**LAB: Analog Input using ADC Part 2**

* **Line Tracking with IR Reflective Sensors-**

**I. Overview**

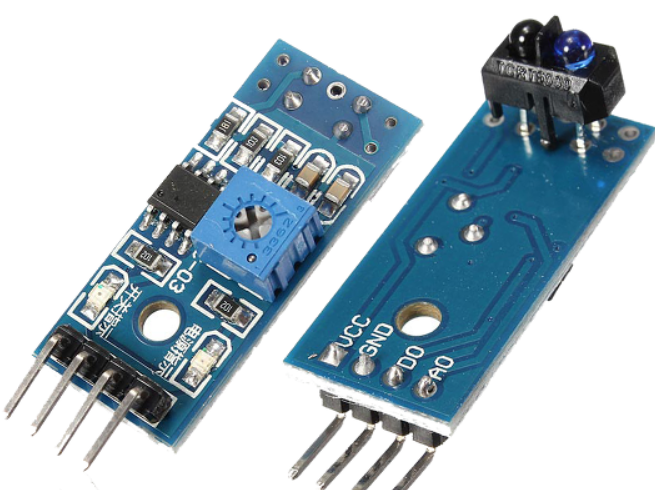
In this lab, we will learn how to configure and use ‘Analog mode’ of GPIO input of the MCU. Then, we will learn how to triggering ADC by using a timer, to control the sampling period of ADC measurements. From acquiring measurements from IR sensors, you are required to create a simple logic to follow a white line on the black track to control your mobile platform.

The objectives of this lab are learning how to

* Read and configure registers of ADC
* Create your own functions for configuring and reading ADC
* Use external timers to trigger ADC

**Preparation**:

* TCRT 5000 IR Reflective sensors. Read more about the sensor especially input and output voltage.
* You need to study the following registers: ADC register in ‘STM Reference Manual pg. 212 - 239’’



**Figure 1. IR Reflective Sensor**

**II. Pre-Lab**

**A. ADC Register**

* List of ADC registers for this and previous LAB (ADC Part I and ADC Part II)

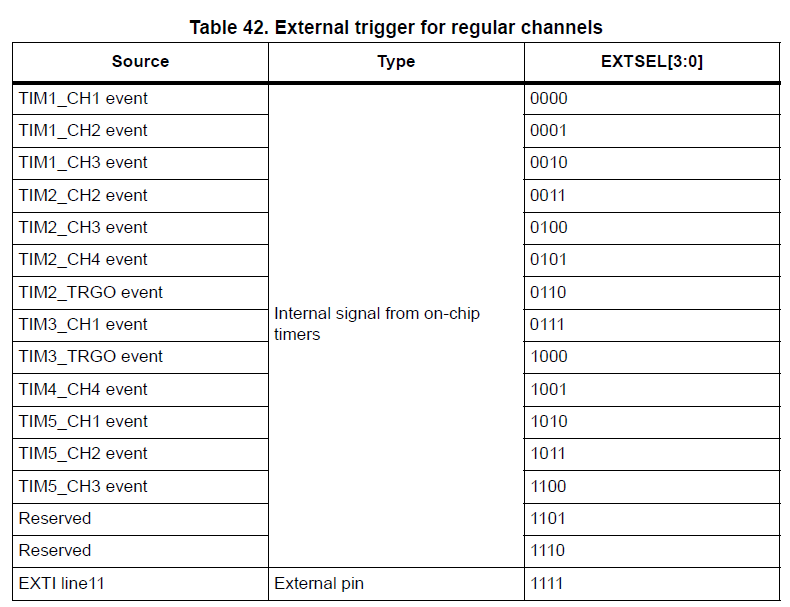
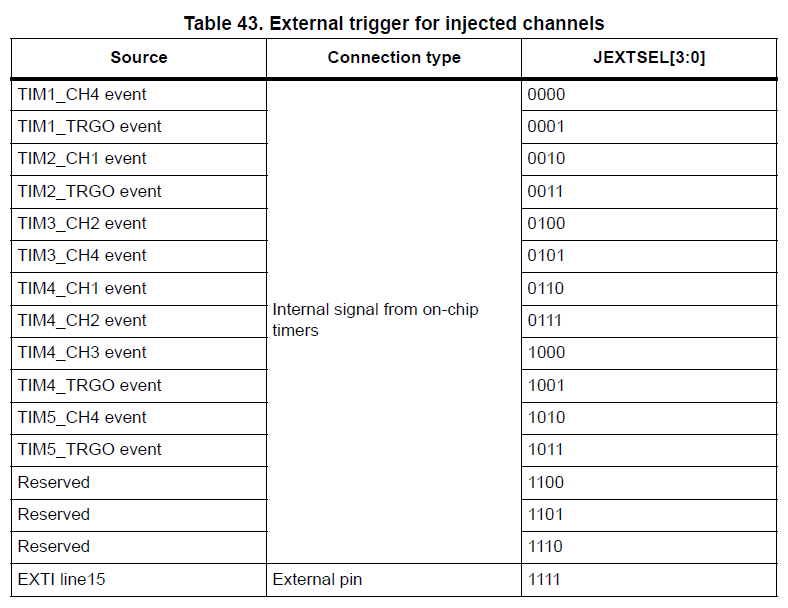
|  |  |  |
| --- | --- | --- |
| Type | Register Name | Description |
| RCC | RCC\_AHB1ENR | Enable the clock of GPIO Port |
| RCC\_APB2ENR | Enable the clock of ADC |
| GPIO | GPIOx\_MODER | Analog |
| GPIOx\_PUPDR | No Pull-Up, No Pull-Down: |
| ADCx | ADCx\_ CCR | Select the prescaler of ADC clock frequency |
| ADCx\_CRx | Start one ADC conversion with resolution setting |
| Choose the data alignment (right/left). |
| ADCx\_SQRx | Configure the sequence of conversation |
| ADCx\_SMPRx | Decide the sampling time of conversation |
| ADCx\_SR | Read the EOC (end of conversion) bit |
| ADCx\_DR | Read the converted data from ADC pin |
|  |  |  |

**B ADC Register for External Trigger**

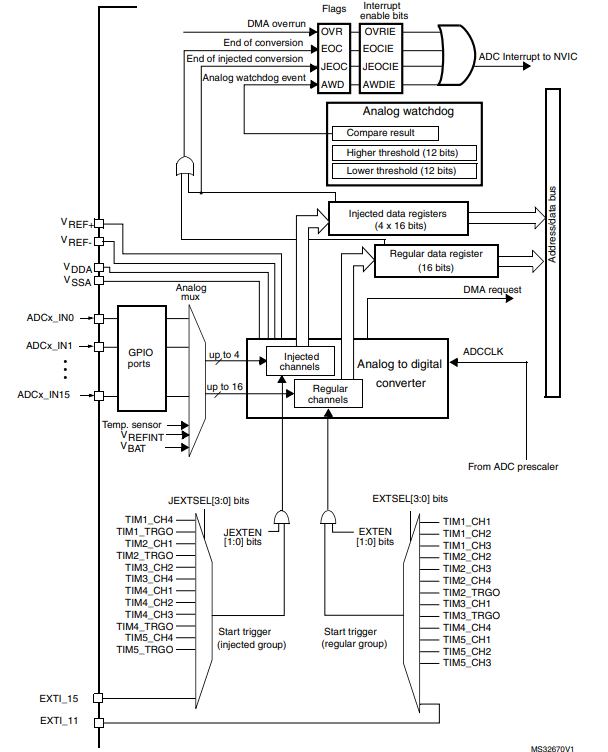
* Registers for ADC Trigger Output(TRGO)

|  |  |  |
| --- | --- | --- |
| Type | Register Name | Description |
| TIMx | TIMx\_CR2->MMS[2:0] | Selects Triger Output in timer |
|  |  |
| ADCx | ADC\_CR2->EXTSEL[3:0] | select which out of 16 possible  events can trigger conversion for the regular and injected groups |
| ADC\_CR2->JEXTSEL[3:0] |
| ADC\_CR2->EXTEN [1:0] | Trigger detection disable  Detection on rising/falling/rising and falling edge |
| ADC\_CR2->JEXTEN [1:0] |

* External Trigger
  + ADC conversion can be triggered by (1) Software (2) External
  + Software Trigger: ADC\_CR->ADSTART
  + Hardware signals such as TIMx can be used to trigger ADC conversions
  + By choosing a timer trigger, we can configure the sampling time of ADC conversion
  + Trigger Source: EXTSEL is used to select which trigger event to use
  + Trigger Polarity: EXTEN is used to select the edge of trigger
    - If EXTEN[1:0]=00, select software trigger. Otherwise choose hardware trigger

* ADC block diagram with External Trigger



* Process of configuring External Trigger (e.g. TIM4)

|  |
| --- |
| **GPIO Pin setting**  **ADC setting**  // same as previous lab   1. Enable ADC peripheral clock (**RCC\_APB2ENR)** 2. Configure clock prescaler. **(ADC\_ CCR)** 3. Configure ADC data resolution and alignment. **(ADC1\_CR1, CR2)** 4. Enable continuous conversion mode **(ADC1\_CR2)** 5. Configure the number of conversions. **(ADC1\_SQR1)** 6. Configure the channel and sequence of conversion. **(ADC1\_SQRn)** 7. Configure the sampling time with choosing the number of cycles. **(ADC1\_SMPRn)**   // External Trigger   1. Select Trigger Source **(ADC1\_CR->EXTSEL)** 2. Select Trigger Polarity **(ADC1\_CR->EXTEN)** 3. Enable ADC and start conversion. **(ADC1\_CR2)**   **TIMx setting**   1. Enable Timer Clock 2. Counting Direction Setting 3. Master Mode Selection Bits MMS[2:0]: choose Trigger output (TRGO) 4. Output Compare Mode 5. Set PSC, ARR 6. Enable Counter |

**B. Register Setting**

**1. GPIO Pin Initialization**

* ADC pin: **Port A pin 0** / Analog / No pull-up & No pull-down.
* Read appendix for ADC pin map

**2. ADC Initialization for External Trigger**

**Common Setting**

* ADC Intialization: Prescaler /2, 12-bit resolution, right alignment, continuous conversation mode, one channel scan in regular group, channel sample time selection 84-cycles
* Use your own functions created in previous lab for GPIO configuration. See (LAB: ADC Part I)

**External Trigger Source Selection**

* **ADC1\_CR2:** External event by Timer3 TRGO event

|  |
| --- |
| *Register map goes here* |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Mask** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* **ADC1\_CR2:** Trigger Detection on Rising Edge

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** |  |  |  |  | **11** | **10** | **9** | **8** |  |  |  |  |  |  | **1** | **0** |
| **Mask** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

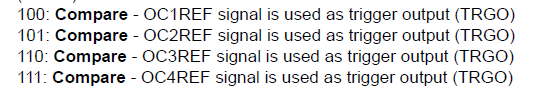
**3. TIMx Initialization for External Trigger for ADC**

**Common Setting**

* TIM3 Intialization: Output compare mode, up-counter, PWM mode\_1, 1 KHz,
* Use your own functions created in previous lab for Timer configuration. See (LAB: Timer/PWM)

**Timer Trigger Setting (TIMx\_CR2)**

* Master Mode Selection ( **MMS[1:0]** ): Configure master mode of timer for synchronization (TRGO)
* For selecting Compare(OC1REF to OC4REF) for ADC trigger, choose MMS[2:0]=100~111



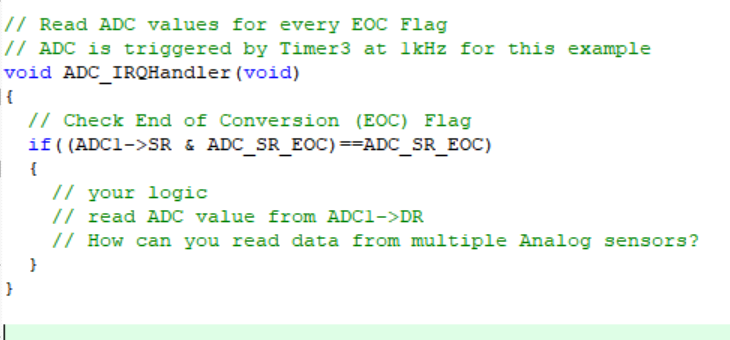
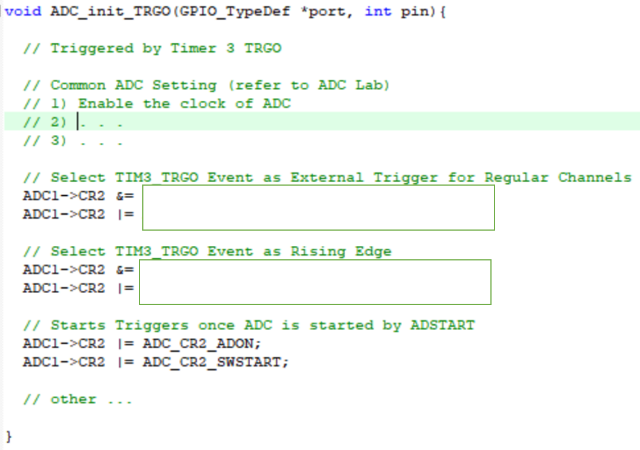
* **TIMx\_CR2:** Timer master mode- Compare OC1REF for TRGO

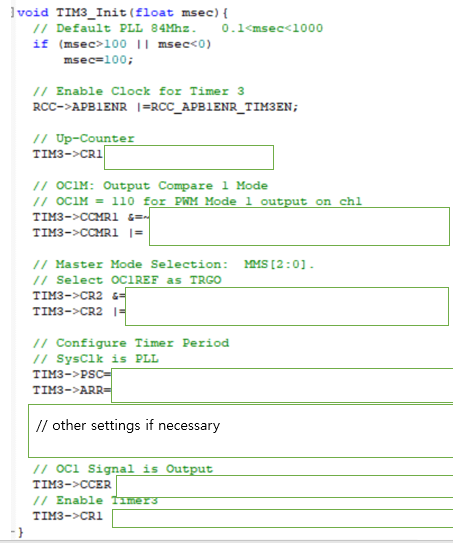
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** |  |  |  |  | **11** | **10** | **9** | **8** |  |  |  |  |  |  | **1** | **0** |
| **Mask** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**IV. Exercise**

1. **Modify your own functions of ADC and Timer initialization for ADC External trigger mode**

* Modify TIM3 initialization to select TRGO as OC1REF. Set it as 1KHz by default, or you can give the timer period.
* Modify ADC initialization to be triggered by Timer 3. Change the sampling time of ADC by changing the Timer 3 period.
* You have used polling method to read ADC in the previous lab. Modify the program to read the ADC data by using interrupt.
* Make sure that any interrupt handler should be as concise as possible.
* Example code:

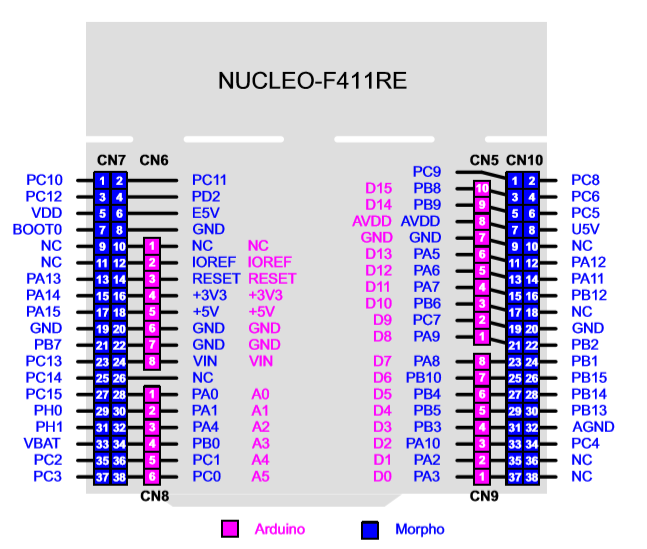


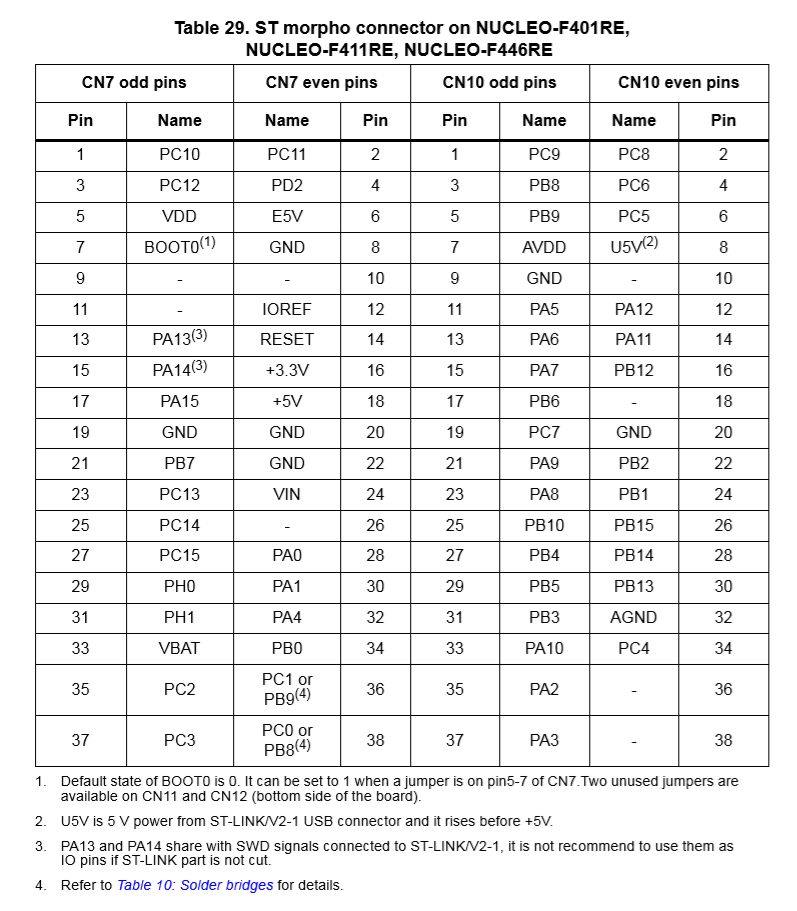
1. **Create a simple program to control RC Car to trace a line using the IR sensors**

* You can choose any ADC pins and choose an appropriate ADC clock frequency
* Set the ADC sampling rate to be below 1KHz, to decrease burden to your CPU.
* Use the analog pin from the IR sensor modules and determine threshold value to differentiate dark and white surface of target
* You can use either regular or injected groups
* Create a logic to trace a white line on dark background surface for your RC car. You will use it in your final project.

**Appendix**

1. **Pin Configuration of NUCLE-F411RE**

****

****

1. **ADC GPIO pinout for STM32f411**

