Volume

1

A-WIT TECHNOLOGIES INC.

… a passion for execution …

Using the C Stamp with the StampPlot Graphing Utility

Version 1.1

A-WIT TECHNOLOGIES INC.

Using the C Stamp with the StampPlot Graphing Utility

Phone (800) 985-AWIT • Fax (800) 985-2948

Table of Contents

Chapter

1

Introduction

Data acquisition software is ideal for receiving and sending data to a microcontroller. Providing information on the spot without having the user stop to take specific points of interest, or sending information to the controller are just some of the luxuries the software offers. With the use of this software and the C Stamp, data can now be received more efficiently.

With the use of SelmaWare's StampPlot Pro, users will be able to view data sent from the C Stamp. This data can then be placed into graphs or stored. Some of the software features are listed below:

Supports Standard User (Free for home & educational institution) and Developers Licenses

- Eight ready-to-run configurations for monitoring and control.
- Plots up to 10 comma-separated analog values.
- Plots over 8 channels of digital data.
- Save plots files and JPG images.
- Template data into other documents.
- Linear or Logarithmic scales and numerous math operations.
- Instructions for drawing, playing WAV files, and placing graphics images.
- Plot Objects allow you to create graphical user interfaces (GUI) for monitoring or interactive control. Examples include: alarming meters, gauges, text boxes, sliders, image boxes, image buttons, progress bars, list boxes, additional plotting areas and more.

- Perform over-the-Internet acquisition and control with the included Serial-TCP gateway software (host PC required).

- Configurations may come from the controller or PC based text file (StampPlot Macro).

Notices

Getting Support

If possible, please check the C Stamp website www.c-stamp.com under SUPPORT for any updates to documentation, changes, or notices that may have become available since your Installation CD was produced. If you continue to have any issues for which a solution is not found in the aforementioned website, please e-mail tech_support@a-wit.com for help.

Getting Started

What you will need for this example:

- SelmaWare's StampPlot Pro
- C Stamp 48-Pin Module

- BOL Kit
- Digital Thermometer Part # CS452000
- 9 Pieces of Copper Wire

After you have downloaded SelmaWare's StampPlot Pro, uncompress (unzip) the installation file. Double click the installation file to start the install wizard. Follow the wizard along selecting the typical installation for the software.

After the software installs it is advised to browse through the help file. Once the program loads, click the 'Help' drop down menu then 'Contents', or press F1 to bring up the help section. It is not required for this example to go through everything, but viewing the help section will give you a better understanding of the software for future projects.

Next go to the CS310XXX (μC 101) Reference Guide Manual and scroll down to Project 8, Digital Thermometer. This will show you how to set up your thermometer on the BOL.

Sample Code

The code uses the **TEMPSIN** and **TEMPS_F** to calculate the temperature in Fahrenheit and assign it to the variable T. Then, using the **WCftoa** command, the floating point T is converted into an ASCII string and assigned to buffer. The second **SEROUT** command is to prevent a constant input of numbers. Without the "\r\n" string, the software will not properly read the data.

Note: A baud rate of 4.8K, with 8 data bits, and no parity is used. In addition, a single LED is lit to signify that the program is running or not. A delay of two seconds is used to separate data being sent.

```
#include  "CS110000.h"

void main(void)
{
// TYPE YOUR CODE HERE AFTER THIS LINE
  RAM BYTE msg[] = "\r";
  int interval;
  RAM BYTE buffer[5];
  float T;
  float baudr = 4.8;
  interval = 2000;

  STPIND(46, HIGH);
```

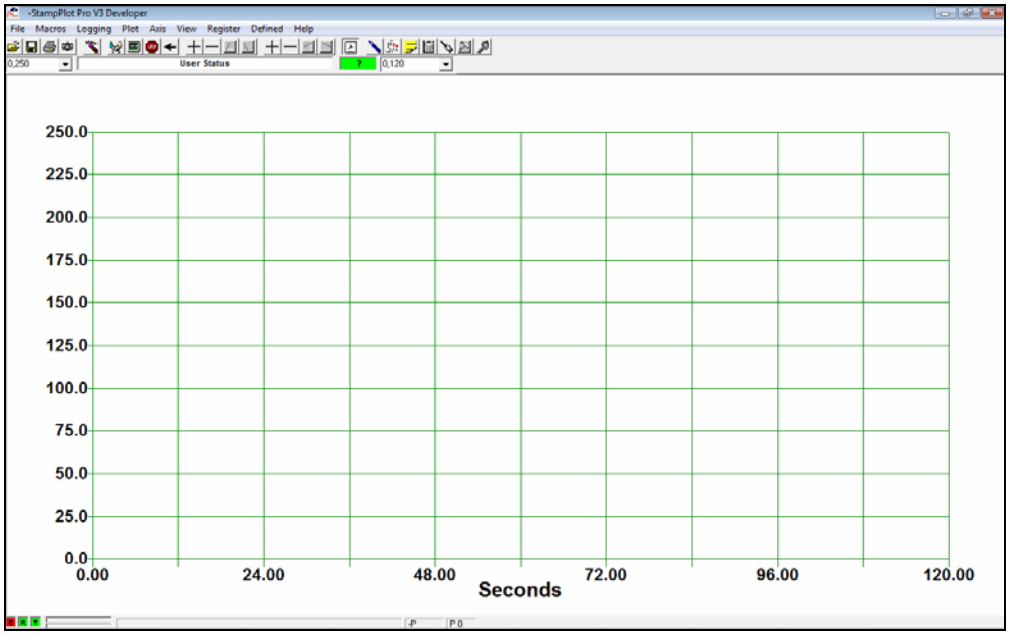
```
  while(TRUE){
    T = TEMPSIN_CS452000(0, TEMPS_F);
    WCftoa(T, buffer);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, buffer, 5);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 2);
    PAUSE(interval);
  }
}
```

Downloading and Running Your Program

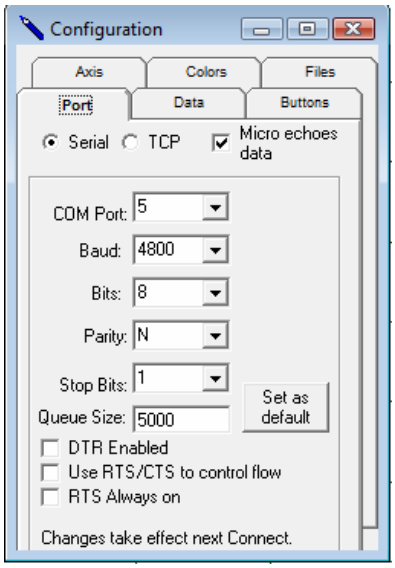
Download and start the program by pushing and letting go of the RESET button while pushing the START button. Then you can let go of the START button.

StampPlot Pro

Start up StampPlot Pro if you have not done so already. You will see eight different configurations to choose from. For the purpose of this example, we will not be using any of the different configurations; instead click 'Run no-Frills plot'. You should then see the following window appear.



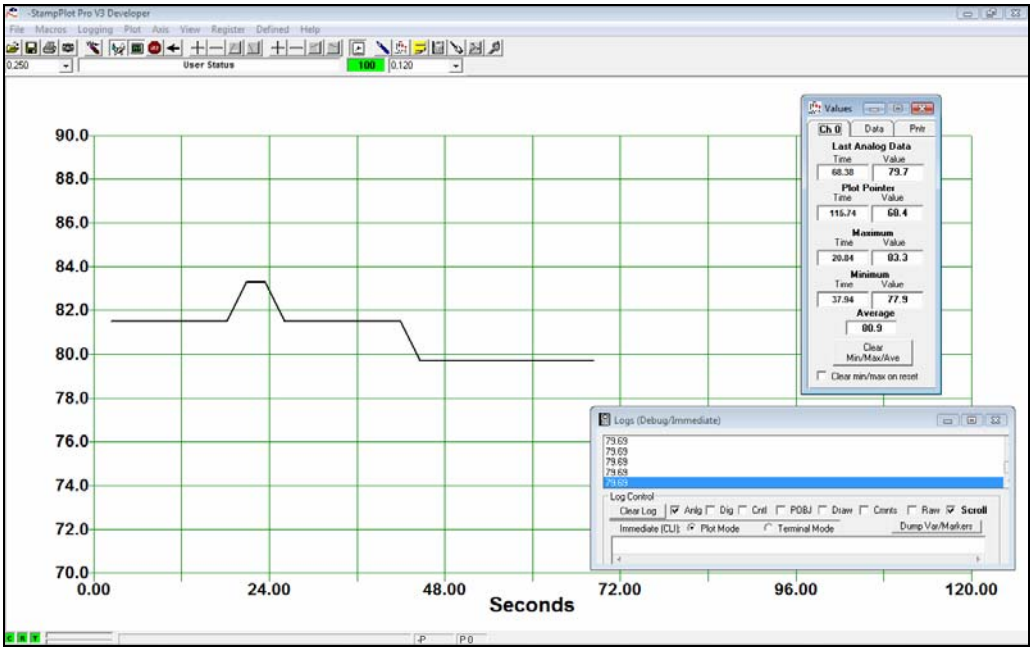
The next step is to connect the C Stamp to the StampPlot Pro. To do this click on the 'View' drop down menu and then 'Configuration', a window should appear that looks like the following.



From here, all of the settings that will be required to receive data can be set. Set your window to look like the one above for the exception of the COM Port. For that selection, set it to the COM Port that you are using.

When you are finished settings the configurations, exit out of the window. To connect to the C Stamp, click the 'Defined' drop down menu and look for 'Connect'. Click the option and you should now be connected to the C Stamp. If an error occurs, make sure that your configurations are correct. Next, go back to the 'Defined' drop down menu and click on 'Plot Data On'. With this turned on now, if your program is on already, a graph will begin to form showing the temperature readings. Note: you may need to adjust the Y-Axis settings. To do this, click the 'Axis' drop down menu and click on 'Analog Max…', and then 'Analog Min…' to set the maximum and minimum values for the Y-Axis.

In addition, if you wish to view the data go 'View' and click 'Values'. This will show you the value of the last point received. Alternatively, you can also click on 'Log (Debug/Immediate)'. Once in this window make sure that 'Anlg' is checked and scroll only. You should see your data start to appear on the screen.



Waveforms

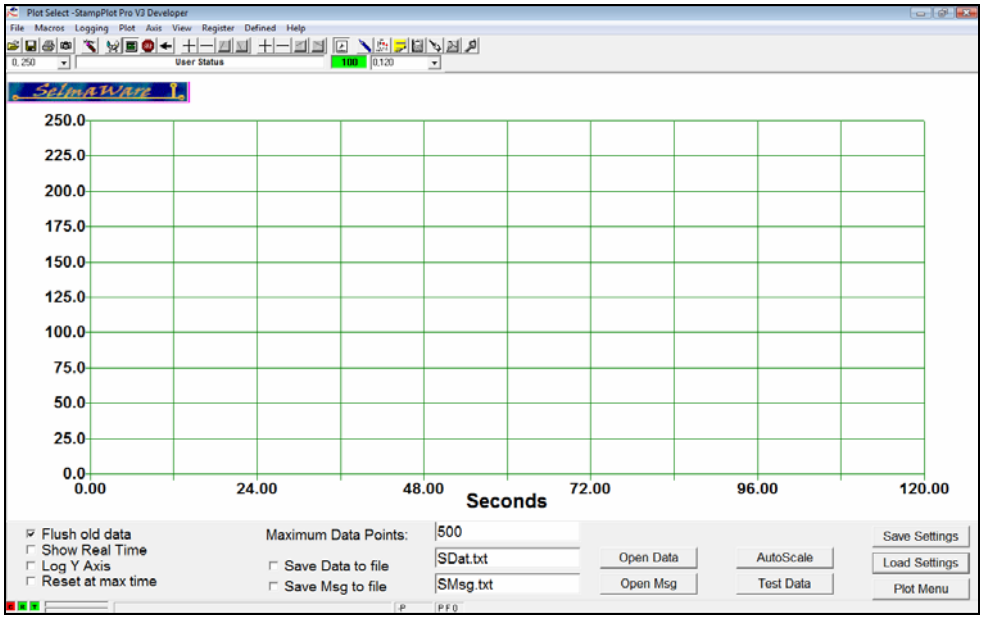
For this example, we will be writing code that forms two different waveforms. In order to begin, first create a new project. After you have created a new project and opened MPLAB, type the following code.

```
#include  "CS110000.h"

void main(void)
{
// TYPE YOUR CODE HERE AFTER THIS LINE
  RAM BYTE test[] = ",";
  RAM BYTE msg[] = "\r";
  float w;
  float x;
  float y;
  float z;
  BYTE a[5];
  BYTE b[5];
  BYTE c[5];
  BYTE d[5];
  float test_1;
  float test_2;
  float test_3;
```

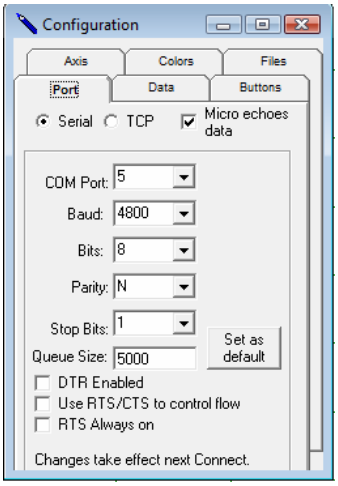
```
  float test_4;

  STPIND(46, HIGH);

  for(x = 0; x < 360.0; x = x + 1){
    PAUSE(10);
    test_1 = (x*PI/180);
    w = WCcos(test_1);
    WCftoa(w, a);
    SEROUT(0, 0, 4.8, 0, 8, 0, 0, a, 5);
    SEROUT(0, 0, 4.8, 0, 8, 0, 0, test, 1);

    PAUSE(10);
    test_2 = (x*PI/180);
    y = WCsin(test_2);
    WCftoa(y, b);
    SEROUT(0, 0, 4.8, 0, 8, 0, 0, b, 5);
    SEROUT(0, 0, 4.8, 0, 8, 0, 0, test, 1);

    PAUSE(10);
    test_3 =(x*PI/180);
    z = WCcos(3*test_3);

    PAUSE(10);
    test_4 = w*y*z;
    WCftoa(test_4, d);
    SEROUT(0, 0, 4.8, 0, 8, 0, 0, d, 5);
    SEROUT(0, 0, 4.8, 0, 8, 0, 0, msg, 1);
  }
  END();
}
```

This code will generate three separate graphs. Theses graphs will be plotted until `x` reaches its limit set by the '**`for`**' loop. Although there are four functions, the second cosine function will not print out to prevent clutter on the graph. However, the result of the second cosine function will be used within the final graph where all of the waveforms are multiplied together. The use of the '**`WCftoa`**' command is to convert the variables into the right format for the StampPlot Pro to read them. In addition, the '**`STPIND`**' command is used to turn on one LED to signify that the C Stamp program is running.

After you have compiled and downloaded your code onto the C Stamp, start up StampPlot Pro. You will see eight different configurations to choose from. For this example, we will be using the 'Standard with Control Objects' plot. When the window loads, you should see the following.



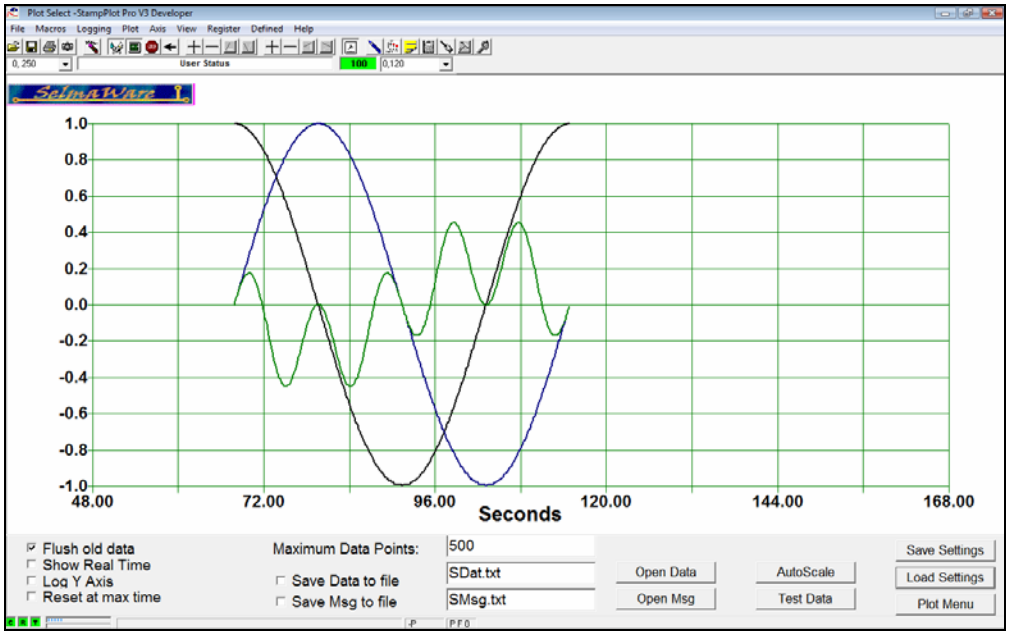
We will now configure the connection for this example. In order to do this, click the configuration button  or click the 'View' drop down menu and then 'Configuration', a window should appear that looks like the following.



From here, all of the settings that will be required to receive data can be set. Set your window to look like the one above for the exception of the COM Port. For that selection, set it to the COM Port that you are using. Also within the 'Data' tab, change the number of Data Points to 5000. This allows us to fully view the graphs without having the max number of data points reached.

Now connect to the C Stamp by pressing the 'Connect to Stamp on COM Port' button  , and by making sure the 'Start Plot' button  is pressed. Next, go to 'Axis' drop down menu and select 'Analog Max…'. Inside change the number to 1. Go back to 'Axis' drop down menu and select 'Analog Min…' this time. Change this number to –1. This will change the range of the graph into something that we will be able to view. Once these tasks are finished, start the C Stamp program.

Make sure that you do not run your C Stamp program until all the above-mentioned tasks have been completed.



Binary Graphing

This section will deal with a simple code that we will use to graph binary values. In order to begin, first create a new project. After you have created a new project and opened MPLAB, type the following code.

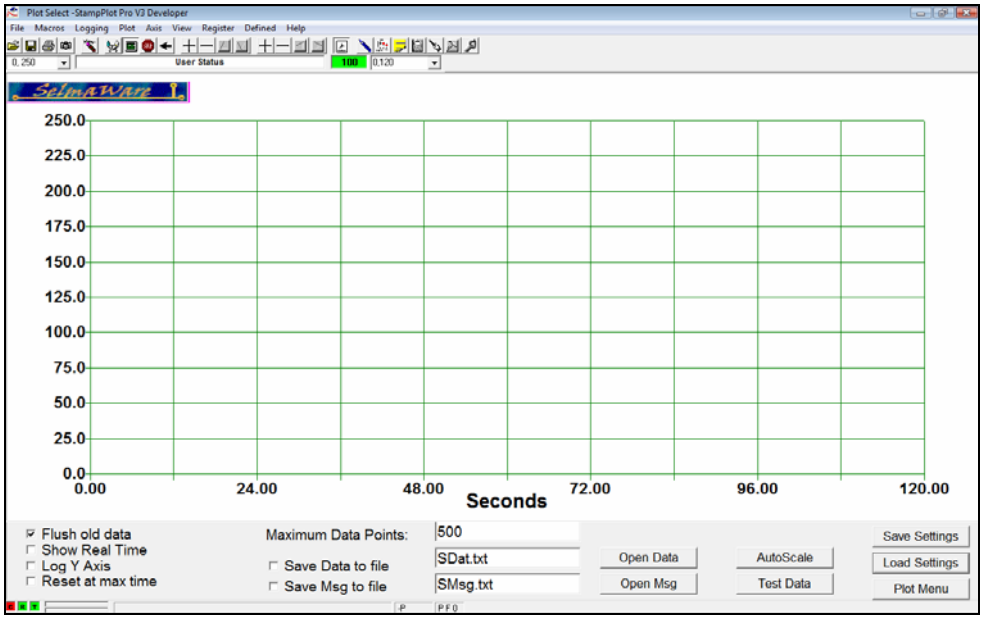
```
#include   "CS110000.h"

void main(void)
{
// TYPE YOUR CODE HERE AFTER THIS LINE
  RAM BYTE msg[] = "\r";
  RAM BYTE test[] = ",";
```

```
  RAM BYTE test_1[] = "%";
  RAM BYTE test_2[] = "!SETD 1,";
  RAM BYTE test_3[] = "!LBLD 1";
  RAM BYTE test_4[] = "Binary Number";
  RAM BYTE test_5[] = "!SETD 2,";
  RAM BYTE test_6[] = "!LBLD 2";
  RAM BYTE test_7[] = "Decimal Number";
  int x;
  float baudr = 4.8;
  BYTE test_8[5];
  BYTE test_9[2];

  STPIND(46, HIGH);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_3, 8);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test, 1);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_4, 14);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);

  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_6, 8);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test, 1);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_7, 14);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, msg,1);

  for(x = 0; x < 32; x = x + 1){
    sprintf(test_8, "%05b", x);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_2, 9);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_8, 5);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_1, 1);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_8, 5);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);

    sprintf(test_9, "%02d", x);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_5, 9);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_9, 2);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_9, 2);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);
    PAUSE(2000);
  }
  END();
}
```

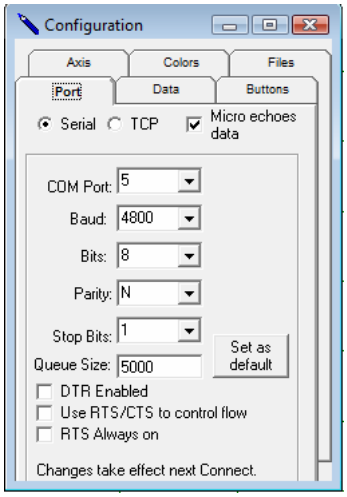
This code generates binary waves, a waveform showing the increasing number, and places that data into an updating window as it is received. All of the different

'**SEROUT**' commands are issued to assign different pieces of data into the 'Values' window. The '!SET 1' and '!LBLD 1' commands Label areas in the 'Values' window, and Sets the specific data that will be received in those areas.

After you have compiled and downloaded your code onto the C Stamp, start up StampPlot Pro. You will see eight different configurations to choose from. For this example, we will be using the 'Standard with Control Objects' plot. When the window loads, you should see the following.

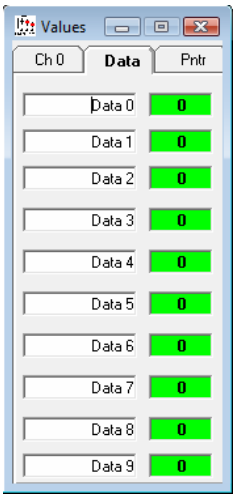


We will now configure the connection for this example. In order to do this, click the configuration button  or click the 'View' drop down menu and then 'Configuration', a window should appear that looks like the following.

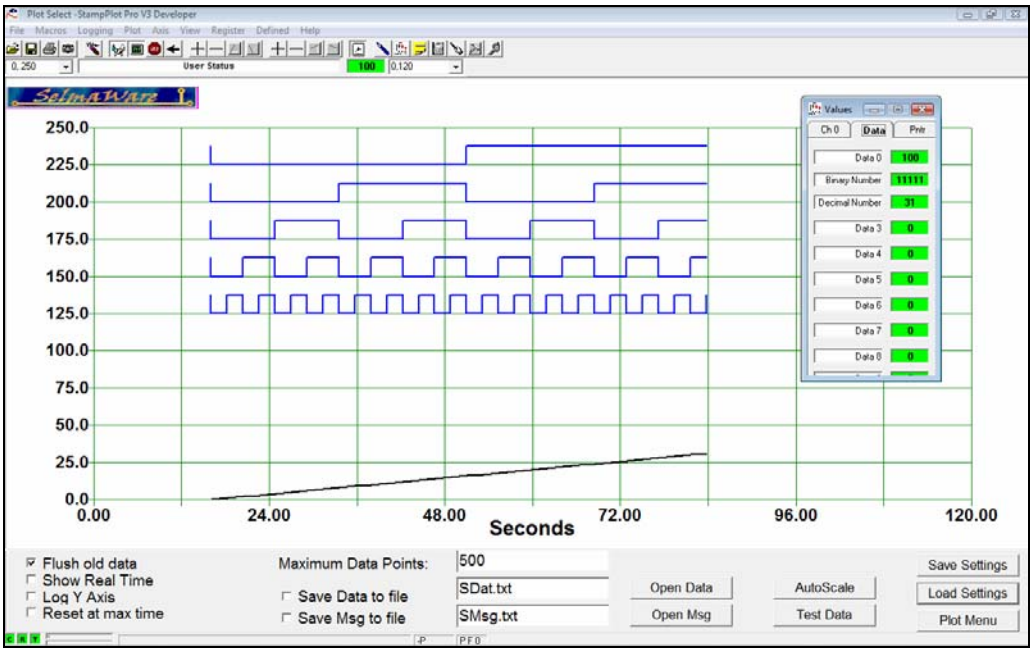


From here, all of the settings that will be required to receive data can be set. Set your window to look like the one above for the exception of the COM Port. For that selection, set it to the COM Port that you are using.

Now we can connect to the C Stamp by pressing the 'Connect to Stamp on COM Port' button  , and by making sure the 'Start Plot' button  is pressed. Once you have successfully connected, open up the 'Values' window. To do so either click the 'View' drop down menu and then click 'Values', or you can click the 'Values' button  . The values window will pop up. Select the 'Data' tab. You should see the following.



Now that you have the window open, and if have successfully downloaded your code and connected to the C Stamp, run your program. You should begin to see the binary graphs form along with the number graph form as well. In addition, the data will be entered into the 'Values' window.



For an added bonus, the following code also incorporates the '**rand**' command to generate random numbers and graphs them as a binary and decimal number. Copy the following code, or add the changes to the previous code. The changes are in italics and underlined.

```
#include   "CS110000.h"

void main(void)
{
// TYPE YOUR CODE HERE AFTER THIS LINE
  RAM BYTE msg[] = "\r";
  RAM BYTE test[] = ",";
  RAM BYTE test_1[] = "%";
  RAM BYTE test_2[] = "!SETD 1,";
  RAM BYTE test_3[] = "!LBLD 1";
  RAM BYTE test_4[] = "Binary Number";
  RAM BYTE test_5[] = "!SETD 2,";
  RAM BYTE test_6[] = "!LBLD 2";
  RAM BYTE test_7[] = "Decimal Number";
```

```
  int x;
  float baudr = 4.8;
  BYTE test_8[7];
  BYTE test_9[3];

  STPIND(46, HIGH);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_3, 8);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test, 1);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_4, 14);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);

  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_6, 8);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test, 1);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, test_7, 14);
  SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);

  while(1){
    x = rand() % 127 + 1;
    sprintf(test_8, "%07b", x);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_2, 9);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_8, 7);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_1, 1);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_8, 7);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);

    sprintf(test_9, "%03d", x);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_5, 9);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_9, 3);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, test_9, 3);
    SEROUT(0, 0, baudr, 0, 8, 0, 0, msg, 1);
    PAUSE(2000);
  }
  END();
}
```

Although not much of the code has changed, with the addition of the '**rand**' command and addition of two more binary places, more numbers will be generated. The code is explained in more detail in the previous section: Binary Graphing. Also, if you do run this program, make sure you change the 'Analog Max…' to 500 so none of the graphs interferes with each other.

Developing Your Own Programs and Projects

Now that you have successfully developed and run your programs, it is easy to move on to more complex and elaborate projects and circuits of your own. Try to mix and match the above projects, and write your own. Also, remember the different types of formatting that are required for StampPlot Pro to read your code.

Questions and Answers

Here we will try to answer some of the most common questions. If you do not see your question here, look to the 'Getting Support' section on how to contact us with your problem. Also, please remember to read the StampPlot Pro Manual for information on how the program works.

Q: No data is being plotted. Why is this?

A: If you do not see any graphs forming, click the 'Autoscale' button. If no data is being plotted, and your C Stamp program is running, click the 'View' drop down menu and then 'Log (Debug/Immediate)'. Inside will give a window of all the data that is being received. Check the 'Anlg' button, if nothing is showing up while your C Stamp is running, return to your code and make sure that you have copied everything correctly.

Terms and Conditions

Quality Assurance

A-WIT has stringent quality control procedures in place to insure the best quality products.

90-Day Limited Warranty

A-WIT Technologies, Inc warrants its products against defects in materials and workmanship for a period of 90 days. If you discover a defect, A-WIT Technologies, Inc. will, at its option, repair, replace, or refund the purchase price. After 90 days, products can still be sent in for repair or replacement, but there will be a $10.00USD minimum inspection/labor/repair fee (not including return shipping and handling charges).

14-Day Money-Back Guarantee

If, within 14 days of having received your product, you find that it does not suit your needs, you may return it for a refund. A-WIT will refund the purchase price of the product in the form of a check, excluding shipping/handling costs, once the product is received. This refund does not apply if the product has been altered or damaged. If you decide to return the products after the 14-day evaluation period, a 20% restocking fee will be charged against a credit.

Disclaimer

Warranty does not apply if the product has been altered, modified, or damaged. A-WIT makes no other warranty of any kind, expressed or implied, including any warranty of merchantability, fitness of the product for any particular purpose even if that purpose is known to A-WIT, or any warranty relating to patents, trademarks, copyrights or other intellectual property. A-WIT shall not be liable for any injury, loss, damage, or loss of profits resulting from the handling or use of the product shipped.

How to Return a Product

When returning, you must first e-mail sales@a-wit.com for a Return Merchandise Authorization number. No packages will be accepted without the RMA number clearly marked on the outside of the package. After inspecting and testing, we will return your product, or its replacement using the same shipping method used to ship the product to A-WIT within 30 days. In your package, please include a daytime telephone number and a brief explanation of the problem.

Please contact our Sales Department at sales@a-wit.com if you have any questions regarding our warranty policy or if you are requesting an RMA number.