EMBEDDED SYSTEMS LABORATORY

LABORATORY MANUAL

ADC on MSP432

AY 22/23

OBJECTIVES

- Learn the difference between polling and interrupts
- Configure and implement hardware interrupts
- Configure the ADC on MSP432 for analog input on an IR-based sensor

EQUIPMENT

- Computer or laptop that supports Code Composer Studio (CCS) 9.3 IDE
- Texas Instrument MSP-EXP432P401R LaunchPad Development Kit
- Micro-USB Cable

NOTE:

- Only students wearing fully covered shoes are allowed in the lab due to safety.
- Bring your laptop with Code Composer Studio installed.
- For your understanding and better quiz preparation, note the given tasks, especially questions or unexpected code behaviour.

Introduction

Analog to Digital Converter (ADC) is an essential part of any embedded system. It helps to obtain data (i.e. temperature, humidity, etc) that can't be directly read on any microcontroller as they are in analog format. To read this type of analog data, we have to convert this analog data into a digital format via an ADC inside the microcontroller.

In this lab session, we shall learn how o use the ADC offered with the MSP432P401R. In the previous lab sessions, we have only handled digital data type on the MSP432P401R, e.g. GPIO, UART, etc.

However, in this lab session, we shall have a brief understanding how we can use ADC module available within MSP432P401R. Lastly, we will develop code that incorporates ADC for the **IR-based sensor** in measuring contrast on the surface.

Introduction to ADC

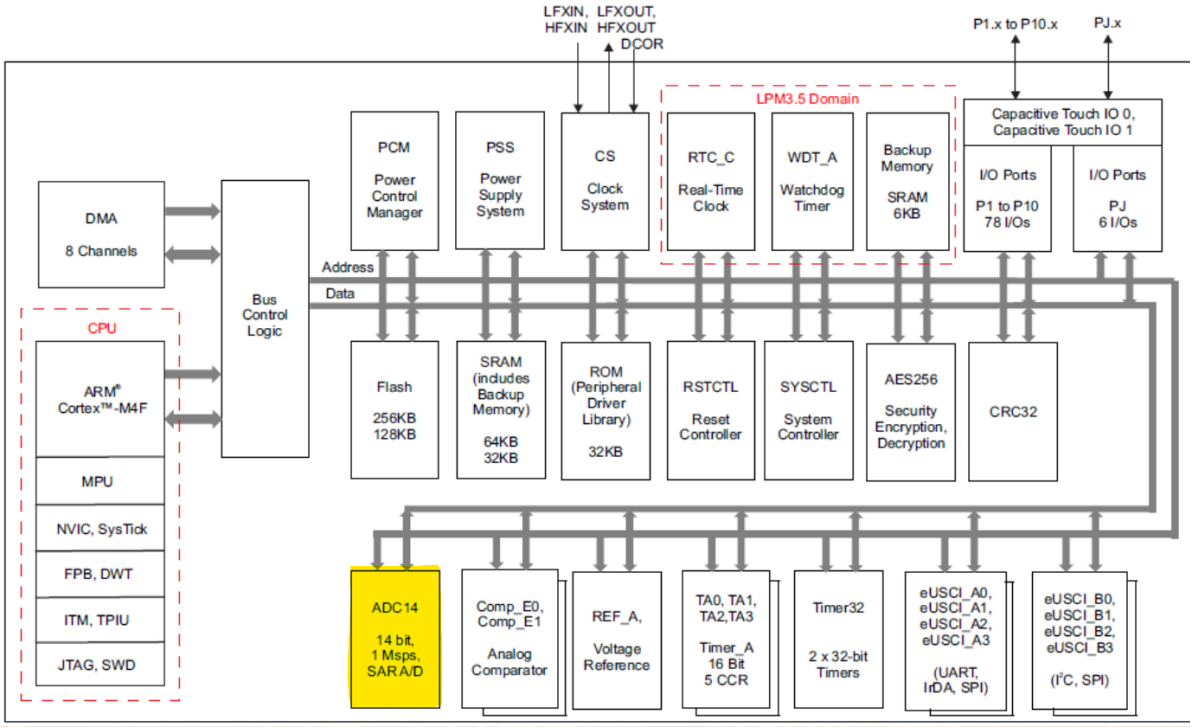

Figure 1: ADC (in yellow) for the MSP432P401R Functional Block Diagram.

Texas Instrument MSP432P401R microcontroller features a precision ADC module with a 14-bit successive-approximation-register (SAR) analog-to-digital conversions, and can support up to a 16-bit precision through software over-sampling. The module implements a 14-bit SAR core, sample select control, and up to 32 independent conversion-and-control buffers. The conversion-and-control buffer allows up to 32 independent analog-to-digital converter (ADC) samples to be converted and stored without any CPU intervention.

Sampling refers to the process of converting a continuous, analog signal to discrete digital numbers. Typically, an Analog to Digital Converter (ADC) would be used to convert voltages to a digital number corresponding to a certain voltage level. The process may be reversed through a Digital to Analog Converter (DAC).

The ADC can be configured by using two control registers, ADC14CTL0 and ADC14CTL1. It resets when ADC14ON = 0, but powers up when ADC14ON = 1. The ADC can be turned off when not in use to save power. For example, during conversion the ADC14ON bit is set to 0, the conversion is abruptly exited and the ADC is powered down. The ADC control bits can

mostly be modified only when ADC14ENC = 0 and the ADC14ENC must be set to 1 before any conversion can take place. The conversion results are always stored in binary unsigned format. The data format bit ADC14DF in ADC14CTL1 allows the user to read the conversion results as binary unsigned or signed binary (2s complement). For more details about Control Register ADC14CTL0 and Control Register ADC14CTL1, please download and look into the datasheet and user guide.

Steps to configure the MSP432P401R ADC module:

- Configure the reference for 3.3V
- Disable the ADC by clearing the ENC bit
- Wait for BUSY to be zero, in case there is a conversion in progress
- Set the ADC mode to S/H pulse mode, sysclk, 32 sample clocks, software trigger
- Set the conversion address, to 12-bit mode and turn on the reference
- Set the 3.3V reference and select Channel 0 (P5.5)
- Conversion for mem register
- Set the ENC bit to enable the ADC registers
- Start conversion
- Wait till conversion completes read the conversion result from the memory

Task 1: Configuring the IR Line sensor

The IR line sensor is a reflective sensor that includes an infrared emitter and phototransistor in a leaded package which blocks visible light. The infrared emission diode constantly emits infrared rays. When the intensity of the reflection is not received or is not large enough, the light-activated diode will be in the off state. However, if the sensor detects an object within the field of view, infrared light is reflected back and activates the diode.

Figure 2 shows how the IR line sensor can be connected to the MSP432 platform. P5.5 have been configured as an ADC input. The accompanying source code can be obtained from xSITe.

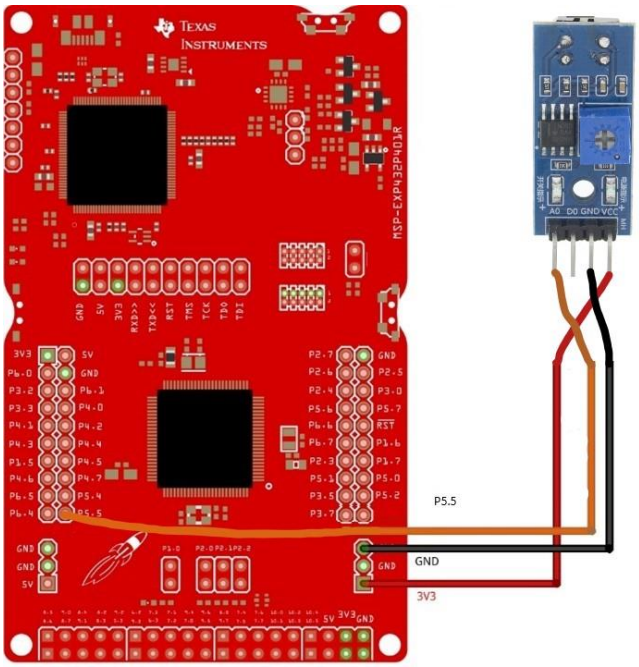

Figure 2: Connecting the IR sensor to MSP432.

The sensor will output a voltage between 0v to 3.3v to P5.5 that has been configured for ADC. The ADC will then convert the voltage into a digital format that can be used by your algorithm.

Submission

No submission. The marks will go to lab quiz 5 which will become 3% rather than 2%.