STM32 Clocks

G V V Sharma*

Contents

***Abstract*—This manual shows how to manage clocks in arm using STM32F103C8T6.**

1 Components

Component	Value	Quantity
Breadboard		1
Resistor	220 Ω	1
STM32F103C8T6		1
Seven Segment Display	Common Anode	1
Jumper Wires		20

TABLE 1.0: Components

Problem 1.1. List all available clocks in the STM32F103C8T6 blue pill.

Solution: See Table 1.1.

Clock	Location	Type	Frequency
HSI	Internal	RC	8Mhz
LSI	Internal	RC	32.768 kHz
HSE	External	Crystal	8Mhz

TABLE 1.1: STM32F103C8T6 Clock Types

1.1 HSE

Problem 1.2. Execute the following program

```
https://github.com/gadepall/
    STM32F103C8T6/blob/master/
    examples/clocks/
    hse_systick_blink.c
```

Problem 1.3. Explain the following instruction

```
RCC->CR =0x00010000;
```

Solution: Fig. 1.3 shows the RCC− >CR register. The above instruction enables the HSE crystal, which is 8 MHz for the STM32F103C8T6.

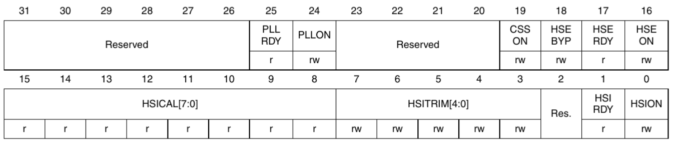

Fig. 1.3: RCC Clock Control Register (RCC− >CR)

Problem 1.4. Explain the following instruction

```
RCC->CFGR =0x00000001;
```

Solution: Fig. 1.4 shows the RCC− >CFGR register. The above instruction makes the HSE as the system clock through SW = 01.

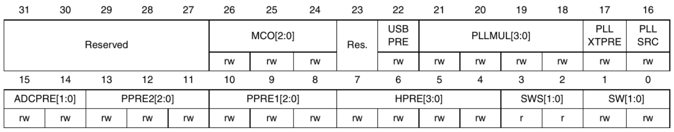

Fig. 1.4: RCC clock Configuration Register (RCC− >CFGR)

Problem 1.5. Verify that HSE is the system clock by checking that SWS = 01.

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. All content in this manual is released under GNU GPL. Free and open source.

2 PLL

Problem 2.1. Make the PLL as the system clock.

Solution:

```
RCC->CFGR  =0x00000010;
```

Problem 2.2. Choose the PLL input as HSE.

```
RCC->CFGR  =0x00010010;
```

Problem 2.3. Enable PLL

Solution:

```
RCC->CR  =0x01010000;
```

Problem 2.4. Execute the following code.

```
https://github.com/gadepall/
    STM32F103C8T6/blob/master/
    examples/clocks/
    pll_systick_blink.c
```

Problem 2.5. Make the PLL output = 24 MHz.