MCU Clock

NAME: ABDELRAHMAN MATARAWY SAYED

“WORKING ON STM32F103C6”

Lab1

- First, we write code to Toggle port A pin 13.
  - Code:

```
-in-depth
Abdelrahman Matarawy
//Mastering_Embedded System online diploma
typedef volatile unsigned int vuint32_t ;
#include <stdint.h>
#include <stdlib.h>
#include <stdio.h>
// register address
#define GPIOA_BASE 0x40010800
#define GPIOA_CRH *(volatile uint32_t *)(GPIOA_BASE + 0x04)
#define GPIOA_ODR *(volatile uint32_t *)(GPIOA_BASE + 0x0C)
int main(void)
{
    //Init GPIOA
    GPIOA_CRH &= 0xFF0FFFFF;
    GPIOA_CRH |= 0x00200000;
    while(1)
    {
        GPIOA_ODR |= 1<<13 ;
        for (int i = 0; i < 5000; i++); // arbitrary delay
        GPIOA_ODR &= ~(1<<13) ;
        for (int i = 0; i < 5000; i++); // arbitrary delay
    }
```

  - As default this code shouldn't be Run as we aren't adjusting the clock of the MCU But in TRM we found that...

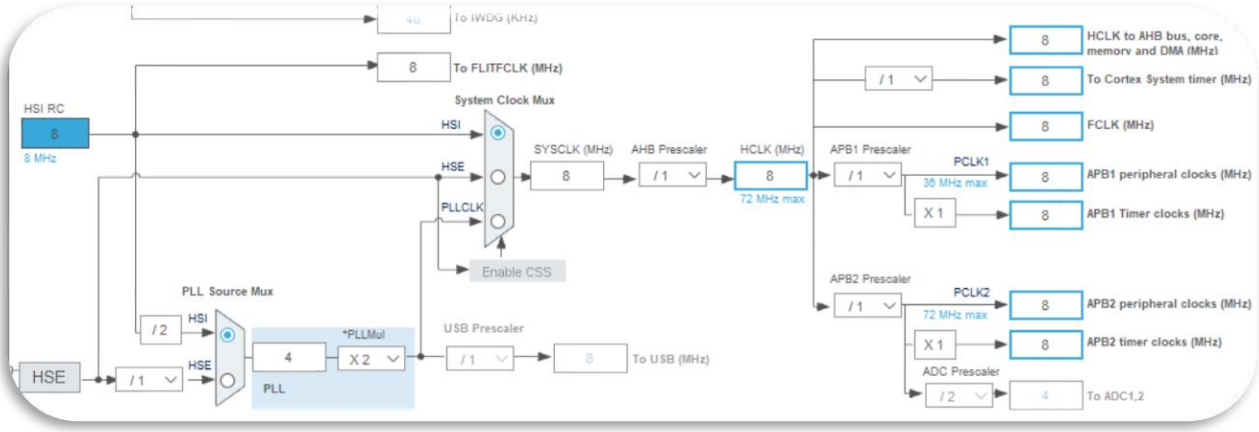
Clock configuration register (RCC_CFGR)

Address offset: 0x04

Reset value: 0x0000 0000

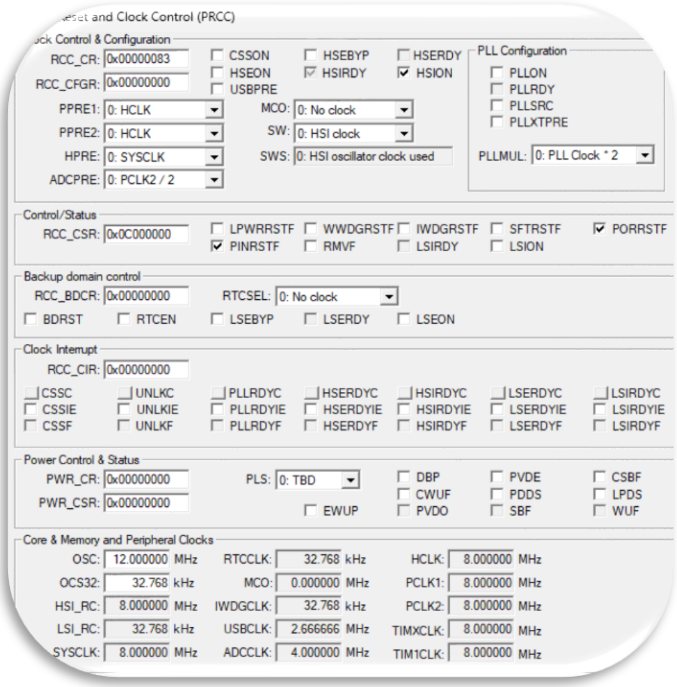
Bits 3:2 **SWS:** System clock switch status
Set and cleared by hardware to indicate which clock source is used as system clock.
00: HSI oscillator used as system clock
01: HSE oscillator used as system clock
10: PLL used as system clock
11: not applicable

Bits 1:0 **SW:** System clock switch
Set and cleared by software to select SYSCLK source.
Set by hardware to force HSI selection when leaving Stop and Standby mode or in case of failure of the HSE oscillator used directly or indirectly as system clock (if the Clock Security System is enabled).
00: HSI selected as system clock
01: HSE selected as system clock
10: PLL selected as system clock
11: not allowed

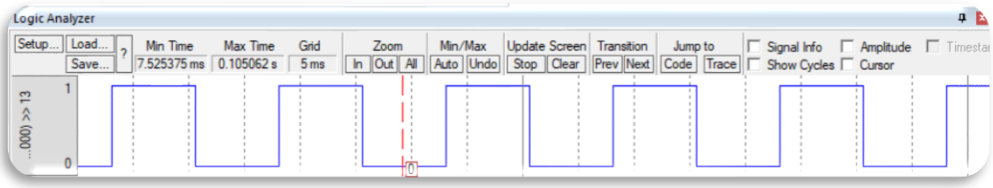


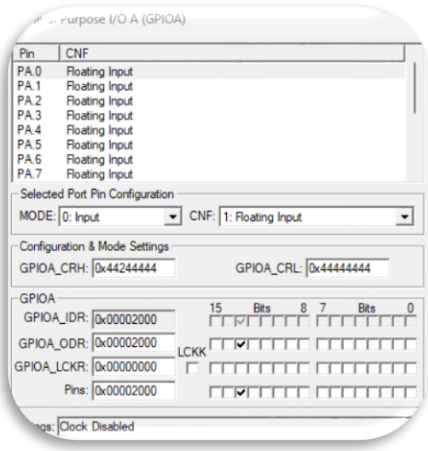
- The Clock system RCC of this code will be as default

▪

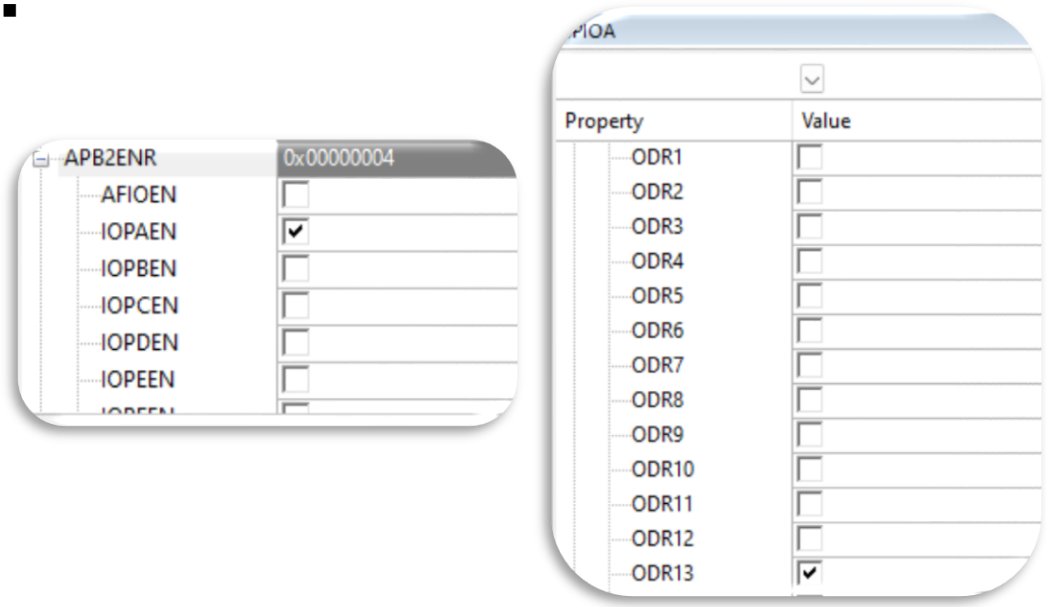


“HIS work as default so HCLK, PCLK1 and PCLK2 work as default 8MHZ”

- Output of this code:
  - 

  - 

- Second, we need to enable Clock.
  - Code:
    -
```c
//Learn-in-depth
//Abdelrahman Matarawy
//Mastering_Embedded System online diploma
typedef volatile unsigned int vuint32_t ;
#include <stdint.h>
#include <stdlib.h>
#include <stdio.h>
// register address
#define GPIOA_BASE 0x40010800
#define GPIOA_CRH *(volatile uint32_t *)(GPIOA_BASE + 0x04)
#define GPIOA_ODR *(volatile uint32_t *)(GPIOA_BASE + 0x0C)

#define RCC_BASE   0x40021000
#define RCC_APB2RSTR  *(volatile uint32_t *)(RCC_BASE + 0x18)

int main(void)
{
    //init CLK for GPIOA
    //  Bit 2 IOPARST: IO port A reset
    //  Set and cleared by software.
    //  0: No effect
    //  1: Reset IO port A
    RCC_APB2RSTR |= 1<<2;
    //Init GPIOA
    GPIOA_CRH &= 0xFF0FFFFF;
    GPIOA_CRH |= 0x00200000;
    while(1)
    {
        GPIOA_ODR |= 1<<13 ;
        for (int i = 0; i < 5000; i++); // arbitrary delay
        GPIOA_ODR &= ~(1<<13) ;
        for (int i = 0; i < 5000; i++); // arbitrary delay
    }
```

- Output:
  - 
APB2ENR
0x00000004
AFIOEN
IOPAEN
IOPBEN
IOPCEN
IOPDEN
IOPEEN
PIOA
Property
Value
ODR1
ODR2
ODR3
ODR4
ODR5
ODR6
ODR7
ODR8
ODR9
ODR10
ODR11
ODR12
ODR13

LAB2

(Change SYSCLK, HCLK, PCLK1 and PCLK2 wit different frequencies)

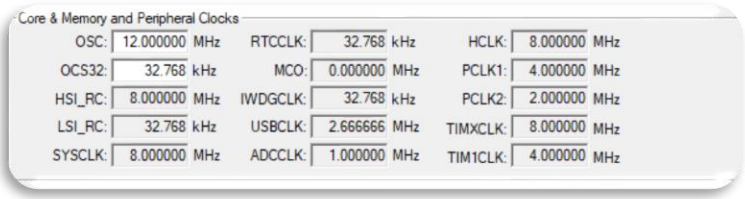
1) Configure Board to run with the Following rates:

1. APB1 Bus frequency 4MHZ
2. APB2 Bus frequency 2MHZ
3. AHB frequency 8 MHZ
4. SysClk 8 MHZ
5. Use only internal HSI_RC

- Code:

```
#define RCC_BASE   0x40021000
#define RCC_APB2RSTR  *(volatile uint32_t *)(RCC_BASE + 0x18)

#define RCC_CFGR      *(volatile uint32_t *)(RCC_BASE + 0x04)

int main(void)
{
    //init CLK for GPIOA
    //  Bit 2 IOPARST: IO port A reset
    //  Set and cleared by software.
    //  0: No effect
    //  1: Reset IO port A
    RCC_APB2RSTR |= 1<<2;

    //init GPIOA
    GPIOA_CRH &= 0xFF0FFFFF;
    GPIOA_CRH |= 0x00200000;

    //  Bits 10:8 PPRE1: APB low-speed prescaler (APB1)
    //  Set and cleared by software to control the division factor of the APB low-speed clock
    //  (PCLK1).
    //  Warning: the software has to set correctly these bits to not exceed 36 MHz on this domain.
    //  0xx: HCLK not divided
    //  100: HCLK divided by 2
    //  101: HCLK divided by 4
    //  110: HCLK divided by 8
    //  111: HCLK divided by 16
    RCC_CFGR |= 0b100 << 8;  // APB1 with prescaler 2 to make CLK APB1 bus with from 4MHZ

    //  Bits 13:11 PPRE2: APB high-speed prescaler (APB2)
    //  Set and cleared by software to control the division factor of the APB high-speed clock
    //  (PCLK2).
    //  0xx: HCLK not divided
    //  100: HCLK divided by 2
    //  101: HCLK divided by 4
    //  110: HCLK divided by 8
    //  111: HCLK divided by 16
    RCC_CFGR |= 0b101 << 11; // APB2 with prescaler 4 to make CLK APB1 bus with from MHZ

    while(1)
    {
        GPIOA_ODR |= 1<<13 ;
        for (int i = 0; i < 5000; i++); // arbitrary delay
        GPIOA_ODR &= ~(1<<13) ;
        for (int i = 0; i < 5000; i++); // arbitrary delay
    }
```

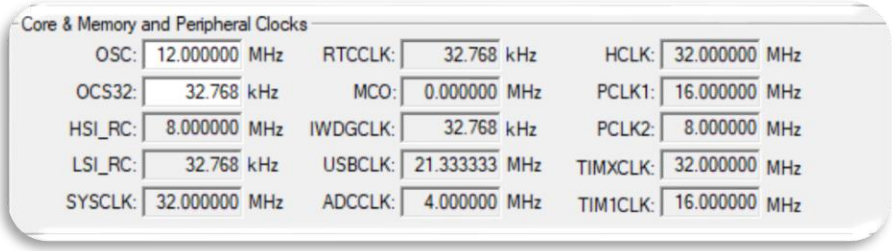
- Output Frequency:



2) Configure Board to run with the Following rates:

1. APB1 Bus frequency 16MHZ
2. APB2 Bus frequency 8MHZ
3. AHB frequency 32 MHZ
4. SysClk 32 MHZ
5. Use only internal HSI_RC

- Output frequency:



- Code:

```
//Bits 1:0 SW: System clock switch
//Set and cleared by software to select SYSCLK source.
//Set by hardware to force HSI selection when leaving Stop and Standby mode or in case of
//failure of the HSE oscillator used directly or indirectly as system clock (if the Clock Sec
//System is enabled).
//00: HSI selected as system clock
//01: HSE selected as system clock
//10: PLL selected as system clock
//11: not allowed
RCC_CFGR |= 0b10 << 0; // to enable PLL

//  Bits 21:18 PLLMUL: PLL multiplication factor
//  These bits are written by software to define the PLL multiplication factor. These bits can
//  written only when PLL is disabled.
//  Caution: The PLL output frequency must not exceed 72 MHz.
//  0000: PLL input clock x 2
//  0001: PLL input clock x 3
//  0010: PLL input clock x 4
//  0011: PLL input clock x 5
//  0100: PLL input clock x 6
//  0101: PLL input clock x 7
//  0110: PLL input clock x 8
RCC_CFGR |= 0b0110 << 18;  // CLK enter to PLL MUX 4MHZ and out 32MHZ

//  Bits 10:8 PPRE1: APB low-speed prescaler (APB1)
//  Set and cleared by software to control the division factor of the APB low-speed clock
//  (PCLK1).
//  Warning: the software has to set correctly these bits to not exceed 36 MHz on this domain.
//  0xx: HCLK not divided
//  100: HCLK divided by 2
//  101: HCLK divided by 4
//  110: HCLK divided by 8
//  111: HCLK divided by 16
RCC_CFGR |= 0b100 << 8;  // APB1 with prescaler 2 to make CLK APB1 bus with freq 16MHZ

//  Bits 13:11 PPRE2: APB high-speed prescaler (APB2)
//  Set and cleared by software to control the division factor of the APB high-speed clock
//  (PCLK2).
//  0xx: HCLK not divided
//  100: HCLK divided by 2
//  101: HCLK divided by 4
//  110: HCLK divided by 8
//  111: HCLK divided by 16
RCC_CFGR |= 0b101 << 11; // APB2 with prescaler 4 to make CLK APB2 bus with freq 8MHZ

//  Bit 24 PLLON: PLL enable
//  Set and cleared by software to enable PLL.
//  Cleared by hardware when entering Stop or Standby mode. This bit can not be reset if the
//  PLL clock is used as system clock or is selected to become the system clock.
//  0: PLL OFF
//  1: PLL ON
```