LAB: Stepper Motor

Date: 2023-11-10

Author/Partner: NohYunKi / KimEunChan

Demo Video: https://youtube.com/shorts/BtoboAa4JtI

PDF version:

Introduction

In this lab, we operate stepper motor with FSM(Finite-State-Machine) mechanism to generate particular form of rectangular signal. To realize FSM into code, we used structure.

Requirement

Hardware

- MCU
  - NUCLEO-F41RE
- Actuator/Sensor/Others:
  - 3Stepper Motor 28BYJ-48
  - Motor Driver ULN2003
  - breadboard

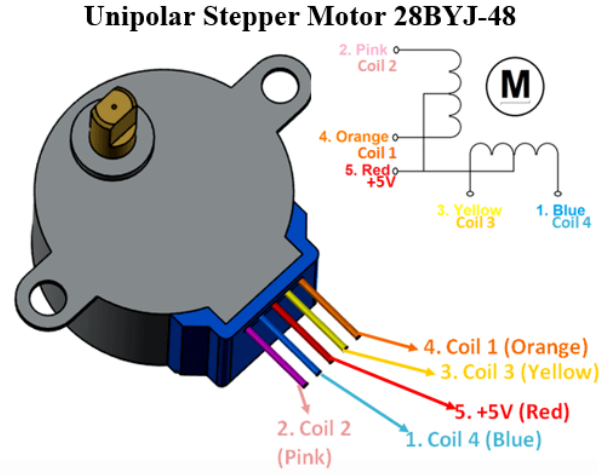
Software

- Keil uVision, CMSIS, EC_HAL library

Problem 1: Stepper Motor

Hardware Connection

We used Unipolar Stepper Motor in this lab. In addition to this to input signal into motor, Stepper Motor Driver is also used.



- Rated Voltage: 5V DC
- Number of Phases: 4
- Stride Angle: 5.625°/64
- Gear ratio: 1/32
- Pull in torque: 300 gf.cm
- Coil: Unipolar 5 lead coil

Figure 1. Unipolar Stepper Motor

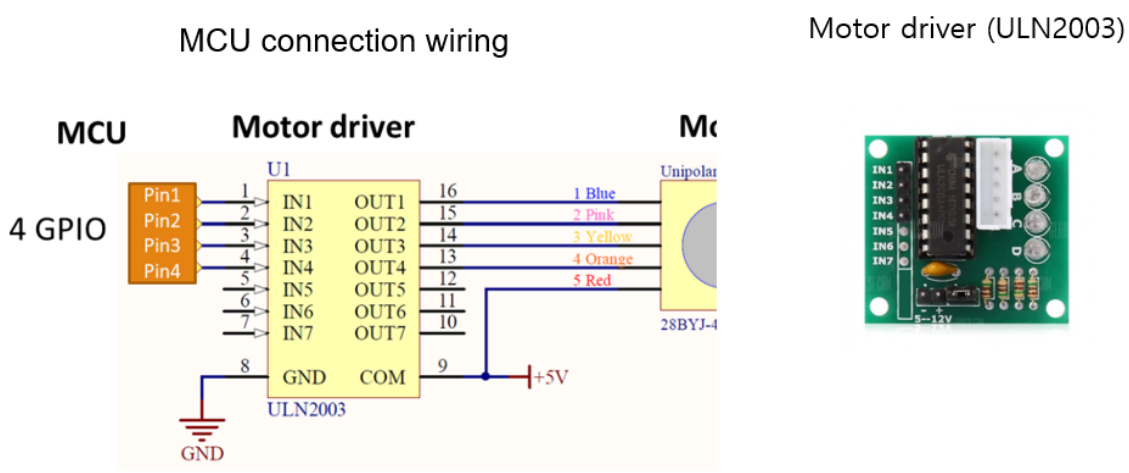


Figure 2. Unipolar Stepper Motor Drvier

Stepper Motor Sequence

Stepper Motor is worked according to the method between magnetic field and current.

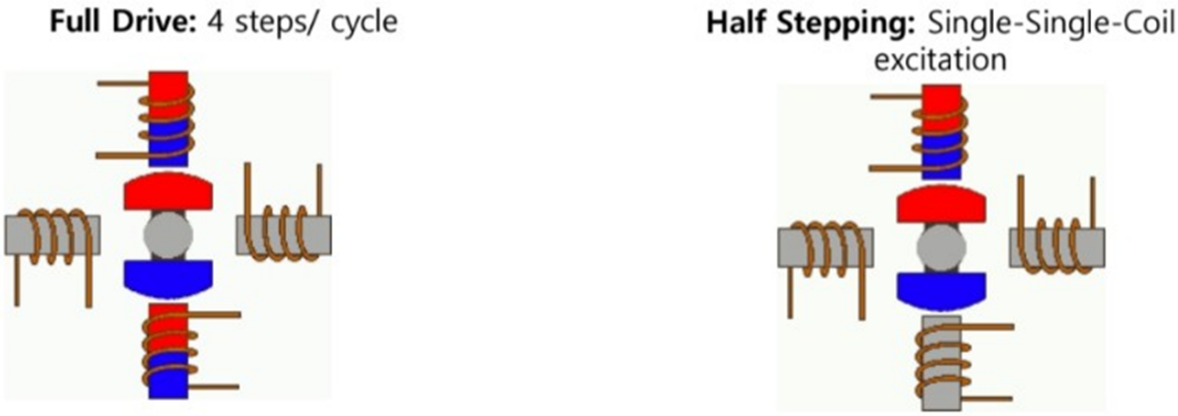


Figure 3. Magnet in Stepper Motor

In this reason, we input current into Stepper Motor to arise movement of magnet. And there are two types of current flow sequence.

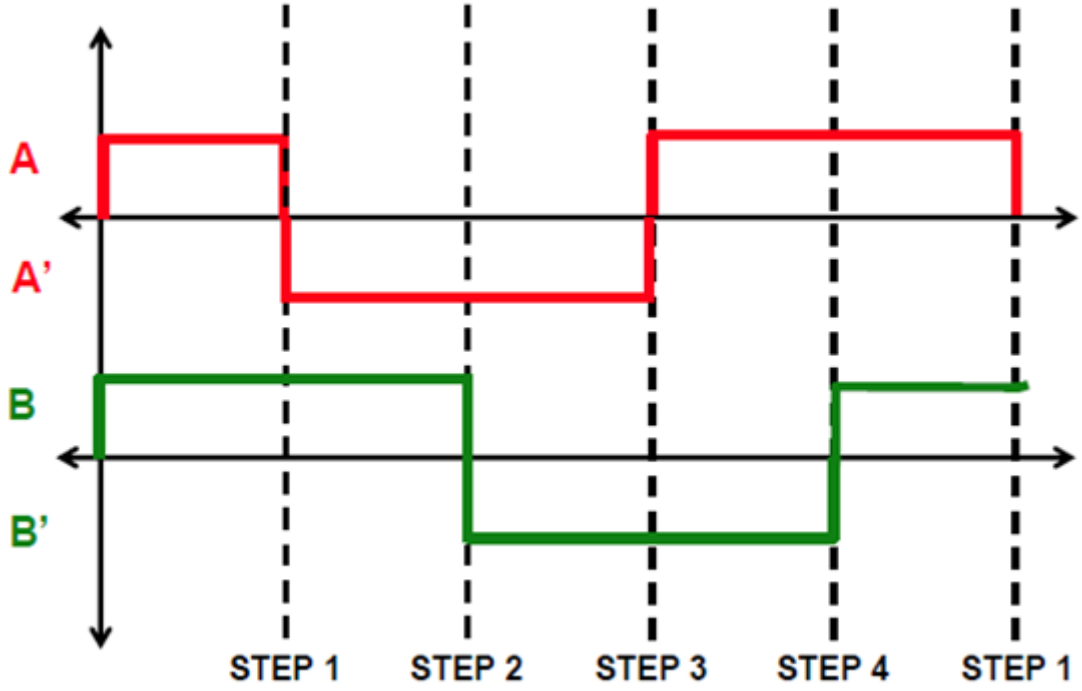


Figure 4. Full-Stepping Sequence

Phase	Port_Pin	Step 1	Step 2	Step 3	Step 4
A	PB_10	1	0	0	1
A'	PB_4	0	1	1	0
B	PB_5	1	1	0	0
B'	PB_3	0	0	1	1

Table 1. Full-Stepping Sequence with Port_Pin

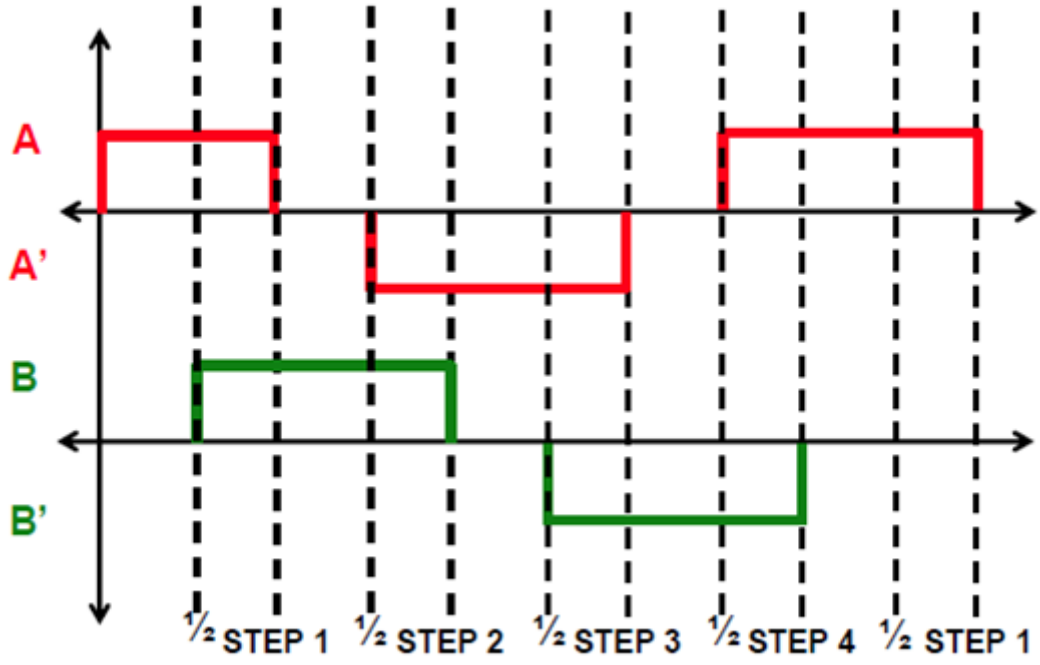


Figure 5. Half-Stepping Sequence

Phase	Port_Pin	1/2 Step 1	Step 1	1/2 Step 2	Step 2	1/2 Step 3	Step 3	1/2 Step 4	Step 4
A	PB_10	1	1	0	0	0	0	0	1
A'	PB_4	0	0	0	1	1	1	0	0
B	PB_5	0	1	1	1	0	0	0	0
B'	PB_3	0	0	0	0	0	1	1	1

Table 2. Half-Stepping Sequence with Port_Pin

Finite State Machine

```
To generate current in this sequence, we used FSM mechanism.
```

State	Previous State(DIR=0)	Subsequent State(DIR=1)	Output(A,A',B,B')
S0	S3	S1	1010
S1	S0	S2	0110
S2	S1	S3	0101
S3	S2	S0	1001

Table 3. FSM of Full-Stepping Sequence

FSM of Half-Stepping Sequence is like that.

State	Previous State(DIR=0)	Subsequent State(DIR=1)	Output(A,A',B,B')
S0	S7	S1	1000
S1	S0	S2	1010
S2	S1	S3	0010
S3	S2	S4	0110
S4	S3	S5	0100
S5	S4	S6	0101
S6	S5	S7	0001
S7	S6	S0	1001

Table 4. FSM of Half-Stepping Sequence

Circuit Diagram

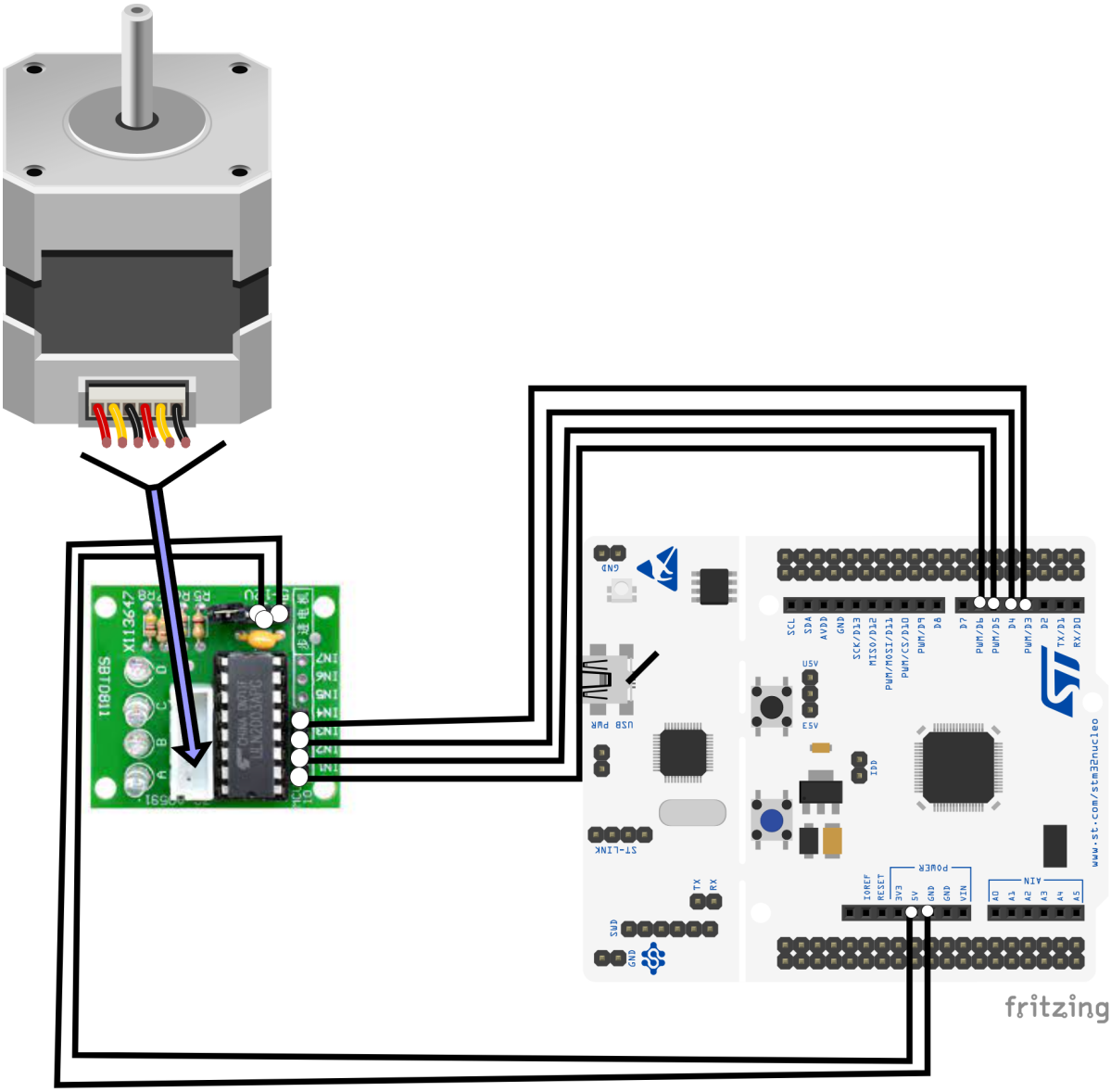


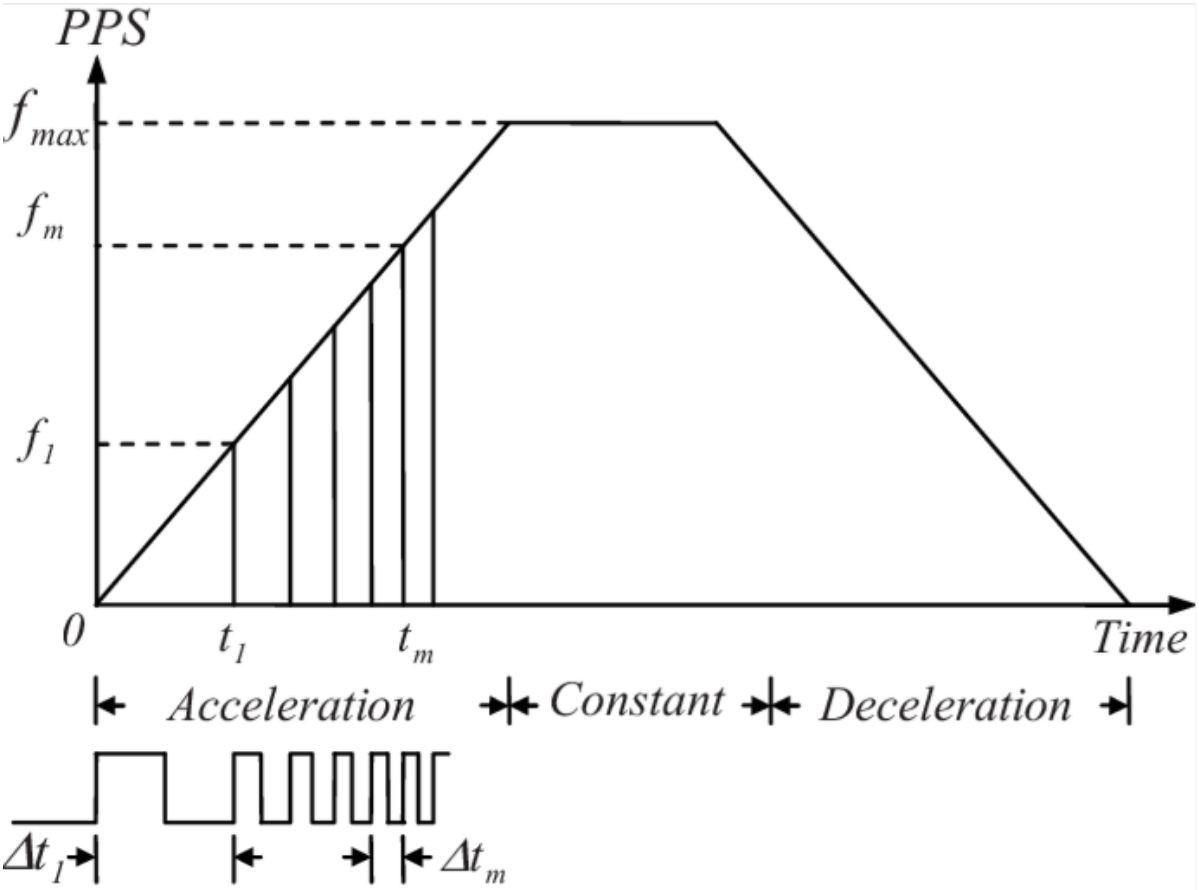
Figure 6. Stepper & Stepper Driver & MCU Circuit Diagram

Configuration

Digital Out	SysTick
PB10, PB4, PB5, PB3 NO Pull-up Pull-down Push-Pull Fast	delay()

Discussion

1. Find out the trapezoid-shape velocity profile for a stepper motor. When is this profile necessary?



This velocity profile will be necessary when this stepper motor is used to accurate work which is related with time or counting. Because if velocity is changed, stepper motor's accurate operation will be also difficulty, we have to know when stepper motor reach at steady state.

2. How would you change the code more efficiently for micro-stepping control? You don’t have to code this but need to explain your strategy.

-

Code

- main

To use timer delay, set sysTick Timer. In addition to this to control stepper motor with PB10,PB4,PB5,PB3 pins, we used 'Stepper_init' function which sets pins' environment. Fianlly, using 'Stepper_step' function to set stepper motor's operation, we can stop with External Interrupt Button signal.

```
/*----------------------------------------------------------------
Hnadong Global University_Embeded Controller_Kim Yung Keun Prof.
Student: NohYunKi
Modified: 23/11/10
----------------------------------------------------------------*/

#include "stm32f411xe.h"
#include "ecGPIO.h"
#include "ecRCC.h"
#include "ecEXTI.h"
#include "ecSysTick.h"
#include "ecStepper.h"
```

```
void setup(void);
void EXTI15_10_IRQHandler(void);
volatile int signal = 0;

int main(void) {
   // Initialiization --------------------------------------------------------
   setup();

   // Inifinite Loop ---------------------------------------------------------
   while(1){
         if(signal == 1){
         Stepper_stop();
         }
         else if(signal == 0) Stepper_step(2048, 1, FULL);  // (Step : 2048,
Direction : 0(CW) or 1(CCW), Mode : FULL or HALF)
     }

}

// Initialiization
void setup(void){

   RCC_PLL_init();                                  // System Clock = 84MHz
   SysTick_init();                                  // Systick init

   EXTI_init(GPIOC, BUTTON_PIN, FALL,0);           // External Interrupt Setting
   GPIO_init(GPIOC, BUTTON_PIN, INPUT);           // GPIOC pin13 initialization

   Stepper_init(GPIOB,10,GPIOB,4,GPIOB,5,GPIOB,3); // Stepper GPIO pin
initialization
   Stepper_setSpeed(10);                              //  set stepper motor speed
}

void EXTI15_10_IRQHandler(void) {
   if (is_pending_EXTI(BUTTON_PIN)) {
          signal = 1;
      clear_pending_EXTI(BUTTON_PIN); // cleared by writing '1'
   }
}
```

Results

Link: https://youtube.com/shorts/BtoboAa4JtI