP STORAGE AREA AND ARREY ID TO 4
6:  Set Active Storage Area (P80)^15304
 1: 1        Final Storage Area 1
 2: 4        Array ID

7:  Real Time (P77)^3758
 1: 1220     Year,Day,Hour/Minute (midnight = 2400)

8:  Resolution (P78)
 1: 1        High Resolution

; CALCULATE AVERAGE FOR ABLATION SENSOR
9:  Average (P71)^9773
 1: 1        Reps
 2: 1        Loc [ ablat_cm  ]

10:  Resolution (P78)
 1: 0        Low Resolution

;CALCULATE AVERAGE FOR UNDERWATER LIGHT, PAR, AND BATTERY
VOLTS (REPS 3)
11:  Average (P71)^11901
 1: 3        Reps
 2: 2        Loc [ uwlight   ]

; COMUNICATE WITH STORAGE MODULE (7)
12:  Serial Out (P96)
 1: 71       Storage Module

;**********************************************************************
;***********PROGRAM FOR PSA-916 AT LAKE
FRYXELL**********************
;**********************************************************************
```

```
    ;*********Internal Swiches in PSA-916*****
            ;1-ON
            ;2-OFF
            ;3-ON
            ;4-ON
            ;5-ON
            ;6-ON
            ;7-OFF
            ;8-OFF



  ;*****************************************
  ; *** PSA-916 & CR10X WIRING AND CONTROLS ***
  ;*****************************************

;Green  Not used - (analog output ground)
;Orange Not used - (analog output)
;Red    12V      - (external power 6-12V) - it goes to SW 12 V
;Black  G        - (Ground)
;White  C4       - (RS-232 External Key Input)
;Blue   C5       - (RS-232 Output/Error Output)

; Control Ports

    ;   CP1 -

    ;   CP2 -

    ;   CP3 - jump wire to SW 12V CTRL (Fryxel, ELB & WLB)

    ;   CP4 - white (input)

    ;   CP5 - blue (output)

;SW
    ;sw 12V CTRL - jump wire to CP3

    ;sw 12V - Red
;12V
    ; red wire from PSA-916
    ; G - black wire from PSA-916
```

```
;---------------------------------------------------------------

;  Start program every 2 hours

*Table 2 Program
  01: 7200      Execution Interval (seconds)

; If is midnight, exicute program

1:  If time is (P92)
 1: 0        Minutes (Seconds --) into a
 2: 1440     Interval (same units as above)
 3: 30       Then Do


     ;*****************************************
     ;************ Turn Port 3 High to **********
     ;************ turn ON Switched 12 V**********
     ;*****************************************

2:  Do (P86)
 1: 43       Set Port 3 High


3:  Excitation with Delay (P22)
 1: 2        Ex Channel
 2: 0000     Delay W/Ex (0.01 sec units)
 3: 500      Delay After Ex (0.01 sec units)
 4: 0000     mV Excitation


     ;*****************************************
     ;****Turn Port 4 High to set trigger high*****
     ;**********  (pause sounder)  ***************
     ;*****************************************


4:  Do (P86)
 1: 44       Set Port 4 High




     ;*********************************
```

```
      ; Reset read ascii values to 00000
      ;**********************************

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

5:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 5        Z Loc [ Uascii_0  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

6:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 6        Z Loc [ Uascii_1  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

7:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 7        Z Loc [ Uascii_2  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

8:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
 3: 8        Z Loc [ Uascii_3  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

9:  Z=F x 10^n (P30)
 1: 48       F
 2: 00       n, Exponent of 10
```

```
 3: 9       Z Loc [ Uascii_4  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

10:  Z=F x 10^n (P30)
 1: 48      F
 2: 00      n, Exponent of 10
 3: 10      Z Loc [ Uascii_5  ]

;---------------------------------------------------------------

; Decimal 48= 0 ASCII

11:  Z=F x 10^n (P30)
 1: 48      F
 2: 00      n, Exponent of 10
 3: 11      Z Loc [ Uascii_6  ]


    ;*************************************************************
    ; Set port 4 low to trigger measurement with Benthos PSA-916
    ;*************************************************************

; * set port 4 low to click under ice sounder

   12:  Do (P86)
    1: 54       Set Port 4 Low


    ;*************************************************************
    ;**********Read Benthos PSA-916 RS232 output*****************
    ;*************************************************************

; Instruction P15 uses CP4 as DTR, CP5 as RX.

; Receives ascii string as binary RS232, 1200 baud (set inside sounder

; with dip switches, see the above). Term char 13 = <cr>. Output is"Rxx.xx(E)<cr><lf>"

; i.e. 8(9) characters (the E and 9 if no echo rec'd).

; P63 identifies 82 (Ascii R) as leading

; tag, and P15 identifies <cr> as end tag. Five values xx.xx (Ascii . = 46)
```

; read into 5 input locations from Uascii_1

; DTR - Data Transmit?
; RTS - Receive to Send
; CTS - Clear To Send
; RX - Receive ?
; TX - Transmit ?
; <cr> - carrage return
; <lf> - line feed

13: Port Serial I/O (P15)
1: 1 Reps
2: 21 8-Bit, RS-232 Binary, 1200 Baud
3: 1 Delay (0.01 sec units) before TX
4: 4 First (RTS/DTR) of Control Ports Used
5: 12 Start Loc for TX [junk]
6: 0 Number of Locs to TX
7: 13 Termination Character for RX
8: 9 Max Characters to RX
9: 200 Time Out for CTS (TX) and/or RX (0.01 sec units)
10: 6 Start Loc for RX [Uascii_1]
11: 1.0 Mult for RX
12: 0.0 Offset for RX

;* * * filter to synchronize the output stream Data output starts with an R.
; 82 = R in ASCII table for alphabethic characters

14: Extended Parameters (P63)
1: 82 Option
2: 0 Option
3: 0 Option
4: 0 Option
5: 0 Option
6: 0 Option
7: 0 Option
8: 0 Option

;***
; Set port 4 high to end measure mode PSA-916
;***

```
    15:  Do (P86)
     1: 44       Set Port 4 High


     ;*****************************************
     ;****Conversion from ASCII to Decimal*******
     ;*****************************************

; Conversion from ascii (Uascii_1 .. _5) to decimal.

; Using subroutine 1 for conversion digit by digit

; First put all readings in same format: xx.xx
; If format is x.xx insert a leading zero after
; bumping digits along one place.


;------------------------ set if format 0X.XXX-------------------------
;If second location is "." than move digit one place to the right


    16:  If (X<=>F) (P89)
     1: 7        X Loc [ Uascii_2  ]
     2: 1        =
     3: 46       F
     4: 30       Then Do


;---------move value from location 4 to location 5-------------

         17:  Z=X (P31)
          1: 9        X Loc [ Uascii_4  ]
          2: 10       Z Loc [ Uascii_5  ]

;---------move value from location 3 to location 4-------------

         18:  Z=X (P31)
          1: 8        X Loc [ Uascii_3  ]
          2: 9        Z Loc [ Uascii_4  ]


;----------move value from location 2 to location 3------------

         19:  Z=X (P31)
          1: 7        X Loc [ Uascii_2  ]
          2: 8        Z Loc [ Uascii_3  ]
```

```
;----------move value from location 1 to location 2------------

        20:  Z=X (P31)
         1: 6       X Loc [ Uascii_1  ]
         2: 7       Z Loc [ Uascii_2  ]


;-----Insert "0" at the first location (decimal 48=0 ascii)-----


        21:  Z=F x 10^n (P30)
         1: 48       F
         2: 00       n, Exponent of 10
         3: 6        Z Loc [ Uascii_1  ]



    22:  End (P95)




     ;*************************************
     ;***Now do digit by digit conversion***
     ;*************************************

;-------------------first digit-------------------------------
; Take value from location Uascii_1 and put it in location num
;(num location is further use in a subroutine).
; Make location multiple to be 10. Reduce this value for 0.1 for
; eqach incoming digit conversion (see end of soubroutine).
; Value in location Convert returned from the end of soubroutine
; and store in location Under.
; After first digit is saved, next digits
; added in next decimal place via decrease in location 'multipl'
; by factor of 10 (see end of soubroutine).

    23:  Z=X (P31)
     1: 6       X Loc [ Uascii_1  ]
     2: 13       Z Loc [ num       ]

    24:  Z=F x 10^n (P30)
     1: 1       F
```

```
 2: 1       n, Exponent of 10
 3: 14      Z Loc [ mult     ]

25:  Do (P86)
 1: 1       Call Subroutine 1

26:  Z=X (P31)
 1: 15      X Loc [ Convert  ]
 2: 16      Z Loc [ Under    ]

;-------------------Now second digit---------------------------------


27:  Z=X (P31)
 1: 7       X Loc [ Uascii_2  ]
 2: 13      Z Loc [ num       ]

28:  Do (P86)
 1: 1       Call Subroutine 1

29:  Z=X+Y (P33)
 1: 16      X Loc [ Under    ]
 2: 15      Y Loc [ Convert  ]
 3: 16      Z Loc [ Under    ]

;-------------------Now third digit---------------------------------
;skipping decimal point at third location (Uascii_3)


30:  Z=X (P31)
 1: 9       X Loc [ Uascii_4  ]
 2: 13      Z Loc [ num       ]

31:  Do (P86)
 1: 1       Call Subroutine 1

32:  Z=X+Y (P33)
 1: 16      X Loc [ Under    ]
 2: 15      Y Loc [ Convert  ]
 3: 16      Z Loc [ Under    ]

;-------------------Now fourth digit---------------------------------

33:  Z=X (P31)
 1: 10      X Loc [ Uascii_5  ]
 2: 13      Z Loc [ num       ]
```

```
    34:  Do (P86)
     1: 1       Call Subroutine 1

    35:  Z=X+Y (P33)
     1: 16      X Loc [ Under     ]
     2: 15      Y Loc [ Convert   ]
     3: 16      Z Loc [ Under     ]

;---------------------------------------------------------------------
; Uascii_6 skipped. This Location is only for "E" or 9 as an error message
; - no signal recived.



    ;***********************************************
    ; ************** OUTPUT ************************
    ;***********************************************



    36:  Do (P86)
     1: 10      Set Output Flag High (Flag 0)



;Setup Storage Area & Arrey ID to 12 for Fryxell, 22 Hoare
;32 ELB and 42 WLB

    37:  Set Active Storage Area (P80)^28635
     1: 1       Final Storage Area 1
     2: 12      Array ID


; Setup Time

    38:  Real Time (P77)^10879
     1: 1220    Year,Day,Hour/Minute (midnight = 2400)

;Setup Resolution

    39:  Resolution (P78)
     1: 0       Low Resolution
```

```
   40:  Sample (P70)^143
    1: 1        Reps
    2: 16       Loc [ Under     ]


; Activate Serial Output: Comunicate with storage area - send
; data to SM4M for storage.

   41:  Serial Out (P96)
    1: 71       Storage Module




    ;*********************************************
    ;************* Turn Port 3 Low to ************
    ;************* turn OFF Swiched 12 V**********
    ;*********************************************


   42:  Do (P86)
    1: 53       Set Port 3 Low


; Closing If statement/command/P92

43:  End (P95)


;*********************************************************
*Table 3 Subroutines
;*********************************************************

; ------ Subroutine for  ascii to decimal conversion -----

; Benthos sounder return depth in ascii format. Convert digit-by-digit

; to decimal here, then save as one number [Convert] for output.


1:  Beginning of Subroutine (P85)
 1: 1        Subroutine 1

;--------------***Decimal 48=0 ASCII ****----------------------
; Take value from location num and if this value is equal 48
```

; than put it in location Convert .

```
    2:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 48       F
     4: 30       Then Do


        3:  Z=F x 10^n (P30)
         1: 0        F
         2: 00       n, Exponent of 10
         3: 15       Z Loc [ Convert   ]


    4:  End (P95)

;--------------***Decimal 49=1 ASCII ****----------------------

    5:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 49       F
     4: 30       Then Do


        6:  Z=F x 10^n (P30)
         1: 1        F
         2: 00       n, Exponent of 10
         3: 15       Z Loc [ Convert   ]


    7:  End (P95)

;--------------***Decimal 50=2 ASCII ****----------------------

    8:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 50       F
     4: 30       Then Do


        9:  Z=F x 10^n (P30)
         1: 2        F
```

```
        2: 00       n, Exponent of 10
        3: 15       Z Loc [ Convert   ]


    10:  End (P95)

;--------------***Decimal 51=3 ASCII ****----------------------

    11:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 51       F
     4: 30       Then Do


        12:  Z=F x 10^n (P30)
         1: 3        F
         2: 00       n, Exponent of 10
         3: 15       Z Loc [ Convert   ]


    13:  End (P95)

;--------------***Decimal 52=4 ASCII ****----------------------

    14:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 52       F
     4: 30       Then Do


        15:  Z=F x 10^n (P30)
         1: 4        F
         2: 00       n, Exponent of 10
         3: 15       Z Loc [ Convert   ]


    16:  End (P95)

;--------------***Decimal 53=5 ASCII ****----------------------

    17:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 53       F
```

```
 4: 30       Then Do


    18:  Z=F x 10^n (P30)
     1: 5        F
     2: 00       n, Exponent of 10
     3: 15       Z Loc [ Convert   ]


 19:  End (P95)

;--------------***Decimal 54=6 ASCII ****----------------------

 20:  If (X<=>F) (P89)
  1: 13       X Loc [ num       ]
  2: 1        =
  3: 54       F
  4: 30       Then Do


    21:  Z=F x 10^n (P30)
     1: 6        F
     2: 00       n, Exponent of 10
     3: 15       Z Loc [ Convert   ]


 22:  End (P95)

;--------------***Decimal 55=7 ASCII ****----------------------


 23:  If (X<=>F) (P89)
  1: 13       X Loc [ num       ]
  2: 1        =
  3: 55       F
  4: 30       Then Do


    24:  Z=F x 10^n (P30)
     1: 7        F
     2: 00       n, Exponent of 10
     3: 15       Z Loc [ Convert   ]


 25:  End (P95)
```

```
;--------------***Decimal 56=8 ASCII ****----------------------


    26:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 56       F
     4: 30       Then Do


        27:  Z=F x 10^n (P30)
         1: 8        F
         2: 00       n, Exponent of 10
         3: 15       Z Loc [ Convert   ]


    28:  End (P95)

;--------------***Decimal 57=9 ASCII ****----------------------

    29:  If (X<=>F) (P89)
     1: 13       X Loc [ num       ]
     2: 1        =
     3: 57       F
     4: 30       Then Do


        30:  Z=F x 10^n (P30)
         1: 9        F
         2: 00       n, Exponent of 10
         3: 15       Z Loc [ Convert   ]


    31:  End (P95)

;*****************************************************
;***save decimal as one number [Cononvert] for output***
;*****************************************************

; take value from Convert, multiple and put back in location Convert
; (rewrite location Convert)
; This value in location Convert returne back in Table 1
; and add it in location Under.

    32:  Z=X*Y (P36)
     1: 15       X Loc [ Convert   ]
```

2: 14 Y Loc [mult]
3: 15 Z Loc [Convert]

33: Z=X*F (P37)
1: 14 X Loc [mult]
2: .1 F
3: 14 Z Loc [mult]

34: End (P95)

End Program

4.4.3 Input storage (*6) locations

-Input Locations-
1 ablat_cm 1 1 1
2 uwlight 1 1 1
3 par 1 1 1
4 battvolts 1 1 1
5 Uascii_0 1 0 1
6 Uascii_1 1 2 3
7 Uascii_2 1 3 2
8 Uascii_3 1 1 2
9 Uascii_4 1 2 2
10 Uascii_5 1 1 2
11 Uascii_6 1 0 1
12 junk 1 1 0
13 num 1 10 4
14 mult 1 2 2
15 Convert 1 5 11
16 Under 1 4 4
17 _____ 0 0 0
18 _____ 0 0 0
19 _____ 0 0 0
20 _____ 0 0 0
21 _____ 0 0 0
22 _____ 0 0 0
23 _____ 0 0 0
24 _____ 0 0 0
25 _____ 0 0 0
26 _____ 0 0 0
27 _____ 0 0 0
28 _____ 0 0 0

-Program Security-
0000
0000
0000
-Mode 4-
-Final Storage Area 2-
0
-CR10X ID-
0
-CR10X Power Up-
3
-CR10X Compile Setting-
3
-CR10X RS-232 Setting-
-1
-DLD File Labels-
0
-Final Storage Labels-
0,4,15304
1,Year_RTM,3758
1,Day_RTM
1,Hour_Minute_RTM
2,ablat_cm_AVG~1,9773
3,uwlight_AVG~2,11901
3,par_AVG~3
3,battvolts_AVG~4

4,12,28635
5,Year_RTM,10879
5,Day_RTM
5,Hour_Minute_RTM
6,Under~16,143

4.4.4 Final storage array definition

Final Storage Label File for: NEW BYW 2007 UPDATED FOR PSA916.csi
Date: 12/17/2007
Time: 17:08:48

4 Output_Table 20.00 Min
1 4 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 ablat_cm_AVG H
6 uwlight_AVG L
7 par_AVG L
8 battvolts_AVG L

12 Output_Table 7200.00 Sec
1 12 L
2 Year_RTM L
3 Day_RTM L
4 Hour_Minute_RTM L
5 Under L

Estimated Total Final Storage Locations used per day 708