



ENG. MOHAMED HAMDY

MASTERING EMBEDDED SYSTEM ONLINE DIPLOMA
WWW.LEARN-IN-DEPTH.COM

**SECOND TERM (UNIT6LESSON3) - SYSTEM CLOCK
CONFIGURATION ON STM32F103C**

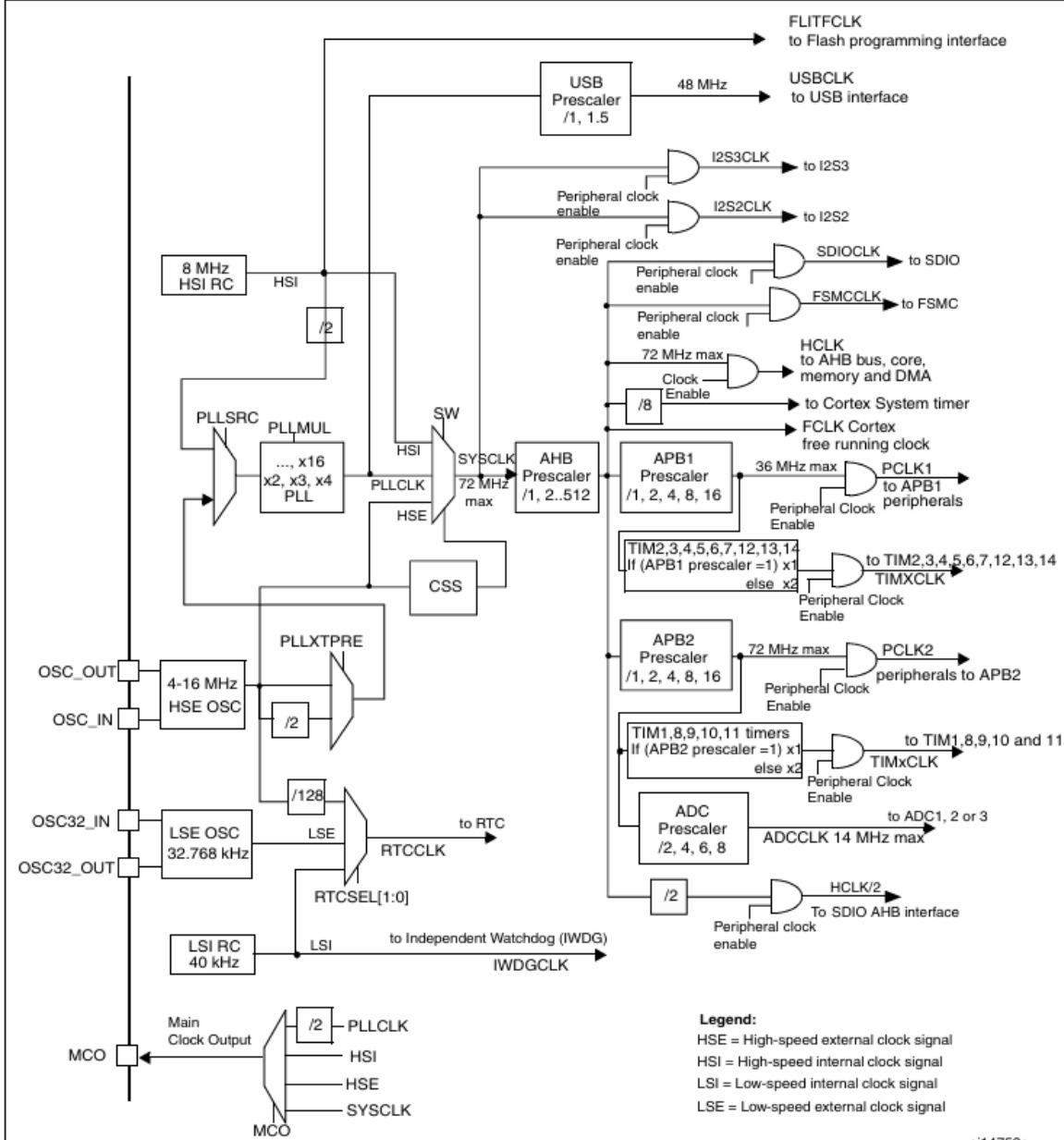
MY PROFILE:

[HTTPS://WWW.LEARN-IN-DEPTH-STORE.COM/PROFILE/MOHAMEDHAMDY-M-H-M-A/PROFILE](https://www.learn-in-depth-store.com/profile/mohamedhamdy-m-h-m-a/profile)

GITHUB REPO:

[HTTPS://GITHUB.COM/MOHAMED-HAMDY-MA/MASTERING_EMBEDDED_SYSTEM.GIT](https://github.com/mohamed-hamdy-ma/mastering_embedded_system.git)

Figure 8. Clock tree



ai14752e

Table 18. RCC register map and reset values

Offset	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								
0x00	RCC_CR	Reserved						PLL RDY	PLL ON	Reserved						CSSON	HSEBYP	HSERDY	HSEON	HSICAL[7:0]						HSITRIM[4:0]				Reserved	HSIRDY	HSION									
	Reset value							0	0							0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1						
0x04	RCC_CFGR	Reserved						MCO [2:0]	Reserved	USBPRE	PLLMUL [3:0]						PLLXTPRE	PLLSRC	ADC PRE [1:0]	PPRE2 [2:0]		PPRE1 [2:0]		HPRE[3:0]				SWS [1:0]		SW [1:0]											
	Reset value							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
0x08	RCC_CIR	Reserved								CSSC	Reserved	PLLRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC	Reserved				PLLRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE	CSSF	Reserved				PLLRDYF	HSIRDYF	LSERDYF								
	Reset value									0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
0x0C	RCC_APB2RSTR	Reserved										TIM11RST				TIM10RST	TIM9RST	Reserved				ADC3RST	USART1RST	TIM8RST	SPI1RST	TIM1RST	ADC2RST	ADC1RST	IOPGRST	IOPFRST	IOPDRST	IOPBRST	IOPARST	Reserved							
	Reset value											0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0						
0x010	RCC_APB1RSTR	Reserved	DACRST	PWRHST	BKPRST	Reserved	CANRST	Reserved	USBRST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	Reserved	SPI3RST	SPI2RST	Reserved				WWDGRST	Reserved				TIM14RST	TIM13RST	TIM12RST	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST					
	Reset value		0	0	0		0		0	0	0	0	0	0	0		0	0					0					0	0	0	0	0	0	0	0	0					
0x14	RCC_AHBENR	Reserved																									SDIOEN	Reserved	FSMCEN	Reserved	CRCEN	Reserved	FLITFEN	Reserved	SRAMEN	DMA2EN	DMA1EN				
	Reset value																										0		0		0		1		1	0	0	0			
0x18	RCC_APB2ENR	Reserved										TIM11 EN				TIM10 EN	TIM9 EN	Reserved				ADC3EN	USART1EN	TIM8EN	SPI1EN	TIM1EN	ADC2EN	ADC1EN	IOPGEN	IOPFEN	IOPDEN	IOPCEN	IOPBEN	IOPAEN	Reserved	Reserved	AFIOEN				
	Reset value											0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x1C	RCC_APB1ENR	Reserved	DACEN	PWREN	BKPEN	Reserved	CANEN	Reserved	USBEN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	Reserved	SPI3EN	SPI2EN	Reserved				WWDGEN	Reserved				TIM14EN	TIM13EN	TIM12EN	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN					
	Reset value		0	0	0		0		0	0	0	0	0	0	0		0	0					0					0	0	0	0	0	0	0	0	0					
0x20	RCC_BDCR	Reserved															BDRST	RTCON	Reserved						RTC SEL [1:0]		Reserved						LSEBYP	LSERDY	LSEON						
	Reset value																0	0							0	0							0	0	0						

An abstract geometric design on the left side of the slide. It features a dark blue background with various geometric shapes and patterns. A white circle is positioned near the top left. Below it, a light blue semi-circle is visible. To the right of the semi-circle, there is a pink triangle with diagonal lines. Further down, there is a pink square with a pattern of concentric lines. At the bottom, there is a pink triangle with a pattern of concentric lines. The overall design is modern and minimalist.

SCREENSHOTS

```

25 #pragma pack(1)
26 typedef struct {
27     uint32_t HSION:1;
28     uint32_t HSIRDY:1;
29     uint32_t :1;
30     uint32_t HSITRIM:5;
31     uint32_t HSICAL:8;
32     uint32_t HSEON:1;
33     uint32_t HSERDY:1;
34     uint32_t HSEBYP:1;
35     uint32_t CSSON:1;
36     uint32_t :4;
37     uint32_t PLLON:1;
38     uint32_t PLLRDY:1;
39     uint32_t :6;
40 }SRCC_CR_t;

```

```

42 #pragma pack(1)
43 typedef struct {
44     uint32_t SW:2;
45     uint32_t SWS:2;
46     uint32_t HPRE:4;
47     uint32_t PPRE1:3;
48     uint32_t PPRE2:3;
49     uint32_t ADC:2;
50     uint32_t PLLSRC:1;
51     uint32_t PLLXTPRE:1;
52     uint32_t PLLMUL:4;
53     uint32_t USBPRE:1;
54     uint32_t :1;
55     uint32_t MCO:3;
56     uint32_t :5;
57 }SRCC_CFGR_t;

```

```

59 #pragma pack(1)
60 typedef struct {
61     uint32_t :2;
62     uint32_t IOPAEN:1;
63     uint32_t :29;
64 }SRCC_APB2ENR_t;
65
66 #define RCC_BASE      0x40021000
67 #define PORTA_BASE    0x40010800
68
69 #define GPIOA_CRH      *(volatile uint32_t*)(PORTA_BASE+0x04)
70 #define GPIOA_ODR      *(volatile uint32_t*)(PORTA_BASE+0x0C)

```

```
147 int main(void)
148 {
149
150     clock_init();
151
152     int i;
153     GPIOA_CRH &= 0xFF0FFFFFFF;
154     GPIOA_CRH |= 0x00200000;
155     while (1)
156     {
157         GPIOA_ODR |= 1<<13;
158         for (i=0; i<5000; i++);
159         GPIOA_ODR &=~(1<<13);
160         for (i=0; i<5000; i++);
161     }
162     return 0;
163 }
```

```

void clock_init(){
    volatile SRCC_CR_t* const RCC_CR = (SRCC_CR_t*)(RCC_BASE+0x00);
    volatile SRCC_CFGR_t* const RCC_CFGR = (SRCC_CFGR_t*)(RCC_BASE+0x04);
    volatile SRCC_APB2ENR_t* const RCC_APB2ENR = (SRCC_APB2ENR_t*)(RCC_BASE+0x18);

    // APB1 Bus frequency 4MHZ
    // APB2 Bus frequency 2MHZ
    // AHB frequency 8 MHZ
    // SysClk 8 MHZ
    // Use only internal HSI_RC

    RCC_CFGR->PPRE1 = 0b100; //100: HCLK divided by 2
    RCC_CFGR->PPRE2 = 0b101; //101: HCLK divided by 4
    RCC_APB2ENR->IOPAEN = 0b1; //Enable PORTA Clock
}

```

Power, Reset and Clock Control (PRCC)

Clock Control & Configuration

RCC_CR: 0x00000083 ☐ CSSON ☐ HSEBYP ☐ HSERDY ☐ HSION
☐ HSEON ☒ HSIIRDY ☒ HSION
☐ USBPRE
PPRE1: 4: HCLK / 2 MCO: 0: No clock
PPRE2: 5: HCLK / 4 SW: 0: HSI clock
HPRE: 0: SYSCLK SWS: 0: HSI oscillator clock used
ADCPRE: 0: PCLK2 / 2
PLL Configuration
☐ PLLON
☐ PLLRDY
☐ PLLSRC
☐ PLLXTPRE
PLLMUL: 0: PLL Clock * 2

Control/Status
RCC_CSR: 0x0C000000 ☐ LPWRRSTF ☐ WWDGRSTF ☐ IWDGRSTF ☐ SFTRSTF ☒ PORRSTF
☒ PINRSTF ☐ RMVF ☐ LSIRDY ☐ LSION

Backup domain control
RCC_BDCR: 0x00000000 RTCSEL: 0: No clock
☐ BDRST ☐ RTCEN ☐ LSEBYP ☐ LSEDRY ☐ LSEON

Clock Interrupt
RCC_CIR: 0x00000000
☐ CSSC ☐ UNLKC ☐ PLLRDYC ☐ HSERDYC ☐ HSIIRDYC ☐ LSEDRYC ☐ LSIRDYC
☐ CSSIE ☐ UNLKIE ☐ PLLRDYIE ☐ HSERDYIE ☐ HSIIRDYIE ☐ LSEDRYIE ☐ LSIRDYIE
☐ CSSF ☐ UNLKF ☐ PLLRDYF ☐ HSERDYF ☐ HSIIRDYF ☐ LSEDRYF ☐ LSIRDYF

Power Control & Status
PWR_CR: 0x00000000 PLS: 0: TBD ☐ DBP ☐ PVDE ☐ CSBF
PWR_CSR: 0x00000000 ☐ EWUP ☐ CWUF ☐ PDDS ☐ LPDS
☐ PVDO ☐ SBF ☐ WUF

Core & Memory and Peripheral Clocks
OSC: 12.000000 MHz RTCCLK: 32.768 kHz HCLK: 8.000000 MHz
OCS32: 32.768 kHz MCO: 0.000000 MHz PCLK1: 4.000000 MHz
HSI_RC: 8.000000 MHz IWDGCLK: 32.768 kHz PCLK2: 2.000000 MHz
LSI_RC: 32.768 kHz USBCLK: 2.666666 MHz TIMXCLK: 8.000000 MHz
SYSCLK: 8.000000 MHz ADCCLK: 1.000000 MHz TIM1CLK: 4.000000 MHz

```

void clock_init(){
    volatile SRCC_CR_t* const RCC_CR = (SRCC_CR_t*)(RCC_BASE+0x00);
    volatile SRCC_CFGR_t* const RCC_CFGR = (SRCC_CFGR_t*)(RCC_BASE+0x04);
    volatile SRCC_APB2ENR_t* const RCC_APB2ENR = (SRCC_APB2ENR_t*)(RCC_BASE+0x18);

    // APB1 Bus frequency 16MHZ
    // APB2 Bus frequency 8MHZ
    // AHB frequency 32 MHZ
    // SysClk 32 MHZ
    // Use only internal HSI_RC

    // PLLSRC & PLLMUL bits can be written only when PLL is disabled.
    RCC_CFGR->PLLSRC = 0b0; //0: HSI oscillator clock / 2 selected as PLL input clock
    RCC_CFGR->PLLMUL = 0b0110; //0110: PLL input clock x 8

    RCC_CR->PLLON = 0b1; //1: PLL ON

    while(!RCC_CR->PLLRDY){
    }

    RCC_CFGR->PPRE1 = 0b100; //100: HCLK divided by 2
    RCC_CFGR->PPRE2 = 0b101; //101: HCLK divided by 4

    RCC_CFGR->SW = 0b10; //10: PLL selected as system clock

    RCC_APB2ENR->IOPAEN = 0b1; //Enable PORTA Clock
}

```

Power, Reset and Clock Control (PRCC)

Clock Control & Configuration

RCC_CR: 0x03000083

RCC_CFGR: 0x00182C0A

PPRE1: 4: HCLK / 2

PPRE2: 5: HCLK / 4

HPRE: 0: SYSCLK

ADCPRE: 0: PCLK2 / 2

CSSON

HSEON

USBPRE

MCO: 0: No clock

SW: 2: PLL clock

SWS: 2: PLL clock used

PLL Configuration

PLLON

PLLRDY

PLLSRC

PLLXTPRE

PLLMUL: 6: PLL Clock * 8

Control/Status

RCC_CSR: 0x0C000000

LPWRRSTF

WWDGRSTF

IWDGRSTF

SFTRSTF

PORRSTF

PINRSTF

RMVF

LSIRDY

LSION

Backup domain control

RCC_BDCR: 0x00000000

RTCSEL: 0: No clock

BDRST

RTCEN

LSEBYP

LSERDY

LSEON

Clock Interrupt

RCC_CIR: 0x00000000

CSSC

UNLKC

PLLRDYC

HSERDYC

HSIRDYC

LSERDYC

LSIRDYC

CSSIE

UNLKIE

PLLRDYIE

HSERDYIE

HSIRDYIE

LSERDYIE

LSIRDYIE

CSSF

UNLKF

PLLRDYF

HSERDYF

HSIRDYF

LSERDYF

LSIRDYF

Power Control & Status

PWR_CR: 0x00000000

PLS: 0: TBD

PWR_CSR: 0x00000000

DBP

PVDE

CSBF

CWUF

PDDS

LPDS

PVDO

SBF

WUF

EWUP

Core & Memory and Peripheral Clocks

OSC: 12.000000 MHz

RTCCLK: 32.768 kHz

HCLK: 32.000000 MHz

OCS32: 32.768 kHz

MCO: 0.000000 MHz

PCLK1: 16.000000 MHz

HSI_RC: 8.000000 MHz

IWDGCLK: 32.768 kHz

PCLK2: 8.000000 MHz

LSI_RC: 32.768 kHz

USBCLK: 21.333333 MHz

TIMXCLK: 32.000000 MHz

SYSCLK: 32.000000 MHz

ADCCLK: 4.000000 MHz

TIM1CLK: 16.000000 MHz


```

void clock_init(){
    volatile SRCC_CR_t* const RCC_CR = (SRCC_CR_t*)(RCC_BASE+0x00);
    volatile SRCC_CFGR_t* const RCC_CFGR = (SRCC_CFGR_t*)(RCC_BASE+0x04);
    volatile SRCC_APB2ENR_t* const RCC_APB2ENR = (SRCC_APB2ENR_t*)(RCC_BASE+0x18);

    // APB1 Bus frequency 16MHZ
    // APB2 Bus frequency 16MHZ
    // AHB frequency 32 MHZ
    // SysClk 32 MHZ
    // Use only internal HSI_RC

    // PLLSRC & PLLMUL bits can be written only when PLL is disabled.
    RCC_CFGR->PLLSRC = 0b0; //0: HSI oscillator clock / 2 selected as PLL input clock
    RCC_CFGR->PLLMUL = 0b0110; //0110: PLL input clock x 8

    RCC_CR->PLLON = 0b1; //1: PLL ON

    while(!RCC_CR->PLLRDY){
    }

    RCC_CFGR->PPRE1 = 0b100; //100: HCLK divided by 2
    RCC_CFGR->PPRE2 = 0b100; //100: HCLK divided by 4

    RCC_CFGR->SW = 0b10; //10: PLL selected as system clock

    RCC_APB2ENR->IOPAEN = 0b1; //Enable PORTA Clock
}

```

Power, Reset and Clock Control (PRCC)

Clock Control & Configuration

RCC_CR: 0x03000083

RCC_CFGR: 0x0018240A

PPRE1: 4: HCLK / 2

PPRE2: 4: HCLK / 2

HPRE: 0: SYSCLK

ADCPRE: 0: PCLK2 / 2

CSSON

HSEON

USBPRE

MCO: 0: No clock

SW: 2: PLL clock

SWS: 2: PLL clock used

PLL Configuration

PLLON

PLLRDY

PLLSRC

PLLXTPRE

PLLMUL: 6: PLL Clock * 8

Control/Status

RCC_CSR: 0x0C000000

LPWRRSTF

WWDGRSTF

IWDGRSTF

SFTRSTF

PORRSTF

PINRSTF

RMVF

LSIRDY

LSION

Backup domain control

RCC_BDCR: 0x00000000

RTCSEL: 0: No clock

BDRST

RTCEN

LSEBYP

LSERDY

LSEON

Clock Interrupt

RCC_CIR: 0x00000000

CSSC

UNLKC

PLLRDYC

HSERDYC

HSIRDYC

LSERDYC

LSIRDYC

CSSIE

UNLKIE

PLLRDYIE

HSERDYIE

HSIRDYIE

LSERDYIE

LSIRDYIE

CSSF

UNLKF

PLLRDYF

HSERDYF

HSIRDYF

LSERDYF

LSIRDYF

Power Control & Status

PWR_CR: 0x00000000

PLS: 0: TBD

DBP

PVDE

CSBF

CWUF

PDDS

LPDS

PVDO

SBF

WUF

EWUP

Core & Memory and Peripheral Clocks

OSC: 12.000000 MHz

RTCCLK: 32.768 kHz

HCLK: 32.000000 MHz

OCS32: 32.768 kHz

MCO: 0.000000 MHz

PCLK1: 16.000000 MHz

HSI_RC: 8.000000 MHz

IWDGCLK: 32.768 kHz

PCLK2: 16.000000 MHz

LSI_RC: 32.768 kHz

USBCLK: 21.333333 MHz

TIMXCLK: 32.000000 MHz

SYSCLK: 32.000000 MHz

ADCCLK: 8.000000 MHz

TIM1CLK: 32.000000 MHz

F:\STUDY\COURSES\Mastering Embedded System Diploma\ASSIGNMENTS\Unit_6_Microcontroller_Architecture\Lesson_3_MCU_Clocks\uVisionProject\U6L3_Lab.uvprojx - µVisio

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Registers

Register	Value
Core	
R0	0x20000000
R1	0x00000002
R2	0x4001080C
R3	0x00002000
R4	0x2000001C
R5	0x00000000
R6	0x00000000
R7	0x200027F0
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x200027F0
R14 (LR)	0x08000203
R15 (PC)	0x08000226
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	140083498
Sec	9.72264369

Logic Analyzer

Setup... Load... Save... Min Time: 0 s Max Time: 4.377609 s Grid: 5 ms Zoom: In Out All Auto Undo Update Screen: Stop Clear Transition: Prev Next Jump to: Code Trace Signal Info: Show Cycles Amplitude: Cursor Timestamps

Disassembly Logic Analyzer

main.c

```

153 GPIOA_CRH &= 0xFFFFFFF;
154 GPIOA_CRH |= 0x00200000;
155 while (1)
156 {
157     GPIOA_ODR |= 1<<13;
158     for (i=0; i<5000; i++);
159     GPIOA_ODR &= ~(1<<13);
160     for (i=0; i<5000; i++);
161 }
162 return 0;
163 }
164

```

General Purpose I/O A (GPIOA)

Pin	CNF
PA.0	Floating Input
PA.1	Floating Input
PA.2	Floating Input
PA.3	Floating Input
PA.4	Floating Input
PA.5	Floating Input
PA.6	Floating Input
PA.7	Floating Input

Selected Port Pin Configuration

MODE: 0: Input CNF: 1: Floating Input

Configuration & Mode Settings

GPIOA_CRH: 0x44244444 GPIOA_CRL: 0x44444444

GPIOA

GPIOA_IDR	GPIOA_ODR	GPIOA_LCKR	Pins
0x00002000	0x00002000	0x00000000	0x00002000

Settings: Clock Enabled

GPIOA

Property	Value
CRL	0x44444444
CRH	0x44244444
IDR	0x00002000
ODR	0x00002000
BSRR	0
BRR	0
LCKR	0

Command

```

*** error 35: undefined line number
BS \\U6L3_LAB\..\Src\main.c\113
^
*** error 35: undefined line number
LA ((PORTA & 0x00002000) >> 13 & 0x2000) >> 13
<
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVER

```

Project Registers

Command

*** error 35: undefined line number

BS \\U6L3_LAB\..\Src\main.c\113

*** error 35: undefined line number

LA ((PORTA & 0x00002000) >> 13 & 0x2000) >> 13

ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVER

t1: 9.72264369 sec L:158 C:1 CAP NUM SCRL OVR: R/W

F:\STUDY\COURSES\Mastering Embedded System Diploma\ASSIGNMENTS\Unit_6_Microcontroller_Architecture\Lesson_3_MCU_Clocks\uVisionProject\U6L3_Lab.uvprojx - µVisio

File Edit View Project Flash Debug Peripherals Tools SVCS Window Help

Registers

Register	Value
Core	
R0	0x20000000
R1	0x00000002
R2	0x4001080C
R3	0x00000000
R4	0x2000001C
R5	0x00000000
R6	0x00000000
R7	0x200027F0
R8	0x00000000
R9	0x00000000
R10	0x00000000
R11	0x00000000
R12	0x00000000
R13 (SP)	0x200027F0
R14 (LR)	0x08000203
R15 (PC)	0x08000248
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	140143518
Sec	9.72451931

Logic Analyzer

Setup... Load... Save... Min Time: 0 s Max Time: 4.379485 s Grid: 5 ms

Zoom: In Out All Auto Undo Update Screen: Stop Clear Transition: Prev Next Jump to: Code Trace Signal Info: Show Cycles Amplitude: Cursor Timestamps

Disassembly Logic Analyzer

main.c

```

153 GPIOA_CRH &= 0xFFFFFFF;
154 GPIOA_CRH |= 0x00200000;
155 while (1)
156 {
157     GPIOA_ODR |= 1 << 13;
158     for (i=0; i<5000; i++);
159     GPIOA_ODR &= ~(1 << 13);
160     for (i=0; i<5000; i++);
161 }
162 return 0;
163 }
164

```

General Purpose I/O A (GPIOA)

Pin	CNF
PA.0	Floating Input
PA.1	Floating Input
PA.2	Floating Input
PA.3	Floating Input
PA.4	Floating Input
PA.5	Floating Input
PA.6	Floating Input
PA.7	Floating Input

Selected Port Pin Configuration

MODE: 0: Input CNF: 1: Floating Input

Configuration & Mode Settings

GPIOA_CRH: 0x44244444 GPIOA_CRL: 0x44444444

GPIOA

GPIOA_IDR: 0x00000000 15 Bits 8 7 Bits 0

GPIOA_ODR: 0x00000000 LCKK

GPIOA_LCKR: 0x00000000

Pins: 0x00000000

Settings: Clock Enabled

GPIOA

Property	Value
CRL	0x44444444
CRH	0x44244444
IDR	0x00000000
ODR	0x00000000
BSRR	0
BRR	0
LCKR	0

Command

```

*** error 35: undefined line number
BS \\U6L3_LAB\..\Src/main.c\113
^
*** error 35: undefined line number
LA ((PORTA & 0x00002000) >> 13 & 0x2000) >> 13
<
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVER

```

Project Registers

Command

*** error 35: undefined line number

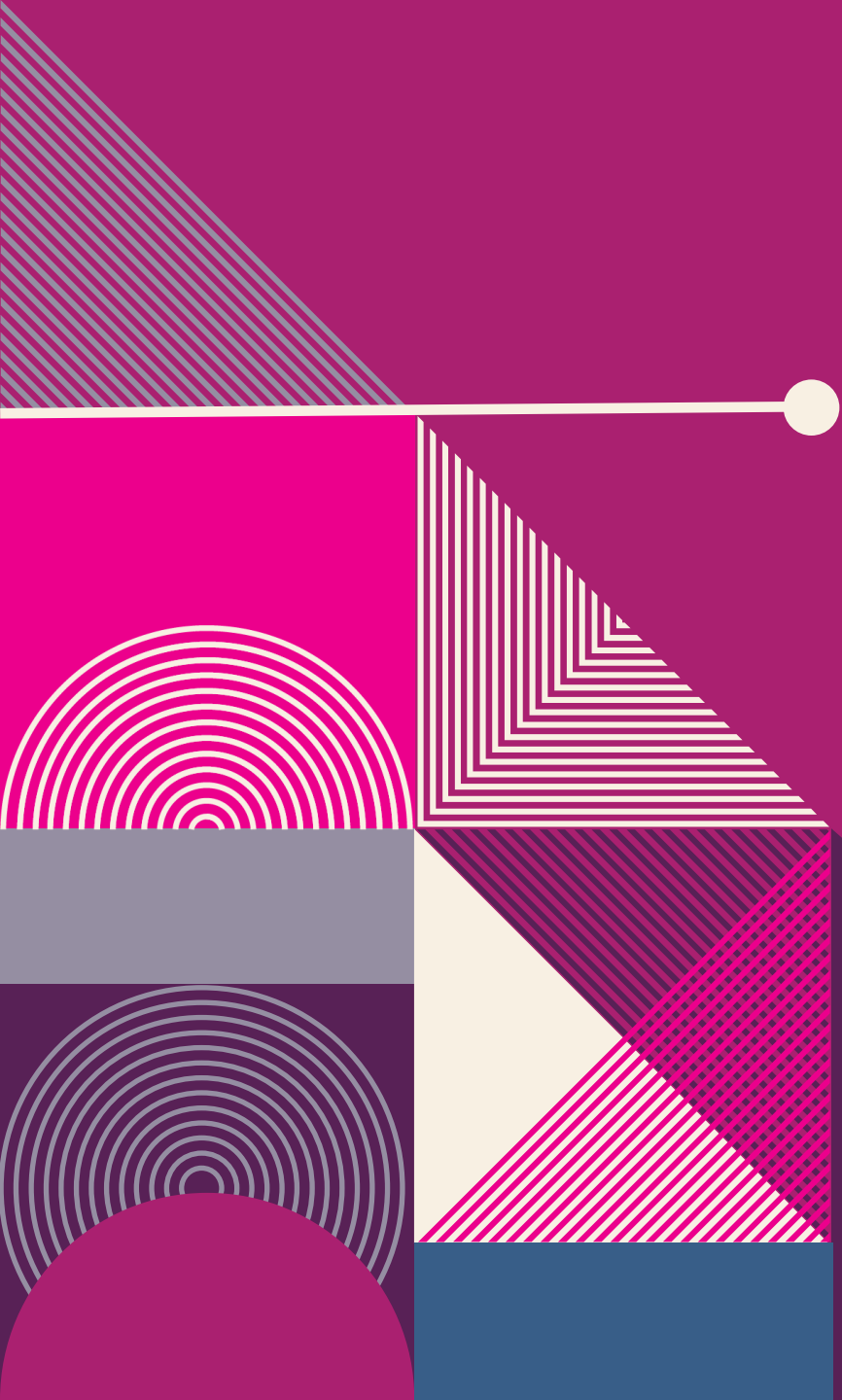
BS \\U6L3_LAB\..\Src/main.c\113

*** error 35: undefined line number

LA ((PORTA & 0x00002000) >> 13 & 0x2000) >> 13

ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVER

t1: 9.72451931 sec L:160 C:1 CAP NUM SCRL OVR/ R/W



THANK YOU

Mohamed Hamdy

Mohamedhamdy.m.h.m.a@gmail.com