Cover Page

Lab 5 Report

Implementation of ADC Setting and Sensor Initialization

ECE4436, DMU

Spring 2024

Wankang Zhai

2232923

contents

1. Overview

In this experiment, we successfully mastered the skills of ADC initialization and finally realized the task of judging distance through the infrared sensor.

1) Implemented Analog signals to digital signals conversion

2) Realized the call of the infrared sensor and completed the distance judgment

3) Read the distance and calculate the distance by activating the interrupt. Description of Tasks and Results

2.Detailed Description and Results

Task1: Initialize ADC and GPIOE

2.1.1 Enable two ports we are going to use

```
void ADC_init(){
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOE);
    SysCtlPeripheralEnable(SYSCTL_PERIPH_ADC0);
    GPIOPinTypeADC(ADC0_BASE, GPIO_PIN_3);
```

Figure 1

In Figure 1, we enable two system ports, namely GPIOE and ADC. GPIOE reads distance information by installing an infrared sensor. The full name of ADC is Analog to Digital Converter. By converting Analog signals into Digital signals, TivaC can successfully read and process them. In this, we use ADC0 module to convert. Here we use GPIO_PIN_3 to activate the ADC function.

2.1.2 Set ADC Sequence

```
ADCSequenceConfigure(ADC0_BASE,1,ADC_TRIGGER_PROCESSOR,0);
ADCSequenceStepConfigure(ADC0_BASE,1,0,ADC_CTL_CH0);
ADCSequenceStepConfigure(ADC0_BASE,1,1,ADC_CTL_CH0);

ADCSequenceStepConfigure(ADC0_BASE,1,2,ADC_CTL_CH0);

ADCSequenceStepConfigure(ADC0_BASE,1,3,ADC_CTL_IE|ADC_CTL_END|ADC_CTL_CH0);
```

Figure 2

- I first set up Sequence Number and Sequence Source in FIGURE2
- Here we need to set AdCSEQUENCE first. We activate adc0_base and adc_trigger through configure, so that we can sleep with ADC PROCESSORIGGER instructions to stimulate interruptions. We use Adcsequestepconfigure instructions to set each STEP. After reading four times, the statement is over. Through the ADCsequEnceStePConfigure function, we saved Figure 3 in the global variables defined before：

```
uint32_t zhai[4];
uint32_t sum = 0;
float adcVal;
float distance;
```

Figure 3

By defining global variables, I implemented the storage of distance information in the zhai int32 type array. At this point, my initialization program is completely completed.

Task2 : Set Main fuction while （1）

2.2.1 find the value and calculate the data.

```
while(1)
{
    int i=0;


        adcVal = zhai[1];


        distance = -1.858203*pow(10,-9)*pow(adcVal, 3) + 1.3214630459*pow(10, -5)*pow(adcVal, 2) - 0.03356491643966*adcVal + 35.536853696418802;
    SysCtlDelay(SysCtlClockGet() / (1000 * 3));

    ADCProcessorTrigger(ADC0_BASE, 1);
    if(distance >= 10.0)
        {
            GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_3|GPIO_PIN_2, GPIO_PIN_3|GPIO_PIN_1);
        }

    if((distance >= 6.0)&& (distance<10.0))
            {
                GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_3|GPIO_PIN_2, GPIO_PIN_3);

            }
    if(distance < 6.0)
            {
                GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_3|GPIO_PIN_2, GPIO_PIN_1);


            }


}
```

Figure4

In wihile (1), we first successfully calculated the distance to the object through the formula, and then we used the ADC ProcessorTrigger function to enter the interrupt. In the interrupt, we executed the function in Figure 5.

```
77
78
79  void
80  tainanle(void)
81  {
82      ADCIntClear(ADC0_BASE,1);
83      ADCSequenceDataGet(ADC0_BASE,1,zhai);
84
85  }
```

Figure 5

We cleared the previously saved data in the ADC, obtained the new data through DataGet, and stored it in the character array of zhai defined previously. Through this method, we can directly use the data stored in the zhai array. Subsequent We then process zhai. We use three judgment statements to determine when exactly what program needs to be called and what LED is lit. Finally, we successfully completed the lighting. And realized the task of lighting up different lights at different distances.

3 Conclusion

In this experiment, we learned all the steps from Analog signal transfer to Digital Signal by initializing the ADC. More importantly, we can call any sensor we want to call and let these sensors serve us. By processing its values, we achieve the purpose we need. In addition, we have strengthened our understanding of interrupts and completed the interrupts in ADC settings by exporting interrupts in setup. Based on the above new knowledge, we completed the task of this lab. We realized the adjustment of the light on and off through different distances. First of all, distance sensing paved the way for our future robot tasks.