

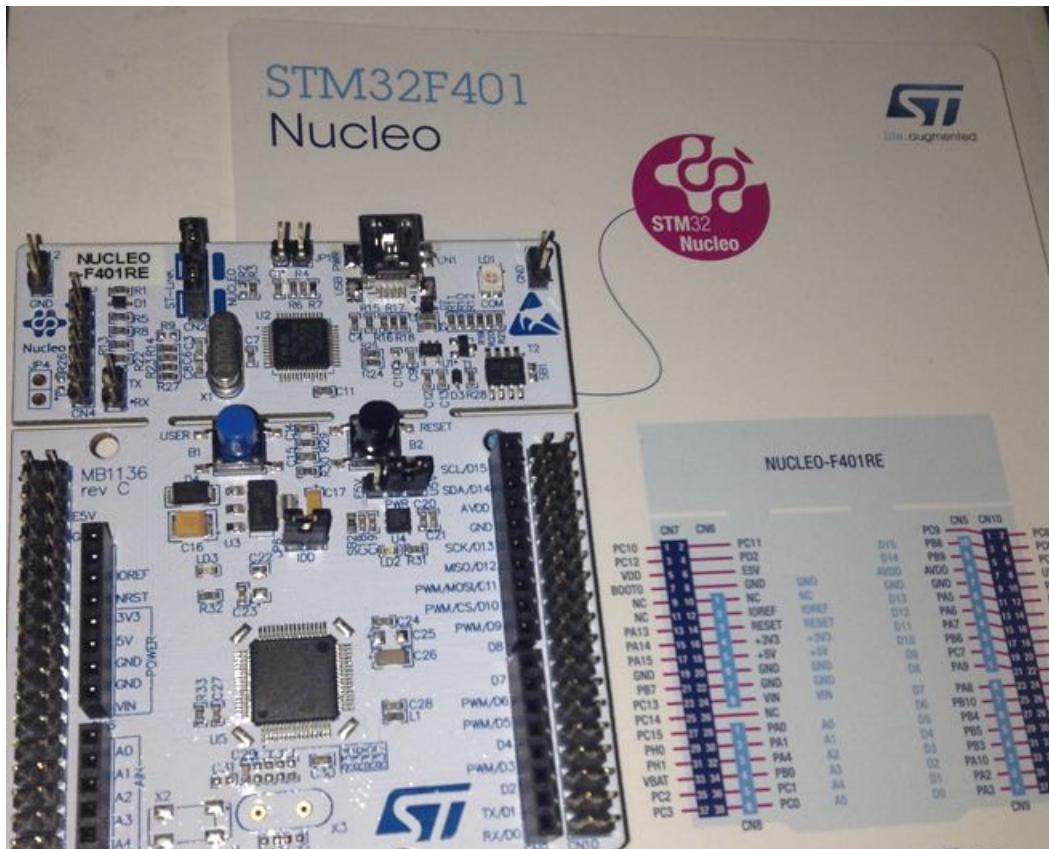
Programming Your Nucleo Board

Updated:Wednesday, March 14, 2018

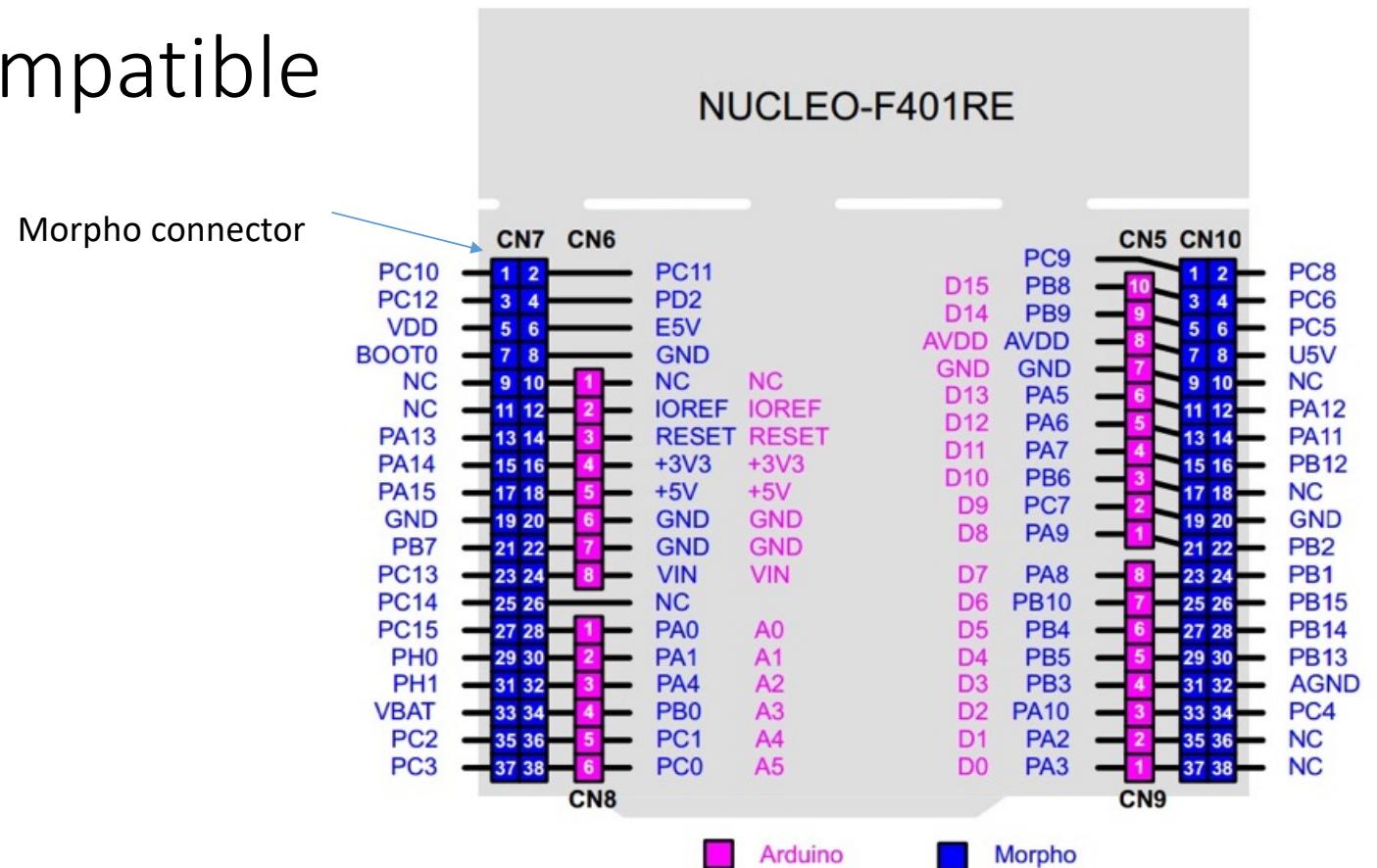
Outline

- Download the provided Zip file on the web!!
- What is Nucleo Board?
- Programing the development board

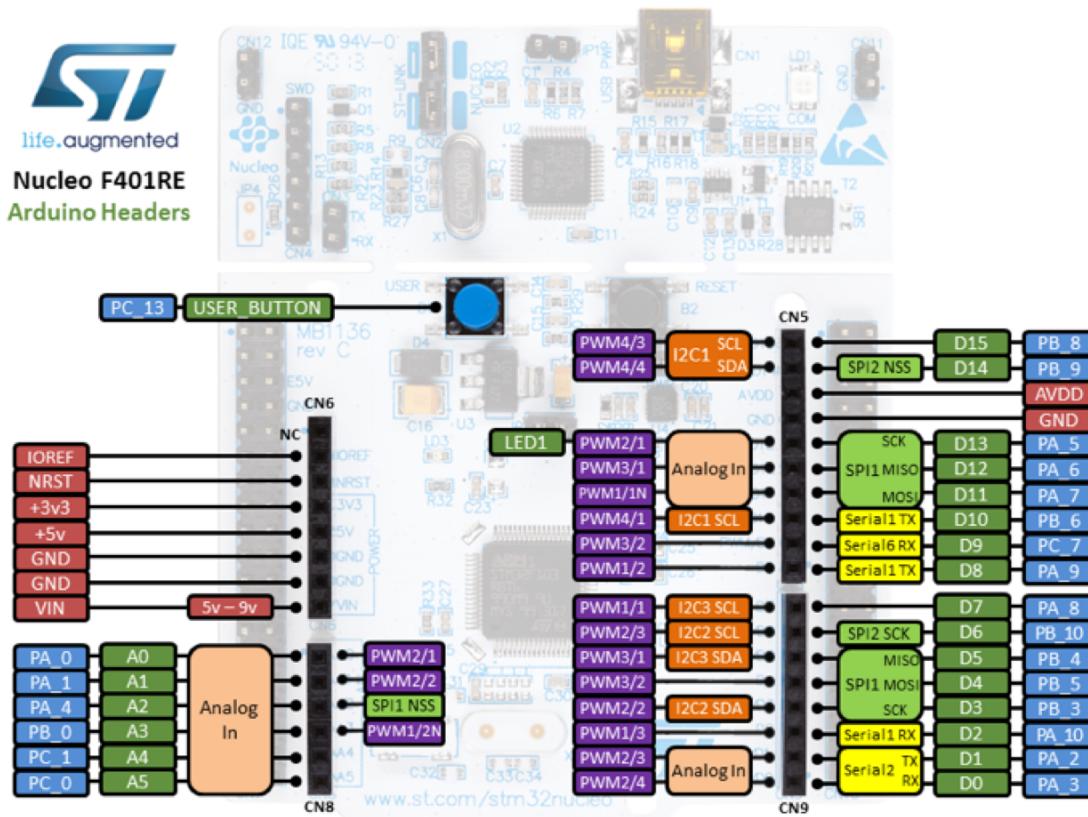
STM32F401



Board IO Pins Ardunio Compatible

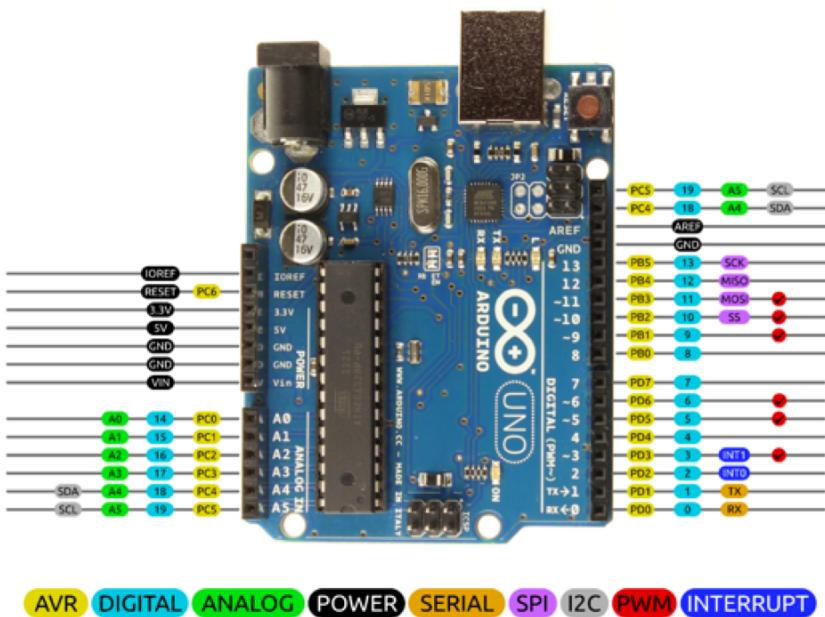


Nucleo Board Pinouts

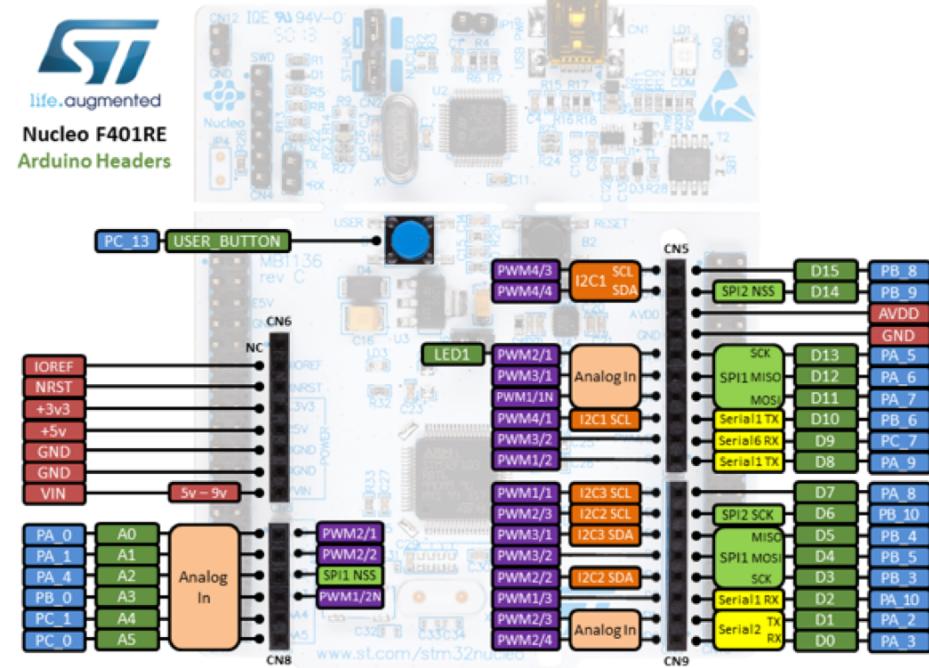


Comparing Arduino and Nucleo Board

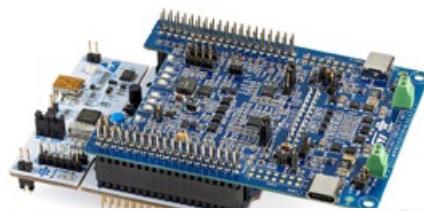
Arduino Uno R3 Pinout



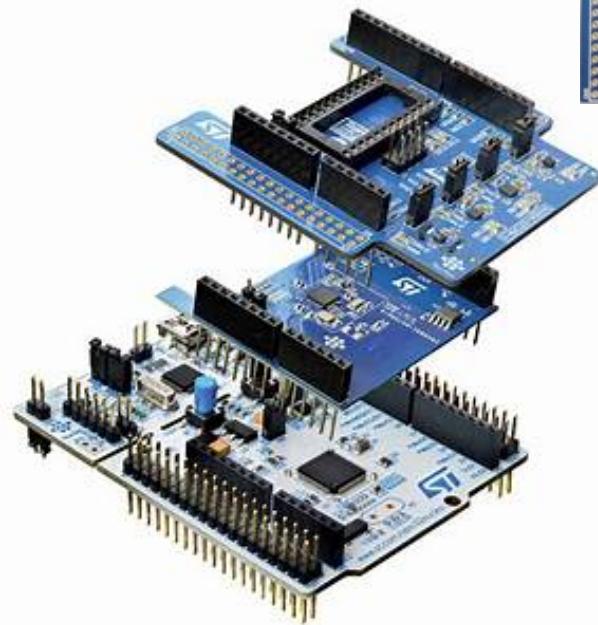
2014 by Bouni
Photo by Arduino.cc



Expansion Shields for Nucleo Board



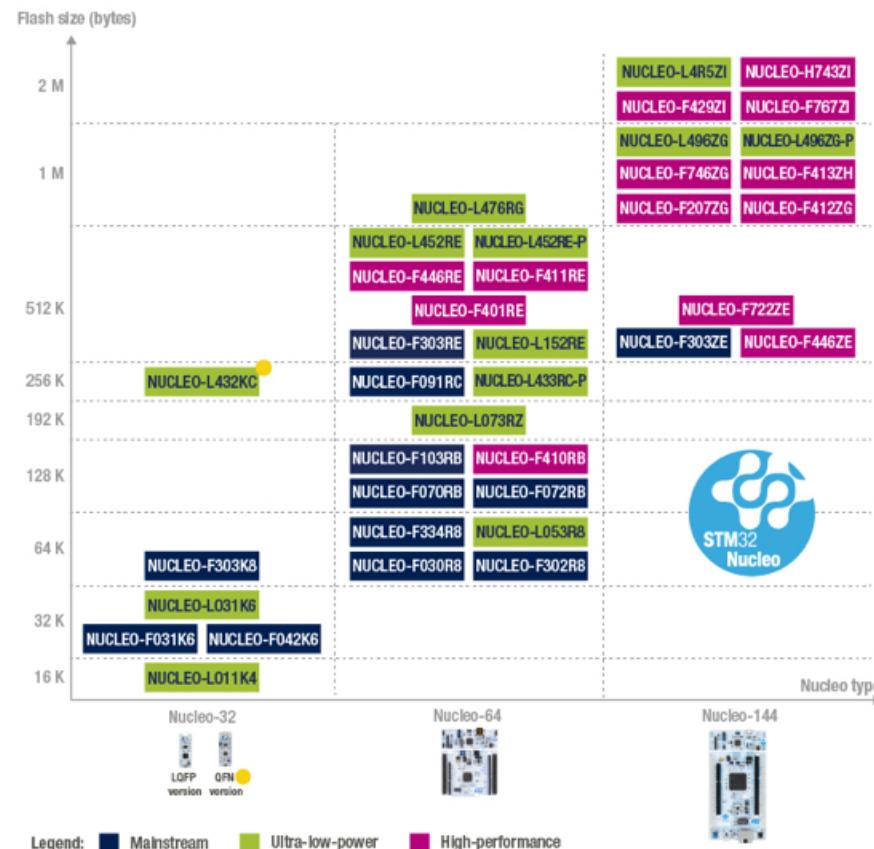
Driver boards



CO Sensor



STM32 Nucleo Ecosystem



<http://www.st.com/en/evaluation-tools/stm32-mcu-nucleo.html?querycriteria=productId=LN1847>

STM32 Nucleo Ecosystem



<http://www.st.com/en/evaluation-tools/stm32-mcu-nucleo.html?querycriteria=productId=LN1847>

STM32F401RE MCU – STM32F401RET6

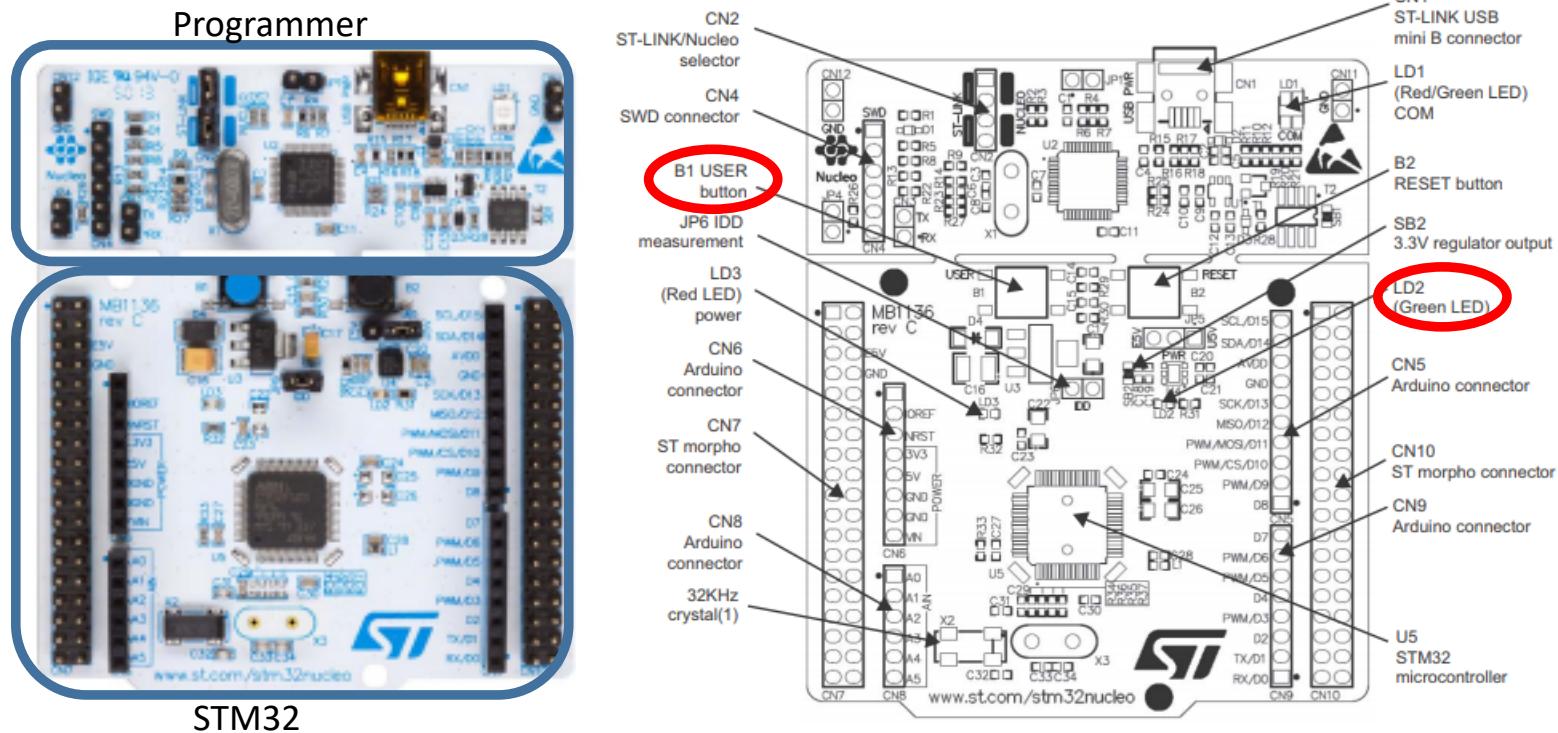


- ARM Cortex-M4 @ 84 MHz
- 512-KB Flash & 96KB SRAM
- 64-Kbyte of CCM (core coupled memory) data RAM
- LCD parallel interface, 8080/6800 modes
- Timer with quadrature (incremental) encoder input
- 5 V-tolerant I/Os
- Parallel camera interface
- True random number generator
- RTC: subsecond accuracy, hardware calendar
- 96-bit unique ID

http://www.st.com/content/st_com/en/products/evaluation-tools/product-evaluation-tools/mcu-eval-tools/stm32-mcu-eval-tools/stm32-mcu-nucleo/nucleo-f401re.html#design-scroll

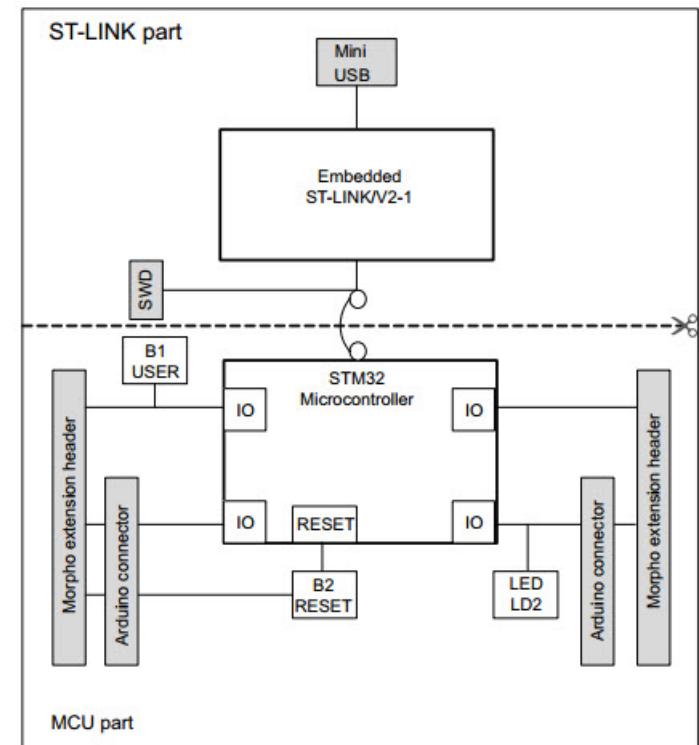
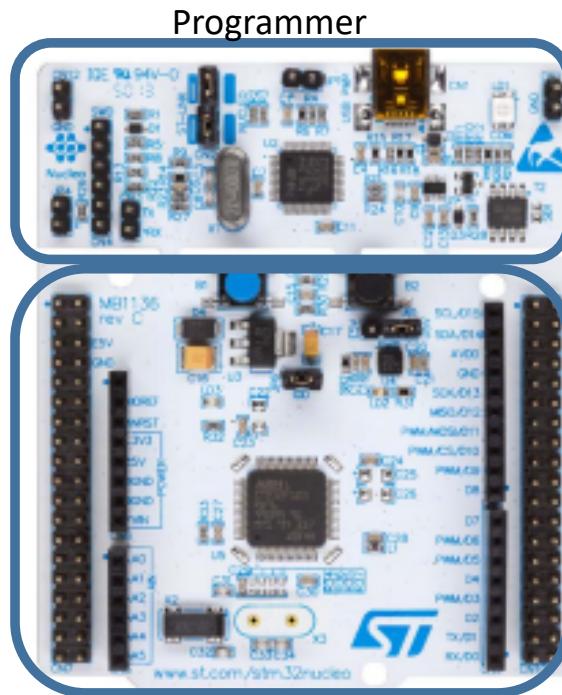
STM32F401RE MCU

Zooming in

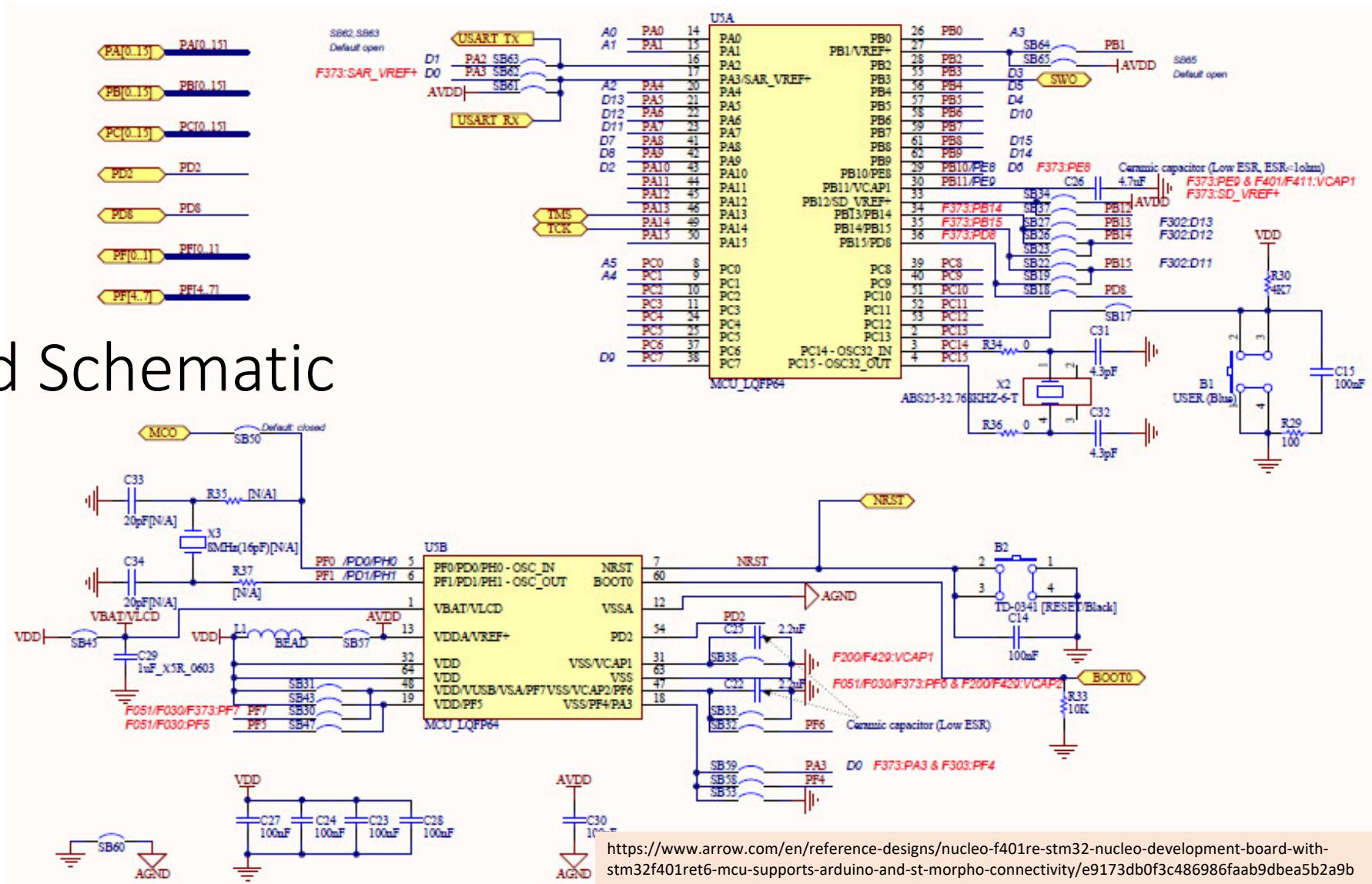


System View

- The ST-LINK allows us to program the board

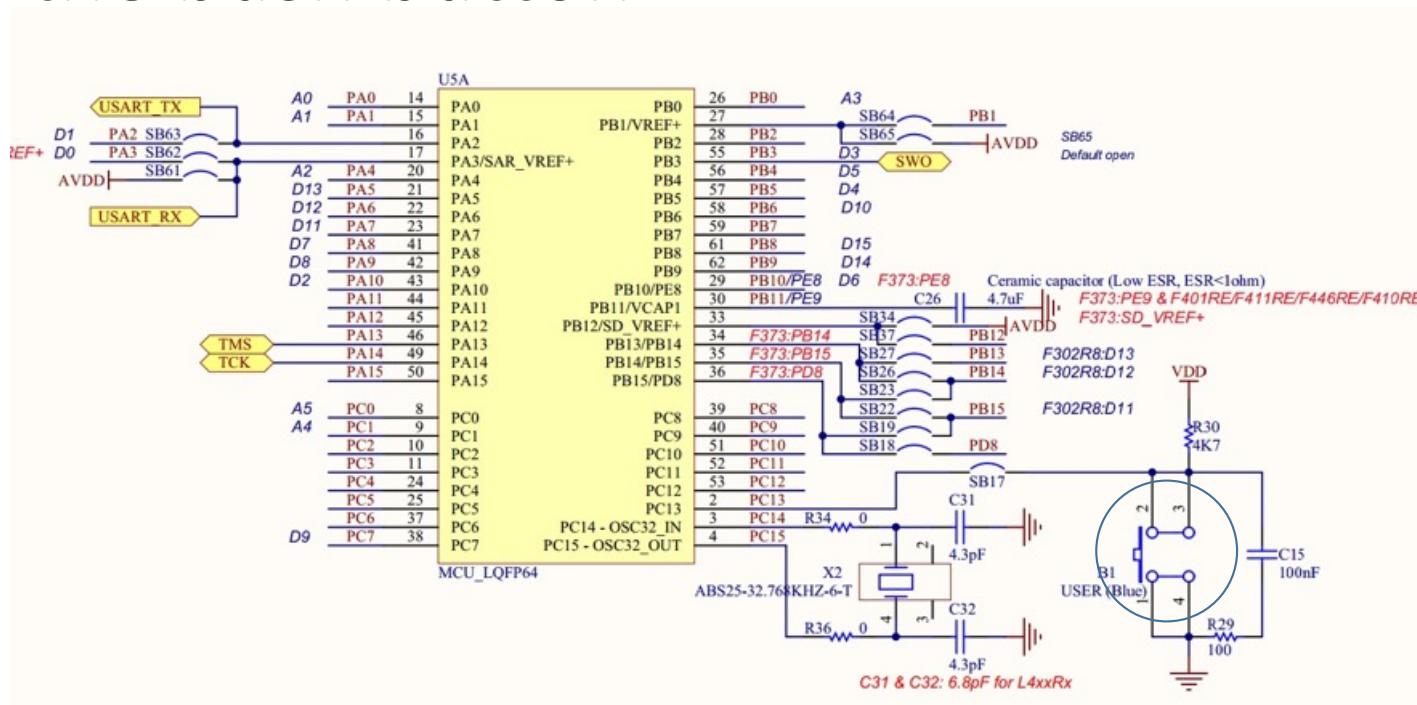


Board Schematic



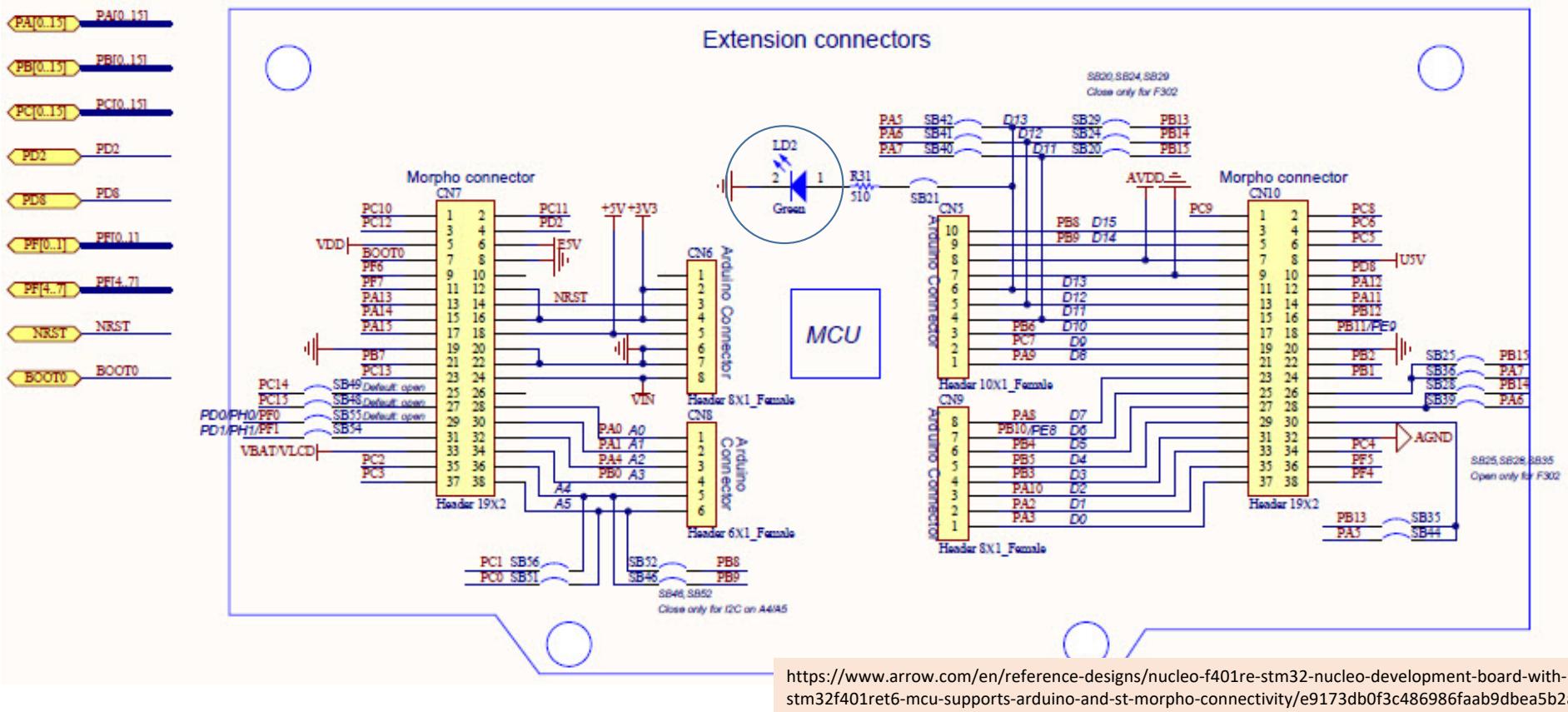
Board Schematic

Use the bush button



<https://www.arrow.com/en/reference-designs/nucleo-f401re-stm32-nucleo-development-board-with-stm32f401ret6-mcu-supports-arduino-and-st-morpho-connectivity/e9173db0f3c486986faab9dbea5b2a9b>

Board Schematic



Answer the following questions:

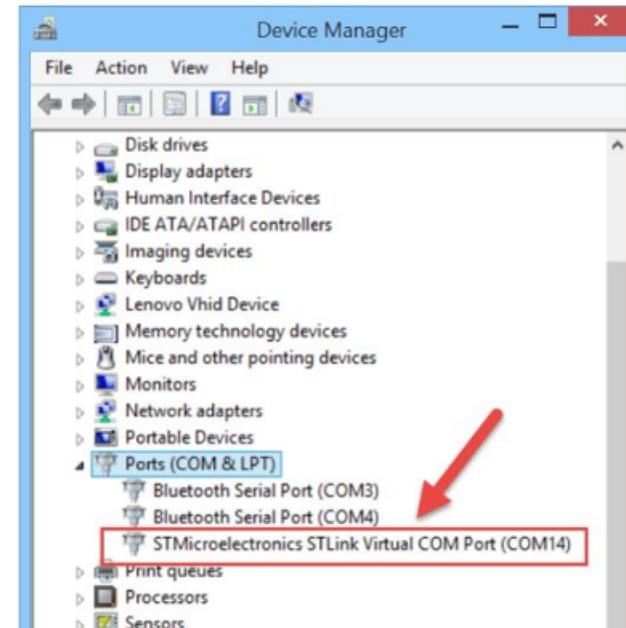
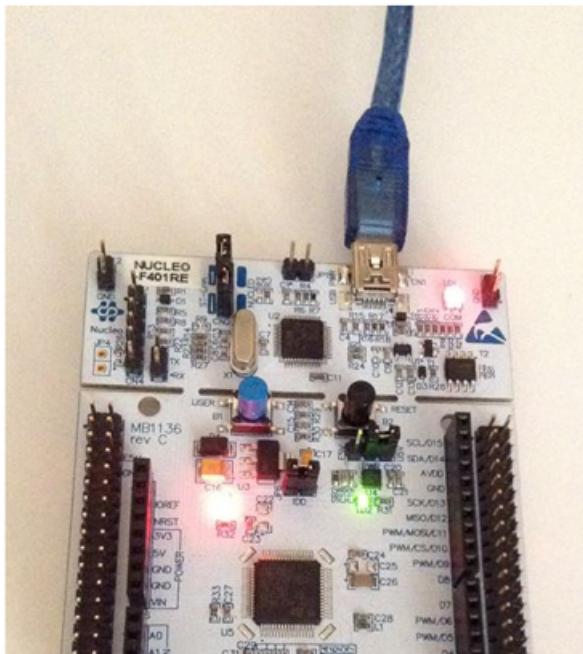
You need to use the board schematic. Make sure you have the board.

1. Can the Nueclo board operate with 3.3V supply?
2. How many LEDs the board have? What are the Labels?
3. How many jumpers does the board have? How are they labeled?
4. Does the board outputs 3.3V?
5. How many buttons are on the board? How are they labeled?
6. How many oscillators are on the board? How are they labeled?
7. What types of package the STM chip uses?
8. How many pins does the MCU have?
9. Which port the BUE button is connected to?
10. PC10 is connected to which PIN on the chip?
11. What is the value of the resistor directly connected to the Green LED?
12. Looking at the board, where are SB29 and SB42 and are they closed or open?
13. Which IO port LD2 is connected to?
14. How many pins (in total) are on each Morpho connector?
15. Which PIN on the Morph connector is connected to PC10? What is the reference designator for this Morpho connector?
16. On your board is JP6 placed? What is it used for (see the schematic)?
17. How many PORTS are available on the board (e.g., Px) ? How many bits does each PORT have?

Checking your Connection

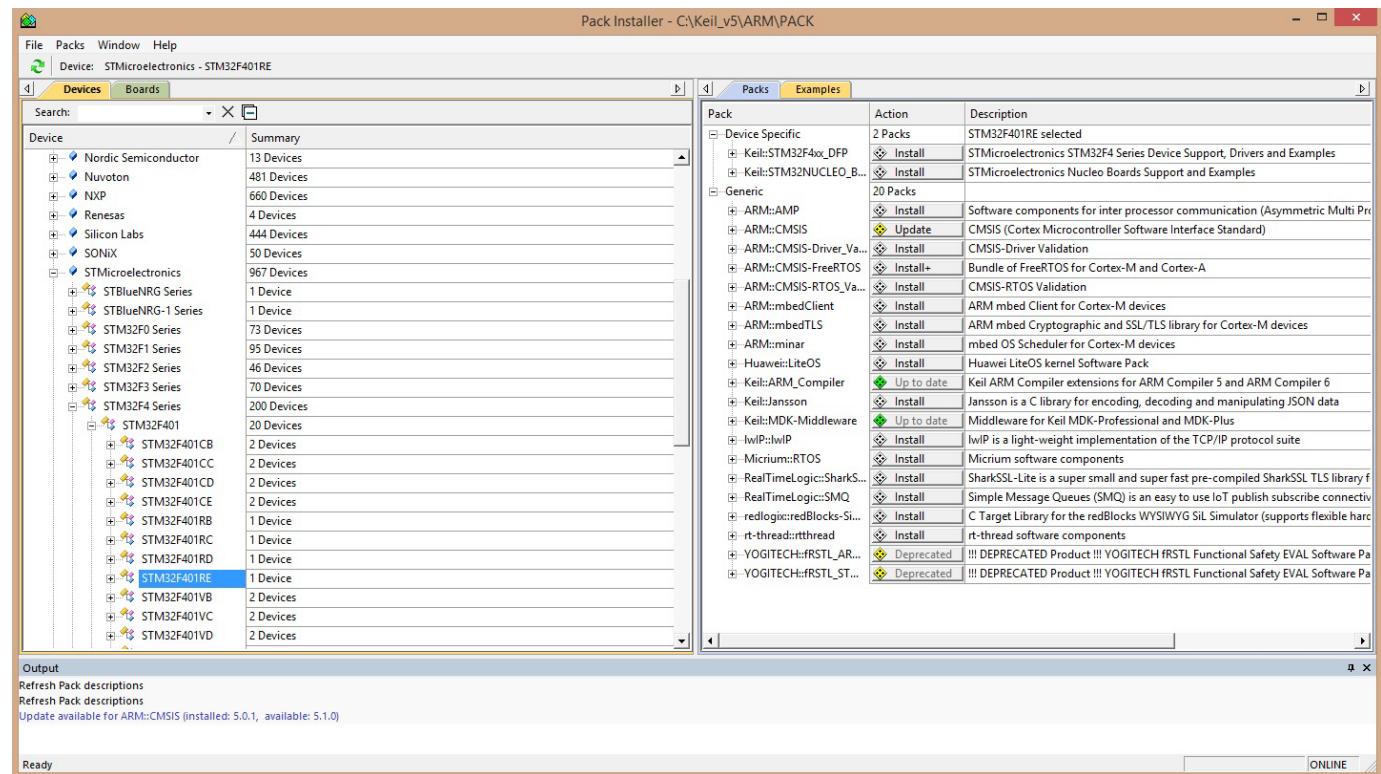
Make sure your computer sees the board

If not make sure you have downloaded the driver – just double click on the provided driver file.



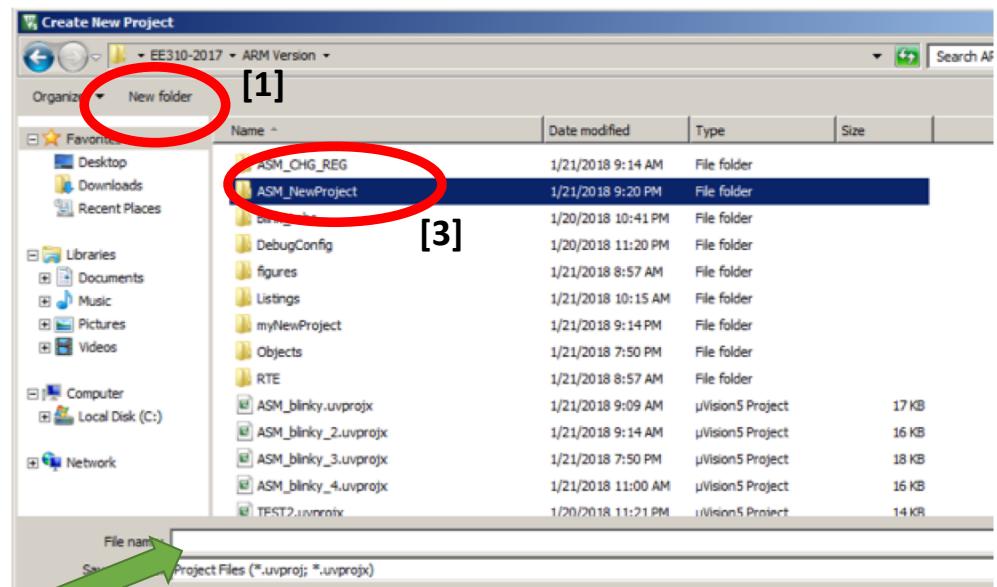
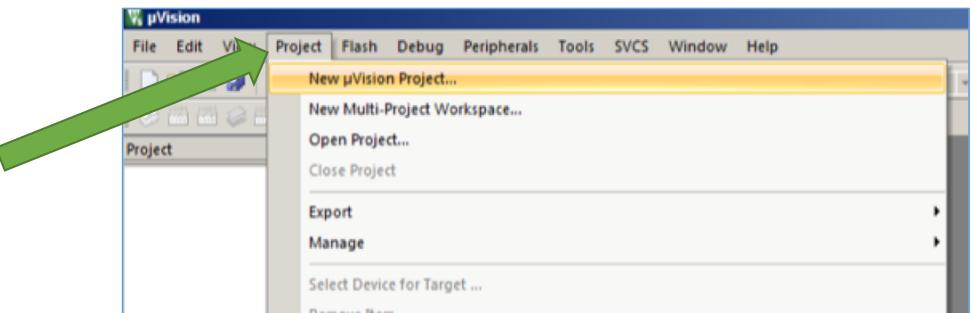
Start a NEW project in your Keil

- On the left side of the window under Devices, click to expand STMicroelectronics.
- Find STM32F4 Series and expand it
- Double click on STM32F401RE and it should add it to the packs list on the right side.
- Click all the buttons that say Install or Update to get all the features/packages that Keil offers.



Start A New Project

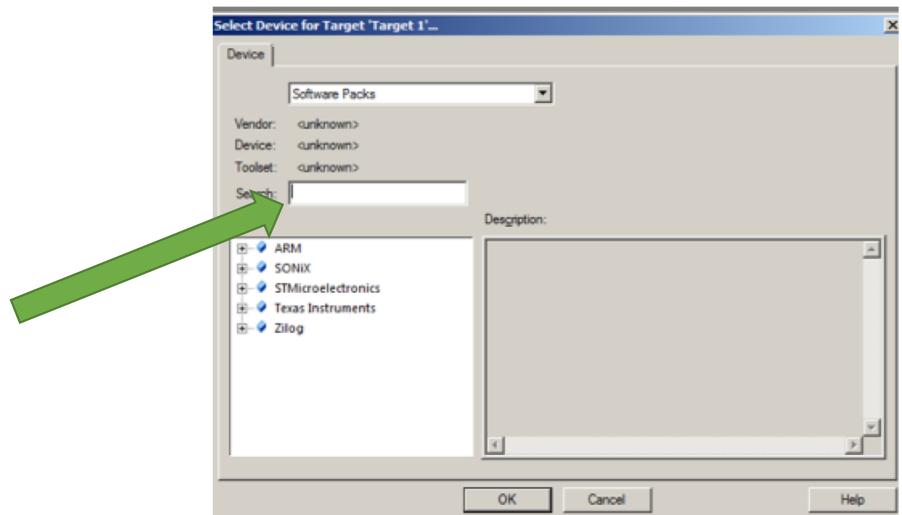
- Open Keil uVision 5 from your Windows
- Make sure all projects are closed:
 - Project → Close Project
- Click on Project → new uVision Project ...
- In the window click on New Folder [1]
- Type the name of the project:
ASM_NewProject [2]
- Go to the new folder that you just created [3]



[2]

Start A New Project-2

- The Select Device window pops up
- In the search type: STM32F401RETx
- Click on the device (STM32F401RETx) that showed up and then click OK



Start A New Project-3

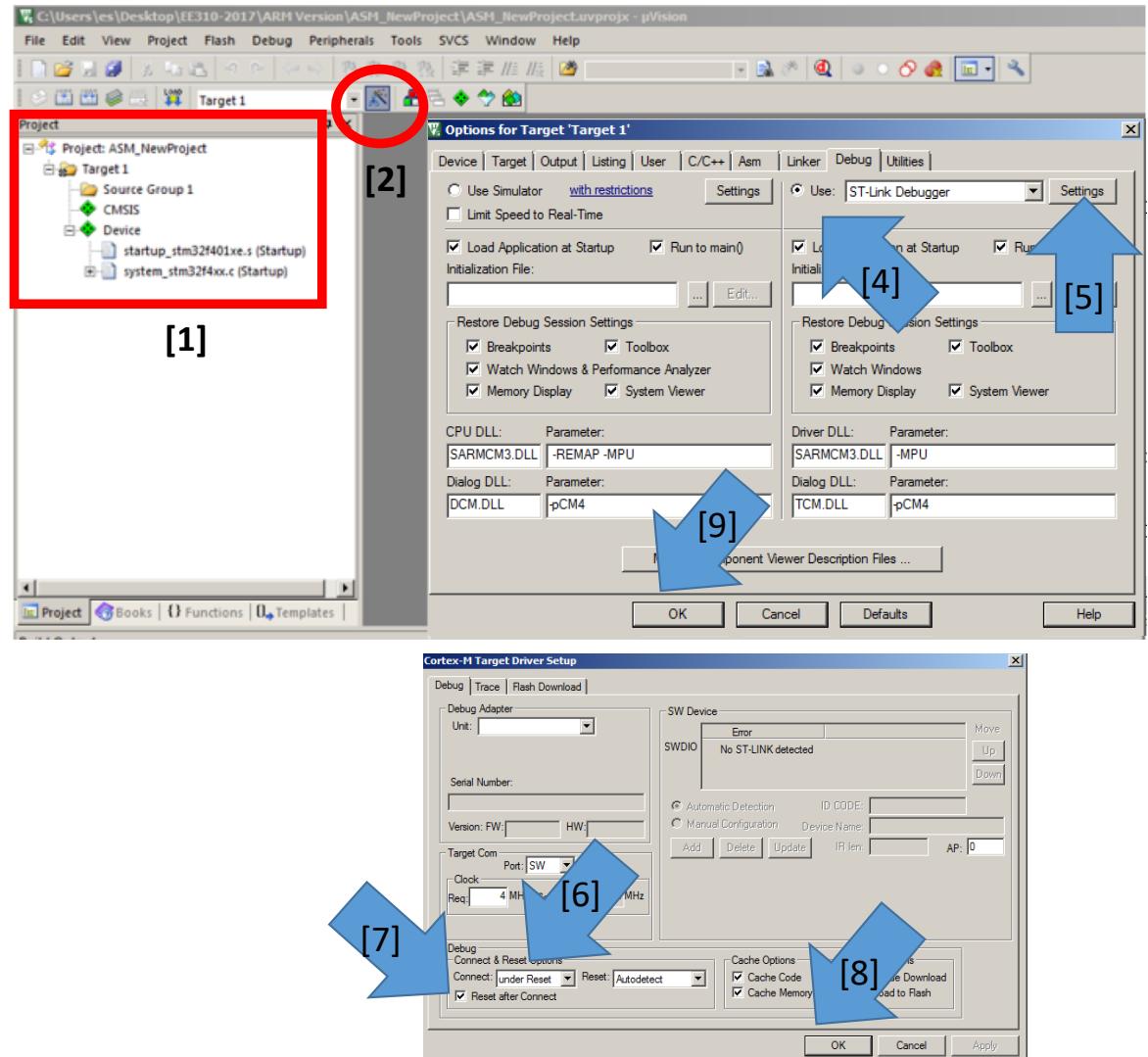
- At this point the Manage Run window pops up
- CHECK CMSIS → CORE and DEVICE → STARTUP boxes – note that the color changes
- Then click OK

If Devices Are Marked as X

Manage Run-Time Environment				
Software Component	Sel.	Variant	Version	Description
+ AMP				Asymmetric Multiprocessing
+ Board Support		STM32F429I-Discove	1.0.0	STMicroelectronics STM32F429I-Discovery Kit
+ CMSIS				Cortex Microcontroller Software Interface Components
CORE	<input checked="" type="checkbox"/>		5.0.2	CMSIS-CORE for Cortex-M_SC000, SC300, ARMv8-M
DSP	<input type="checkbox"/>		1.5.2	CMSIS-DSP Library for Cortex-M_SC000, and SC300
+ RTOS (API)			1.0.0	CMSIS-RTOS API for Cortex-M_SC000, and SC300
+ RTOS2 (API)			2.1.1	CMSIS-RTOS API for Cortex-M_SC000, and SC300
+ CMSIS Driver				Unified Device Drivers compliant to CMSIS-Driver Specifications
+ CMSIS Driver Validation				Run API test for enabled drivers
+ CMSIS RTOS Validation				CMSIS-RTOS Validation Suite
+ Compiler				Compiler Extensions for ARM Compiler 5 and ARM Compiler 6
+ CycloneCommon				Common Files
+ CycloneCrypto				Cryptographic Library
+ CycloneSSL				SSL/TLS Library
+ CycloneTCP				Dual IPv4/IPv6 Stack
+ Data Exchange				Software Components for Data Exchange
+ Device				Startup, System Setup
Startup	<input checked="" type="checkbox"/>		2.6.0	System Startup for STMicroelectronics STM32F4 Series
+ STM32Cube Framework (API)			1.0.0	STM32Cube Framework
+ STM32Cube HAL				STM32F4xx Hardware Abstraction Layer (HAL) Drivers
+ File System				File Access on various storage devices
+ Graphics				User Interface on graphical LCD displays
+ Graphics Display				Display Interface including configuration for emWIN
+ Network				Network lwIP Bundle
+ RTOS				
+ SMQ				
+ Security				
+ SharkSSL				
+ Simulation				Components used for simulation purposes
+ USB				USB Communication with various device classes
+ mbed				
+ wolfSSL			3.9.0	wolfSSL: SSL/TLS and Crypt Library

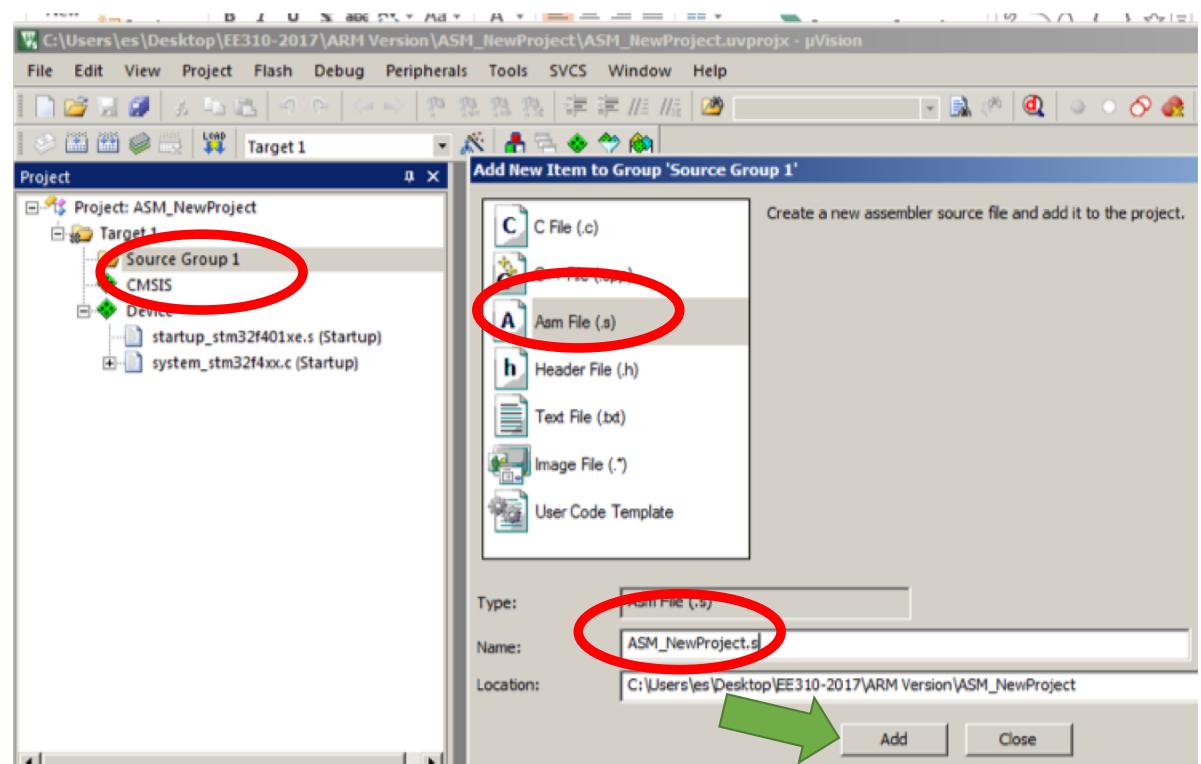
Start A New Project-4

- At this point your Project window should show the CMSIS and DEVICE – these are the CMSIS system initialization files and startup files in order for the RUN TIME to operate [1]
- Click on the OPTIONS FOR TARGET [2]
- Click on DEBUG tab [3]
- CHECK and Select the USE ST-LINK [4]
- Click on Setting [5]**
- In the CORTEX-M4 Target Drive
 - Choose Under Reset [6]
 - Check RESET AFTER CONTROL Box [7]
 - Click ok [8]
- In the OPTIONS window, Click on OK [9]
- In the PROJECT window click on Source Group 1 and select ADD NEW ITEM....
- In the new window select



Start A New Project-5

- In the PROJECT window click on Source Group 1 and select ADD NEW ITEM....
- In the new window select ASM files
- Then type the name of the ASM file: ASM_NewProject.s
- Click on ADD



Start A New Project

- In the project window click on ASM_NewProject.s under the Source Group 1 folder
- Open the file
- Type in the following code
- Save the file
- Press F7 and make sure there are no errors (as shown below) – Note: If you do not see the TIME ELAPSED then you need to make sure you BUILD ALL

```
assembling startup_stm32f401xe.s...compiling
system_stm32f4xx.c...linking...Program Size: Code=300 RO-data=404
RW-data=0 ZI-data=1536 ".\Objects\workingProject.axf" - 0 Error(s),
0 Warning(s).Build Time Elapsed: 00:00:01
```

- Under DEBUS tab → Click on the START/STOP DEBUG SESSION
- Under DEBUS tab → Click on RUN

```
Running with Code Size Limit: 32KLoad
"C:\Users\es\Desktop\EE310-
2017\Objects\workingProject.axf" *** Restricted Version with
32768 Byte Code Size Limit*** Currently used: 704 Bytes (2%)
```



```
1 ;*****
2 ; Program ASM_NewProject
3 ; Changing registers (STM32F401RE)
4 ; Written By:
5 ; Date: 8/25/17
6 ;*****
7 ; Constants
8 P EQU 0x4
9 Q EQU 0x3
10 R EQU 0x0
11 ;*****
12 ; Program Section
13 ;*****
14 ;LABEL DIRECTIVE VALUE COMMENT
15 AREA main, CODE, READONLY
16 THUMB
17 EXPORT __main
18
19 __main
20     MOV r0,#4      ;load 4 into r0
21     MOV r1,#P      ;load 5 into r1
22     ADD r2,r0,r1  ;add r0 to r1 and put the result in r2
23
24 __loop
25     B __loop
26
27 END             ;end of program
```

Make sure you change
your STARTUP file

UNCOMMENT these lines (192 & 193):

The screenshot shows the MDK-ARM IDE interface with several windows open:

- Project:** Shows the project structure with "Mann Test" selected.
- Disassembly:** Shows assembly code for address 0x08000192. Lines 192 and 193 are highlighted with a green box and labeled "UNCOMMENT these lines (192 & 193)".

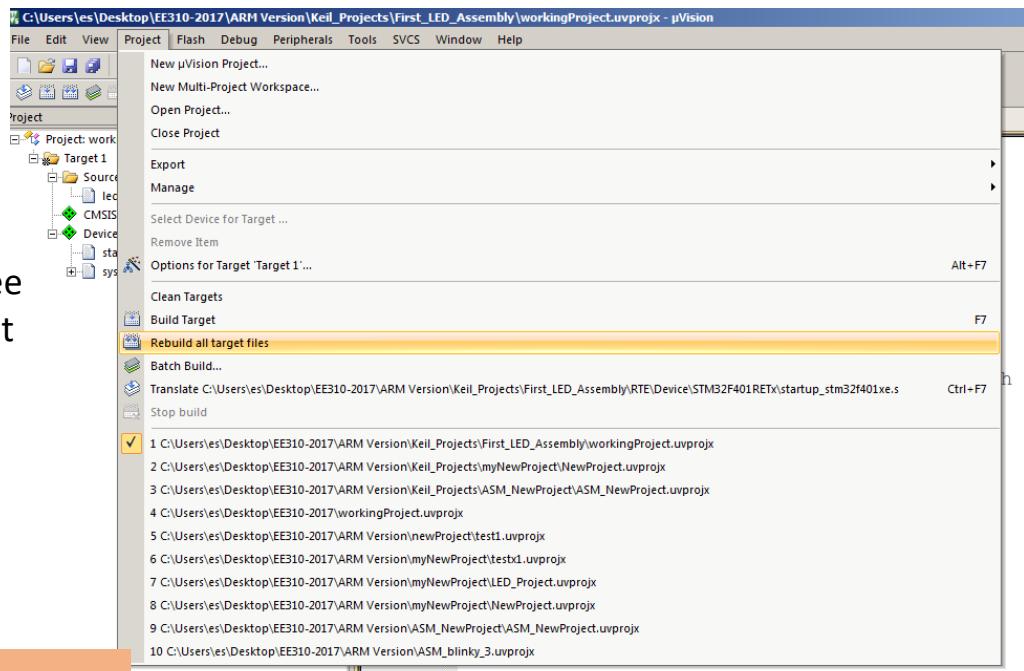
```
0x08000192 0800 DCW 0x0800
194:          LDR R0, =_main
195:          LDR r0,[pc,#32] ; @0x080001B8
196:          BX R0
197:          ENDP
198: ; Dummy Exception Handlers (infinite loops which can be modified)
199:
200: NMI_Handler PROC
201: EXPORT NMI_Handler [WEAK]
202: B .
203: ENDP
204: HardFault_Handler\ PROC
205:
```
- Text Editor:** Shows the source code for "startup_stm32f401xe.s". Lines 192 and 193 are also highlighted with a green box.

```
189 IMPORT SystemInit
190 IMPORT __main
191
192 ;LDR R0, =SystemInit
193 ;BLX R0
194 LDR R0, =__main
195 BX R0
196 ENDP
197
198 ; Dummy Exception Handlers (infinite loops which can be modified)
199
200 NMI_Handler PROC
201 EXPORT NMI_Handler [WEAK]
202 B .
203 ENDP
204 HardFault_Handler\ PROC
205
```
- Command:** Shows the command line interface with the current working directory set to "C:\Users\Manthan Gajjar\Documents\MDK\Boards\STM32F401\Projects\Mann Test\Target 1\Source Group 1\test-ICL0.s". It displays memory usage information:

```
Load "C:\Users\Manthan Gajjar\Documents\MDK\Boards\STM32F401\Projects\Mann Test\Target 1\Source Group 1\test-ICL0.s"
*** Restricted Version with 32768 Byte Code Size Limit
*** Currently used: 594 Bytes (1%)
```
- Memory 1:** Shows a memory dump window with an empty address field.
- Simulation:** Shows simulation status with time 0.0000.

IMPORTANT: Build All Target Files

Also, make sure you build the entire project. You should see Build Time Elapsed to ensure the project was properly built (see next slide):



```
assembling startup_stm32f401xe.s...compiling
system_stm32f4xx.c...linking...Program Size: Code=300 RO-
data=404 RW-data=0 ZI-data=1536  ".\Objects\workingProject.axf"
- 0 Error(s), 0 Warning(s).Build Time Elapsed:  00:00:01- 0 \
```

Modify the program....

1. Change the delay value and see how the blinking LED changes
2. In Keil, which the program is running, click on System View button and see how the registers change
3. Change the program such that the LED ON time is much shorter than the OFF time – in this case we are creating different delays for ON and OFF

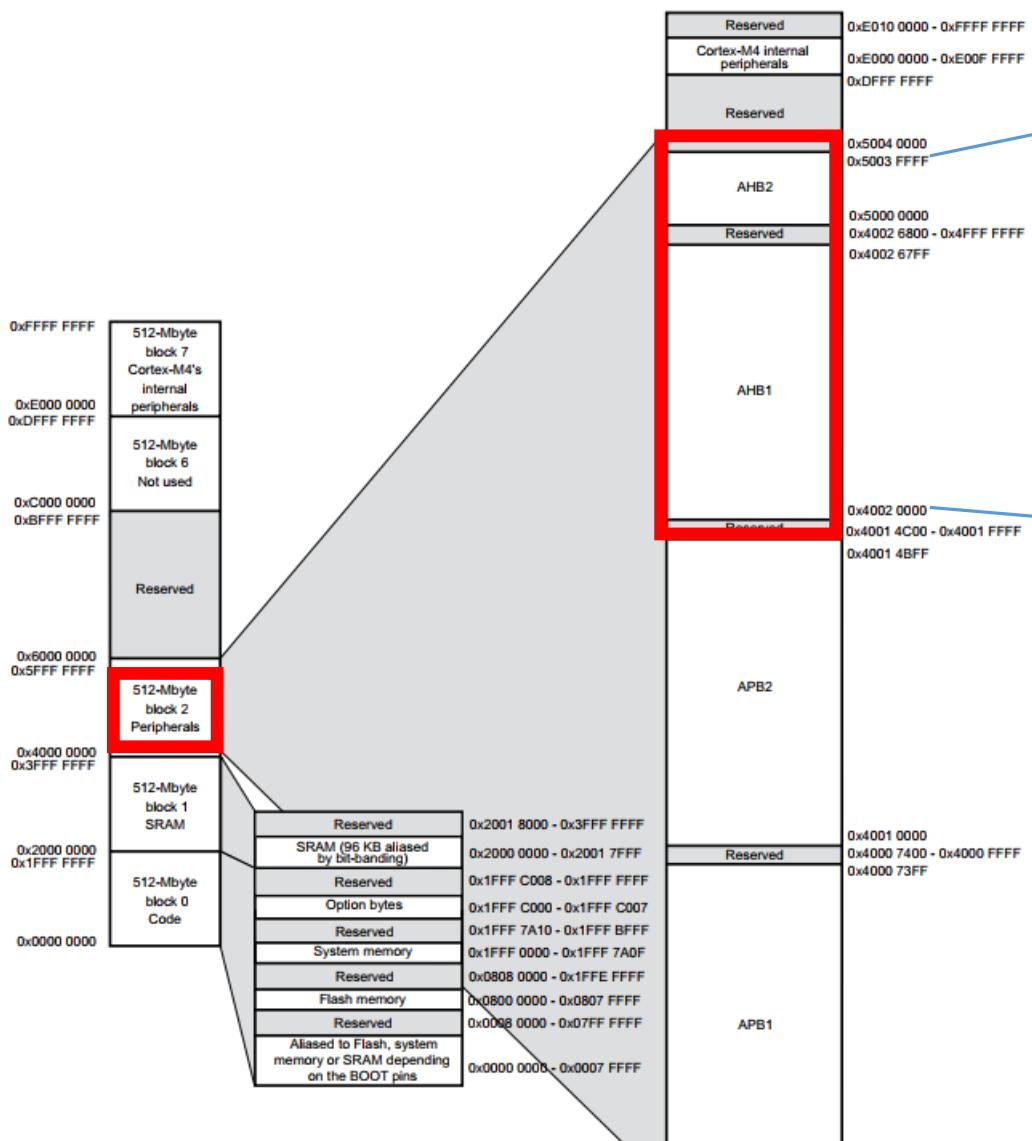


Table 1. STM32F411xC/E register boundary addresses

Boundary address	Peripheral	Bus	Register map
0x5000 0000 - 0x5003 FFFF	USB OTG FS	AHB2	Section 22.16.6: OTG_FS register map on page 744
0x4002 6400 - 0x4002 67FF	DMA2		Section 9.5.11: DMA register map on page 194
0x4002 6000 - 0x4002 63FF	DMA1		
0x4002 3C00 - 0x4002 3FFF	Flash interface register		Section 3.8: Flash interface registers on page 58
0x4002 3800 - 0x4002 3BFF	RCC		Section 6.3.22: RCC register map on page 133
0x4002 3000 - 0x4002 33FF	CRC		
0x4002 1C00 - 0x4002 1FFF	GPIOH		Section 4.4.4: CRC register map on page 68
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		
0x4002 0400 - 0x4002 07FF	GPIOB	AHB1	
0x4002 0000 - 0x4002 03FF	GPIOA		
0x4001 4C00 - 0x4001 FFFF			Section 8.4.11: GPIO register map on page 160
0x4001 4BFF			

Programming GPIO Ports in STM32F411xC/E

- Table 1 gives the boundary addresses of the **peripherals**
- See Section 2.3 of the Reference Manual

Table 1. STM32F411xC/E register boundary addresses

Boundary address	Peripheral	Bus	Register map
0x5000 0000 - 0x5003 FFFF	USB OTG FS	AHB2	Section 22.16.6: OTG_FS register map on page 744
0x4002 6400 - 0x4002 67FF	DMA2		Section 9.5.11: DMA register map on page 194
0x4002 6000 - 0x4002 63FF	DMA1		Section 3.8: Flash interface registers on page 58
0x4002 3C00 - 0x4002 3FFF	Flash interface register		Section 6.3.22: RCC register map on page 133
0x4002 3800 - 0x4002 3BFF	RCC		Section 4.4.4: CRC register map on page 68
0x4002 3000 - 0x4002 33FF	CRC		
0x4002 1C00 - 0x4002 1FFF	GPIOH		
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		
0x4002 0400 - 0x4002 07FF	GPIOB		
0x4002 0000 - 0x4002 03FF	GPIOA		

Click on the Reference Manual to go to the appropriate page

Table 26. GPIO register map and reset values

Offset	Register	Reset value	0x00	GPIOA_MODER	31
			0x00	Reset value	0 MODER[15:1:0]
			0x00	GPIOB_MODER	30
			0x00	Reset value	0 MODER[15:1:0]
			0x00	GPIOx_MODER (where x = C..E and H)	29
			0x00	Reset value	0 MODER[14:1:0]
			0x04	GPIOx_OTYPER (where x = A..E and H)	28
			0x04	Reset value	0 MODER[14:1:0]
			0x08	GPIOx_OSPEEDER (where x = C..E andH)	27
			0x08	Reset value	0 MODER[13:1:0]
			0x08	GPIOA_OSPEEDER	26
			0x08	Reset value	0 MODER[13:1:0]
			0x08	GPIOB_OSPEEDER	25
			0x08	Reset value	0 MODER[12:1:0]
			0x08	GPIOA_OSPEEDER	24
			0x08	Reset value	0 MODER[11:1:0]
			0x08	GPIOB_OSPEEDER	23
			0x08	Reset value	0 MODER[11:1:0]
			0x08	GPIOA_OSPEEDER	22
			0x08	Reset value	0 MODER[10:1:0]
			0x08	GPIOB_OSPEEDER	21
			0x08	Reset value	0 MODER[10:1:0]
			0x08	GPIOA_OSPEEDER	20
			0x08	Reset value	0 MODER[9:1:0]
			0x08	GPIOB_OSPEEDER	19
			0x08	Reset value	0 MODER[9:1:0]
			0x08	GPIOA_OSPEEDER	18
			0x08	Reset value	0 MODER[8:1:0]
			0x08	GPIOB_OSPEEDER	17
			0x08	Reset value	0 MODER[8:1:0]
			0x08	GPIOA_OSPEEDER	16
			0x08	Reset value	0 MODER[7:1:0]
			0x08	GPIOB_OSPEEDER	15
			0x08	Reset value	0 MODER[7:1:0]
			0x08	GPIOA_OSPEEDER	14
			0x08	Reset value	0 MODER[6:1:0]
			0x08	GPIOB_OSPEEDER	13
			0x08	Reset value	0 MODER[6:1:0]
			0x08	GPIOA_OSPEEDER	12
			0x08	Reset value	0 MODER[5:1:0]
			0x08	GPIOB_OSPEEDER	11
			0x08	Reset value	0 MODER[5:1:0]
			0x08	GPIOA_OSPEEDER	10
			0x08	Reset value	0 MODER[4:1:0]
			0x08	GPIOB_OSPEEDER	9
			0x08	Reset value	0 MODER[4:1:0]
			0x08	GPIOA_OSPEEDER	8
			0x08	Reset value	0 MODER[3:1:0]
			0x08	GPIOB_OSPEEDER	7
			0x08	Reset value	0 MODER[3:1:0]
			0x08	GPIOA_OSPEEDER	6
			0x08	Reset value	0 MODER[2:1:0]
			0x08	GPIOB_OSPEEDER	5
			0x08	Reset value	0 MODER[2:1:0]
			0x08	GPIOA_OSPEEDER	4
			0x08	Reset value	0 MODER[1:1:0]
			0x08	GPIOB_OSPEEDER	3
			0x08	Reset value	0 MODER[1:1:0]
			0x08	GPIOA_OSPEEDER	2
			0x08	Reset value	0 MODER[0:1:0]
			0x08	GPIOB_OSPEEDER	1
			0x08	Reset value	0 MODER[0:1:0]

Table 1. STM32F411xC/E register boundary addresses

Boundary address	Peripheral	Bus	Register map
0x5000 0000 - 0x5003 FFFF	USB OTG FS	AHB2	Section 22.16.6: OTG_FS register map on page 744
0x4002 6400 - 0x4002 67FF	DMA2	AHB1	Section 9.5.11: DMA register map on page 194
0x4002 6000 - 0x4002 63FF	DMA1		Section 3.8: Flash interface registers on page 58
0x4002 3C00 - 0x4002 3FFF	Flash interface register		Section 6.3.22: RCC register map on page 133
0x4002 3800 - 0x4002 3BFF	RCC		Section 4.4.4: CRC register map on page 68
0x4002 3000 - 0x4002 33FF	CRC		
0x4002 1C00 - 0x4002 1FFF	GPIOH		
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		Section 8.4.11: GPIO register map on page 160
0x4002 0400 - 0x4002 07FF	GPIOB		
0x4002 0000 - 0x4002 03FF	GPIOA		

Offset:

0x4002 0000 + 0x08 = GPIOA_OSPEEDER

Table 26. GPIO register map and reset values

Offset	Register	Field	Description	Type	Reset value
0x00	GPIOA_MODER	MODER[15:1]	Mode for Pin 15	Mod	31
		MODER[15:1]	Mode for Pin 15	Mod	30
0x00	GPIOB_MODER	MODER[15:1]	Mode for Pin 15	Mod	29
		MODER[14:1]	Mode for Pin 14	Mod	28
0x00	GPIOx_MODER (where x = C..E and H)	MODER[13:1]	Mode for Pin 13	Mod	27
		MODER[13:1]	Mode for Pin 13	Mod	26
0x04	GPIOx_OTYPER (where x = A..E and H)	MODER[12:1]	Mode for Pin 12	Mod	25
		MODER[11:1]	Mode for Pin 11	Mod	24
0x08	GPIOx_OSPEEDER (where x = C..E andH)	MODER[10:1]	Mode for Pin 10	Mod	23
		MODER[10:1]	Mode for Pin 10	Mod	22
0x08	GPIOA_OSPEEDER	OSPEEDR[15:1]	OSPEEDR[15:1]	Mod	21
		OSPEEDR[14:1]	OSPEEDR[14:1]	Mod	20
0x08	GPIOB_OSPEEDER	OSPEEDR[13:1]	OSPEEDR[13:1]	Mod	19
		OSPEEDR[12:1]	OSPEEDR[12:1]	Mod	18
0x08	OSPEEDR[11:1]	OSPEEDR[11:1]	OSPEEDR[11:1]	Mod	17
		OSPEEDR[10:1]	OSPEEDR[10:1]	Mod	16
0x08	OSPEEDR[9:1]	OSPEEDR[9:1]	OSPEEDR[9:1]	Mod	15
		OSPEEDR[8:1]	OSPEEDR[8:1]	Mod	14
0x08	OSPEEDR[7:1]	OSPEEDR[7:1]	OSPEEDR[7:1]	Mod	13
		OSPEEDR[6:1]	OSPEEDR[6:1]	Mod	12
0x08	OSPEEDR[5:1]	OSPEEDR[5:1]	OSPEEDR[5:1]	Mod	11
		OSPEEDR[4:1]	OSPEEDR[4:1]	Mod	10
0x08	OSPEEDR[3:1]	OSPEEDR[3:1]	OSPEEDR[3:1]	Mod	9
		OSPEEDR[2:1]	OSPEEDR[2:1]	Mod	8
0x08	OSPEEDR[1:1]	OSPEEDR[1:1]	OSPEEDR[1:1]	Mod	7
		OSPEEDR[0:1]	OSPEEDR[0:1]	Mod	6

Table 1. STM32F411xC/E register boundary addresses

Boundary address	Peripheral	Bus	Register map
0x5000 0000 - 0x5003 FFFF	USB OTG FS	AHB2	Section 22.16.6: OTG_FS register map on page 744
0x4002 6400 - 0x4002 67FF	DMA2	AHB1	Section 9.5.11: DMA register map on page 194
0x4002 6000 - 0x4002 63FF	DMA1		Section 3.8: Flash interface registers on page 58
0x4002 3C00 - 0x4002 3FFF	Flash interface register		Section 6.3.22: RCC register map on page 133
0x4002 3800 - 0x4002 3BFF	RCC		Section 4.4.4: CRC register map on page 68
0x4002 3000 - 0x4002 33FF	CRC		Section 8.4.11: GPIO register map on page 160
0x4002 1C00 - 0x4002 1FFF	GPIOH		
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		
0x4002 0400 - 0x4002 07FF	GPIOB		
0x4002 0000 - 0x4002 03FF	GPIOA		

More Examples

RCC_AHB1ENR
GPIOA_MODER
GPIOA_ODR

EQU
EQU
EQU

0x40023830
0x40020000
0x40020014

Configure the GPIOA

STEP 1: Setting up the GPIOx (A-H)

- The STM32F401RETx microcontroller on the Nucleo board has several general purpose input/output (GPIO) ports named Port A, B, and C. Each port can have up to 16 pins, and each port has associated with it the following set of registers:
- **GPIO port mode register (GPIOx_MODER) – port mode (In/Out/Analog In)**
- **GPIO port output type register (GPIOx_OTYPER)**
- **GPIO port output speed register (GPIOx_OSPEEDR)**
- **GPIO port pull-up/pull-down register (GPIOx_PUPDR)**
- **GPIO port input data register (GPIOx_IDR) – In INPUT then what is INPUT state?**
- **GPIO port output data register (GPIOx_ODR) – If OUTPUT what is the port OUTPUT state (H/L)**
- **GPIO port bit set/reset register (GPIOx_BSRR)**
- **GPIO port configuration lock register (GPIOx_LCKR)**
- **GPIO alternate function low register (GPIOx_AFRL)**
- **GPIO alternate function high register (GPIOx_AFRH)**
- **GPIO Port bit reset register (GPIOx_BRR)**

See Reference Manual
Section 8.4 GPIO registers

where the 'x' in each register name acronym represents the port i.e. the GPIOx_MODER associated with port A is called GPIOA_MODER.

GPIO register map and reset values (Reference Manual)

- '00'-> input mode, which allows the GPIO pin to be used as an input pin,
- '01'-> Output mode, which allows the GPIO pin to be used as an output pin,
- '11'-> Analog mode, which allows the GPIO pin to be used as an Analog input pin and finally,
- '10'-> Alternate function mode which allow the GPIO pins to be used by peripherals such as the UART, SPI e.t.c. It is important to note that if a pin's MODE is set to alternate function, any GPIO settings for that pin in the GPIO registers will be overridden by the peripheral.

16 PINS on the PORTS

See Table 26

GPIO register map and reset values (Reference Manual)

- '00'-> input mode, which allows the GPIO pin to be used as an input pin,
- '01'-> Output mode, which allows the GPIO pin to be used as an output pin,
- '11'-> Analog mode, which allows the GPIO pin to be used as an Analog input pin and finally,
- '10'-> Alternate function mode which allow the GPIO pins to be used by peripherals such as the UART, SPI e.t.c. It is important to note that if a pin's MODE is set to alternate function, any GPIO settings for that pin in the GPIO registers will be overridden by the peripheral.

; Set GPIOA PA05 to be an OUTPUT

LDR r5, = GPIOA MODER

LDR r6, [r5]

```
ORR r6, #0x400 ; 0000 0000 0000 0000 0000 0100 0000 0000  
STR r6, [r5]      ; 15 14 13 12 11 10 9 8  7 6  5 4  3 2  1  0
```

16 PINS on the PORTS

See Table 26

GPIO register map and reset values (Reference Manual)

- '00'-> input mode, which allows the GPIO pin to be used as an input pin,
- '01'-> Output mode, which allows the GPIO pin to be used as an output pin,
- '11'-> Analog mode, which allows the GPIO pin to be used as an Analog input pin and finally,
- '10'-> Alternate function mode which allow the GPIO pins to be used by peripherals such as the UART, SPI e.t.c. It is important to note that if a pin's MODE is set to alternate function, any GPIO settings for that pin in the GPIO registers will be overridden by the peripheral.

```
; Set up Push Button PC13 to be digital input
LDR r5, = GPIOC_MODER
LDR r6, [r5]
AND r6, #0xFFFFCFFF
STR r6, [r5]
```

16 PINS on the PORTS

See Table 26

GPIO register map and reset values (Reference Manual)

```
; Main Program Starts Here:  
Read from IDR to see if button is pressd:  
    LDR r0, [r1]  
    ANDS r0, r0, #0x2000  
    BEQ __pressed ; Z = 1, button pushed  
  
; Load address of GPIOA_ODR into general reg 0  
; Load      0000 0000 0010 0000 --> PA05 = Set high = 0x00020  
; Load      0000 0000 0000 0000 --> PA05 = Set low = 0x0
```

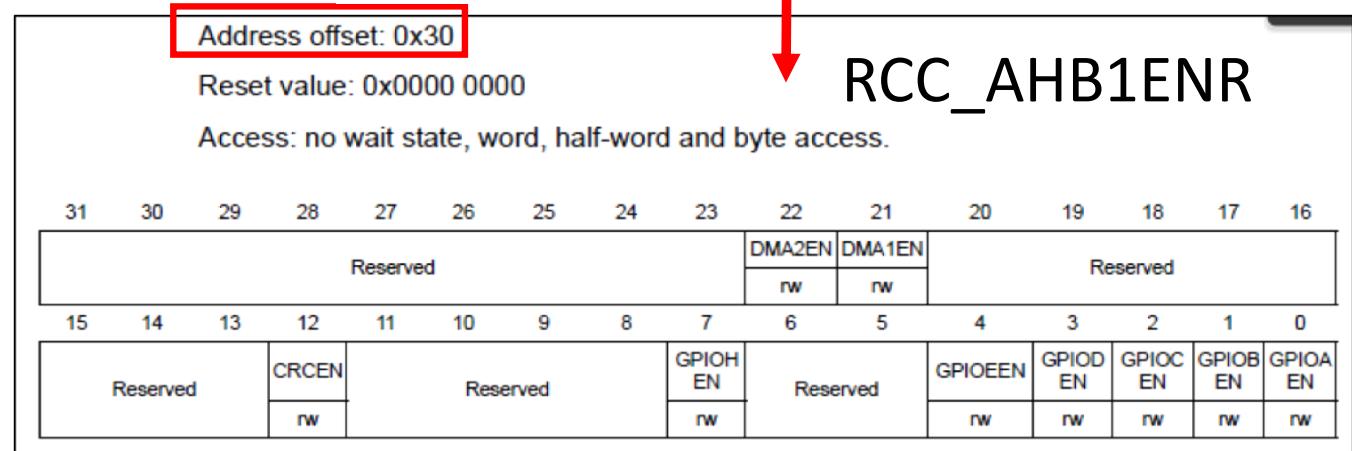
Configure the GPIOA

STEP 2: Clock Setting for GPIOx

- RCC AHB1 peripheral clock enable register (RCC_AHB1ENR)
- Section 6.3.9

Table 1. STM32F411xC/E register boundary addresses

Boundary address	Peripheral	Bus	Register map
0x5000 0000 - 0x5003 FFFF	USB OTG FS	AHB2	Section 22.16.6: OTG_FS register map on page 744
0x4002 6400 - 0x4002 67FF	DMA2		Section 9.5.11: DMA register map on page 194
0x4002 6000 - 0x4002 63FF	DMA1		Section 3.8: Flash interface registers on page 58
0x4002 3C00 - 0x4002 3FFF	Flash interface register		Section 6.3.22: RCC register map on page 133
0x4002 3800 - 0x4002 3BFF	RCC		Section 4.4.4: CRC register map on page 68
0x4002 3000 - 0x4002 33FF	CRC		
0x4002 1C00 - 0x4002 1FFF	GPIOH		
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		
0x4002 0400 - 0x4002 07FF	GPIOB		
0x4002 0000 - 0x4002 03FF	GPIOA		



Configure the GPIOA

STEP 2: Clock Setting for GPIOx

- RCC AHB1 peripheral clock enable register (RCC_AHB1ENR)
- Section 6.3.9
- Loading 0000 0000 0000 0001

Bit 0 GPIOAEN: IO port A clock enable
Set and cleared by software.
0: IO port A clock disabled
1: IO port A clock enabled

```
; Clock Port A and Port C
LDR r5, = RCC_AHB1ENR
LDR r6, [r5]
ORR r6, #0x5
STR r6, [r5]
```

RCC_AHB1ENR																
Address offset: 0x30																
Reset value: 0x0000 0000																
Access: no wait state, word, half-word and byte access.																
31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																
Reserved									DMA2EN	DMA1EN	Reserved				RW	
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0																
Reserved		CRCEN	Reserved			GPIOH EN	Reserved		GPIOEEN	GPIOD EN	GPIOC EN	GPIOB EN	GPIOA EN	RW		
		rw				rw			rw	rw	rw	rw	rw	rw		

Answer the following questions:

You need to use the reference manual.

1. What is the address of GPIOB_OTYPER?
2. What is the address of RCC_AHB1ENR?
3. If the value of register 0x4002 0414 is 0x2 what does it mean?
4. Referring to Figure 16, what does 5V tolerant mean?
5. Referring to Table 23, if PUPDR & MODER are both set to 11, what is the I/O configuration?
6. Referring to Table 23, what are PP and PU?

Looking At The Peripherals

The screenshot shows the IAR Embedded Workbench interface. The assembly code window displays the following sequence of instructions:

```
33 LDR r6, [r5] ; Read
34 ORR r6, #0x5 ; Load
35 STR r6, [r5] ; store
36
37 ;; Set GPIOA PA05 to be an OUTPUT
38 LDR r5, =GPIOA_MODER ; Load
39 LDR r6, [r5] ; Read
40 ORR r6, #0x400 ; Friend
41 STR r6, [r5] ; store
42
```

An orange arrow points from the assembly code window to the right-hand peripheral register viewer. The register viewer is titled "GPIOA" and shows the following properties and their values:

Property	Value
-MODER	0xA8000400
-OTYPE	0
-OSPEEDR	0x0C000000
-PUPDR	0x64000000
-IDR	0x00008028
-ODR	0x00000020
-BSRR	0
-LCKR	0
-AFRL	0
-AFRH	0

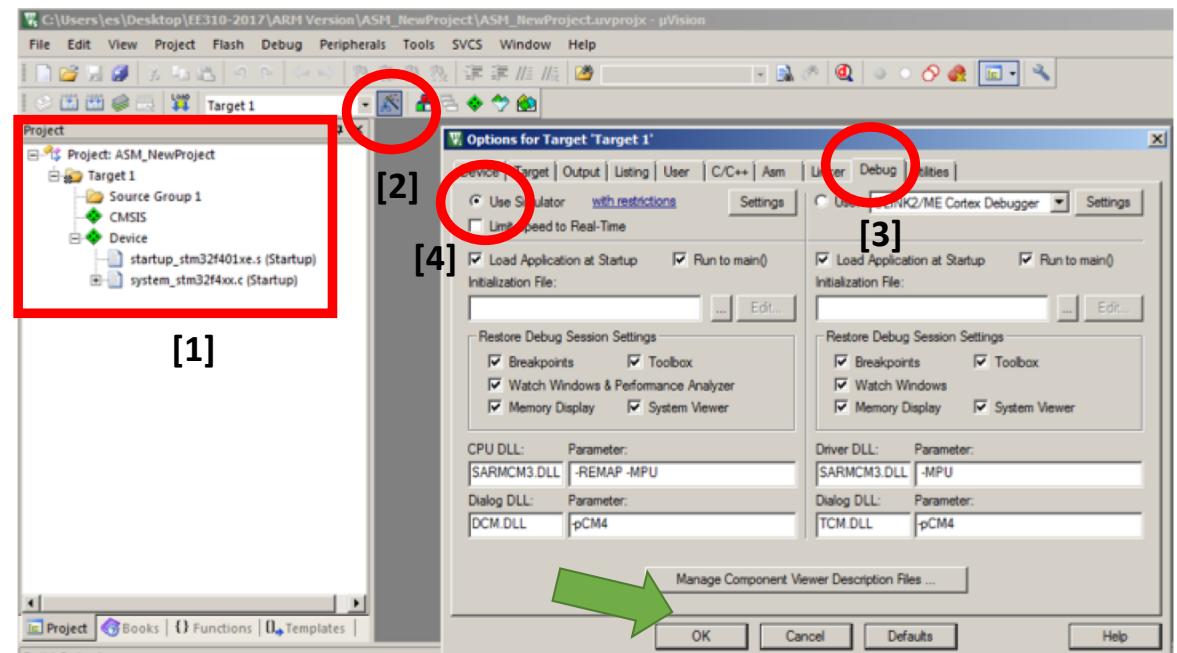
References

- The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors – Chapter 15

If You Experience Any Issues....

If You Get Compiling Errors

- At this point your Project window should show the CMSIS and DEVICE – these are the CMSIS system initialization files and startup files in order for the RUN TIME to operate [1]
- Click on the OPTIONS FOR TARGET [2]
- Click on DEBUG tab [3]
- CHECK the SIMULATOR [4]
- Click on OK
- In the PROJECT window click on Source Group 1 and select ADD NEW ITEM....
- In the new window select



If Devices Are Marked as X

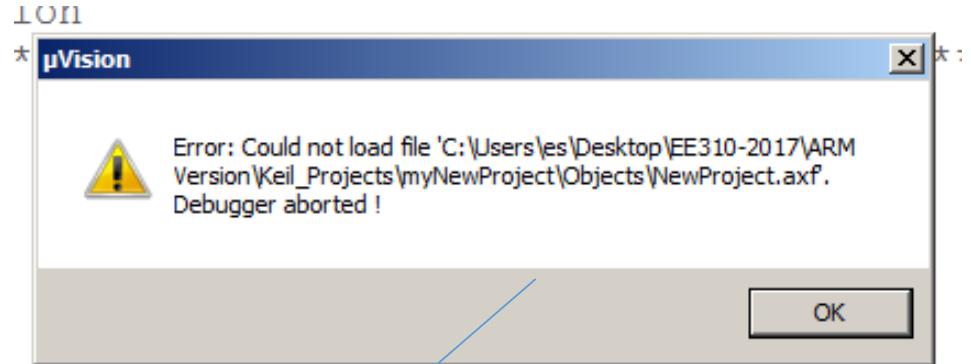
Re-do this section

Software Component	Sel.	Variant	Version	Description
+ AMP				Asymmetric Multiprocessing
+ Board Support		STM32F429I-Discovery	1.0.0	STMicroelectronics STM32F429I-Discovery Kit
- CMSIS				Cortex Microcontroller Software Interface Components
+ CORE			5.0.2	CMSIS-CORE for Cortex-M, SC000, SC300, ARMv8-M
+ DSP			1.5.2	CMSIS-DSP Library for Cortex-M, SC000, and SC300
+ RTOS (API)			1.0.0	CMSIS-RTOS API for Cortex-M, SC000, and SC300
+ RTOS2 (API)			2.1.1	CMSIS-RTOS API for Cortex-M, SC000, and SC300
+ CMSIS Driver				Unified Device Drivers compliant to CMSIS-Driver Specifications
+ CMSIS Driver Validation			1.0.0	Run API test for enabled drivers
+ CMSIS RTOS Validation				CMSIS-RTOS Validation Suite
+ Compiler			1.2.0	Compiler Extensions for ARM Compiler 5 and ARM Compiler 6
+ CycloneCommon			1.7.8	Common Files
+ CycloneCrypto			1.7.8	Cryptographic Library
+ CycloneSSL			1.7.8	SSL/TLS Library
+ CycloneTCP			1.7.8	Dual IPv4/IPv6 Stack
+ Data Exchange				Software Components for Data Exchange
- Device				Startup, System Setup
+ Startup	<input checked="" type="checkbox"/>		2.6.0	System Startup for STMicroelectronics STM32F4 Series
+ STM32Cube Framework (API)			1.0.0	STM32Cube Framework
+ STM32Cube HAL				STM32F4xx Hardware Abstraction Layer (HAL) Drivers
+ File System		MDK-Pro	6.9.8	File Access on various storage devices
+ Graphics		MDK-Pro	5.36.6	User Interface on graphical LCD displays
+ Graphics Display				Display Interface including configuration for emWIN
+ Network		IwIP	1.4.1	Network IwIP Bundle
+ RTOS		RT-Thread	2.1.2	rt-thread
+ SMQ				
+ Security				
+ SharkSSL				
+ Simulation				Components used for simulation purposes
+ USB		MDK-Pro	6.11.0	USB Communication with various device classes
+ mbed				
+ wolfSSL		wolfSSL	3.9.0	wolfSSL: SSL/TLS and Crypt Library

AFX Error: Missing .afx file

- Make sure you do not have multiple projects in the same directory

```
*** Using Compiler 'V5.06 update 5 (build 528)', folder: 'C:\Keil_v5\ARM\ARMCC\Bin' assembling  
Brach_Part_2.s...Brach_Part_2.s(79): warning: A1447W: Missing END directive at end of file, but  
found a label named END"Brach_Part_2.s" - 0 Error(s), 1 Warning(s).Error: Could not load file  
'C:\Users\es\Desktop\EE310-2017\ARM Version\Keil_Projects\myNewProject\Objects\NewProject.axf'.  
Debugger aborted !
```

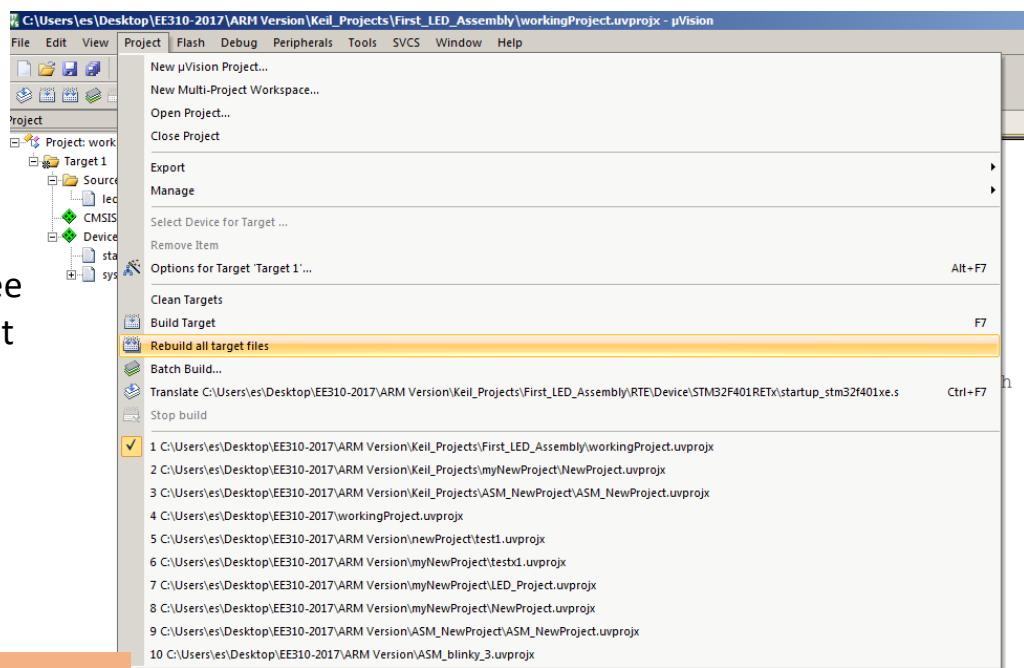


- Also, make sure you build the entire project. You should see Build Time Elapsed to ensure the project was properly built (see next slide):

```
*** Using Compiler 'V5.06 update 5 (build 528)', folder:  
'C:\Keil_v5\ARM\ARMCC\Bin'Rebuild target 'Target 1' assembling  
startup_stm32f401xe.s...assembling led2.s...compiling  
system_stm32f4xx.c...linking...Program Size: Code=300 RO-  
data=404 RW-data=0 ZI-data=1536 ".\Objects\workingProject.axf"  
- 0 Error(s), 0 Warning(s).Build Time Elapsed: 00:00:02
```

Build All Target Files

Also, make sure you build the entire project. You should see Build Time Elapsed to ensure the project was properly built (see next slide):



```
*** Using Compiler 'V5.06 update 5 (build 528)', folder:  
'C:\Keil_v5\ARM\ARMCC\Bin'Rebuild target 'Target 1' assembling  
startup_stm32f401xe.s...assembling led2.s...compiling  
system_stm32f4xx.c...linking...Program Size: Code=300 RO-  
data=404 RW-data=0 ZI-data=1536 ".\Objects\workingProject.axf"  
- 0 Error(s), 0 Warning(s).Build Time Elapsed: 00:00:02
```