MoToScope Mainter Document

Author: Lauren Kosub
Date: December 17th, 2019

Abstract: *The MoToScope combines the functionality of difference hardware components on the STM32L4x6 microcontroller in order to read, store, and perform operations on different types of wave data just as an oscilloscope would. The purpose of this manual is to help embedded systems engineers get a better understanding of the design of the device, to document design choices, and to aid others who may wish to work on or modify this device.*

<u>Table of Contents</u>

System Overview

The MoToScope software provides a front end for interacting with a backend digital storage oscilloscope (DSO). The DSO is made up of different pieces of hardware that all work together in order to capture information about waveforms.

System Components

The MoToScope utilizes four hardware components in order to perform data collection, conversion, and storage: A general purpose timer, an analog to digital converter, a direct memory access unit, and a PA1. Each individual component in set up to work together in order to process waveform data.

TIM2

TIM2 is a general purpose timer that consists of a 16-bit of 32-bit auto reload counter driven by a programmable prescaler. It also has interrupt/DMA generation on different trigger events (counter overflow for example), which is helpful for scheduling data capturing. For the purpose of this project, the TIM2 is primarily used as a trigger for ADC1 to perform its operations. TIM2 is configured in the TRIGGER_TIM2 function in MoT_system/MoT_DMA_Comm.S.The TIM2 counter goes from 0 to sample rate + trigger delay. TIM2 run continuously in repetitive waveform sampling mode in order to ensure data collection is continuous. TIM2 only runs once when triggered with scan through mode’s step option and waits to be run again the next time the scan through’s step option is called. TIM2 is started by scan through’s start mode and can be stopped with scan through’s stop mode. TIM2 is the default trigger. TIM2 registers and bit meanings in each register can be found in the RM0351 reference manual.

PA1

PA1 is configured as analog input and used in this context as another option for a trigger. While the TIM2 is an internal option for a trigger, PA1 offers an external trigger option for the user. PA1 is configured in the TRIGGER_PA1 function in MoT_system/MoT_DMA_Comm.S. With PA1 manual triggering, the sampling is based on when the user enables PA1. PA1 allows the user to access triggering from the board directly, external from the software and hardware configuration. More information about configuring PA1 can be found in the RM0351 reference manual.

ADC1

The STM32 microcontroller has three analog to digital converters (ADCs), and the MotoScope uses ADC1 (master) in order to convert an analog input to a processable digital output every time the trigger is enabled. If editing ADC configurations, ADC must first be disabled (ADEN must be equal to 0). ADC registers and bit meanings in each register can be found in the RM0351 reference manual.

DMA1

The direct memory access (DMA) controller is a busmaster and system peripheral used to perform programmable data transfers between memory mapped peripherals and/or memories upon the control of an off-loaded CPU. The DSO uses DMA1's channel one for access to on-chip memory-mapped devices, namely flash memory, SRAM, and peripherals. All DMA channels are independently configurable, meaning each channel is associated with a DMA request. The DMA is configured to be in circular buffer mode, which supports continuous flow of data. DMA registers and bit meanings in each register can be found in the RM0351 reference manual.

LCD Display

In an attempt to make the MoToScope unique, the LCD display is configured such that it is turned on whenever the MoToScope is run and the display write the word “SCOPE” for the user to see and ensure they are running the correct code. LCD registers and bit meanings in each register can be found in the RM0351 reference manual.

RED LED

The Red LED is enabled in main whenever the waveform surpases expected bounds defined in software.

Putting it all together

The general purpose timer TIM2 is used to generate an ADC triggering event. The output (from timer) is on signal TIMx_TRGO, TIMx_TRGO2 or TIMx_CCx event and the input (to ADC) is on signal EXT[15:0], JEXT[15:0]. The ADC is set to receive the TIM2 interrupts and start a conversion whenever the TIM2 interrupt fires. When the conversion is complete, the ADC raises an interrupt. The DMA is connected to trigger a memory copy when the ADC has an interrupt. The LCD display operates independent of the other configurations.

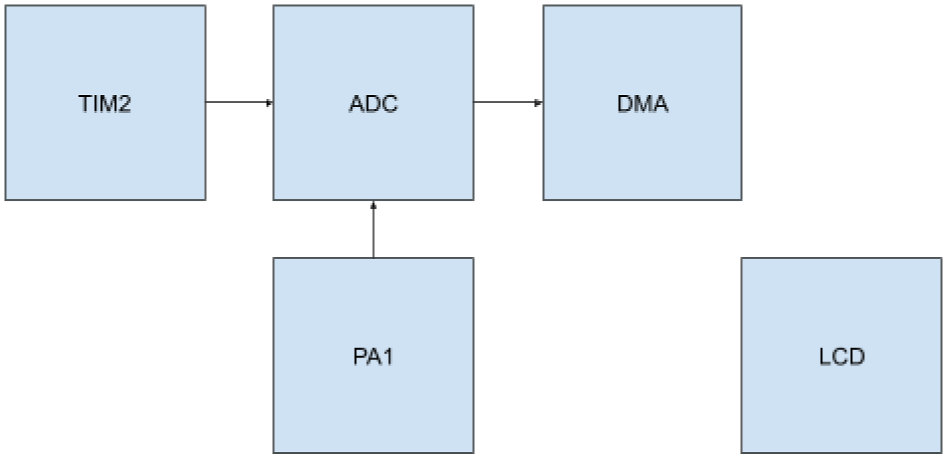

Figure 1: ADC listens for an interrupt either from TIM2 or PA1 depending on trigger configuration. DMA listens for an interrupt from ADC.

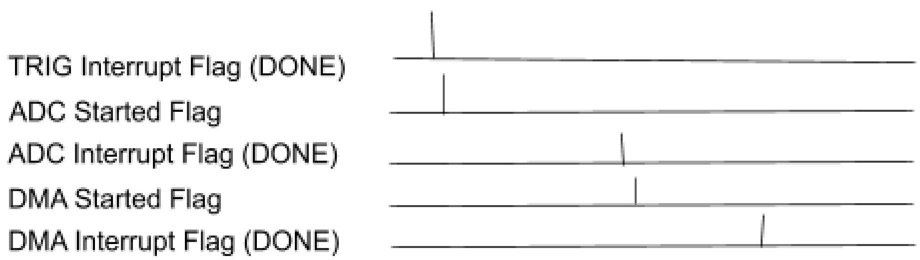

Figure 2: Order of Operations for Hardware Components

Initial Configuration

Instead of passing parameters and configuring the MoToscope with new values right off the bat, I decided to have a default configuration for users. This configuration is run during the MotoScope initialization command and sets the following information:

1. TIM2 is set as the trigger with an initial delay of 0 seconds and a count of X seconds. This count is what represents the sample rate. Once TIM2 is done counting, it triggers an interrupt.
2. TIM2 is enabled in NVIC and the TIM2 priority is set to a mid-level interrupt priority
3. the ADC is set up to listen to TIM2 and wait for the interrupt. Once the ADC sees TIM2's interrupt, then the ADC begins converting analog to digital
4. The DMA is reset, disabled, and cleared. This is done to ensure we are at a blank slate in order to do more configurations. The peripheral register address is set to point at the ADC and the memory address is set to point at a global variable in main called adc_data. adc_data is an array of uint8_t that stores the digital waveform data. This array is initially configured to be of size 2048 and can be modified by changing the sample size variable. adc_data is a circular buffer, so the DMA is configured to be in circular mode.

Oscilloscope Modes

The MoToScope has three different modes: scan through mode, repetitive waveform sampling mode, or single-shot mode. Scan through mode allows the user to start and stop the sampling of the oscilloscope at will as well as step through the values of the waveform in chunks of size sample size at 00 their own pace. This is configured to work with the PA1 as trigger so that the user can more accurately trigger. Repetitive waveform sampling mode is the default mode and it simply grabs waveform data continuously and reports that data to main

Scan Through Mode

Scan through mode works by enabling the trigger (TIM2) and disabling the trigger after it has counted up to its ARR value and caused an interrupt for ADC/DMA.

Repetitive Waveform Monitoring Mode

Repetitive Waveform Monitoring Mode works by having the trigger run continuously. The trigger starts the ADC each time it triggers. Once the ADC is done, it triggers the DMA to write the data to a global variable in main. This process will continuously occur such that the trigger is being trigger constantly/consistently and the data is being updated constantly.

Single-Shot Waveform Monitoring Mode

Single-shot waveform monitoring mode stops the trigger, starts the trigger, waits until the trigger's interrupt flag is on, and then stops the trigger. The trigger's interrupt is only completed once such that the trigger only enables the ADC/DMA once and data is only written back to the program once.