EXPERIMENT 3

SERIAL COMMUNICATION USING UART AND ADC INTERFACING

OBJECTIVE

To establish UART communication between a KL25 Freedom Board and a computer for transmitting digital and analog data. The experiment is divided into two parts:

1. Transmitting data through UART.
2. Reading analog data using the ADC and transmitting it through UART.

THEORY

The KL25 Freedom Board features GPIO (General Purpose Input/Output) pins and an Analog to digital converter (ADC) module, enabling digital and analog data acquisition. UART (Universal Asynchronous Receiver-Transmitter) is a commonly used serial communication protocol for transmitting data between devices. In this experiment, UART communication will be set up between the KL25 board and a computer via the OpenSDA port. The KL25Z MCU provides three UART interfaces (UART 0, 1, and 2). Here UART0 is connected to the open SDA interface which in turn is connected to the miniUSB port of the KL25Z board. Characters including numbers are passed as ASCII characters through the UART interface.

The ADC module is a 16-bit successive approximation type ADC. It can be configured to give 8-bit, 12-bit, 10-bit and 8-bit single ended conversion. In this experiment we will use 8-bit single ended conversion. The VREFH for the ADC is 3.3V and VREFL is 0V.
The ADC value in Volts is given as

$$ADC(V) = ADC\,(reg) * (VREFH / 256)$$

Where $ADC\ (reg)$ is the 8-bit ADC output value obtained from the ADC data register.

Connect a potentiometer to the ADC as shown in Fig. 1. It's one end is connected to 3.3V and the other end to GND. The output pin is connected to the ADC pin on KL25Z board.

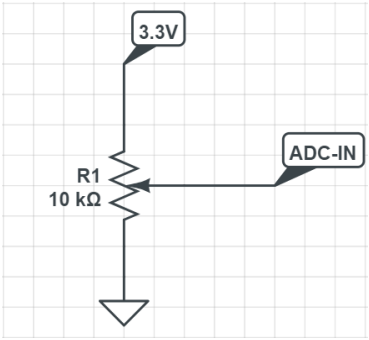

Fig. 1 Potentiometer to KL25Z circuit diagram

PROCEDURE

Part 1: Transmitting characters

1. Refer to the user manual to determine which pins are designated as serial port pins on the KL25Z Freedom Board.
2. Determine the baud rate for UART communication, considering how the baud rate configuration affects communication speed. Give the system clock (Baud clock) 20.971 MHz and an oversampling ratio (OSR) as per default settings.
3. Write a C program to configure the serial port for data communication, including setting up the transmitter and receiver. Transmit the text “Hello World” through UART.
4. Connect the KL25 Freedom Board to the computer via USB. In a serial terminal application (eg. Putty), assign the port name (e.g., COM1) and set the baud rate to 9600 to view the transmitted data.

Part 2: Transmitting Analog Data

1. Connect the potentiometer as shown in Fig. 1 to an ADC pin (eg. PTB0/ADC0_SE8)
2. Configure the ADC for sampling the analog input and converting it to digital format.
3. Once the analog data is converted, transmit it through UART using the configured UART communication settings.
4. Ensure proper connection between the KL25 Freedom Board and the computer via USB, with the appropriate port name and baud rate settings maintained in the device manager.
5. Vary the potentiometer setting and observe the voltage change from 0V to VREFH.