Lab 03

Developed by: Muhammad Imran and Peng Cheng

Date : 2015-03-02

Course : Programming Embedded Systems (ET014G) Mid Sweden University, Sweden.

This is a real-time application and you will be able to understand in details the working of simple real time embedded systems. In this lab, you will develop a PWM function generator with analog control and verify the functionality by using oscilloscope. The design need to produce PWM without any glitch while applying the updated settings

PWM is one of a techniques which provides digital-to-analog conversion. In this technique, duty cycle of a square wave output from microcontroller is varied to offer a changing DC output. The potential applications include controlling speed of DC motors, dimming of LEDs, encode digital transmission [1][2].

This lab requires the initialization of the microprocessor into faster clock speed (66 MHz). At least two interrupts will be required. (1) one for push button (higher priority), (2) one for periodical ADC data reading by TC interrupt.

The datasheet of AT32UC3A (page 699) shows Successive Approximation Register (SAR) 10-bit Analog-to-Digital Converter (ADC) has been used in AVR32 microcontroller. For ADC, the integrated multiplexer offers up to eight independent analog inputs channels. The PINs PA21-PA28 can be connected to ADC-AD[0] to ADC-AD[7] channels which are multiplexed to SAR ADC. Each channel can be individually enabled and disable. The schematic of EVK1100 shows that temperature, photoresistor and potentiometer sensors are connected to channels PA21, PA22 and PA 23 respectively.

Preparatory

Difference between EIC and INTC?

_____

_____

Number of channels in PWM?

_____

_____

Which clock source is required to be running when GPIO interrupts are handled? _____

tc_example3.c shows waveform generation, identify the structure that accepts parameters for the waveform generation._____

Why is the variation in duty cycling important and where can you use it?

_____

_____

_____

Task

Develop a PWM function generator with analog control and functionality to display frequency and duty cycle. The components required are potentiometer, ADC, TC, PWM, LCD display and interrupts.

One push button should be used to switch between the setting of the frequency and that of the duty cycle of the PWM output.

You should use ADC with TC interrupt to periodically log potentiometer reading and use this reading to set either the frequency or the duty cycle of the PWM output.

LCD display is used to show the latest setting of the frequency and the duty cycle.
Frequency range: from 100Hz to 100kHz, which are corresponding to 1 to 1000 from the 10bit ADC reading.

Duty cycle: from 0 to 100 %, which are corresponding to 0 to 1000 from the 10bit ADC reading.

ADC sampling rate: 50Hz, this means that all of your code should be able to execute once within 20ms.

Example app:
All the example apps in the following driver folders.
DRIVERS/PWM
DRIVERS/ADC
DRIVERS/INTC
DRIVERS/PM
DRIVERS/TC

Note:
Check the issues with PWM and ADC modules in errata of UC3A0512 datasheet.
Try to make the ISR codes shorter and faster to execute by doing as much as possible in main function instead.

Reference:

[1]. Dawoud Shenouda Dawoud and R. Peplow, "Digital System Design- Use of Microcontroller", University of Kwa-Zulu Natal, South Africa, April 2010.

[2]. Richard H. Barnett, Richard Barnett, Sarah Cox, Larry O'Cull, "Embedded C Programming and the Atmel AVR, 2nd Edition", Delmar, Cengage Learning 2007.