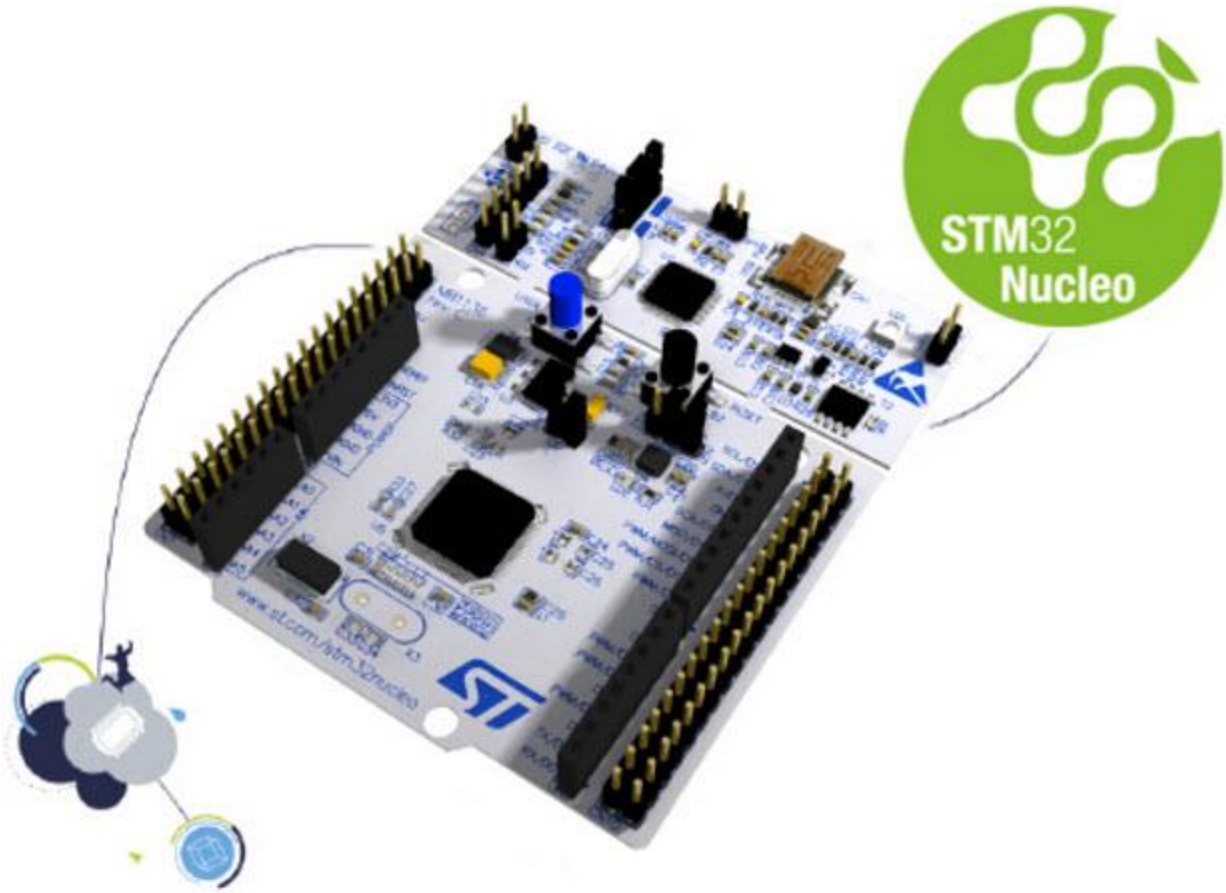
Technical report

Assignment 5:PWM input

By Veselin Atanasov



Teachers: Rene Bakx, Robert Verhijen

Date:12-11-2024

Contents

Introduction

This assignment features setting up timers to PWM input and reading the values of the HC-SR04 ultrasonic sensor and the Parallax 360 high speed feedback servo.

Description

Setting up the timer to read the ultrasonic sensor:

The timer I have chosen to set up to read the value is TIM2. For that reason, PA0 the pin connected to the echo pin on the sensor, is configured as AF1, which means that it is connected to TIM2 CH1. First the system clock is configured in the SysClock_Config() function to 72Mhz. After those initial steps are complete, timer settings have to be set. The very first step of configuring the timer is to enable its clock source. The source clock of TIM2 is APB1. Then the prescaler is set to 71 so a frequency of 1Mhz is achieved. The frequency of 1Mhz makes calculations easier, because a tick happens every microsecond. After this, CC1 is set to capture when a rising edge event happens. To measure the duty cycle, we also need to detect when a falling edge event occurs. For that reason CC2 is set up as input and is configured to store when a falling edge event occurs. The input signal may have noise, so for accurate readings it is crucial to filter the input signal and to synchronize it with the clock counter. To filter the input signal the trigger selection bits(TS) in the SMCR register have to be set to filtered timer input 1 (101 in binary). There is also a need to reset the CC1 and CC2 values as well as the timer counter after each pulse, so the SMS bits in the SMCR register are configured to Reset mode(100 in binary). After all the desired settings are done, the only thing that has to be done to finalize the initialization of the timer is to enable the counter.

In order for the ultrasonic sensor to read distance, it also needs a trigger signal which sends a ultrasonic pulse. To then calculate the distance the time for the pulse to come back to the sensor is measured. Usually for good measurements a pulse needs to be sent with a duty cycle of 10 microseconds, but since for this application there is no need to measure distance so often, a pulse will be sent every 20 milliseconds using the millis function on pin PA4. After the signal is sent, the sensor will send a burst of 8 signals to the STM. Depending on the width of the signals, the distance value can be calculated. To calculate the value in cm the CCR2 value(the register where the duty cycle is measured) has to be divided by 58, if inches is the desired measurement the CCR2 value has to be divided by 148. After every measurement the value is sent to the user via the serial monitor.

Reading the servo position:

The timer configuration is exactly the same as TIM2, the difference here is that TIM3 is used. The feedback connector of the servo is connected to PB4 on the STM32. The principle is pretty much the same. The value is still stored in the CCR2 register, the only difference is the transition from raw data to the actual value. To calculate the angle position of the servo, I used the formula provided by the Parallax 360 feedback servo datasheet.

Duty Cycle = 100% x (tHigh / tCycle).

Duty Cycle Min = 2.9%.

Duty Cycle Max= 91.7%.

$$\text{Angular position in units full circle} = \frac{(\text{Duty Cycle - Duty Cycle Min}) \times \text{units full circle}}{\text{Duty Cycle Max - Duty Cycle Min} + 1}$$

Conclusion

Setting up the timers for PWM input is a long and complicated procedure, without using external resources beyond the datasheets it is really difficult to achieve correct settings. The transition of raw data to actual value is always described in the datasheets of the sensors/actuators.

References

STM 32 Datasheet

Input presentation

HC-SR04 datasheet

Parallax 360 feedback servo datasheet