Code Example-ADCINC

Project Name: Example_ADCINC
Programming Language: C
Associated Part Families: CY8C24x23, CY8C27x43, CY8C29x66, CY8C24x94, CY8C21x34
Software Version: PSoC® 1 Designer™ 5.2
Related Hardware : CY3210 PSoCEval1 Board

Project Objective

This project demonstrates the operation of the ADCINC user module in PSoC® 1.

Overview

In this project, an analog input voltage is measured using the ADCINC user module, converted into voltage and displayed on an LCD display.

User Module List and Placement

The following table lists the user modules used in this project and the hardware resources occupied by each user module.

User Module	Placement
ADCINC	ASC10(Integrator) DBB00(PWM)
PGA	ACB00
LCD	P2[0] to P2[6]

User Module Parameter Settings

The following tables show the user module parameter settings for each of the user modules used in the project.

PGA		
Parameter	**Value**	**Comments**
Gain	1.000	PGA acts as a buffer between ADCINC positive input and P0 [1].
Input	AnalogColumn_InputMUX_0	Output of AnalogColumn_InputMUX_0 is connected to the input of PGA.
Reference	VSS	Reference for the PGA is set to VSS.

ADCINC		
Parameter	**Value**	**Comments**
DataFormat	Unsigned	As we are measuring a unipolar signal, the output is set to unsigned.
Resolution	14 bit	ADC resolution used is 14 bits.
Data Clock	VC1	Column clock for ADCINC is 2 MHz
Clock Phase	Normal	See Note
PosInput	ACB00	Output of ACB00 block is connected to the positive input of ADCINC.
NegInput	ASD11	Not used.
NegInputGain	Disconnected	Not used
PulseWidth	1	Pulse width of PWM is one cycle of data clock.
PWM Output	None	Not used.

Note:

The clock to the digital block of ADCINC should be equal to the column clock of analog block in which the integrator part of ADCINC has been implemented.

If the input to the ADC is from a CT Block or from direct port pin, then set this value to Norm. Or if the input is from another SC block, then set this value to swapped.

More details on the ClockPhase and column clock can be found in the article PSoC1 ADCs – The Five Golden Rules.

LCD		
Parameter	**Value**	**Comments**
LCDPort	Port_2	Port 2 is used to connect the LCD
BarGraph	Disable	Bar graph is not used in this code example

Global Resources

Important Global Resources		
Parameter	**Value**	**Comments**
CPU Clock	SysClk/2	Sets the CPU frequency to 12 MHz
VC1	12	Divide 24 MHz system clock by 12 to get 2 MHz clock which is the column clock to ADCINC.
Analog Power	SC On / Ref High	SC Blocks are On and Reference power is High. See note.
Ref Mux	(Vdd/2) +/- (Vdd/2)	Ref High = 5 V Ref Low = 0 V AGND = 2.5 V

Note:

The analog reference power should be set to the maximum power used by any analog resource. More details on this setting can be found in the article PSoC 1 ADCs – The Five Golden Rules

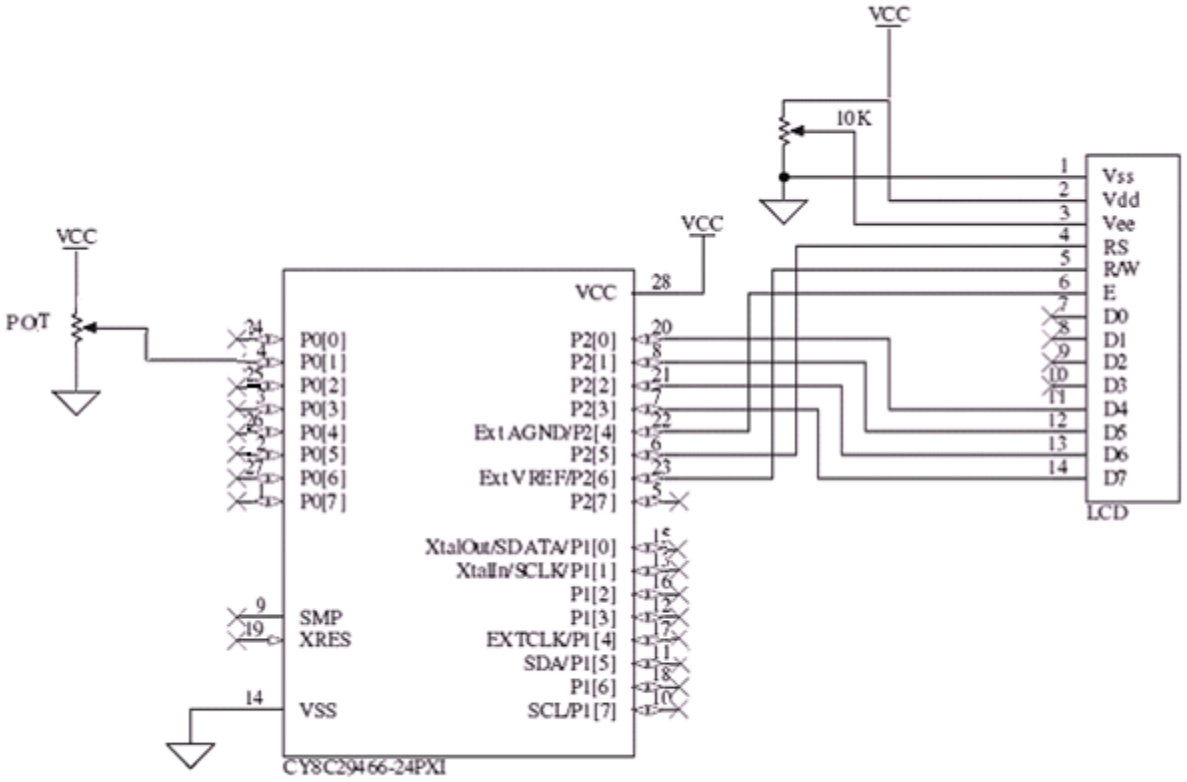
All other global resources are left at their default, as they are not specific to this project.

Pin Configuration

Pinouts			
Pin	**Select**	**Drive**	**Direction**
Port 0_1	Analog Input	High Z Analog	Input
Port2 _0-Port2_6	StdCPU	Strong	Output

Hardware Connections

The following figure shows the schematic of the project.



The input comes from a potentiometer, which provides a signal of 0 to VCC. The input is connected to P0[1]. The LCD is connected to Port 2.

The project can be tested using "CY3210 – PSoC Eval1" evaluation board. To test the project using the CY3210 board, make the following connections.

- Connect P0[1] of J6 to VR of J5
- Connect the 2x16 LCD display to header J9

Operation

Upon program execution, all hardware settings from the device configuration are loaded into the device and *main.c* is executed.

The following operations are performed inside *main.c*.

1. Enable global interrupts.
2. Start PGA in high power mode.
3. Start ADCINC in high power mode.
4. Start LCD.
5. Run the ADCINC in continuous sampling mode.
6. Check if ADC data is available. If yes, go to the step 7. If not, repeat step 6.
7. Read the ADC data and clear the status flag.
8. Convert the ADC result to voltage by multiplying the ADC data by the constant SCALE_FACTOR. Calculate SCALE_FACTOR by using formula

   SCALE_FACTOR = Input Voltage / $2^{Resolution}$.

   Enter this value in the SCALE_FACTOR macro defined along with the global variables.

   For this code example, the input voltage is 5 V and the resolution is 14 bits. Hence the SCALE_FACTOR is 0.00030517578125.
9. Set the LCD cursor to Row 0 Column 0 and display the string “Input Voltage”.
10. Set LCD cursor to Row 1 Column 0 and clear the previously displayed value.
11. Set LCD cursor to Row 1 Column 0.
12. Convert the float value obtained in step 8 to ASCII using ftoa() function and display the string returned by the ftoa() function on LCD.
13. Display the character “V” on the LCD in current cursor position.
14. Go to step 6.

Testing the Project

To test the project using CY3210 PSoCEval1 board, perform the following steps.

- Make the connections as shown in.
- Hardware Connections section.
- Vary the potentiometer and observe the voltage being displayed on the LCD.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
http://www.cypress.com/