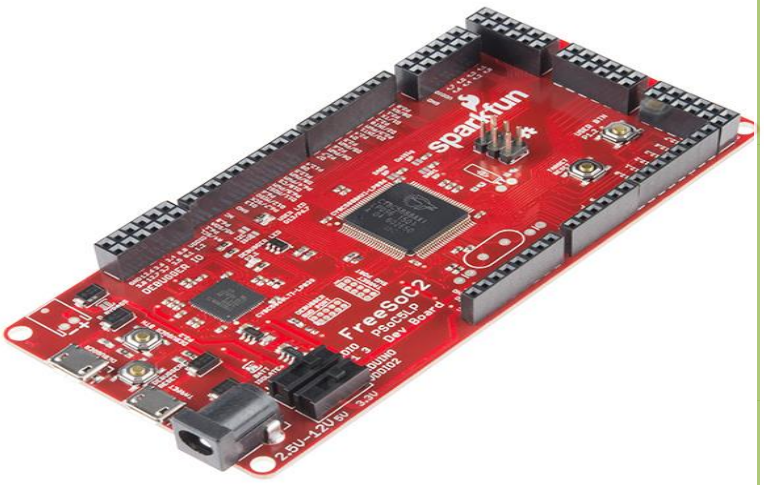
2015

TENET TECHNETRONICS

Sensing analog inputs with FreeSoc2 using Cypress Creator



Author: Gurudatta Palankar

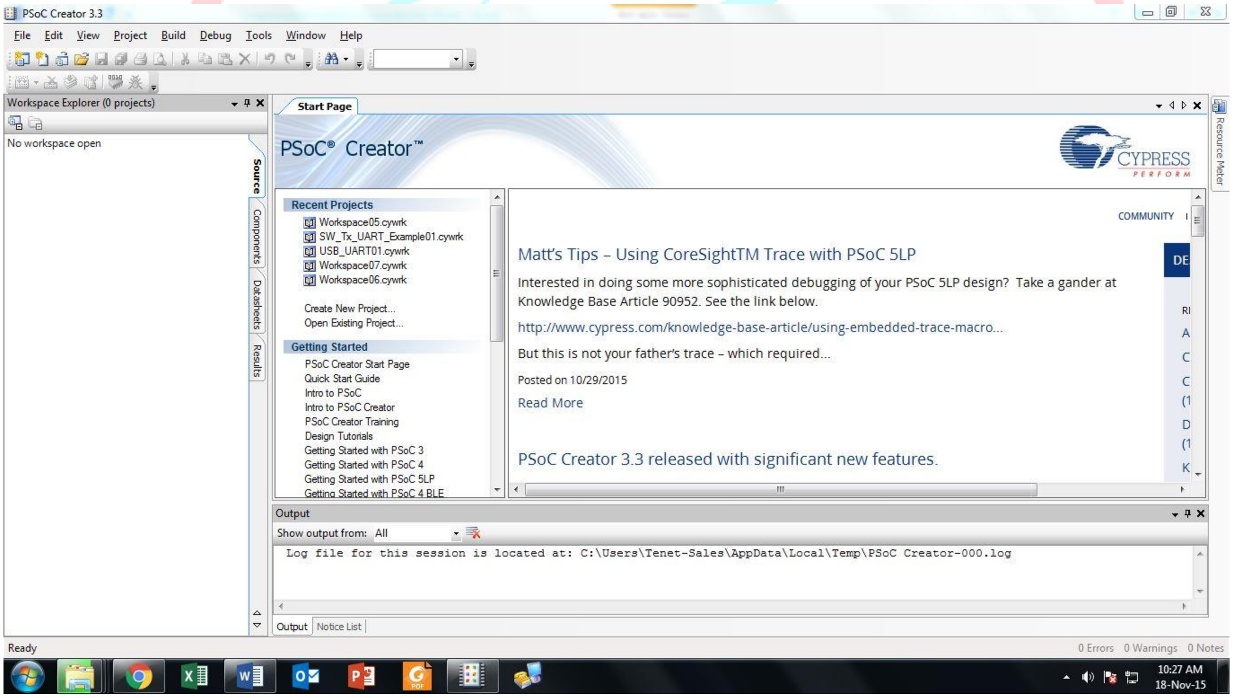
Reviewers:

Version: 1.0

Introduction:

The FreeSoC2 microcontroller based on the PSoC 5LP (Programmable System on a Chip) brings together features of the programmable devices and microcontroller-type systems on chips into one package. By placing a programmable fabric between the peripherals and the pins, the FreeSoC2 allows any function to be routed to any pin! Moreover, the on-board PSoC includes a number of programmable blocks which allow the user to define arbitrary digital and analog circuits for their specific application. To get the most out of the device, you will need to use the PSoC Creator IDE.

Step 1: Open **PSOC creator IDE.**



Matt's Tips – Using CoreSightTM Trace with PSoC 5LP

Interested in doing some more sophisticated debugging of your PSoC 5LP design? Take a gander at Knowledge Base Article 90952. See the link below.

http://www.cypress.com/knowledge-base-article/using-embedded-trace-macro...

But this is not your father's trace – which required...

Posted on 10/29/2015

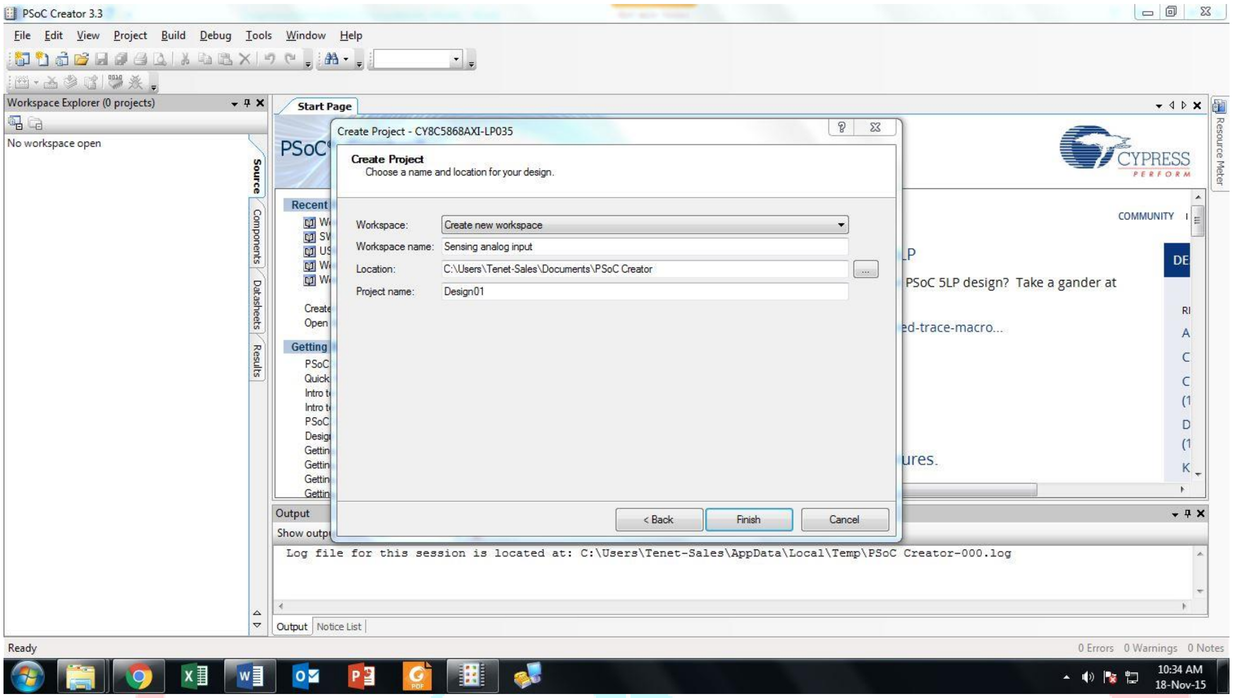
Read More

PSoC Creator 3.3 released with significant new features.

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator-000.log
```

Figure 1

Step 2: File-> new project -> design -> PSoC 5LP design & save with desired name.


PSoC Creator 3.3
File Edit View Project Build Debug Tools Window Help
Workspace Explorer (0 projects)
No workspace open
Start Page
Output
< Back
Finish
Cancel
Ready
0 Errors 0 Warnings 0 Notes
10:34 AM
18-Nov-15

Create Project - CY8C5868AXI-LP035

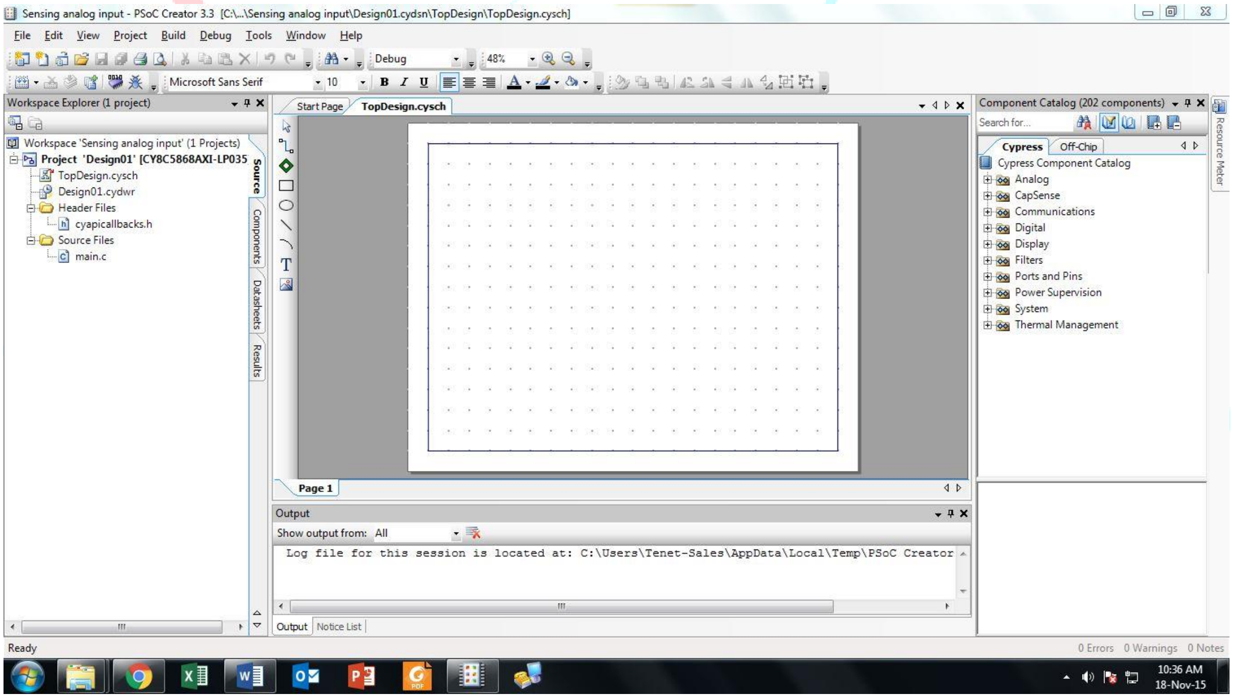
Create Project
Choose a name and location for your design.

Workspace:	Create new workspace
Workspace name:	Sensing analog input
Location:	C:\Users\Tenet-Sales\Documents\PSoC Creator
Project name:	Design01

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator-000.log
```

Figure 2

Step 3: Open **TopDesign.cysch** from workspace explorer.

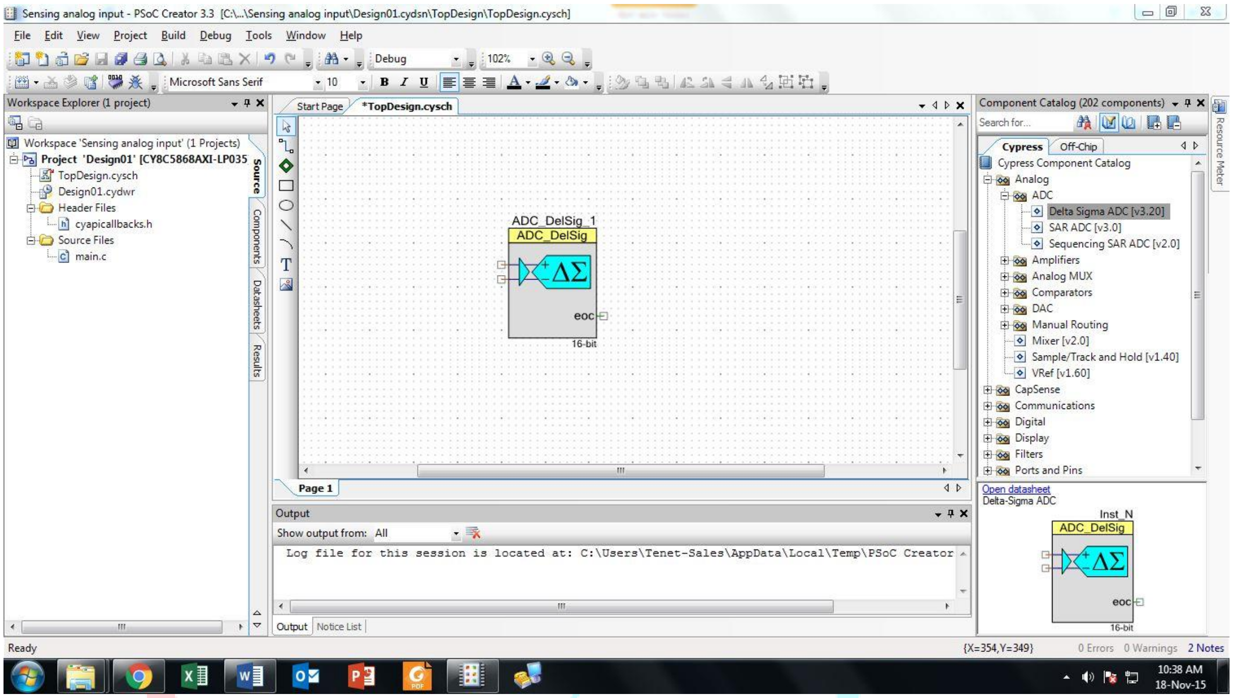

Sensing analog input - PSoC Creator 3.3 [C:\...\Sensing analog input\Design01.cydsn\TopDesign\TopDesign.cysch]
File Edit View Project Build Debug Tools Window Help
Debug
48%
Microsoft Sans Serif
10
Workspace Explorer (1 project)
Start Page
TopDesign.cysch
Component Catalog (202 components)
Search for...
Cypress
Off-Chip
Cypress Component Catalog
Analog
CapSense
Communications
Digital
Display
Filters
Ports and Pins
Power Supervision
System
Thermal Management
Page 1
Output
Show output from: All
Ready
0 Errors 0 Warnings 0 Notes
10:36 AM
18-Nov-15

Workspace 'Sensing analog input' (1 Projects)
- Project 'Design01' [CY8C5868AXI-LP035]
  - TopDesign.cysch
  - Design01.cydwr
  - Header Files
    - cyapicallbacks.h
  - Source Files
    - main.c

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator
```

Figure 3

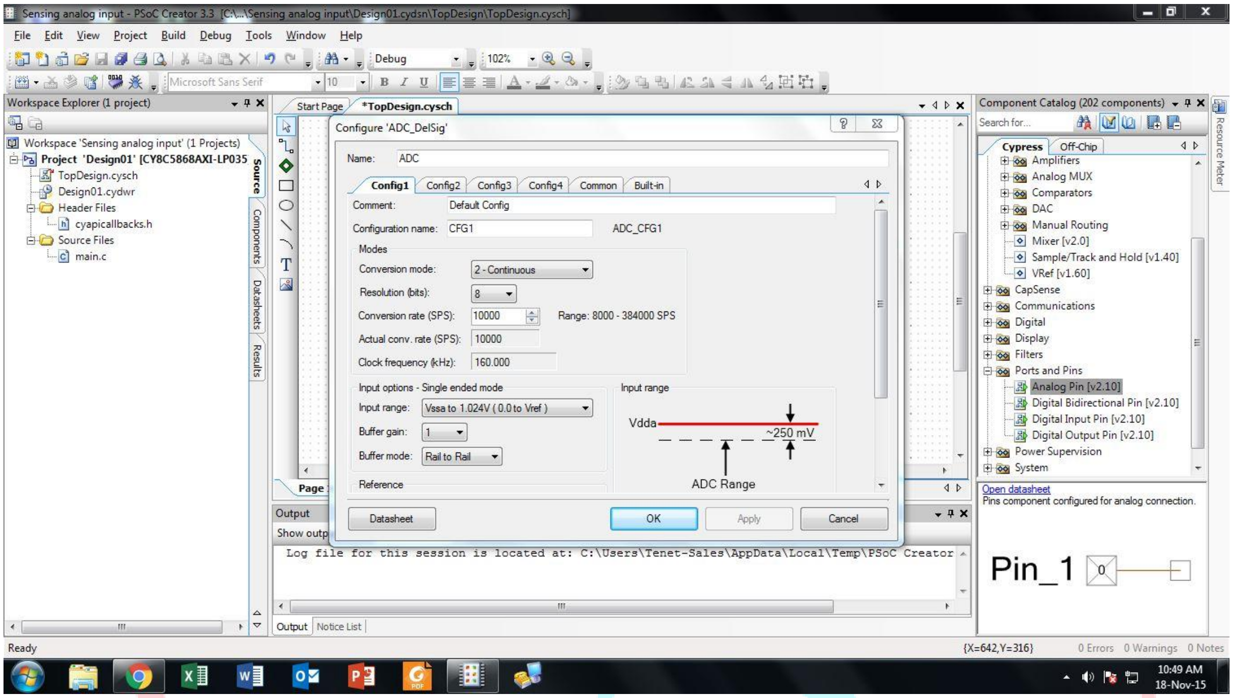
Step 4: To sense analog input we need Analog to Digital Converter (ADC). Let us select a Delta-Sigma ADC from Component catalog on right side of the window. Drag the ADC onto the workspace.



```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator
```

Figure 4

Step 5: Double click on the Delta-Sigma ADC and change the name if you wish to. Configure the ADC as Single ended input mode and 8-bit resolution.


Sensing analog input - PSoC Creator 3.3 [C:\...\Sensing analog input\Design01.cydsn\TopDesign\TopDesign.cysch]

Configure 'ADC_DelSig'

Name: ADC

Comment: Default Config

Configuration name: CFG1 ADC_CFG1

Modes

Conversion mode: 2 - Continuous

Resolution (bits): 8

Conversion rate (SPS): 10000 Range: 8000 - 384000 SPS

Actual conv. rate (SPS): 10000

Clock frequency (kHz): 160.000

Input options - Single ended mode

Input range: Vssa to 1.024V (0.0 to Vref)

Buffer gain: 1

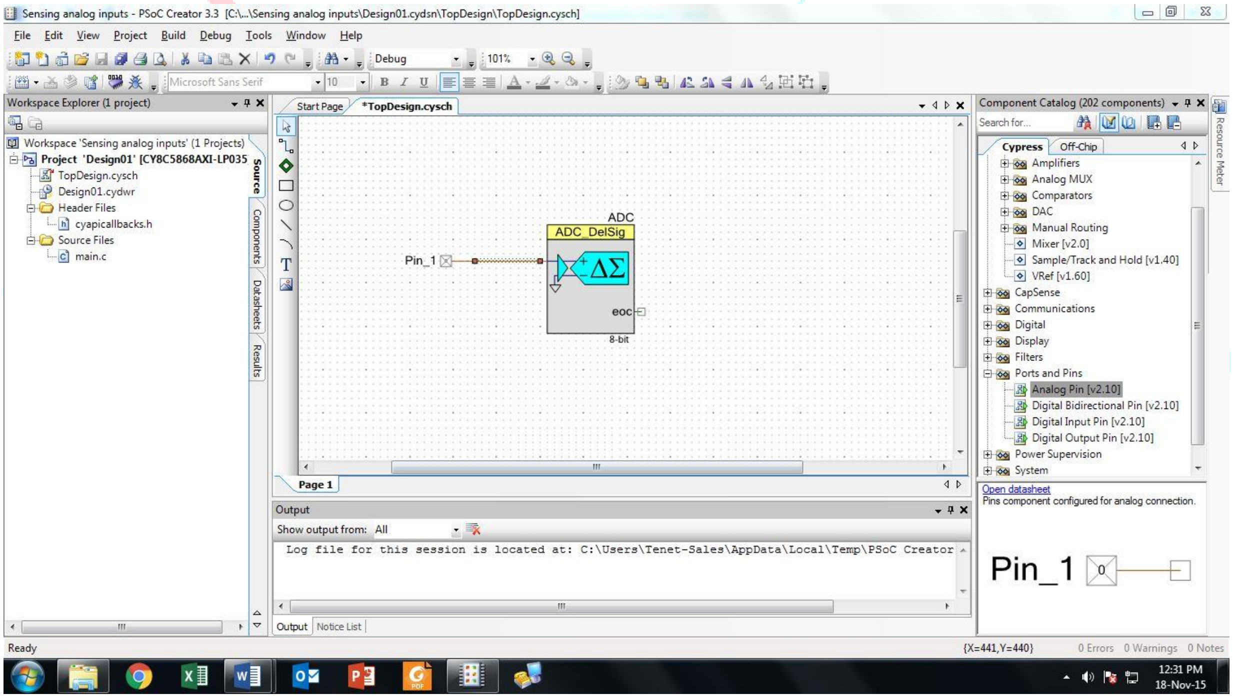
Buffer mode: Rail to Rail

Reference

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator
```

Figure 5

Step 6: Similarly drag the analog input from component catalog. Keep default configurations. Connect the analog input to ADC.

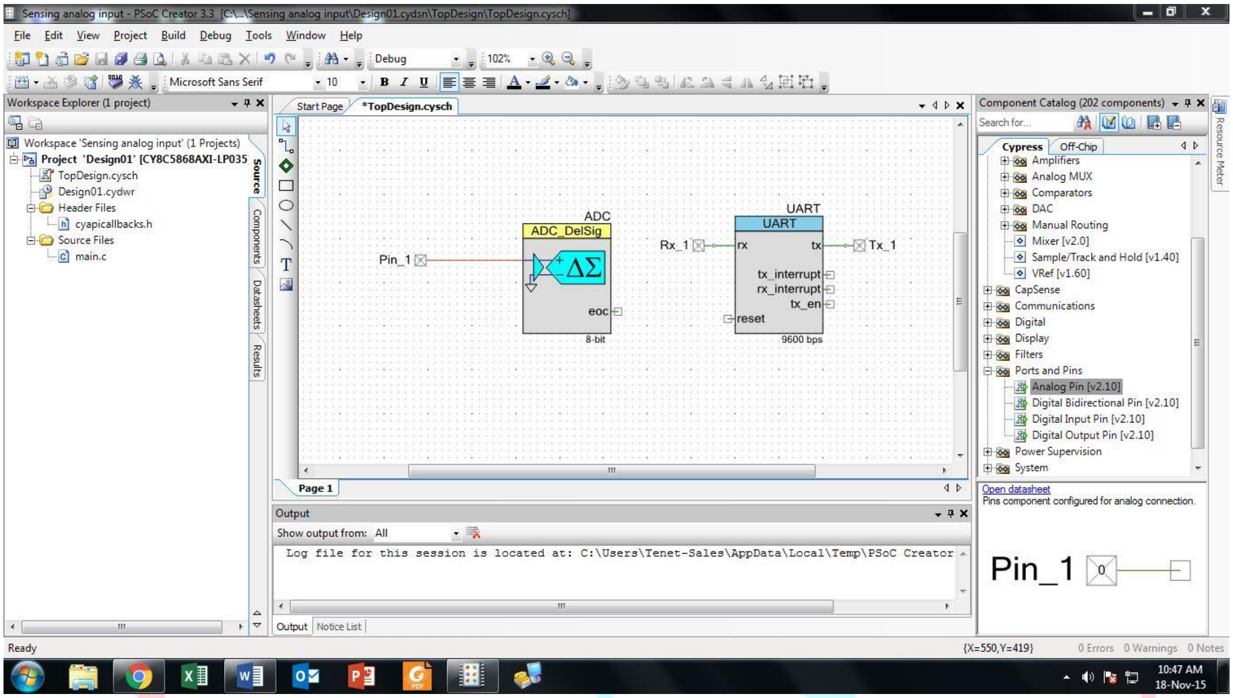

Sensing analog inputs - PSoC Creator 3.3 [C:\...\Sensing analog inputs\Design01.cydsn\TopDesign\TopDesign.cysch]

Show output from: All

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator
```

Figure 6

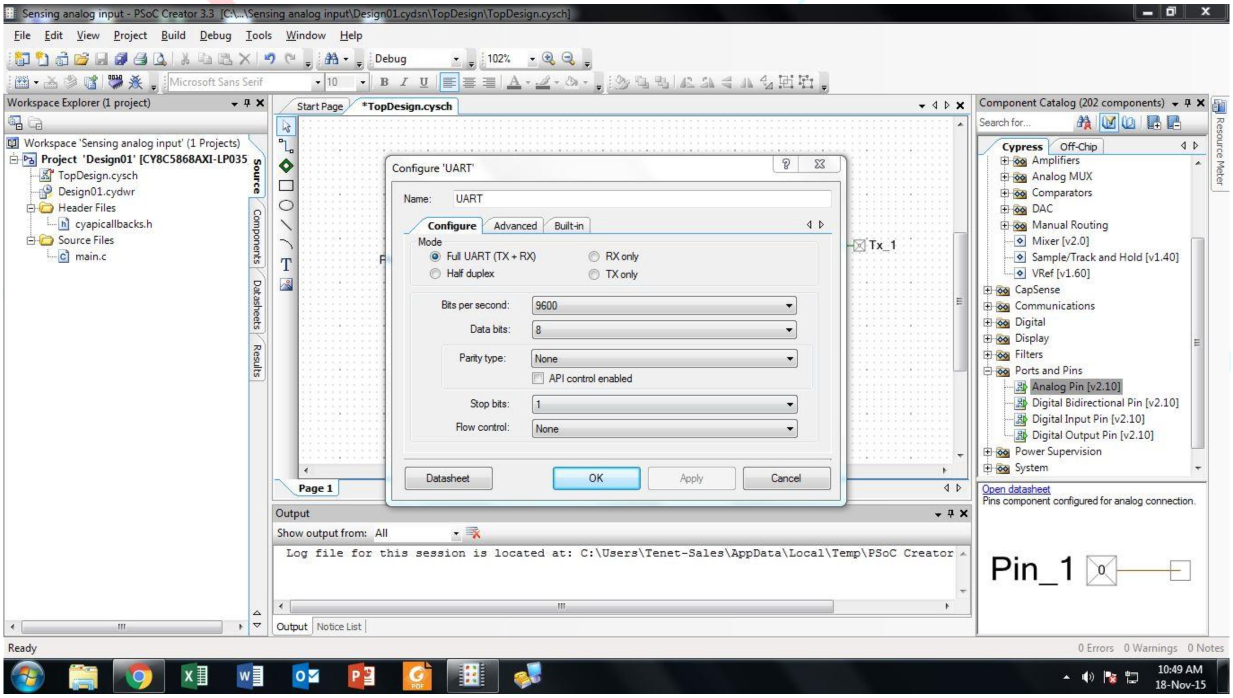
Step 7: Similarly we need UART for serial communication. It is required to read the digital values serially from ADC and transmits them serially onto a terminal (screen).


Sensing analog input - PSoC Creator 3.3 [C:\...\Sensing analog input\Design01.cydsn\TopDesign\TopDesign.cysch]
File Edit View Project Build Debug Tools Window Help
Workspace Explorer (1 project)
Workspace 'Sensing analog input' (1 Projects)
Project 'Design01' [CY8C5868AXI-LP035]
TopDesign.cysch
Design01.cydwr
Header Files
cyapicallbacks.h
Source Files
main.c
Start Page
*TopDesign.cysch
ADC
ADC_DelSig
Pin_1
eoc
8-bit
UART
Rx_1
rx
tx
Tx_1
tx_interrupt
rx_interrupt
tx_en
reset
9600 bps
Page 1
Component Catalog (202 components)
Cypress
Off-Chip
Amplifiers
Analog MUX
Comparators
DAC
Manual Routing
Mixer [v2.0]
Sample/Track and Hold [v1.40]
VRef [v1.60]
CapSense
Communications
Digital
Display
Filters
Ports and Pins
Analog Pin [v2.10]
Digital Bidirectional Pin [v2.10]
Digital Input Pin [v2.10]
Digital Output Pin [v2.10]
Power Supervision
System
Open datasheet
Pins component configured for analog connection.
Pin_1
Output
Show output from: All
Ready
{X=550,Y=419}
0 Errors 0 Warnings 0 Notes
10:47 AM
18-Nov-15

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator
```

Figure 7

Step 8: Configure UART as 8-bit and baud rate of 9600 bps.


Sensing analog input - PSoC Creator 3.3 [C:\...\Sensing analog input\Design01.cydsn\TopDesign\TopDesign.cysch]
Configure 'UART'
Configure
Advanced
Built-in
Datasheet
OK
Apply
Cancel
Component Catalog (202 components)
Pin_1
10:49 AM
18-Nov-15

Name: UART

Mode: ● Full UART (TX + RX) ○ RX only ○ Half duplex ○ TX only

Bits per second: 9600

Data bits: 8

Parity type: None

□ API control enabled

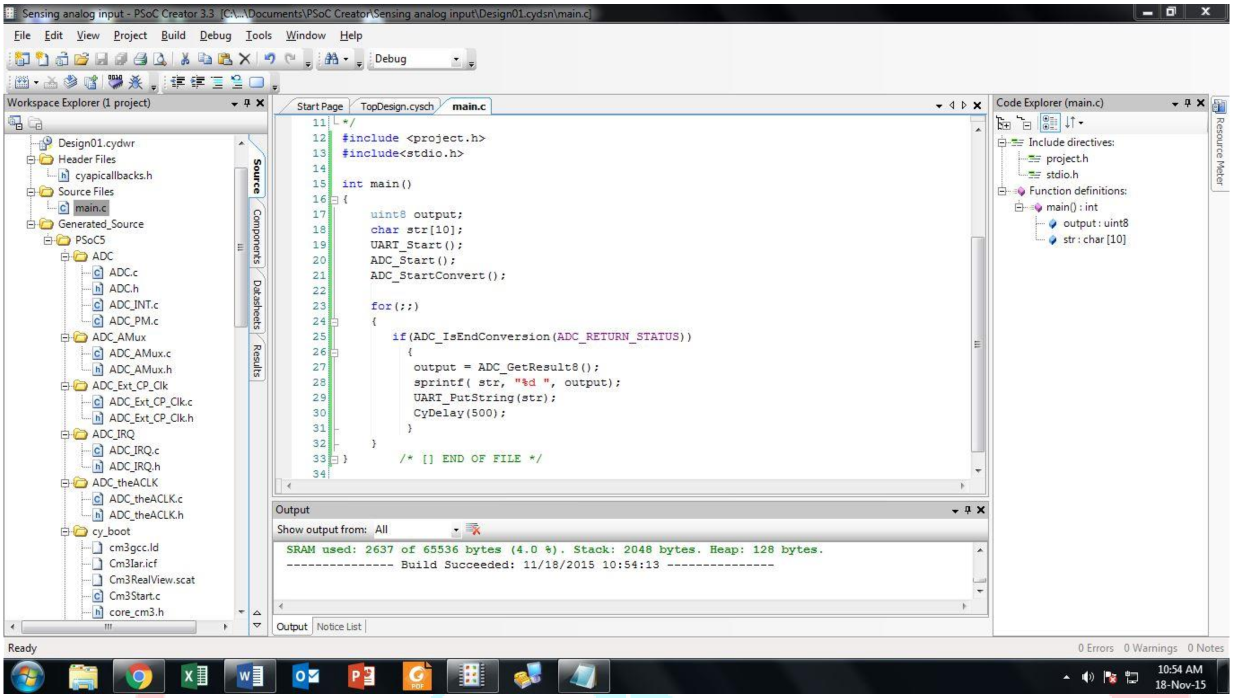
Stop bits: 1

Flow control: None

```
Log file for this session is located at: C:\Users\Tenet-Sales\AppData\Local\Temp\PSoC Creator
```

Figure 5

Step 9: Click on **main.c** from Workspace Explorer. Write the code and Build it.



```
*/
#include <project.h>
#include<stdio.h>

int main()
{
    uint8 output;
    char str[10];
    UART_Start();
    ADC_Start();
    ADC_StartConvert();

    for(;;)
    {
        if(ADC_IsEndConversion(ADC_RETURN_STATUS))
        {
            output = ADC_GetResult8();
            sprintf( str, "%d ", output);
            UART_PutString(str);
            CyDelay(500);
        }
    }
}    /* [] END OF FILE */
```

```
SRAM used: 2637 of 65536 bytes (4.0 %). Stack: 2048 bytes. Heap: 128 bytes.
--------------- Build Succeeded: 11/18/2015 10:54:13 ---------------
```

Figure 6

Code:

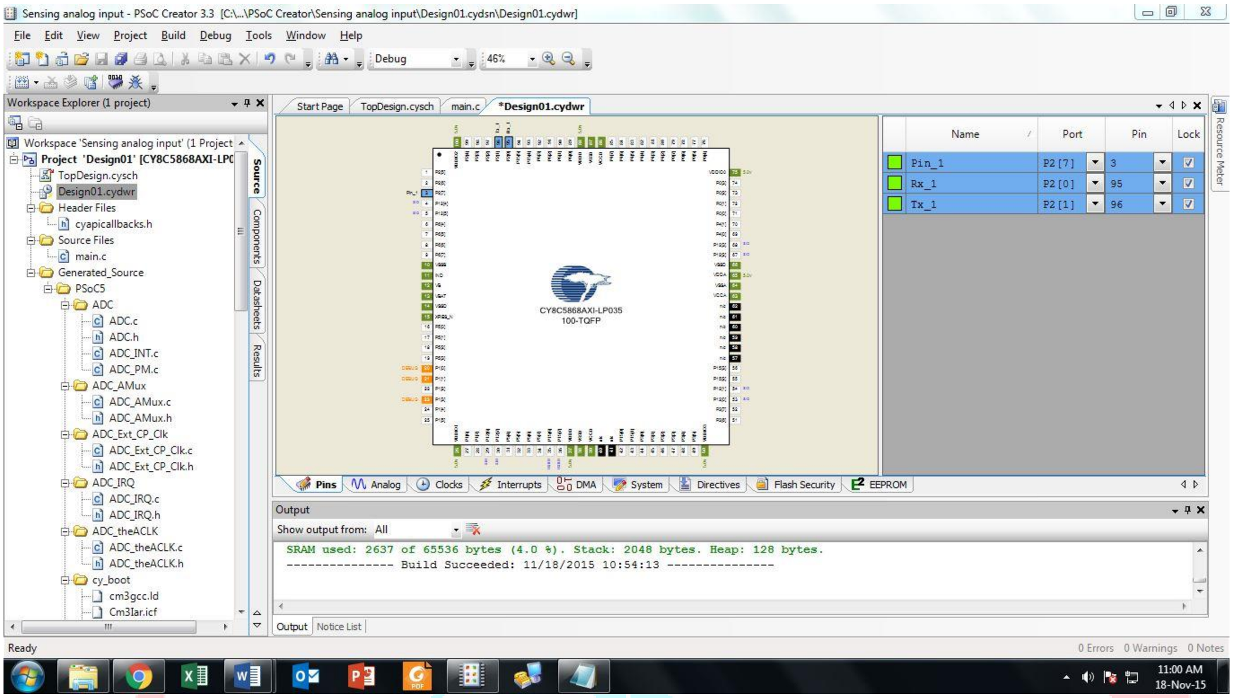
```
#include <project.h>
#include<stdio.h>

int main()
{
    int8 output;
    char str[10];
    UART_Start();
    ADC_Start();
    ADC_StartConvert();

    for(;;)
    {
       if(ADC_IsEndConversion(ADC_RETURN_STATUS))
         {
          output = ADC_GetResult8();
          ADC_CountsTo_Volts(output);
          sprintf( str, "%d ", output);

          UART_PutString(str);
          CyDelay(1000);
         }
    }
}/* [] END OF FILE */
```

Step 10: Click on **Design01.cydwr** from Workspace Explorer.



Sensing analog input - PSoC Creator 3.3 [C:\...\PSoC Creator\Sensing analog input\Design01.cydsn\Design01.cydwr]

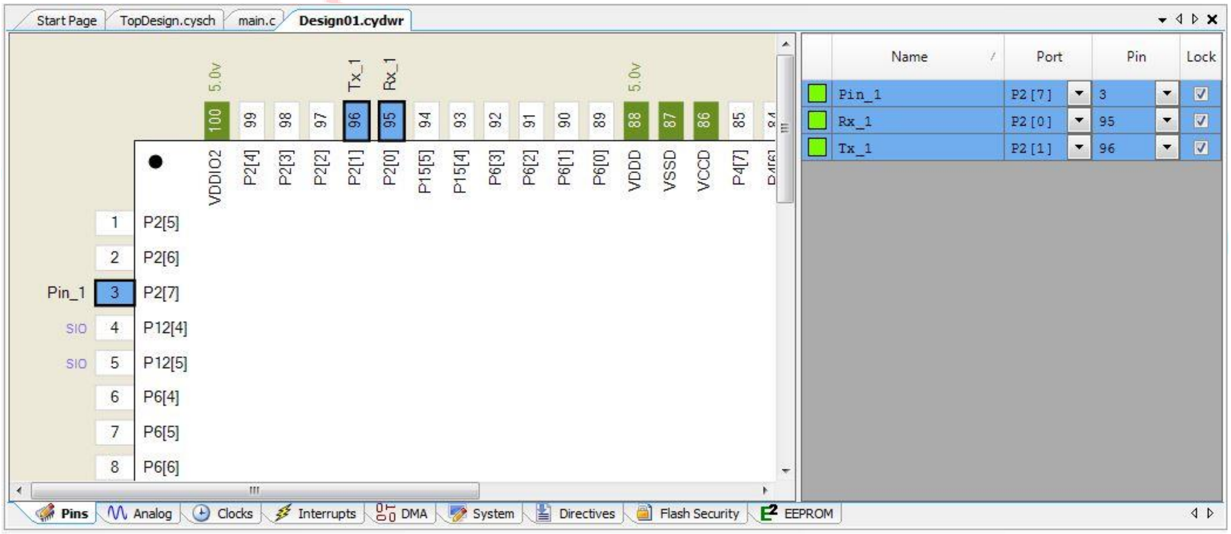
Workspace 'Sensing analog input' (1 Project)
- Project 'Design01' [CY8C5868AXI-LP0
  - TopDesign.cysch
  - Design01.cydwr
  - Header Files
    - cyapicallbacks.h
  - Source Files
    - main.c
  - Generated_Source
    - PSoC5
      - ADC
        - ADC.c
        - ADC.h
        - ADC_INT.c
        - ADC_PM.c
      - ADC_AMux
        - ADC_AMux.c
        - ADC_AMux.h
      - ADC_Ext_CP_Clk
        - ADC_Ext_CP_Clk.c
        - ADC_Ext_CP_Clk.h
      - ADC_IRQ
        - ADC_IRQ.c
        - ADC_IRQ.h
      - ADC_theACLK
        - ADC_theACLK.c
        - ADC_theACLK.h
      - cy_boot
        - cm3gcc.ld
        - Cm3Iar.icf

Name	Port	Pin	Lock
🟩 Pin_1	P2[7]	3	☑
🟩 Rx_1	P2[0]	95	☑
🟩 Tx_1	P2[1]	96	☑

```
SRAM used: 2637 of 65536 bytes (4.0 %). Stack: 2048 bytes. Heap: 128 bytes.
--------------- Build Succeeded: 11/18/2015 10:54:13 ---------------
```

Figure 10

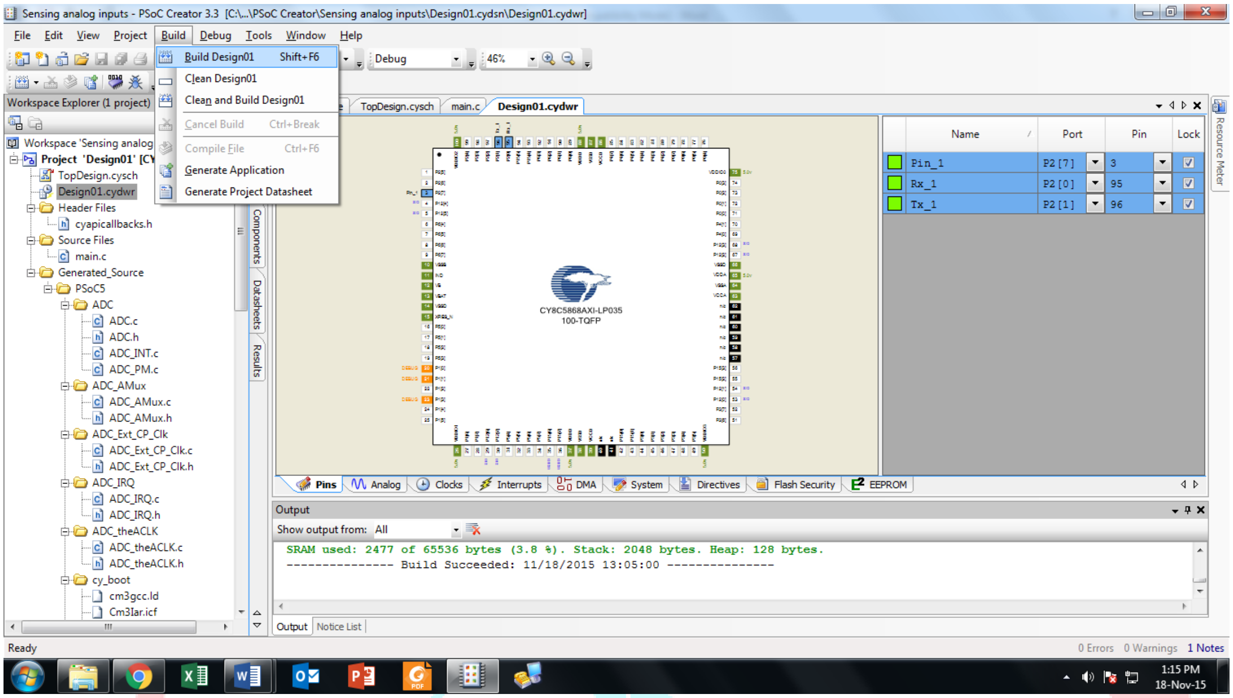
Step 11: Finally, assign pins to desired ports. Pin P2.0 is RX and P2.1 is TX for target IC. Then we connect the two pins to RX/TX pins on the debugger/programmer which is 12.6 (RX) and 12.7 (TX). Note that the RX pin from the PSoC 5LP MCU should be connected to the TX pin of the debugger. And the P2.1 TX pin should be connected to the debugger's RX pin (12.6).



Name	Port	Pin	Lock
🟩 Pin_1	P2[7]	3	☑
🟩 Rx_1	P2[0]	95	☑
🟩 Tx_1	P2[1]	96	☑

Figure 7

Step 12: Click on **Build** on menu bar and build the project, check for errors.

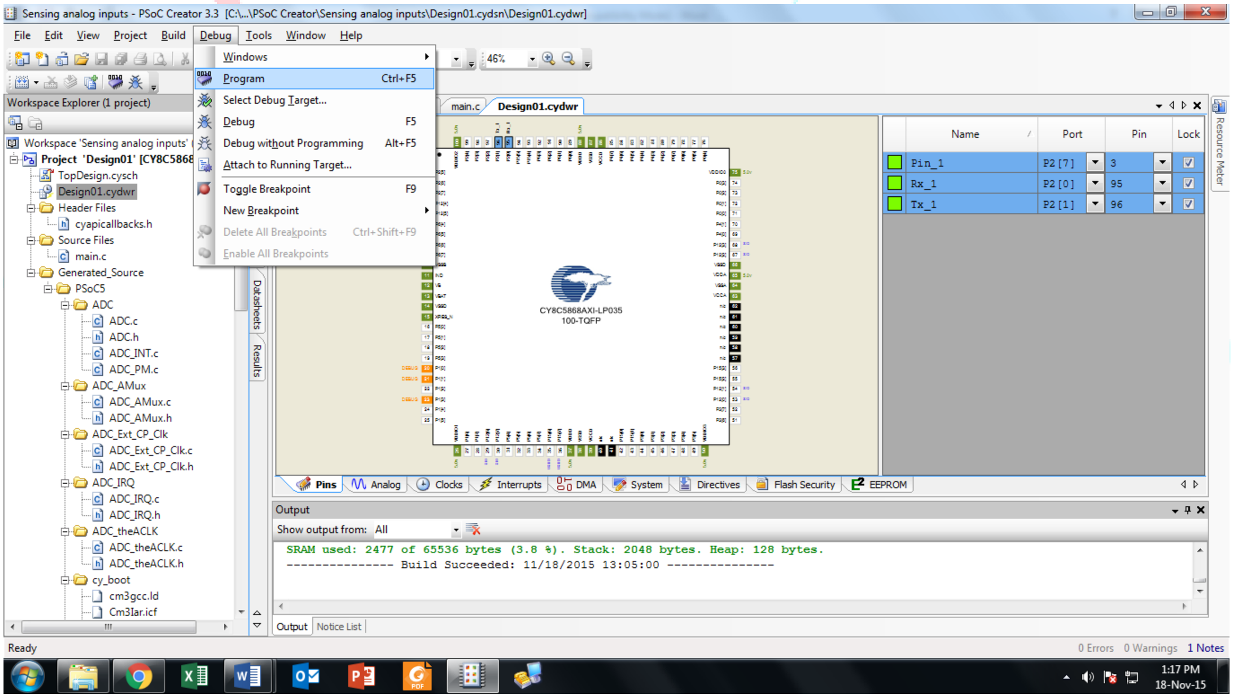


Name	Port	Pin	Lock
🟩 Pin_1	P2 [7]	3	☑
🟩 Rx_1	P2 [0]	95	☑
🟩 Tx_1	P2 [1]	96	☑

```
SRAM used: 2477 of 65536 bytes (3.8 %). Stack: 2048 bytes. Heap: 128 bytes.
--------------- Build Succeeded: 11/18/2015 13:05:00 ---------------
```

Figure 8

Step 13: Connect **Micro B** cable to debugger port of Freesoc2. To upload the file to Freesoc2 click on **Debug --> Program.**

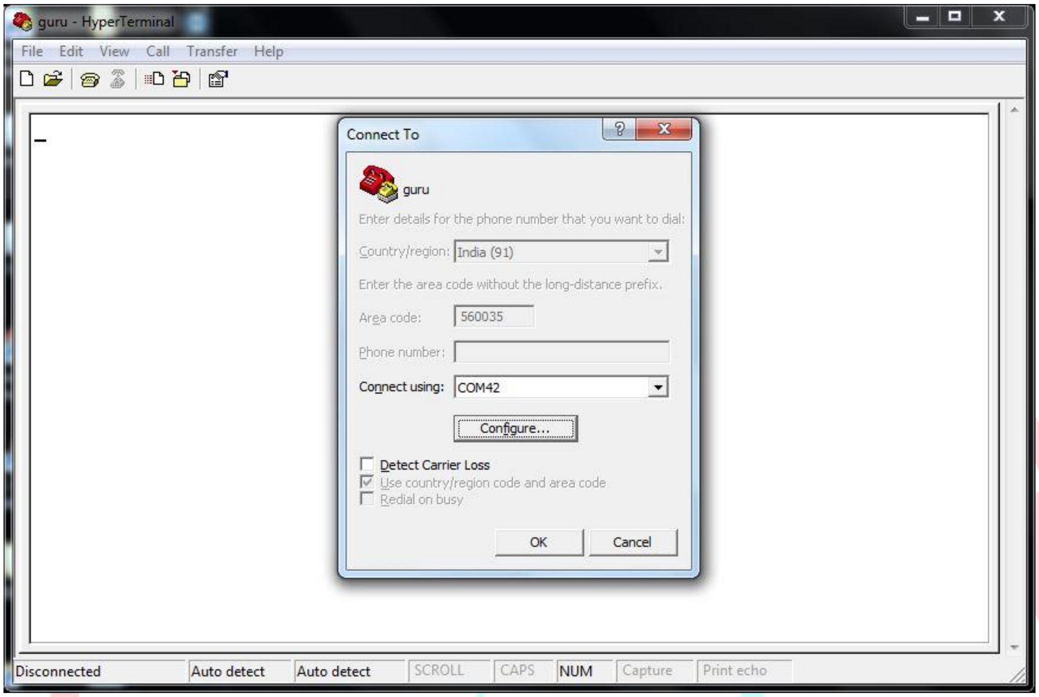


Name	Port	Pin	Lock
🟩 Pin_1	P2 [7]	3	☑
🟩 Rx_1	P2 [0]	95	☑
🟩 Tx_1	P2 [1]	96	☑

```
SRAM used: 2477 of 65536 bytes (3.8 %). Stack: 2048 bytes. Heap: 128 bytes.
--------------- Build Succeeded: 11/18/2015 13:05:00 ---------------
```

Figure 9

Step 14: Now to output the analog input onto screen we need a terminal. A terminal used to display via COM ports. You can use any terminal such as Putty, Hyper-Terminal, X-CTU and so on. I have used here Hyper-Terminal. Select appropriate COM port.



Connect To

guru

Enter details for the phone number that you want to dial:

Country/region: India (91)

Enter the area code without the long-distance prefix.

Area code: 560035

Phone number:

Connect using: COM42

Configure...

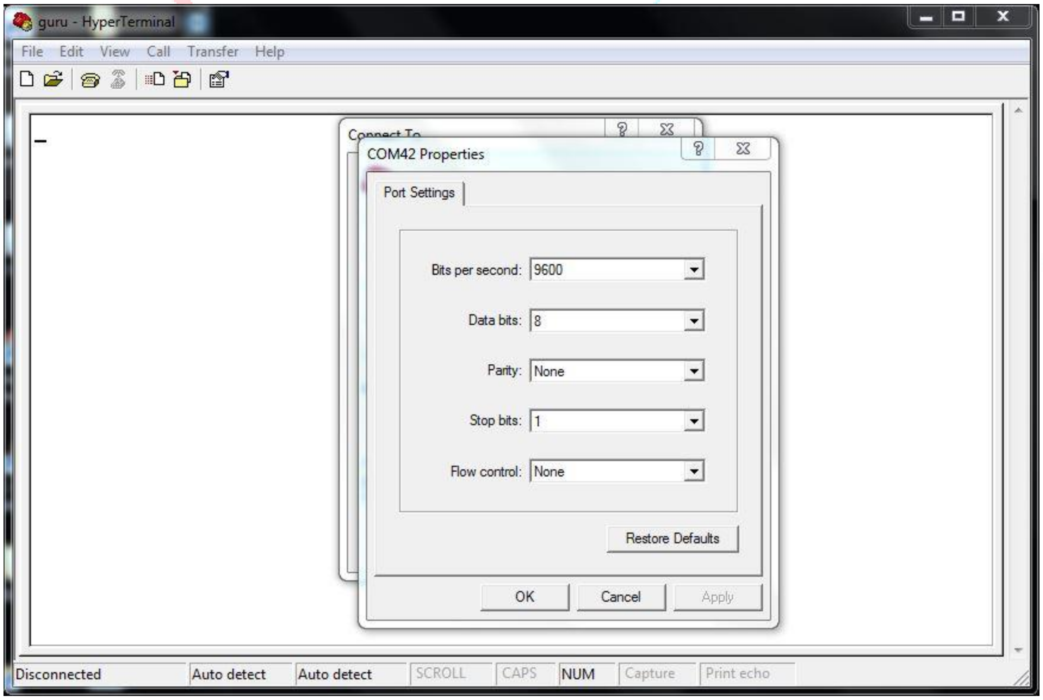
☐ Detect Carrier Loss
☑ Use country/region code and area code
☐ Redial on busy

OK Cancel

Disconnected | Auto detect | Auto detect | SCROLL | CAPS | NUM | Capture | Print echo

Figure 14

Step 15: Configure as 8-bit and baud rate of 9600 bps.



COM42 Properties

Port Settings

Bits per second: 9600

Data bits: 8

Parity: None

Stop bits: 1

Flow control: None

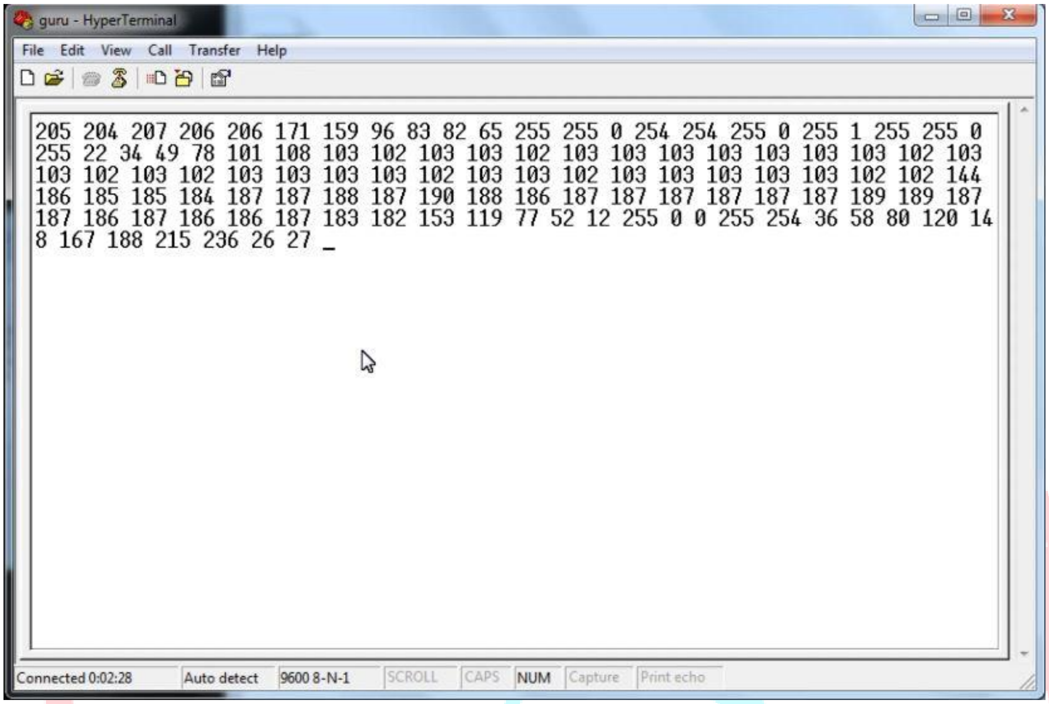
Restore Defaults

OK Cancel Apply

Disconnected | Auto detect | Auto detect | SCROLL | CAPS | NUM | Capture | Print echo

Figure 15

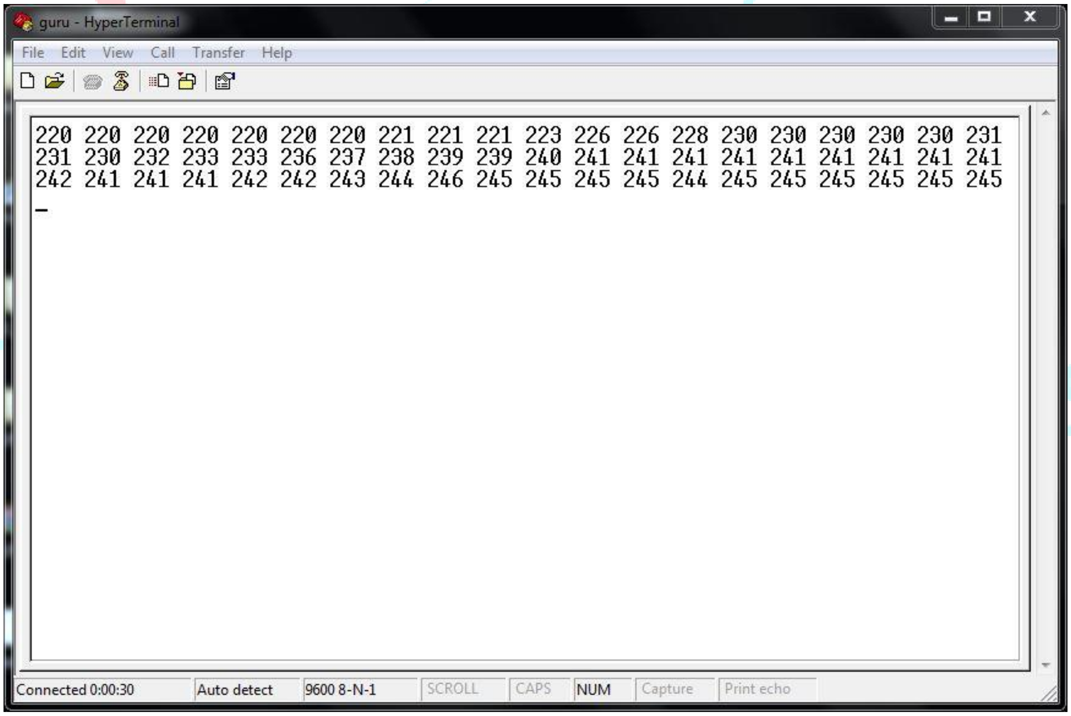
Step 16: Figure below shows the output of an LDR values (analog input).



```
205 204 207 206 206 171 159 96 83 82 65 255 255 0 254 254 255 0 255 1 255 255 0
255 22 34 49 78 101 108 103 102 103 103 102 103 103 103 103 103 103 103 102 103
103 102 103 102 103 103 103 103 102 103 103 102 103 103 103 103 103 102 102 144
186 185 185 184 187 187 188 187 190 188 186 187 187 187 187 187 187 189 189 187
187 186 187 186 186 187 183 182 153 119 77 52 12 255 0 0 255 254 36 58 80 120 14
8 167 188 215 236 26 27 _
```

Figure 16

Similarly we can use potentiometer as an analog input, as shown below



```
220 220 220 220 220 220 220 221 221 221 223 226 226 228 230 230 230 230 230 231
231 230 232 233 233 236 237 238 239 239 240 241 241 241 241 241 241 241 241 241
242 241 241 241 242 242 243 244 246 245 245 245 245 244 245 245 245 245 245 245
_
```

Figure 17

CIRCUIT EXPLAINATION:

The UART module we just added is typically used to achieve a serial communication link using some of the GPIO pins. In this case we want to be able to transfer our data through the USB interface to our PC to avoid the need of any additional cables and equipment.

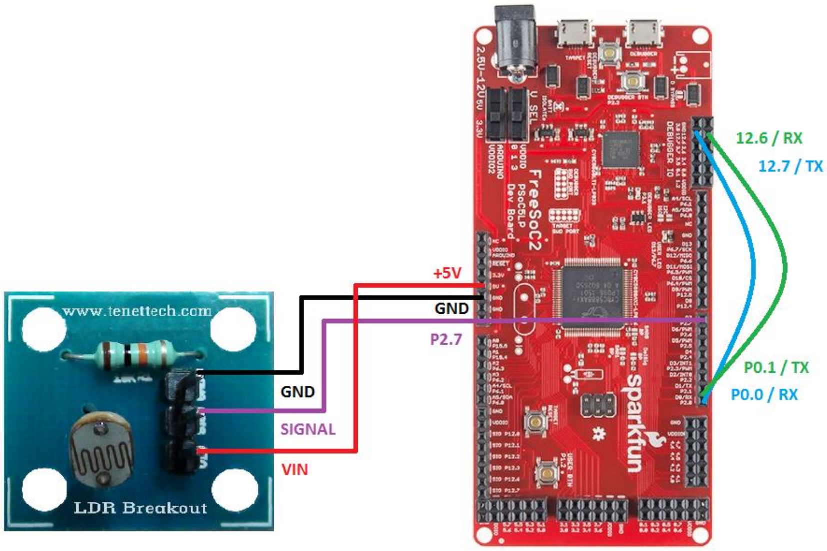


Figure 18

Pin P2.0 is RX and P2.1 is TX. We connect the two pins to RX/TX pins on the debugger/programmer which is 12.6 (RX) and 12.7 (TX). Note that the RX pin from the PSoC 5LP MCU should be connected to the TX pin of the debugger. And the P2.1 TX pin should be connected to the debugger's RX pin (12.6).

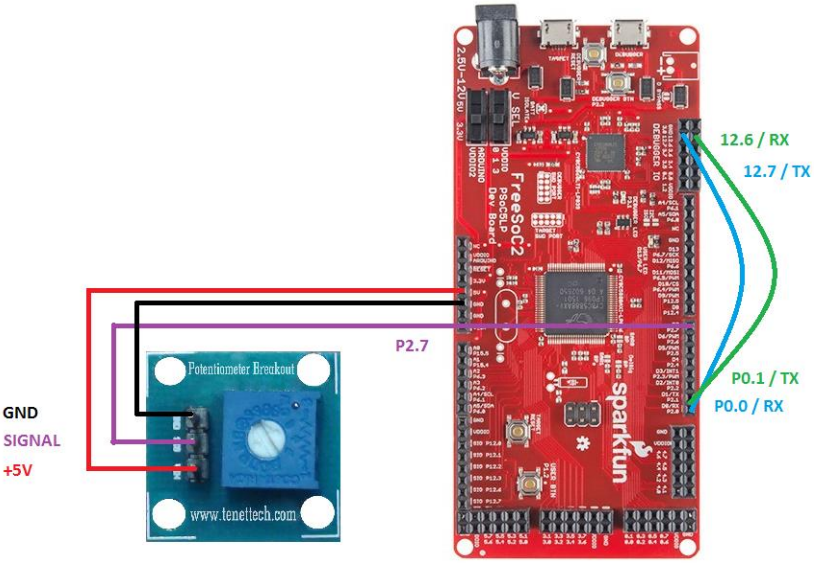


Figure 19

OUTPUT:

Reading LDR values

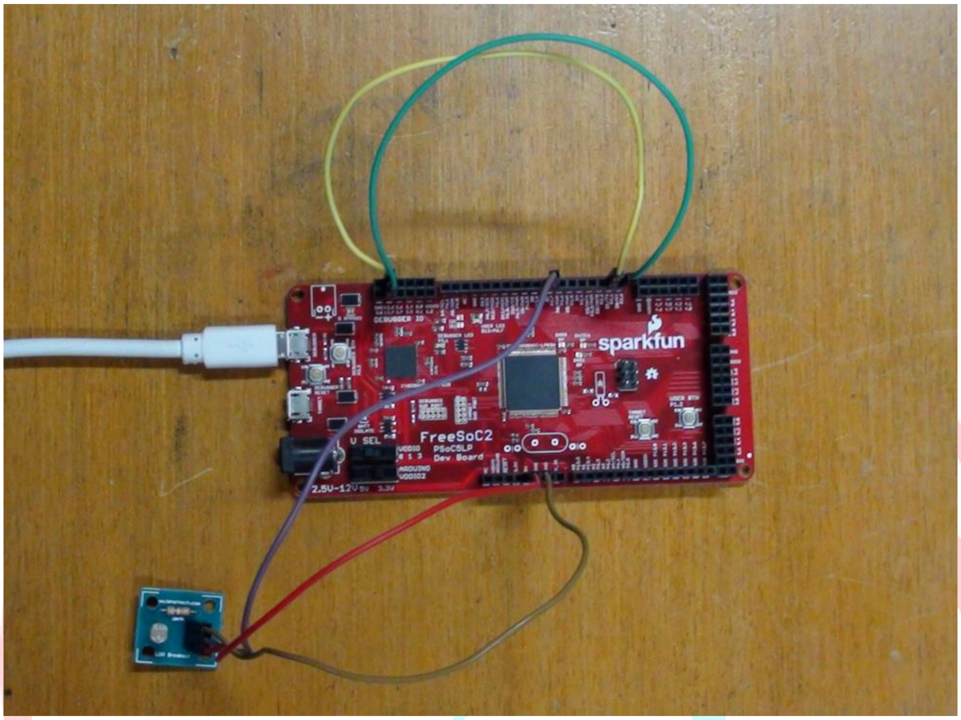


Figure 19

Reading Potentiometer values

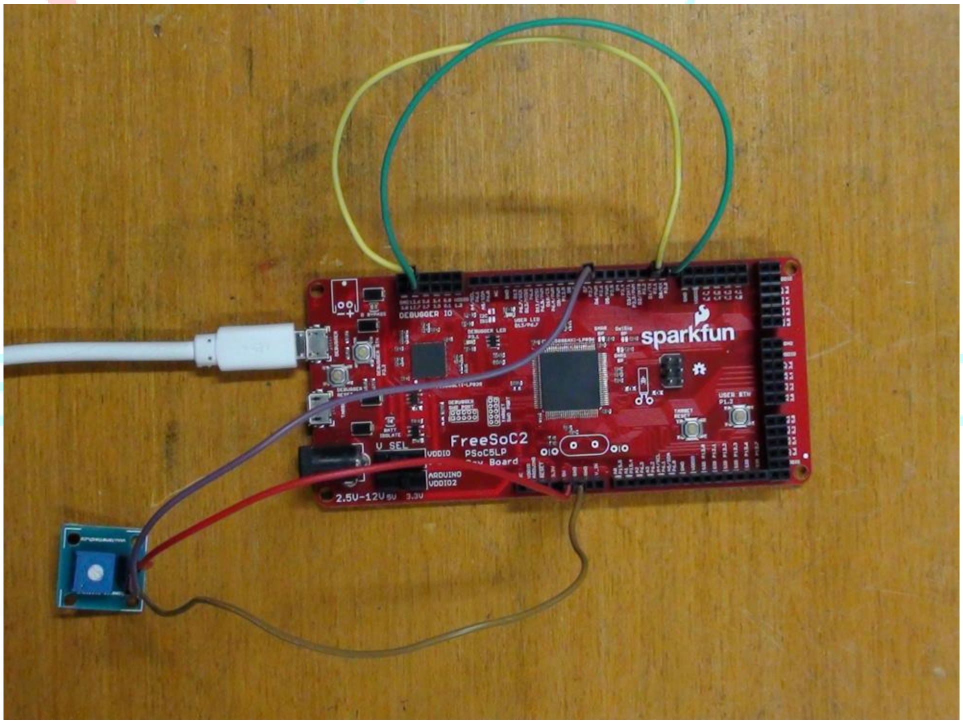


Figure 20

For product link:

1. http://www.tenettech.com/product/7241/freesoc2-development-board-psoc5lp
2. http://tenettech.com/product/6655/universal-gpio-board

For more information please visit: www.tenettech.com

For technical query please send an e-mail: info@tenettech.com