Lab 8: ADC and Data Acquisition

Name: ______________________ ID: __________ Section: ________

Objective

To introduce Cortex M4 Analog-to-Digital-Converter (ADC) programming to measure analog voltage signal.

In-Lab

Task 1: To design a data acquisition system for light intensity monitor using LDR in Keil

Task 2: To design DAQ in Energia Environment

Task 3: To Measure Analog Voltage Signal using Potentiometer

1 Analog-to-digital Converters (ADC)

ADC is an important component when it comes to dealing with digital systems communicating with real-time signals. Analog signals are signals that have a continuous sequence with continuous values (there are some cases where it can be finite). These types of signals can come from sound, light, temperature and motion. Digital signals are represented by a sequence of discrete values where the signal is broken down into sequences that depend on the time series or sampling rate as shown in Fig. 1.

2 Data Acquisition

Data acquisition is the process of sampling signals that measure real world physical conditions and converting the resulting samples into digital numeric values that can be manipulated by a computer. Data acquisition systems, abbreviated by the initials DAS, DAQ, or DAU, typically convert analog waveform into digital values for processing. The components of data acquisition systems include:

- Sensors - to convert physical parameters to electrical signals.
- Signal conditioning circuitry - to convert sensor signals into a form that can be converted to digital values.
- Analog-to-digital converters - to convert conditioned sensor signals to digital values.

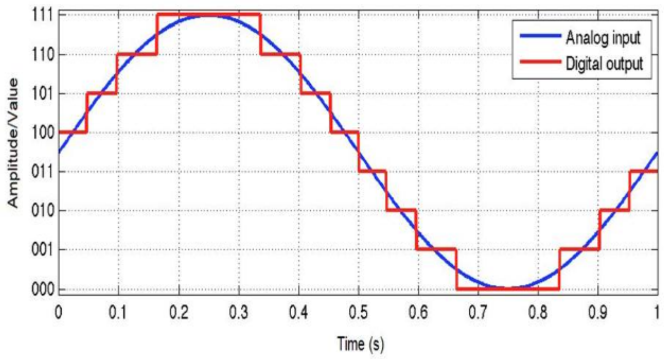

Figure 1: Digital Sampling from Analog Signal

3 Analog-to-Digital Conversion in TM4C123

Microcontrollers cant read values unless its digital data. This is because microcontrollers can only see 'levels' of the voltage, which depends on the resolution of the ADC and the system voltage.

ADCs follow a sequence when converting analog signals to digital. They first sample the signal, then quantify it to determine the resolution of the signal, and finally set binary values and send it to the system to read the digital signal. Two important aspects of the ADC are its **sampling rate** and **resolution**.

TM4C123 ADC has two identical successive Approximation Register (SAR) architecture based ADC modules such as ADC0 and ADC1 which share 12 analog input channels as shown in the Fig. 2.

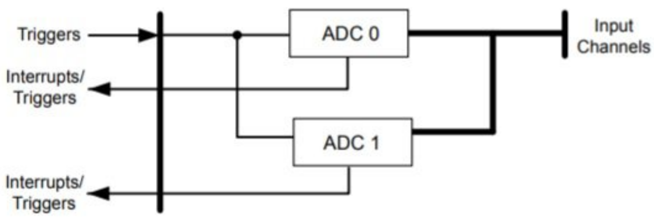

Figure 2: TM4C123 ADC Channel

Each ADC input channel supports **12-bit** conversion resolution. Furthermore, it has a built-in temperature sensor and a programmable sequencer to sample multiple input analog sources. The table in Fig. 3 shows the GPIO pins which are used for alternate functions with analog input channels.

Analog Chnnel	Pin Name	Pin Number
AN0	PE3	6
AN1	PE2	7
AN2	PE1	8
AN3	PE0	9
AN4	PD3	64
AN5	PD2	63
AN6	PD1	62
AN7	PD0	61
AN8	PE5	60
AN9	PE4	59
AN10	PB4	58
AN11	PB5	57

Figure 3: ADC Channels on TivaC Board

3.1 ADC Resolution

TM4C123 microcontroller has 12-bit resolution means the ADC converts the analog values between 0 to 3.3 volts (3.3v is the reference for TM4C123) into discrete digital values in the range of 0 to $2^{12} - 1$ or 0 to 4095. That means, if digital value is 0, the analog input to ADC channel is zero and if digital value is 4095, the analog input to ADC channel is 3.3 volts.

Based on the analog voltage, the value will increase in steps.

The $Voltage/step$ or the resolution can be calculated using the following formula.

$$Voltage/step = \frac{Reference\ Voltage}{2^{12}}$$

The reference voltage is usually 3.3 V. Hence, the smallest change in voltage that can be detected is

$$Voltage/step = \frac{3.3}{4096} = 0.8056\ mV/Step$$

Now, the Input Analog Voltage can be calculated as follows:

$$Input\ Voltage = \frac{ADC\ Value}{ADC\ Resolution} \times Reference\ Voltage$$

We will be using above formula to detect input voltage at ADC pin on our board, so make sure to understand it clearly.

3.2 ADC Sample Sequencers

TM4C123 microcontroller has two ADC modules that are ADC0 and ADC1. Each analog to digital converter has a separate sample sequencer (SS0, SS1, SS2 and SS3) which can sample different input channels. All these sequencers offer different numbers of samples that they can capture and the length of FIFO. TM4C123GH6PM microcontroller supports sampling rate from 125 KSPS to 1 MSPS.

Sequencer	Number of Samples	Depth of FIFO
SS3	1	1
SS2	4	4
SS1	4	4
SS0	8	8

Figure 4: ADC Sampling Sequencer

4 Light Dependent Resistor (LDR) Sensor for ADC Sampling

In this lab, we will use LDR (light dependent resistor) also known as photoresistor as sensor to detect light intensity in a room. It is a passive component that decreases resistance with respect to receiving luminosity (light) on the component's sensitive surface. In the dark, a photoresistor can have a resistance as high as several megaohms (MΩ), while in the light, a photoresistor can have a resistance as low as a few hundred ohms.
The change in resistance can not be measured directly by the microcontroller, thus it requires some signal conditioning circuitry to convert ldr (sensor) signal into a form that can be read by the microcontroller. Using sensor in VDR configuration is the simplest option to detect resistance change. This change in resistance is reflected in change in voltage across the sensor. And hence voltages can be read through ADC channel of Microcontroller.

5 TivaC LaunchPad with Energia

In Energia IDE, unlike Kiel uVision, we can use TivaC LaunchPad pins for various peripherals without the need to activate Ports using registers or specifying function of the pin. But it also comes with limitations of usability and programmable scope of the board. In Energia, we can refer to Pins of TivaC directly using numeric digit like 1,2,3.. and so on. Pin map for the EK-TM4C123GXL LaunchPad is given in Fig. 8 with Black Columns under J1, J2, J3 and J4.

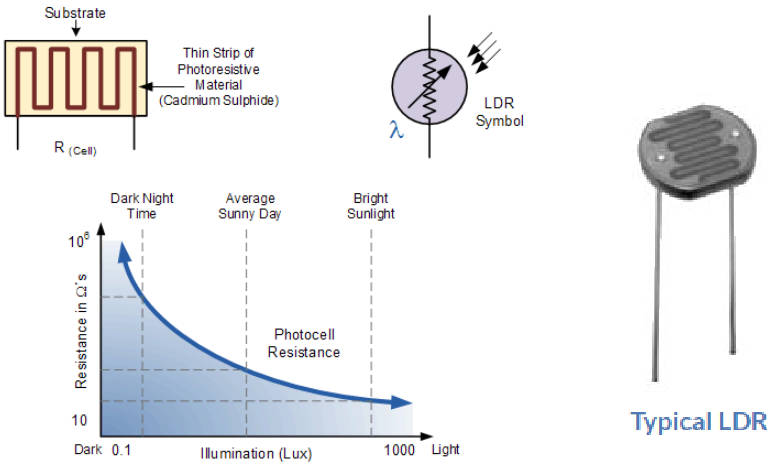

Figure 5: LDR Overview

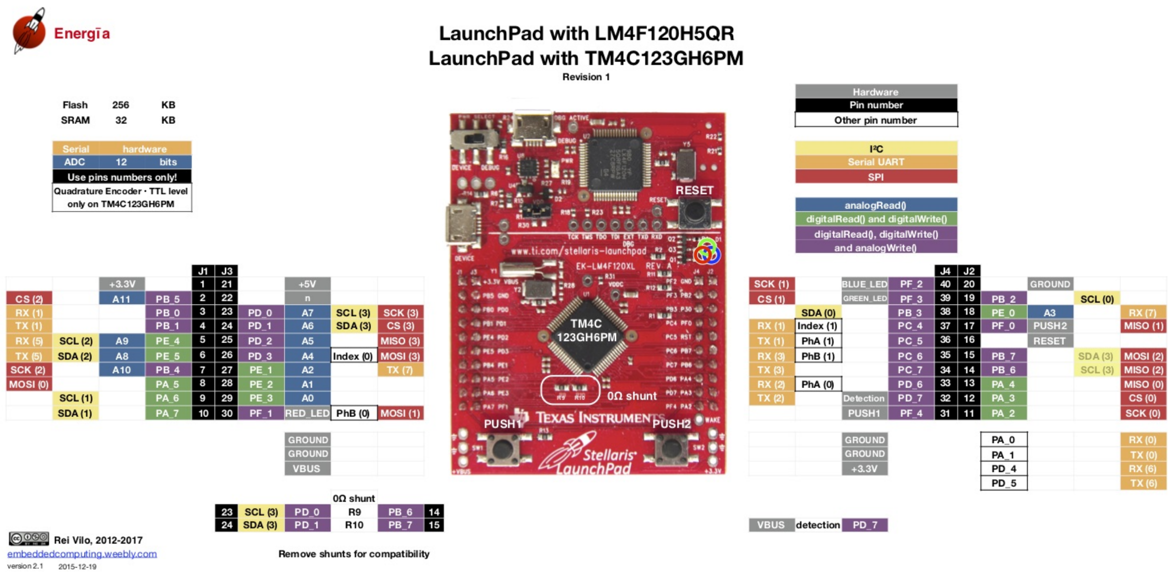

Figure 6: Pin Configuration of TivaC for Energia (Front)

In-Lab Tasks

Task 1: To design a data acquisition system for light intensity monitor using LDR in Keil

In this task, we will read the LDR value and will display it using a 7-Segment display. Use the task01.c file provided on LMS to make connection between TivaC and 7-Segment Display on PortB pins. Once done, make the connection of LDR as shown in Fig. 7, provide 3.3V to LDR with TivaC and use its PE3 pin for reading analogue values.

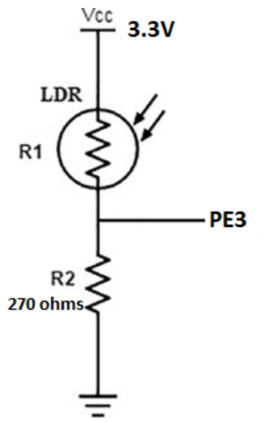

Figure 7: Circuit Connection for LDR with TivaC

Create a new Keil Project and set it up for the TivaC board as performed in previous labs, do not forget to check 'CORE' and 'Startup' in "Manage Run-Time Environment". Also select "Stellaris ICDI" in debug from "Options from Target".

Follow the steps next to setup/initialize ADC of tivaC on-board with LDR and display the results on 7-Segment in the code provided.

1. Add the Hex code (0-F) for the 7-Segment digits in digitPattern[ ] array.
2. Turn on the bus clock for GPIOB by setting the bit 1 of the RCGCGPIO register (Refer to page 406 of the datasheet).
3. Enable all of the GPIOB pins as digital output pins by setting all the bits of the GPIO DEN and GPIO DIR registers (refer to page 682 and page 663 of the datasheet for this).
4. Turn on the bus clock for ADC Module by setting the RCGCADC register. Set the appropriate bit to enable ADC0 (Refer to page 352 of the datasheet).
5. Turn on the bus clock for GPIOE by setting the bit 4 of the RCGCGPIO register (Refer to page 406 of the datasheet). We use GPIOE because AN0 takes analog signals from the PE3 pin of PORTE.

6. **Alternate function of PE3:** To enable alternate function of PE3 pin as an analog input, three registers are used:

    a **AFSEL** This register selects alternate functions of each GPIO pin. Because each pin is multiplexed with different peripherals. Set the Bit 3 (For PE3) of the AFSEL register (Refer to page 672 of the datasheet).

    b **DEN** This register enables or disables the digital functionality of the corresponding GPIO port. Set the Bit 3 (For PE3) of the DEN register (Refer to page 682 of the datasheet).

    c **AMSEL** This register is used to enable analog function of GPIO pins. We are using PE3 pin for AN0. Therefore, we must enable PE3 analog function by setting the 3rd bit of AMSEL register (Refer to page 687 of the datasheet).

7. **Sample Sequencer** It is part of ADC modules of TM4C123 microcontroller. It is used to get ADC samples and stores the conversion results in FIFO.

    a **ACTSS** Active sample sequencer register is used to enable or disable sample sequences SS0, SS1, SS2 and SS3. Disable the SS3 by setting the appropriate bit of ADCACTSS Register (Refer to page 821 of the datasheet).

    b **EMUX** ADCEMUX register selects the trigger option to start sampling of an input signal for each sample sequencer. Trigger event sources are processor, PWM, analog comparators, external interrupts, and software. The default trigger option is by software. Enable the relevant bit for SS3 Trigger select (Refer to page 833 of the datasheet).

    c **SSMUXn** (Refer to page 851 of the datasheet).

8. Now in the while loop, the first few commands read the ADC value between range of 0-4095 on PE3, you can refer to documentation of each in the datasheet for detailed understanding.

9. Under the heading 'Control Output', we have to map ADC value of 0-4095 to digits A-F on 7-Segment. Since we have total of 16 possible digits, we get 4096/16 = 256. Hence, divide your adc_ value by 256 to get corresponding digit between A to F and then use GPIOB->Data command to write the corresponding digit from digitPattern[ ] array.

Write your brief and complete understanding of how the registers value are set and how it has helped us achieve the desired results. Also write your observation of output displayed on 7-Segment and what it represent regarding LDR, and how was it done?

Provide your clear circuit image below

Provide your code here with appropriate comments below (Get the circuit demonstration checked with RA within Lab)

Task 2: To Measure Analog Voltage Signal using Potentiometer

In this task, we will learn how to use the analog to digital module (ADC) of TM4C123GH6PM Microcontroller using TM4C123G Tiva C Launchpad. Firstly, we will learn to configure ADC modules and sample sequencer of TM4C123 using configuration registers. For demonstration purposes, we will measure analog voltage by using one of the analog input pins of TM4C123GH6PM microcontroller

ADC Configuration and Initialization steps

we will discuss the different registers that are used to configure ADC input channels and sample sequencers of TM4C123 Tiva C Launchpad.

Followings are the steps to enable ADC module of TM4C123 Tiva C Launchpad:

1 - Enable Clock to ADC: First, we enable the clock to ADC module and to the GPIO pin which is used as an analog input. RCGCADC register is used to enable the clock to ADC0 and ADC1 modules. Bit0 of RCGCADC enables ADC0 and bit1 of RCGCADC register enables ADC1. Setting the corresponding bit enables and clearing disables the clock to the corresponding analog to digital converter.
For example, we want to use ADC0, this line will enable clock to ADC0:

2- RCGCGPIO register is used to enable clock to the related GPIO port pin which will be used to measure analog input. For example, we are using analog channel zero or AN0. As we mentioned earlier, AN0 takes analog signals from the PE3 pin of PORTE. Setting the 5th bit of RCGCGPIO register enables the clock to PORTE of TM4C123 microcontroller.

3 Configure PORT as an Analog Pin The next step is to configure the analog channel corresponding GPIO pin as an analog pin. To enable alternate function of PE3 pin as an analog input, three registers are used.

a **AFSEL:** This register selects alternate functions of each GPIO pin. Because each pin is multiplexed with different peripherals. This line enables the alternate function of PORTE pin 3 or PE3.

b **DEN :** This register enables or disables the digital functionality of the corresponding GPIO port. This line disables the digital functionality of PE3 pin.

c **AMSEL:** This register is used to enable analog function of GPIO pins. We are using PE3 pin for AN0.Therefore, we must enable PE3 analog function by setting the 3rd bit of AMSEL register.

3 - Sample Sequencer Configuration: The sample sequencer is part of ADC modules of TM4C123 microcontroller. It is used to get ADC samples and stores the conversion results in FIFO.
Activate ADC SS: ADCACTSS (active sample sequencer) register is used to enable or disable sample sequences SS0, SS1, SS2 and SS3. For example, we will be using SS3 in this tutorial. Setting ASEN3 bit of ADCACTSS register enables the SS3 and clearing it disables the SS3.

Before configuring ADC channel, we must disable the sample sequencer by clearing bit3 like this:

4 - Sampling Option ADCEMUX register selects the trigger option to start sampling of an input signal for each sample sequencer. Trigger event sources are processor, PWM, analog comparators, external interrupts, and software. The default trigger option is by software. This line selects a software event as a start conversion trigger.

5 - Analog Channel Selection TM4C123G microcontroller provides 12 analog channels. ADCSSMUXn registers (where n=0, 1,2,3) select analog channels for sample sequencers. For example, if we want to use SS3 and analog channel A0, this line will select AN0 analog channel for sample sequencer 3 or SS3.

6 - ADC Sampling Rate Setting ADCPC register is used to set the sampling rate of ADC. The Analog to digital converter module of the TM4C123G microcontroller supports a sampling rate from 125 KSPS to 1 MSPS. First four bits of the ADCPC register are used to set the sampling rate. This line sets the sampling rate to 250ksps.

After ADC initialization and configuration, set the SS3 bit of ADCPSSI register to start ADC conversion for sample sequencer 3.

If you are using ADC1 or other sample sequencers, select the corresponding bit of ADCPSSI register.

ADCRIS register provides raw interrupt signal for each sample sequencer on sample conversion completion. INR3 bit of ADCRIS register raw interrupt status of SS3. If you are using a polling method to get ADC value, you can keep polling this bit and read the data whenever the INR3 bit becomes one.

ADCSSFIFO0,ADCSSFIFO1, ADCSSFIFO2 and ADCSSFIFO3 registers contain the conversion result of ADC sample sequencers for SS0, SS1, SS2 and SS3 respectively. 12 least significant bits of these registers store conversion results.

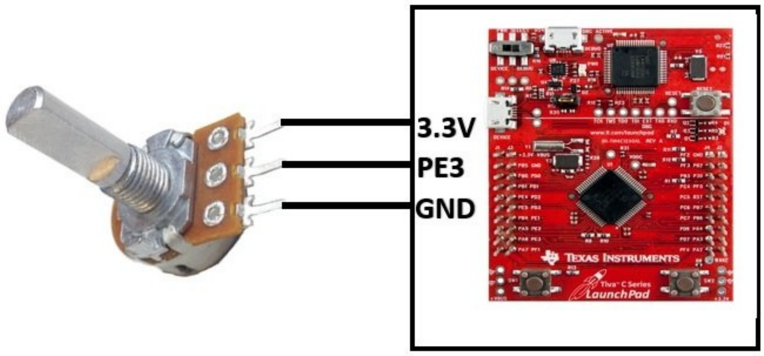

Figure 8: Pin Configuration of Potentiometer with TivaC

Provide your code here with appropriate comments below (Get the circuit demonstration checked with RA within Lab)

Attach Screen-shot of your readings displaying Voltages

Task 3: To design DAQ in Energia Environment

In this task, we will use the Energia to read analogue values from LDR from previously built circuit and will use that to control brightness of TivaC built-in RED LED. Use the file 'task02.ino' to perform following tasks:

1. Set the pin connections using Energia Pin Mapping in Fig. 8.
2. Setup Serial Communication using Serial.begin() at 115200 Baud Rate.
3. Setup pinModes for analogInPin and analogOutPin.
4. Read the ADC value at analogInPin and store it in variable ADCValue using analogRead().
5. Use the map() function to map ADC value from 0-4095 to 0-255 and store it in variable 'brightness'. Read about map() function here:
   https://www.arduino.cc/reference/en/language/functions/math/map/
6. Write the brightness to analogOutPin using analogWrite().
7. On serial monitor, display the ADC value and corresponding voltage level detected between 0-3.3V using the knowledge of section 3.1 "ADC Resolution" covered in theory above. Use Serial.print("") command Serial.println("") commands to write string on monitor.

Write your brief and complete understanding of how the code works. How does the map() function works? Write your observation about the output on LED.

Provide your code here with appropriate comments below (Get the circuit demonstration checked with RA within Lab)

Attach Screen-shot of Energia Serial Monitor Readings displaying ADC Value and Corresponding Voltage from LDR

6 Assessment Rubrics

Marks distribution

		LR1	LR2	LR4	LR5	LR9
	Task 1	10 points	10 points	10 points	10 points	10 points
	Task 2	-	10 points	05 points	10 points	
In-lab	Task 3	-	10 points	05 points	10 points	
Total Marks 100						

Marks obtained

		LR1	LR2	LR4	LR5	LR9
	Task 1					
	Task 2	-				
In-lab	Task 3	-				
Total Marks 100						