VIRGINIA TECHNOLOGIES INC.

APPLICATION NOTE #ECI-100

ECI-1 PROGRAMMING EXAMPLE FOR CAMPBELL SCIENTIFIC DATALOGGERS

AUTOSCANNING SINGLE PORT

Program auto scans all channels (0-9) on *port #1* on power-up. Records all ECI-1 devices found on the port in a local table. All found devices are tested, and data recorded in both INPUT LOCATIONS & FINAL STORAGE.

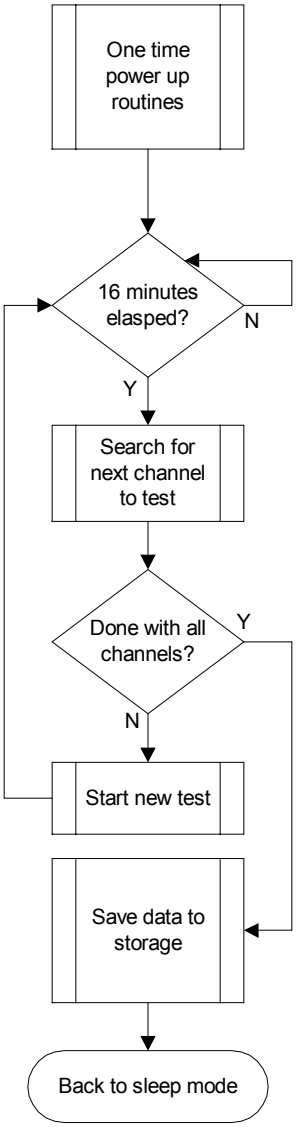
NOTE: ***It is assumed that the reader is familiar with the PC208 v3 software available from Campbell Scientific on their website: www.campbellsci.com.***

~~ All examples are provided as-is, with no warranty given or implied ~~

DOCUMENT CONTROL

REVISION	BY	DATE	REVISION NOTES
Original	SRC	04/01/2004	

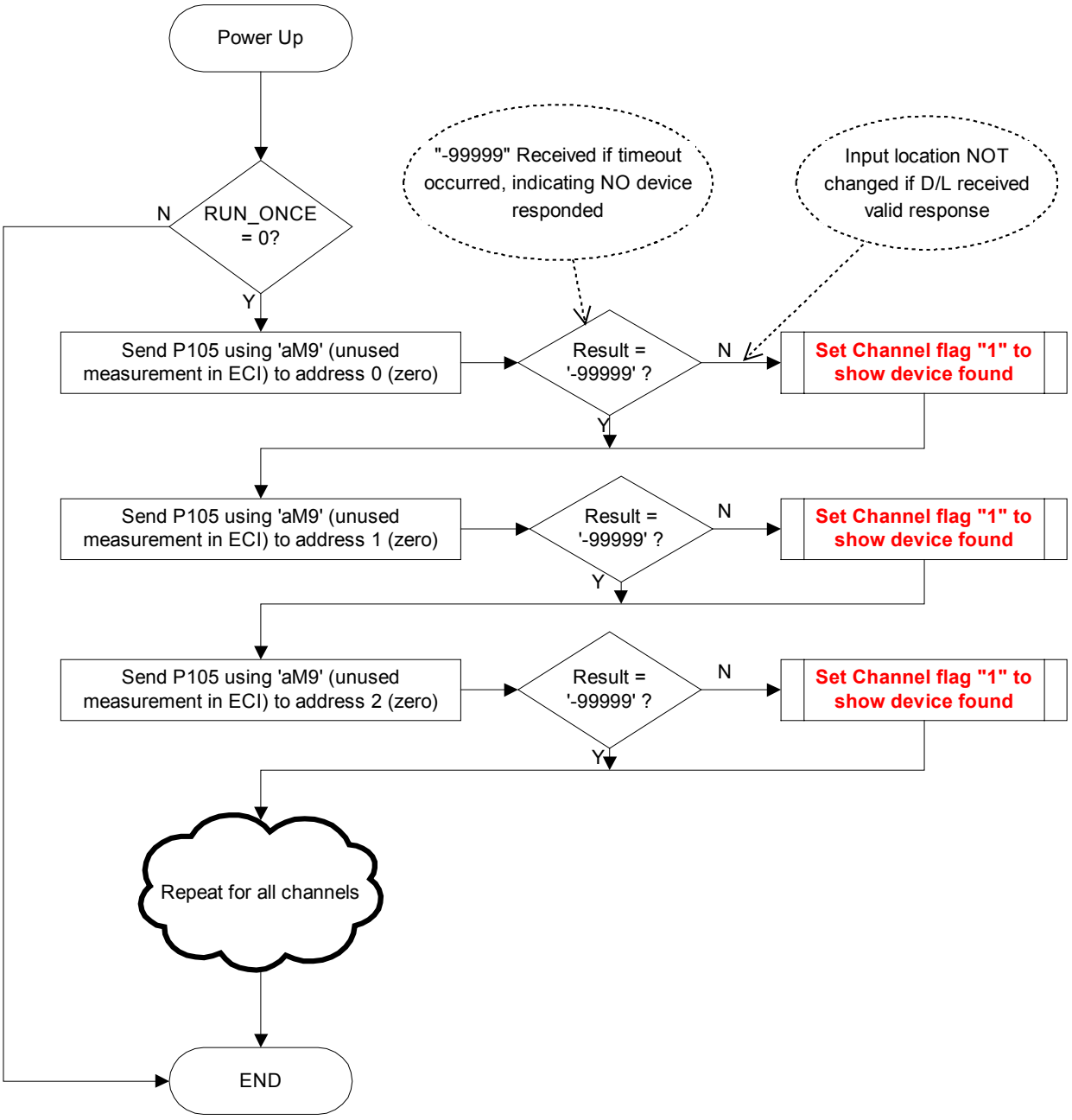
TOP LEVEL OVERVIEW



This flow chart shows the overall objective of the example program. The heart of the example is in the "Power-Up Routines" section. On power-up, the code looks at which ports are enabled, and then cycles through all ten (0-9) channels to looks for ECI devices.

Due to the structure of the Campbell Scientific software, variables cannot be used for the addresses in the P105 command. Therefore, each test for channels 0 to 9 is duplicate code with only the address changed. This means to cycle through 10 channels requires a lot of duplicate code.

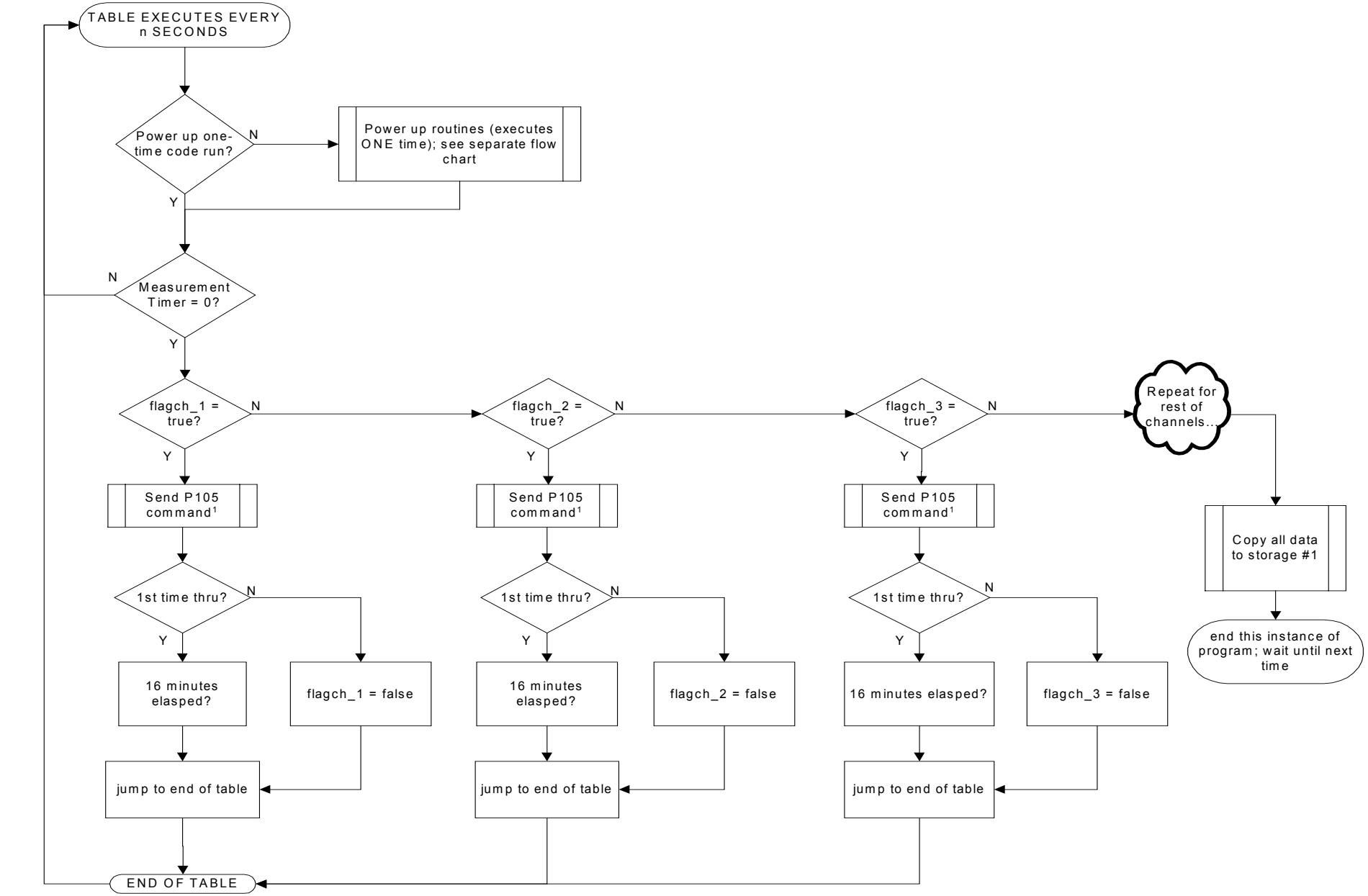
FIGURE 1



After the initial power up, all channels for port #1 have been tested and the data logger stores this data into an array input locations. The program will cycle through all channels and test the channel's "flag" to see if a device was found. If a device was found, a P105 command is sent.

The main flow chart (FIG. 2) shows the flow of the program during execution.:

FIGURE 2

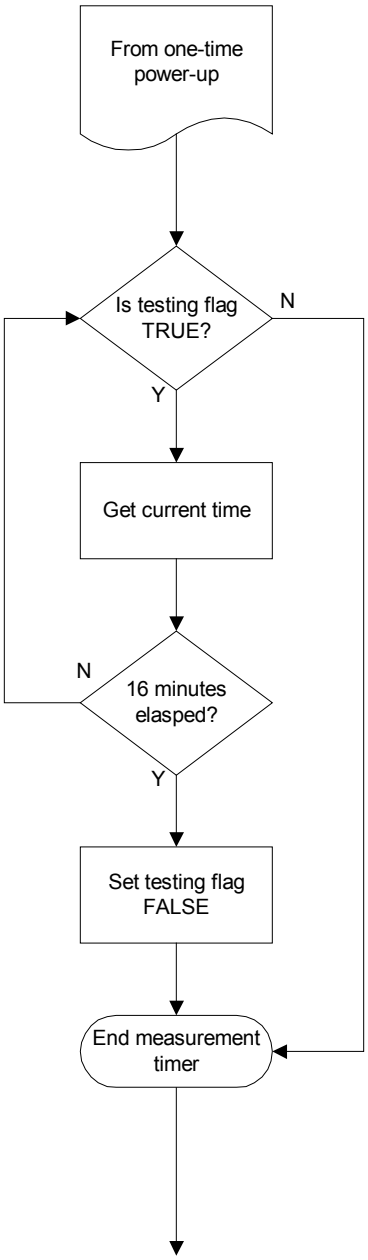


[1] Note: The program does NOT wait for a result with this command; this is why the measurement timer is used.

See FIG. 3 for expanded measurement timing.

FIGURE 3:

Expanded Measurement Test:



If a test is being performed, this measurement test will see if the 16 minutes has elapsed. If not, the program jumps to the end and doesn't do anything further. When the time has elapsed, the testing flag is cleared, and the program flow continues to the channels testing sections.

STORED DATA

The data are stored in INPUT LOCATIONS, and at the end of the program copied to FINAL STORAGE area #1. This method allows the user to change the order of what is stored, or the way it is stored. All channels' data is stored, and the data logger information has been captured.

In this example, we assume that multiple data loggers will be used in a project. The ARRAYID's used are 1-10, to represent the PORT. To make the data more understandable, this example stores the data (in final storage) ***as one large string*** in the flowing format (color coded for readability):

PORT# : DATALOGGER ID : YEAR : DAYS INTO YEAR : HOURS INTO DAY : BATTERY VOLTAGE : TEMPERATURE : FIVE READINGS FOR CHANNEL #1 : FIVE READINGS FOR CHANNEL #2 : FIVE READINGS FOR CHANNEL #3 : … : FIVE READINGS FOR CHANNEL #10

Here is an example (it is one continuous string of data broken up here for clarity):

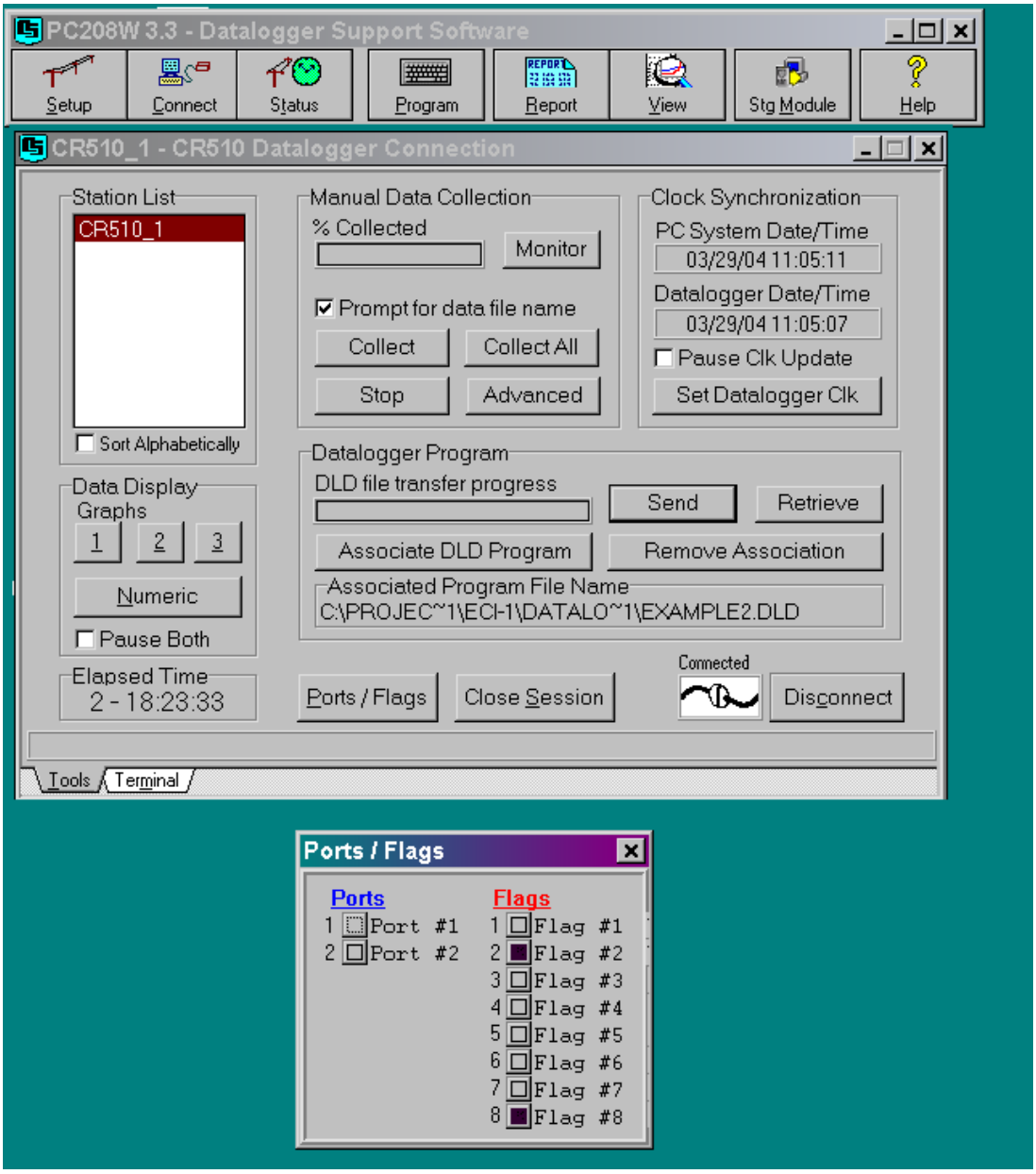
```
1,100,2004,86,2040,12.163,26.836,
.1,.11,1,12.3,0,
0,0,0,0,0,
.4328,0,-99999,25.3,-.7237,
0,0,0,0,0,
0,0,0,0,0,
.5984,0,99999,24.3,-.6308,
0,0,0,0,0,
0,0,0,0,0,
0,-111,-444,26.5,-.0002,
0,0,0,0,0
```

The sets of "0"'s (zeros) represent non-existent ECI instruments. The zeros are inserted as place holders, so importing this data into other applications (Excel, etc.) will always cause channels to line up! Otherwise, we won't know which channels we are looking at, if there is no uniformity.

The reason for this particular choice of format is that the ARRAY_ID range is 0 – 511, and is always printed first. We then find some relationship between our data and this range of values to use this to our advantage. We decided to use this field as PORTS, so you can look at the string and read "port 1 of logger #100 on this date had this data…"

You can readily change the storage format to suite your needs. The code is well documented and it's easy to see where and how the changes are made. However, if you can use the above format, you might wish to do so, as future examples VTI may post will probably use it too – examples like: a small program to create reports; Excel importation application, etc.

USING PC208 AND THE EXAMPLE



Clicking **"CONNECT" brings up the connection screen. From the connection screen, you can open the "PORTS" screen, which shows the flags.**

COMPILING FOR OTHER CAMPBELL DATALOGGERS

The first line of the program has:
;{CR510}

You can change the "CR510" to several other models and click COMPILE:

;{CR10X} , ;{CR23X} , ;{21X} , ;{CR500} , ;{CR7} , ;{CR10}