

STM32G4 Mixed Signal MCU

Hands-On Workshop

Karthik Chidambaranathan

Jason Stover

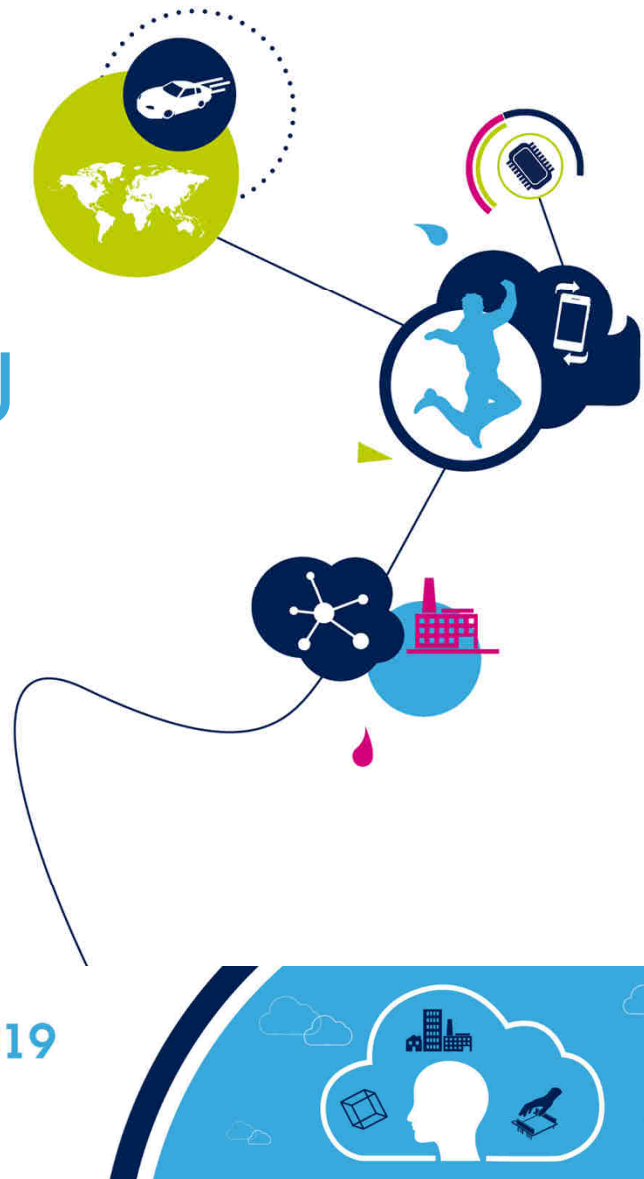
Bruno Montanari

The STMicroelectronics Team



Technology Tour 2019

Boston, MA | November 5



You need to have administration rights on your Laptop Computer to be able to install drivers and software and also to do the workshop.

System & Software requirements for Laptop Computer:

- Windows® 7 and later or MacBook running Windows using Parallels, VM Fusion
- Install Java™ Run Time Environment for 1.8.0 or later. If Java™ is not installed on your computer, we are providing the installer for the latest Java on the USB drive.

Minimum Recommended Hardware Requirements for Laptop Computer:

- Type A USB port
- 2+ GHz processor
- 4 GB of system memory (RAM)
- 10 GB of available disk space

Note: For laptops with only USB Type C ports, you will need a Type A (female) to Type C (male) adapter.

- ❑ STM32CubeMX: [version 5.3.0 minimum](#)
- ❑ STM32CubeG4: [version 1.1.0 minimum](#)
- ❑ STM32CubeProgrammer: [version 2.1.0 minimum](#)
- ❑ Keil MDK-ARM: [version 5.28a minimum](#)
- ❑ Keil MDK-ARM License: [Professional Evaluation License Add On](#)
- ❑ Tera Term [or equivalent terminal emulator](#)
- ❑ Java Runtime Environment: [1.8.0 minimum](#)

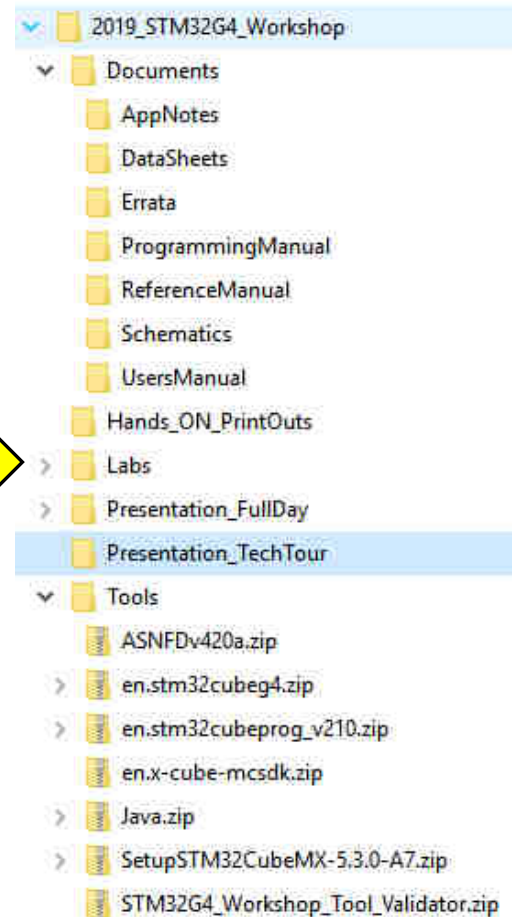
STM32G4 Workshop Installation Guide - Overview

4

- ❑ Copy the contents of your thumb drive or download to your **C:** root
- ❑ If not done automatically, name the top folder
2019_STM32G4_Workshop

This is the Labs folder. You will be here many times.

Note: We don't recommend copying to a deep path such as your Desktop or Documents because long path names can create build issues with some compilers.

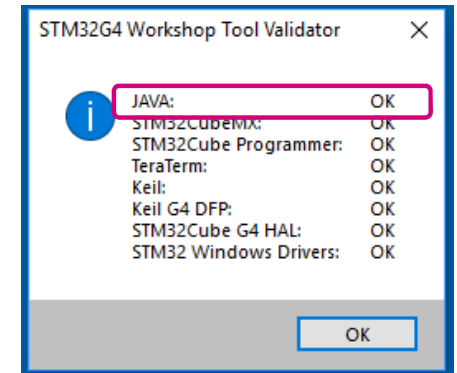


STM32G4 Workshop Installation Guide – Java JRE 1.8.0+

5

Java™ Run Time Environment 1.8.0 or later is required to run Cube software tools.

To check if you have the correct version of Java installed please launch the app “**STM32G4_Workshop_Tool_Validator_vx.x.exe**” on the USB drive.



If needed, the Java installer is located in the **Java** directory:

To install Java right Click on “**jre-8u201-windows-i586.exe**” and Run as administrator.



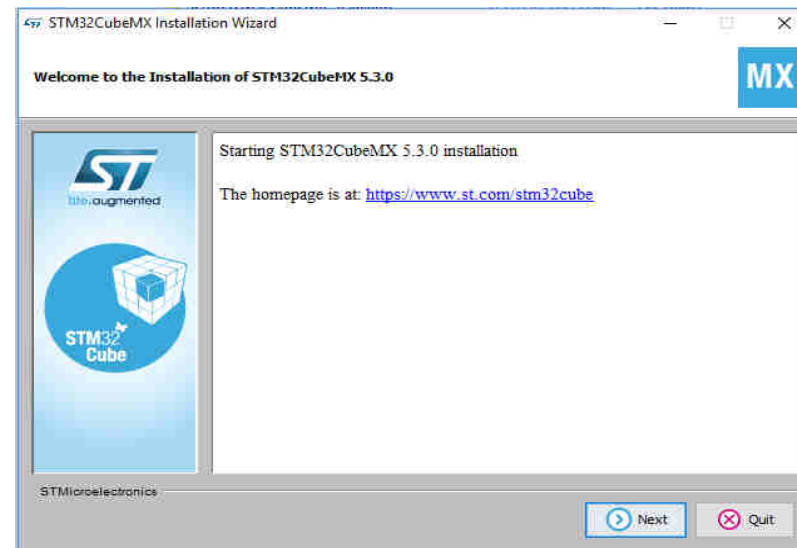
STM32G4 Workshop Installation Guide – CubeMX v5.1.0+

6

Unzip **en.SetupSTM32CubeMX-5.3.0.zip** and you will see this:

Name	Type	Compressed size	Password ...	Size
SetupSTM32CubeMX-5.3.0.app	File folder			
Readme.html	Chrome HTML Document	3 KB	No	7 KB
SetupSTM32CubeMX-5.3.0.exe	Application	168,022 KB	No	169,294 KB
SetupSTM32CubeMX-5.3.0.linux	LINUX File	5 KB	No	14 KB

Right Click on **SetupSTM32CubeMX-5.3.0.exe** and Run as administrator:



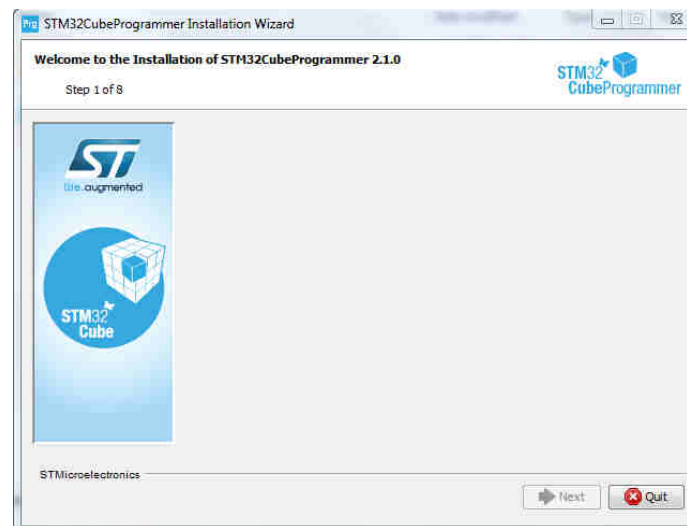
STM32G4 Workshop Installation Guide – CubeProgrammer v2.1.0

7

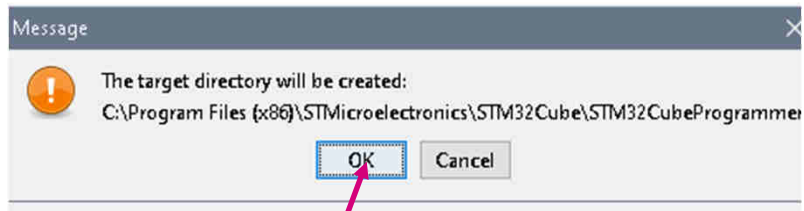
Unzip **en.stm32cubeprog_v2.1.0.zip** and you will see this:

Name	Type	Compressed size	Pass...	Size
SetupSTM32CubeProgrammer-2.1.0.app	File folder			
SetupSTM32CubeProgrammer-2.1.0.exe	Application	182,709 KB	No	183,604 KB
SetupSTM32CubeProgrammer-2.1.0.linux	LINUX File	327 KB	No	742 KB

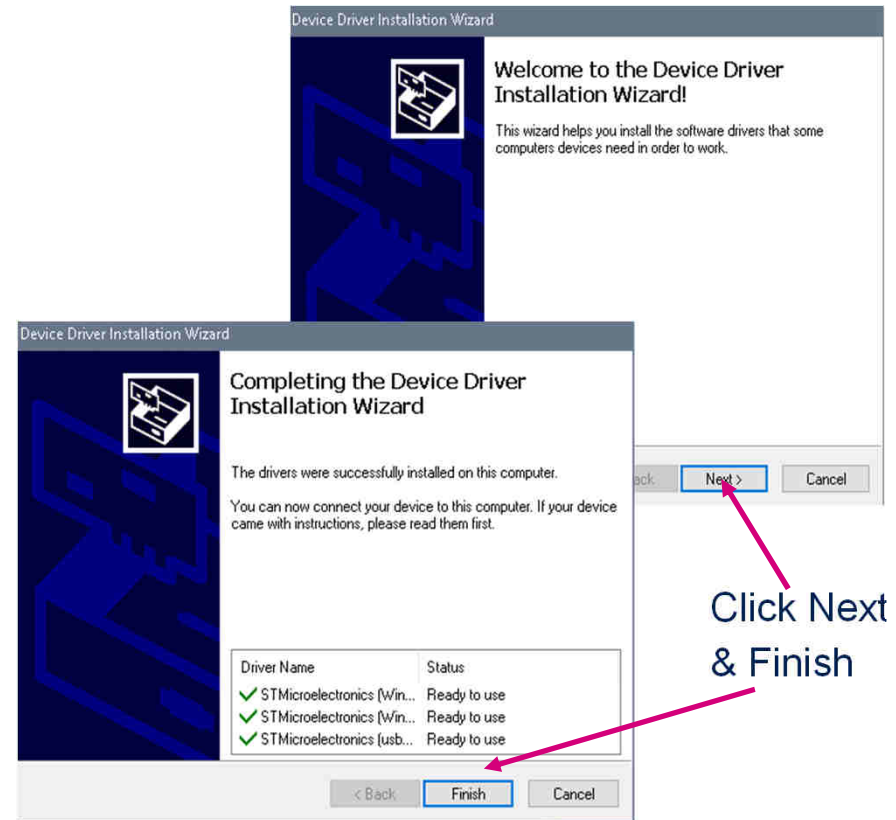
Right Click on **SetupSTM32CubeProgrammer-2.1.0.exe** and Run as administrator:



Install the Windows drivers if prompted



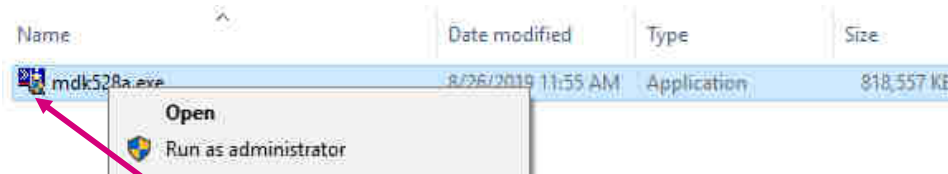
Click OK to create the Directory
(if the directory does not already exist)



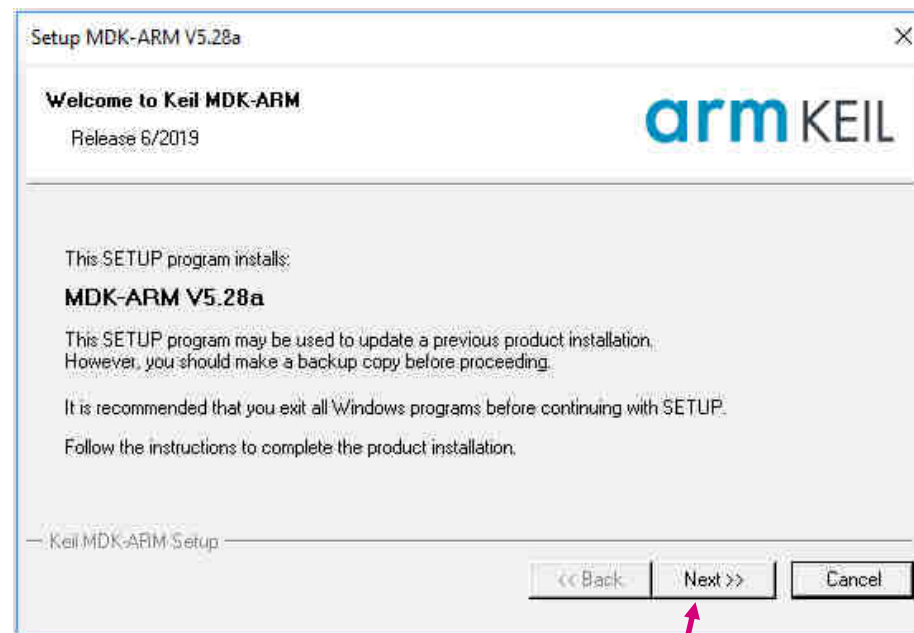
Click Next
& Finish

STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

9



Right Click on **mdk528a.exe** and Run as administrator:

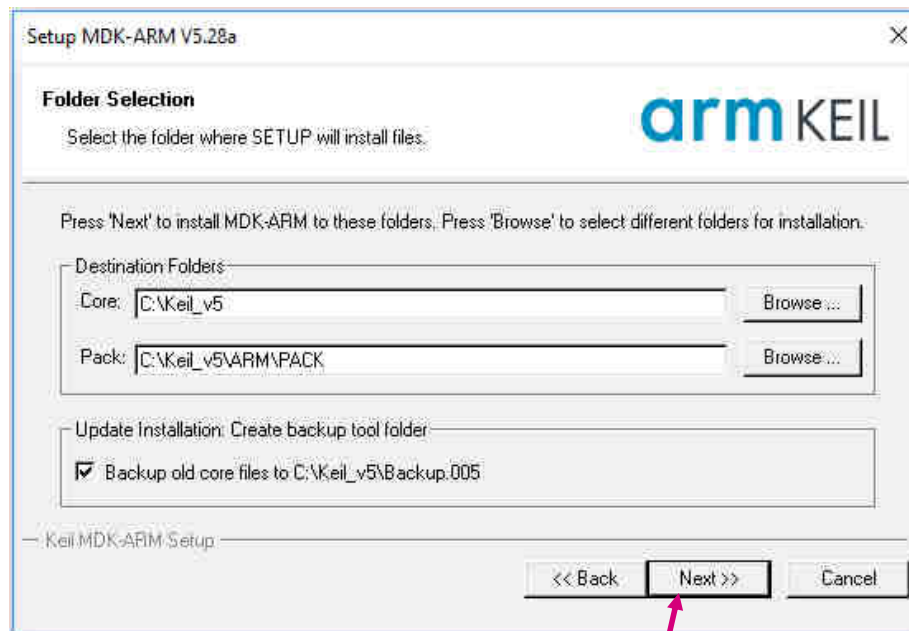


Click Next

STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

10

Leave the default folder locations as is

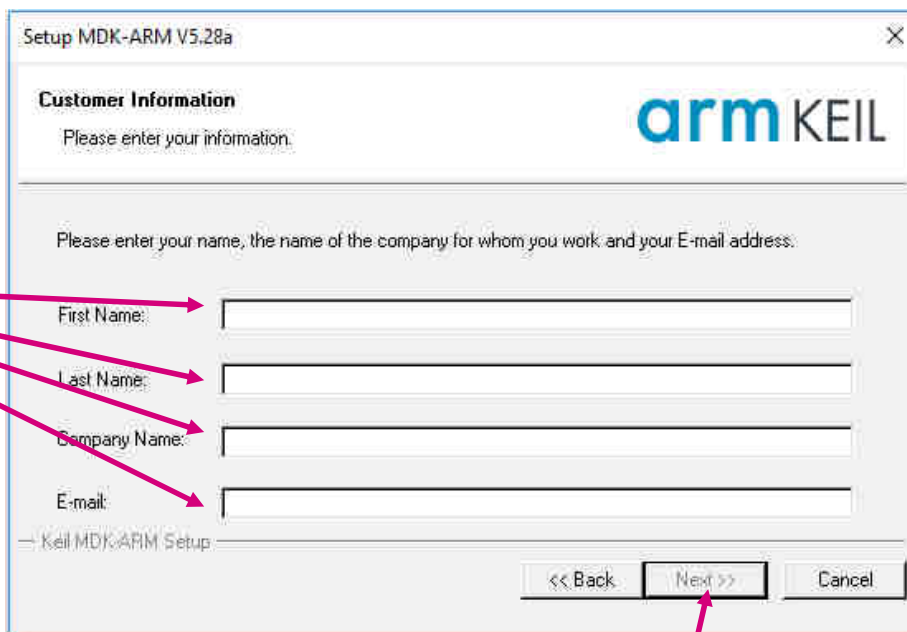


Click Next

STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

11

Enter your contact
information



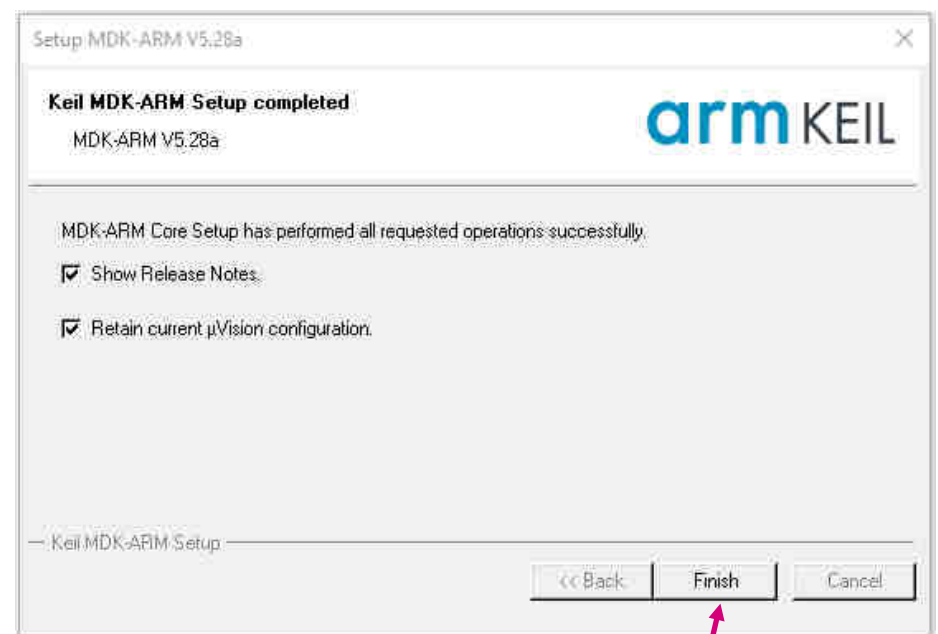
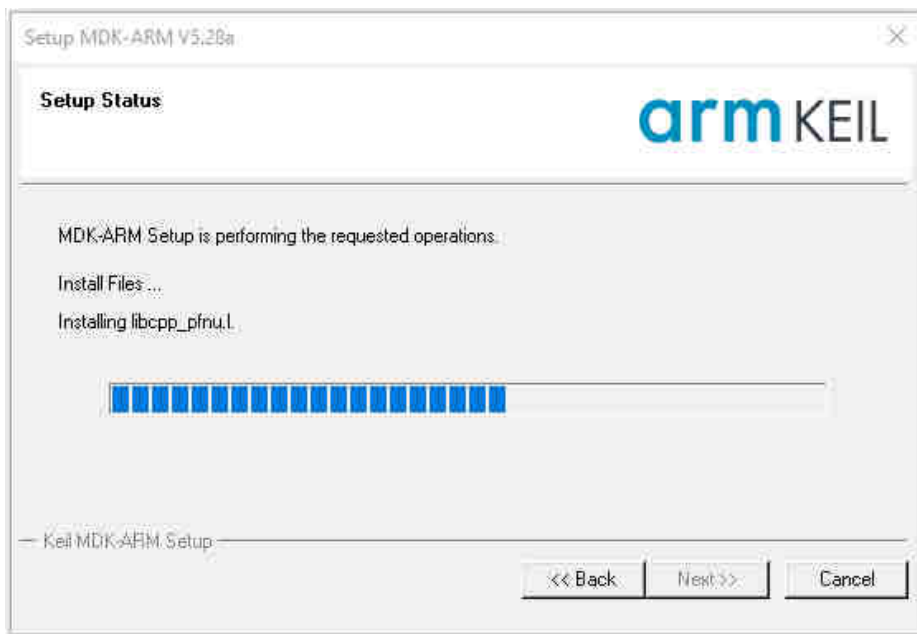
The image shows a screenshot of the 'Setup MDK-ARM V5.28a' window. The window has a title bar with a close button. Inside, there's a section titled 'Customer Information' with the instruction 'Please enter your information.' and the 'arm KEIL' logo. Below this, another instruction says 'Please enter your name, the name of the company for whom you work, and your E-mail address.' There are four text input fields labeled 'First Name:', 'Last Name:', 'Company Name:', and 'E-mail:'. At the bottom, there are three buttons: '<< Back', 'Next >>', and 'Cancel'. A pink arrow points from the 'Enter your contact information' text to the input fields, and another pink arrow points from the 'Click Next' text to the 'Next >>' button.

Click Next

STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

12

The installation should run and look like this:

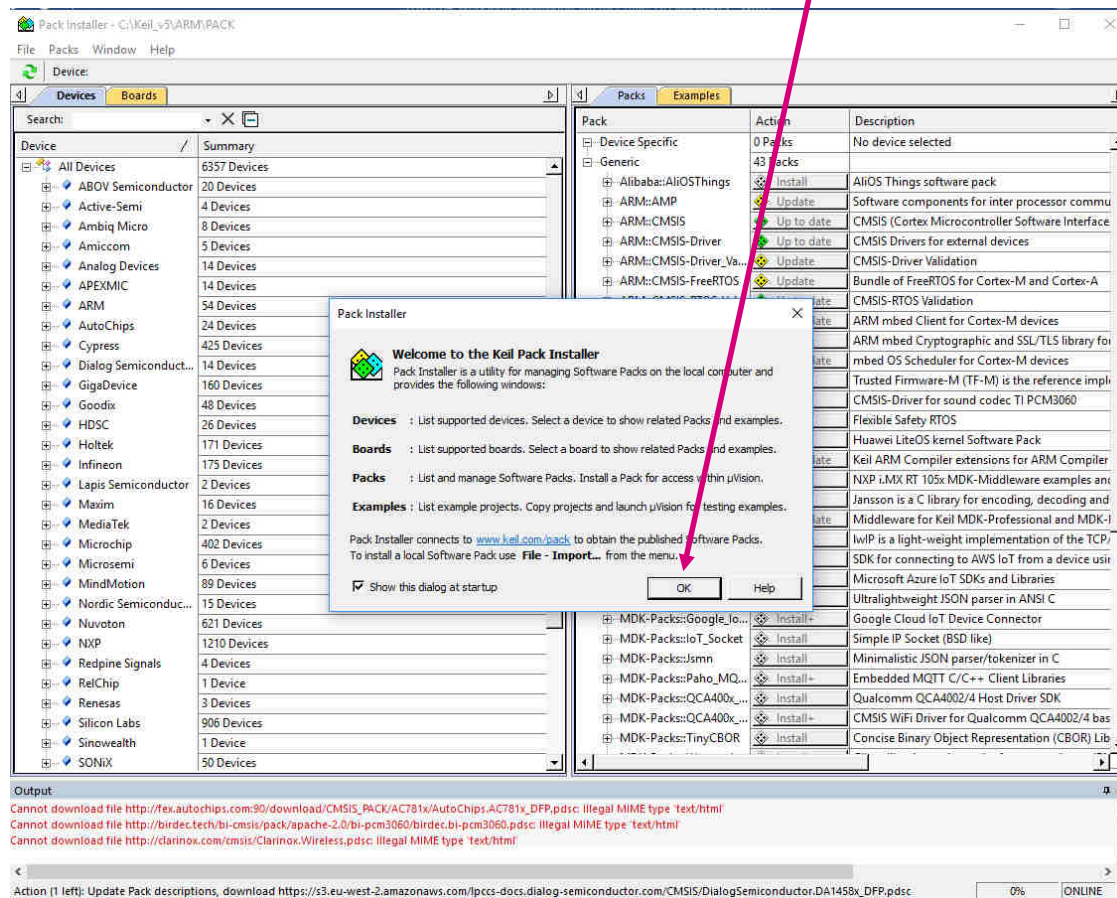


Once the installation has completed click Finish

STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

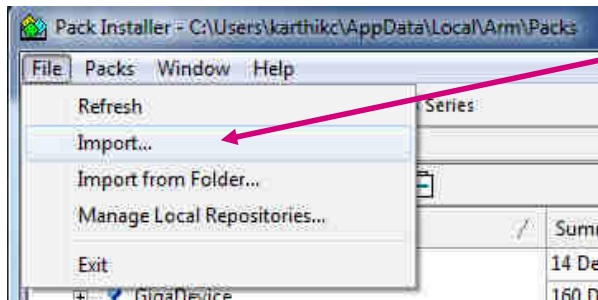
13

The pack installer should automatically start, click OK to close the welcome screen



STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

14

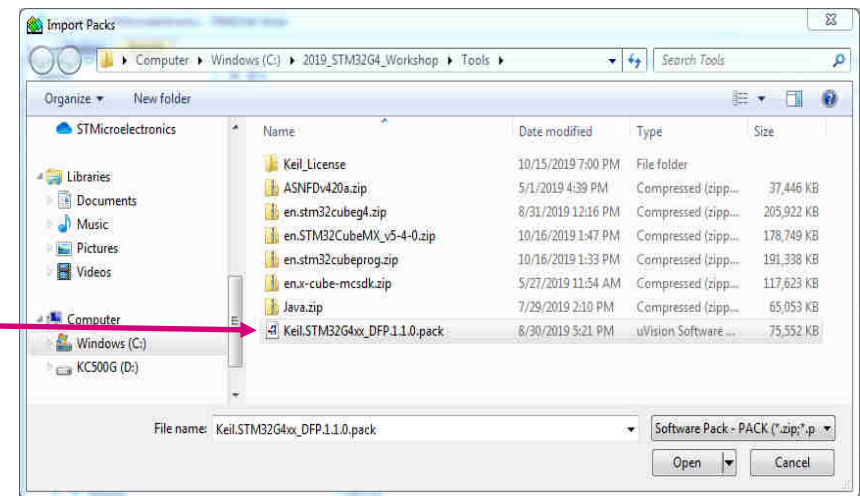


1 Click on File > Import

2

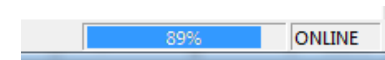
Choose this file to import:

C:\2019_STM32G4_Workshop\Tools\Keil.STM32G4xx_DFP.1.1.0.pack



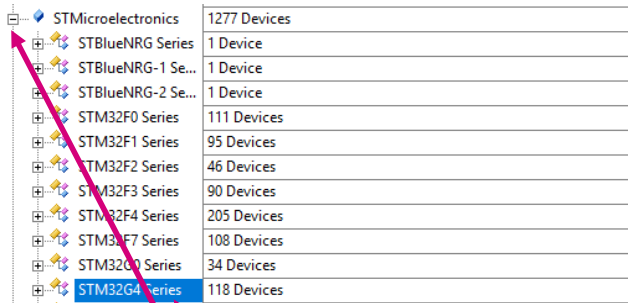
3

Wait until progress bar on bottom right corner is completed to 100%

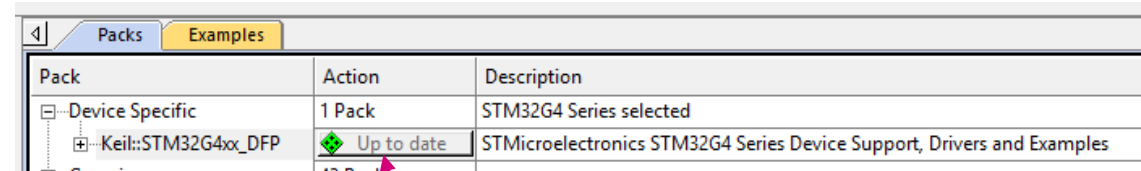


STM32G4 Workshop Installation Guide – Keil MDK-ARM v5.28a

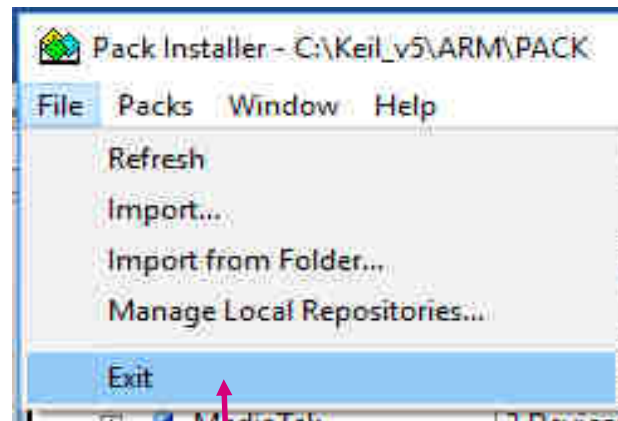
15



- 1 On the “Device” (left) side of the screen scroll down and click on the “+” to expand “STMicroelectronics” and then click on the “STM32G4 Series” to select it



- 2 On the “Pack” (right) side of the screen, next to the “Keil: STM32G4xx_DFP”, once complete it should show “Up to date”



- 3 Click on “File” and then “Exit” to quick the pack installer, installation is now complete

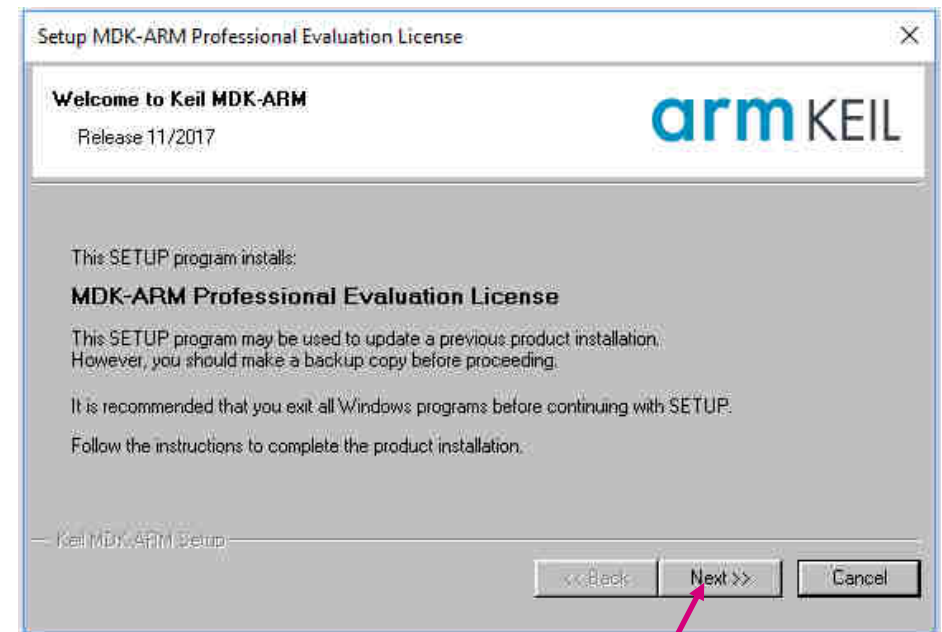
Keil MDK-ARM License Installation

16



C:\2019_STM32G4_Workshop\Tools\Keil_License\
MDK-ARM_Pro_Eval_AddOn_11-2019.exe

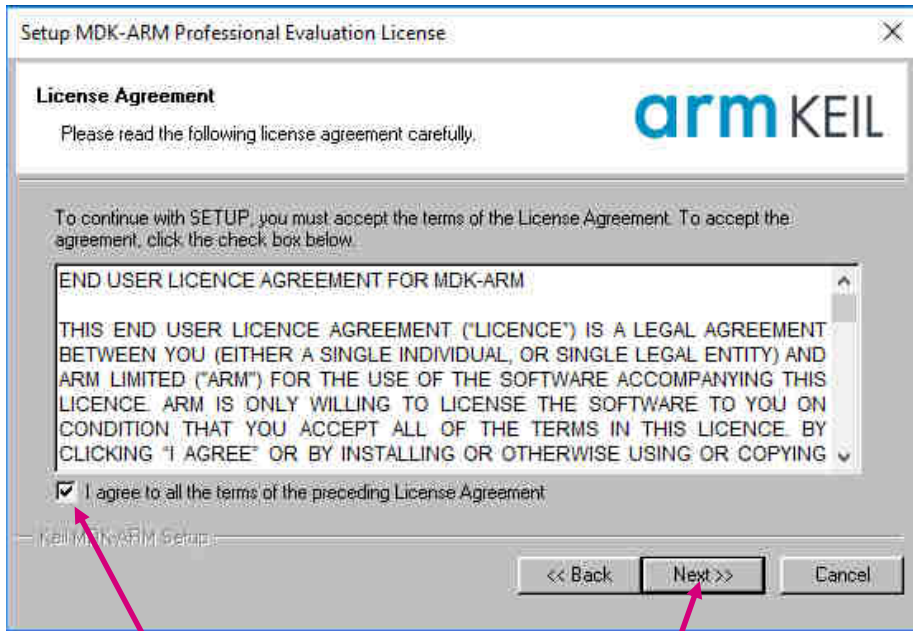
Right click and click “Run as administrator” to install
the Professional Evaluation **License Add On**



Click **Next**

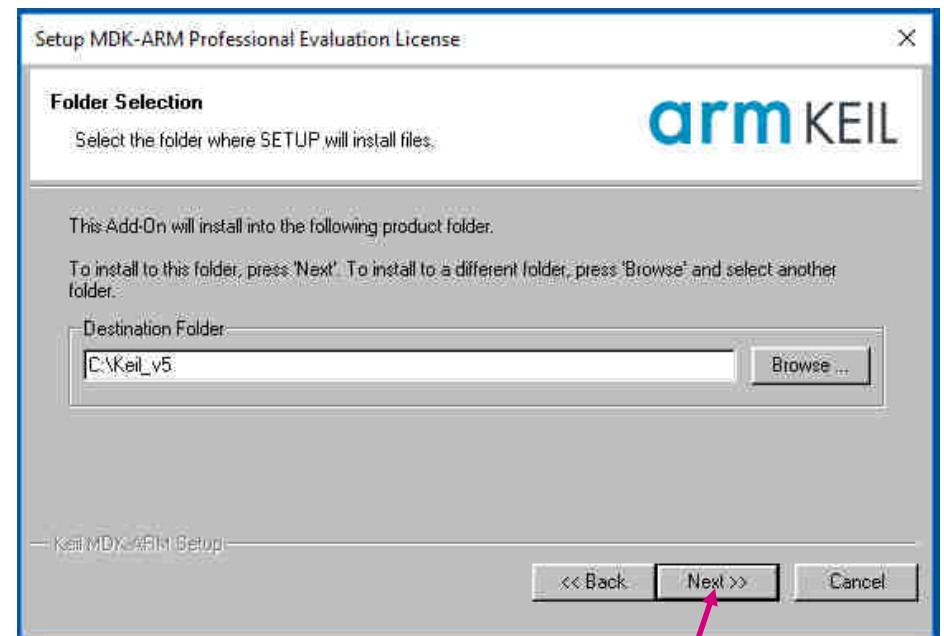
Keil MDK-ARM License Installation

17



Click the check box to agree to the terms

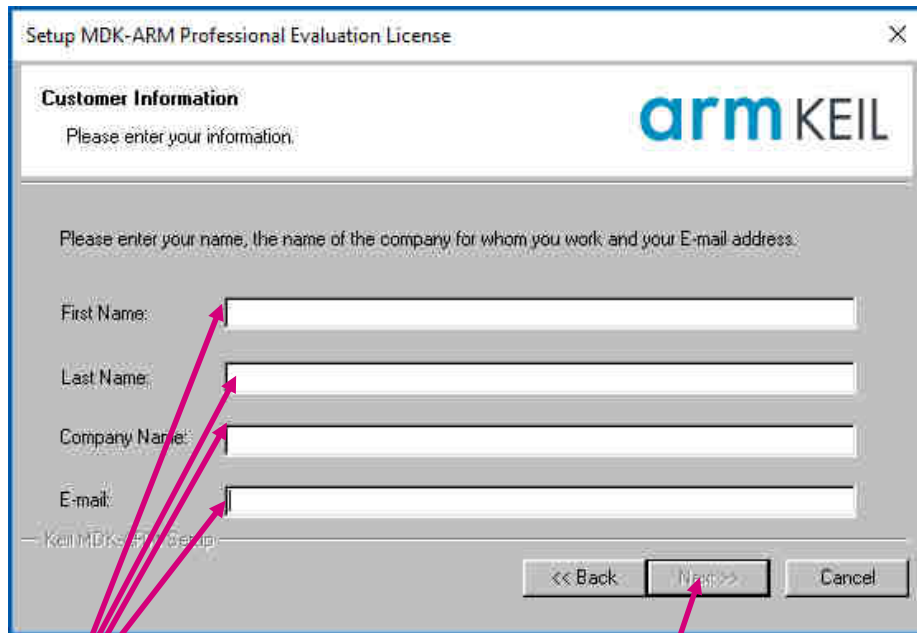
Click **Next**



Click **Next**

Keil MDK-ARM License Installation

18



Setup MDK-ARM Professional Evaluation License

Customer Information
Please enter your information.

arm KEIL

Please enter your name, the name of the company for whom you work and your E-mail address.

First Name:

Last Name:

Company Name:

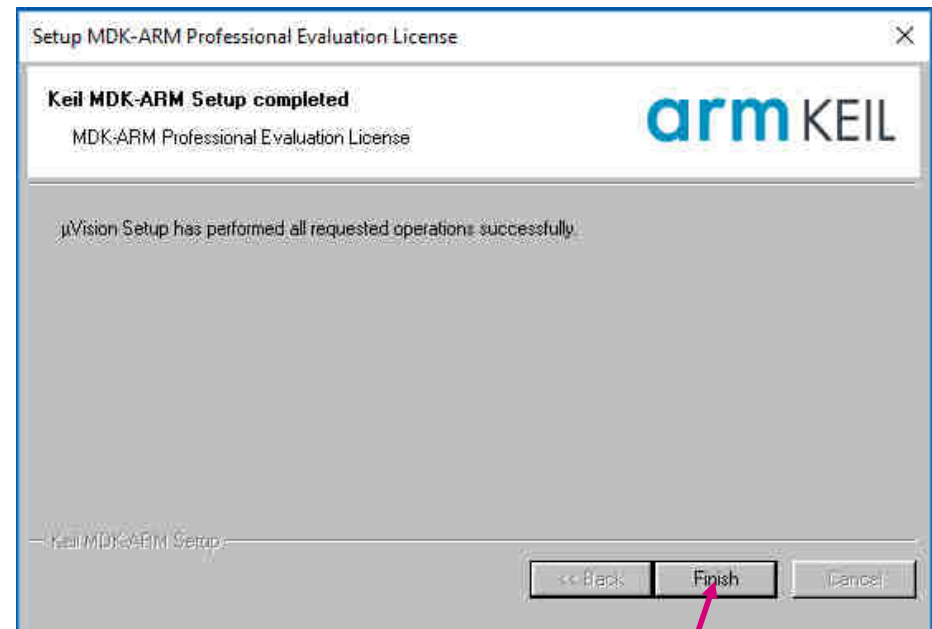
E-mail:

— Keil MDK-ARM Setup —

<< Back Next >> Cancel

Enter your contact information

Click **Next**



Setup MDK-ARM Professional Evaluation License

Keil MDK-ARM Setup completed
MDK-ARM Professional Evaluation License

arm KEIL

µVision Setup has performed all requested operations successfully.

— Keil MDK-ARM Setup —

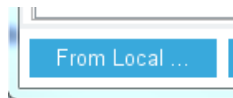
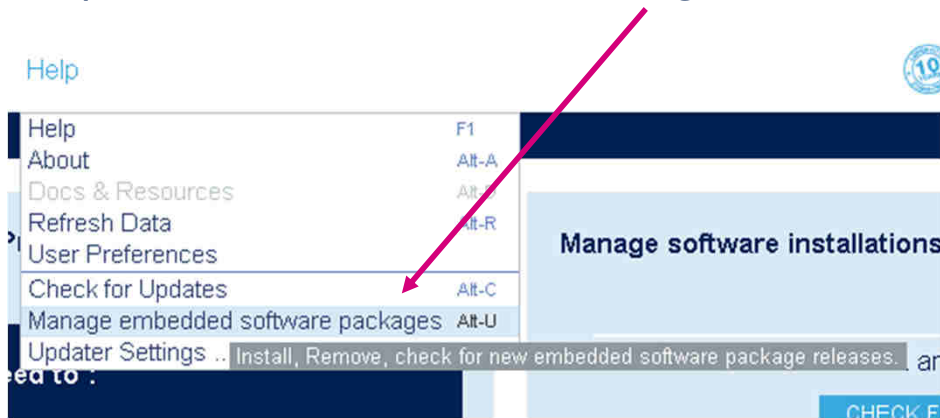
<< Back Finish >> Cancel

Click **Finish**

STM32G4 Workshop Installation Guide – CubeG4, v1.1.0

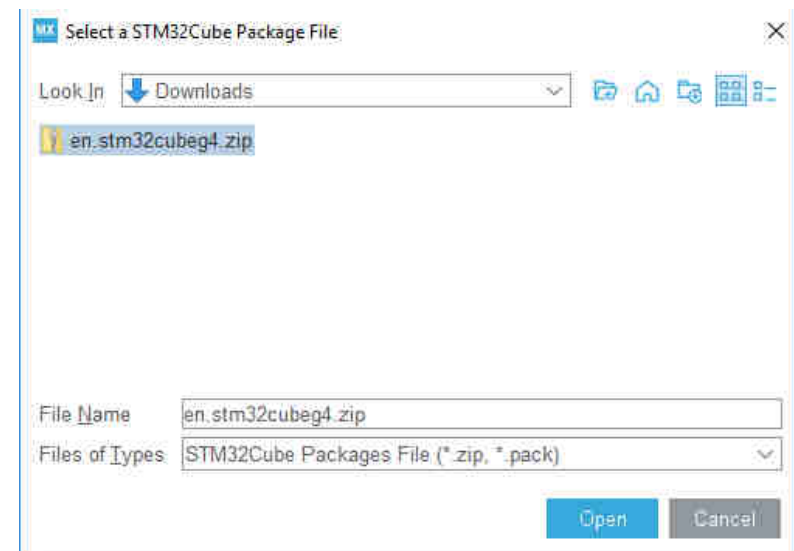
19

Open CubeMX and select **Manage embedded software packages**

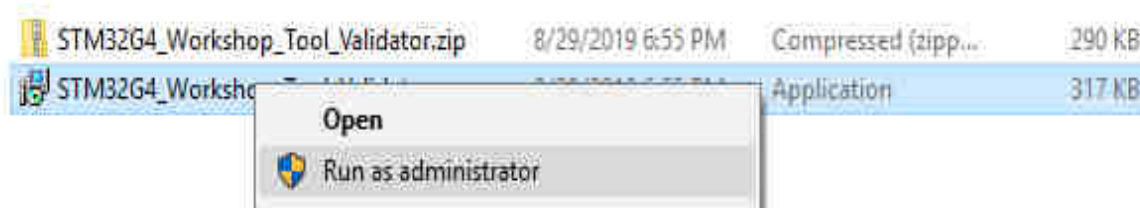


Select **From Local...**

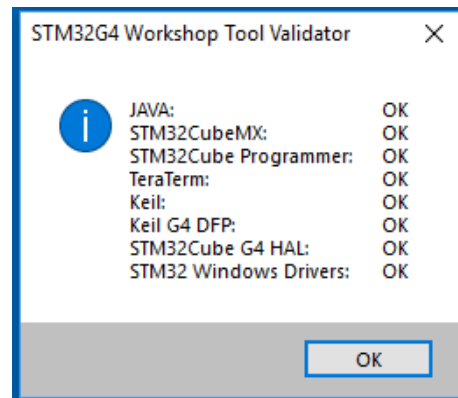
Select **en.stm32cubeg4.zip**



Run the ***STM32G4_Workshop_Tool_Validator.exe***

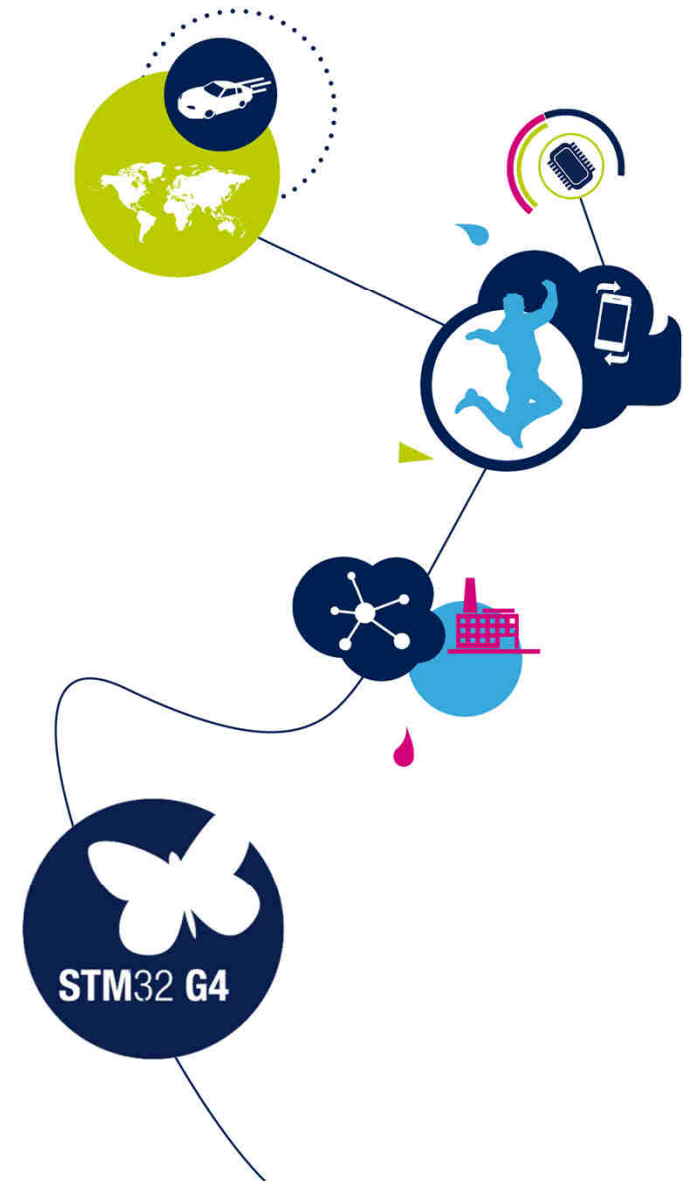


See a workshop proctor for any installation issues



STM32G4 mixed-signal MCU workshop

*Introduction to the STM32G4
Mainstream Series
Mixed Signal MCU*





STM32G4 mixed-signal MCU workshop

22

Agenda

Software Installation

Introduction to the STM32G4 Series

Hands On: STM32G474 Discovery Kit Demo

Hands On: DAC Signal Generation

Hands On: Programmable Gain Amplifier

Break

Hands On: CORDIC Hardware Accelerator

Filter Math Accelerator (FMAC) Overview

Hands On: Dual Bank Flash Live Firmware Upgrade

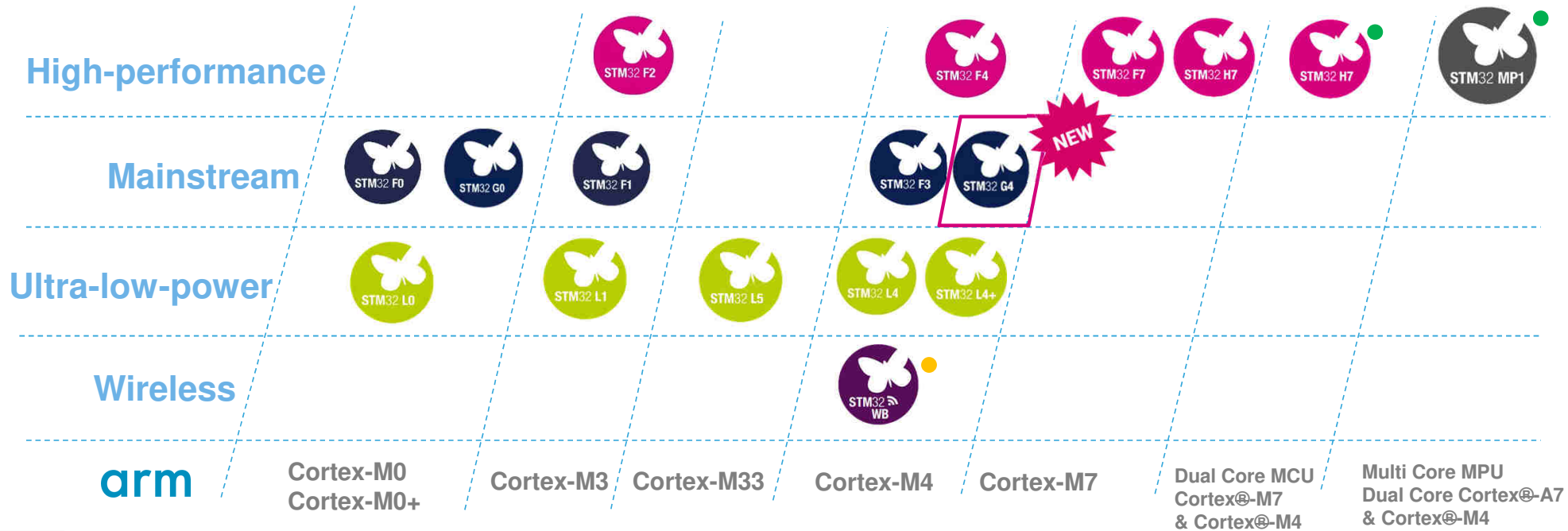
Wrap Up



STM32 portfolio

23

17 product series / More than 50 product lines



Note ● : Cortex-M0+ Radio Co-processor, ● : Cortex-M4 Co-processor

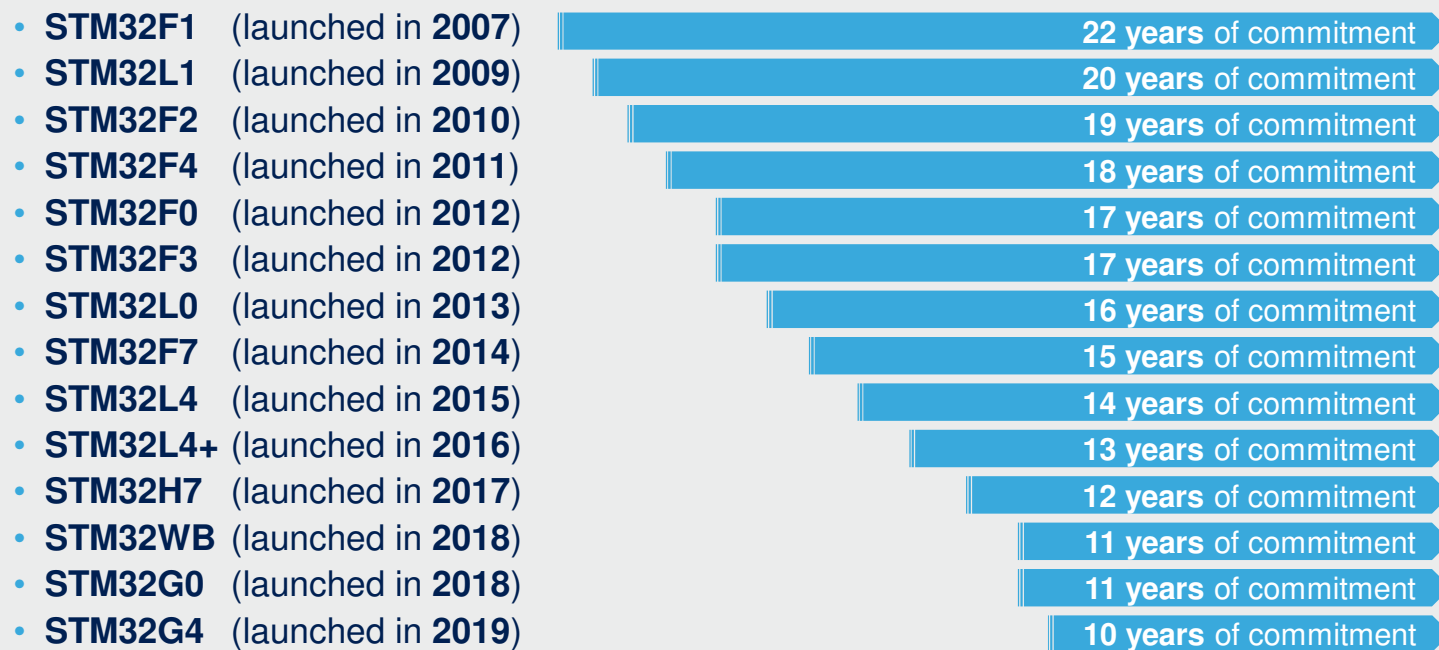
STM32 Rolling Longevity Commitment

24

Longevity commitment is renewed every year



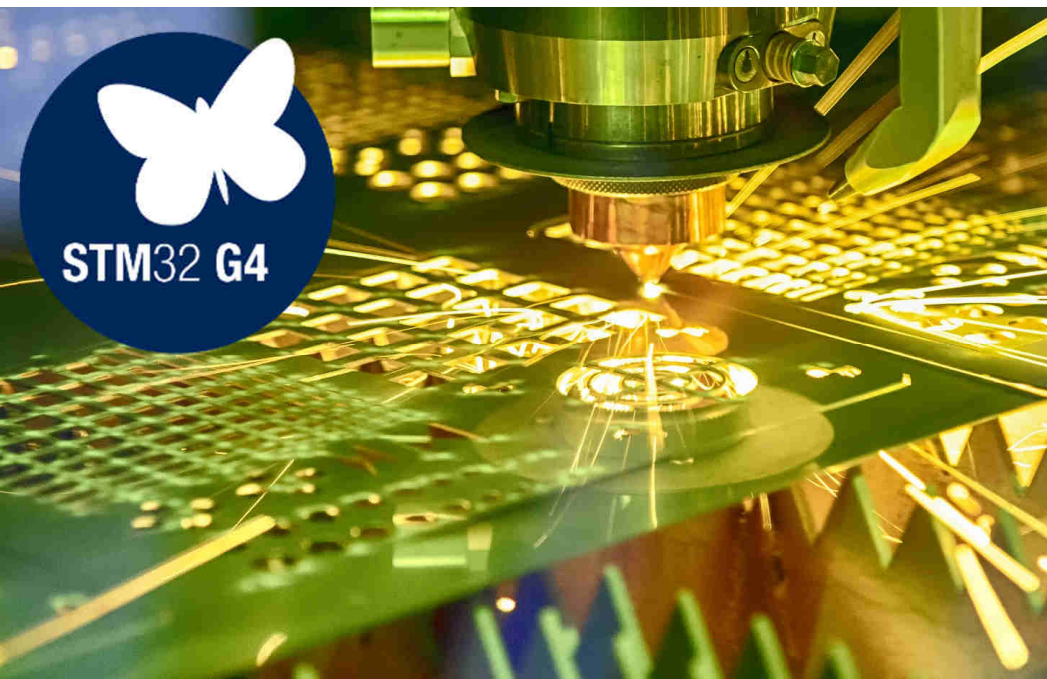
starting January
1st 2019
→ Until 2029



STM32G4 Series

25

Ideal for applications requiring MCU with advanced and rich analog peripherals



- Control applications (Motor Control...)
- Industrial equipment
- Instrumentation and Measurement
- Digital Power
 - Digital SMPS (switch mode power supply)
 - PFC (power factor correction)



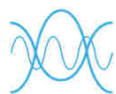
STM32G4 Series – Key Messages

26



Performance

- Arm® Cortex®-M4 at 170 MHz
- 213 DMIPS and 550 CoreMark® results
- Better dynamic power consumption (163µA/MHz)
- ART Accelerator™ (dynamic cache)
- Mathematical accelerators
- CCM-SRAM Routine Booster (static cache)



Rich Integrated Analog and Digital

- Op-Amps (Built-in gain), DACs, Comparators
- 12-bit ADCs 4Msps with hardware oversampling
- CAN-FD (flexible data rate – 8Mbps bit rate)
- High resolution timer (184 ps)
- USB type-C Power Delivery3.0
- 1% RC accuracy [-5°..90°C], 2% full T° range

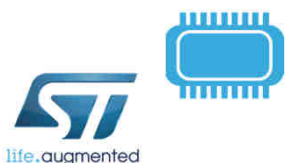


Safety and security focus

- Dual Bank Flash with ECC (error code correction)
- Securable Memory Area
- Hardware encryption AES-256
- SIL, Class-B
- SRAM with Parity bit

} Secure Live Upgrade

} Functional safety design packages



Complete portfolio

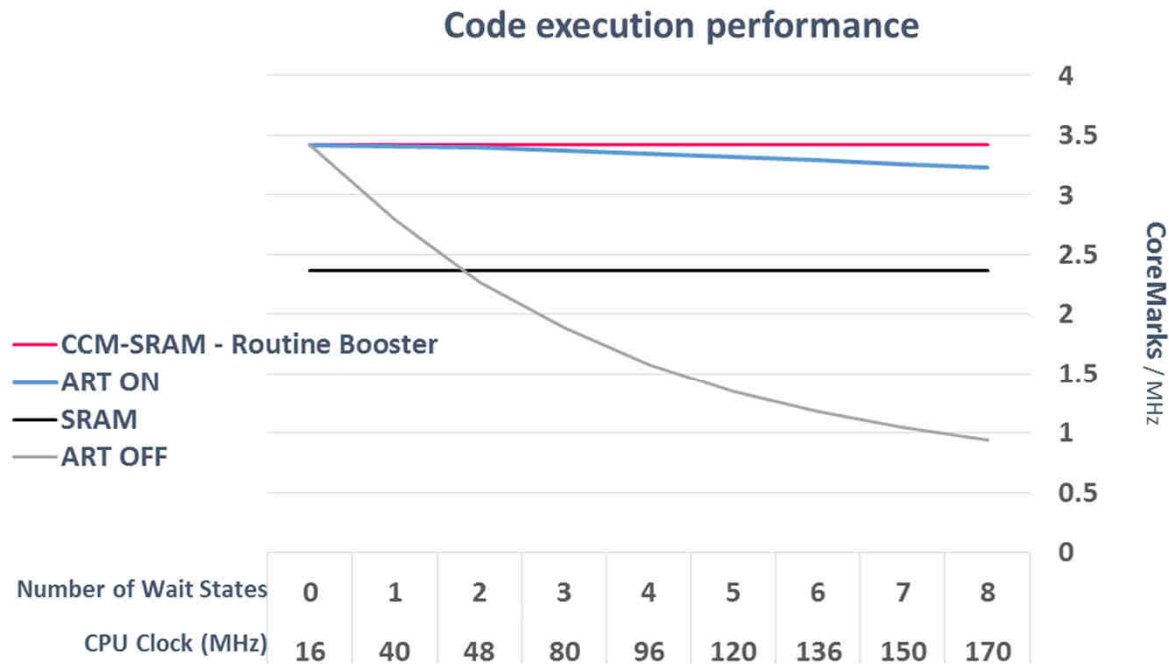
- Complements existing STM32F3 Series portfolio
- From -40°C up to 85 or 125°C devices
- From 32- up to 128-pin
- From 32KB to 512KB Flash



Greater Performance

27

Pure 170 MHz CPU performance (Arm® Cortex®-M4) with 3 accelerators



Arm Cortex-M4 with **FPU**

Up to 170 MHz CPU frequency

Up to 213 DMIPS and 550 CoreMark® results

3 different HW accelerators:

- **ART accelerator** (~dynamic cache)
→ Full code acceleration (average)
- **Routine Booster CCM-SRAM**
(~static cache) → determinism preserved
- **Mathematical** (CORDIC + FMAC)





Mathematical Accelerators

28

Function acceleration and CPU offload

1. CORDIC (Trigo)

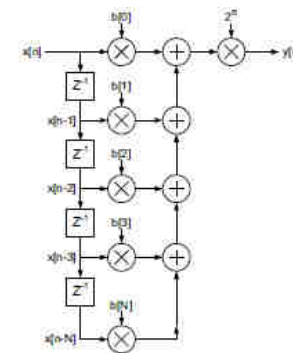
- Very helpful for Field Oriented Motor Control method (FOC)

- Vector rotation (polar to rectangular): Sin, Cos
- Vector translation (rectangular to polar): Atan2, Modulus
- Sinh, Cosh, Exp
- Atan, Atanh
- Square root
- Ln

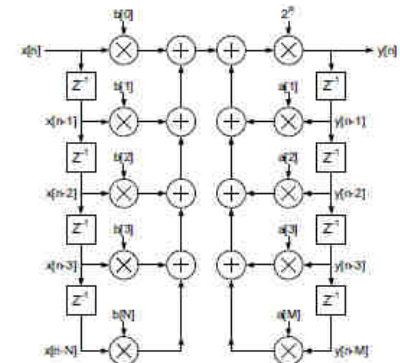
2. Filter Math Accelerator (FMAC)

- Can be used to create
 - 3p3z Compensator ($\rightarrow$ Digital power)
 - Sigma Delta modulator
 - Noise Shaper

FIR filter



IIR filter

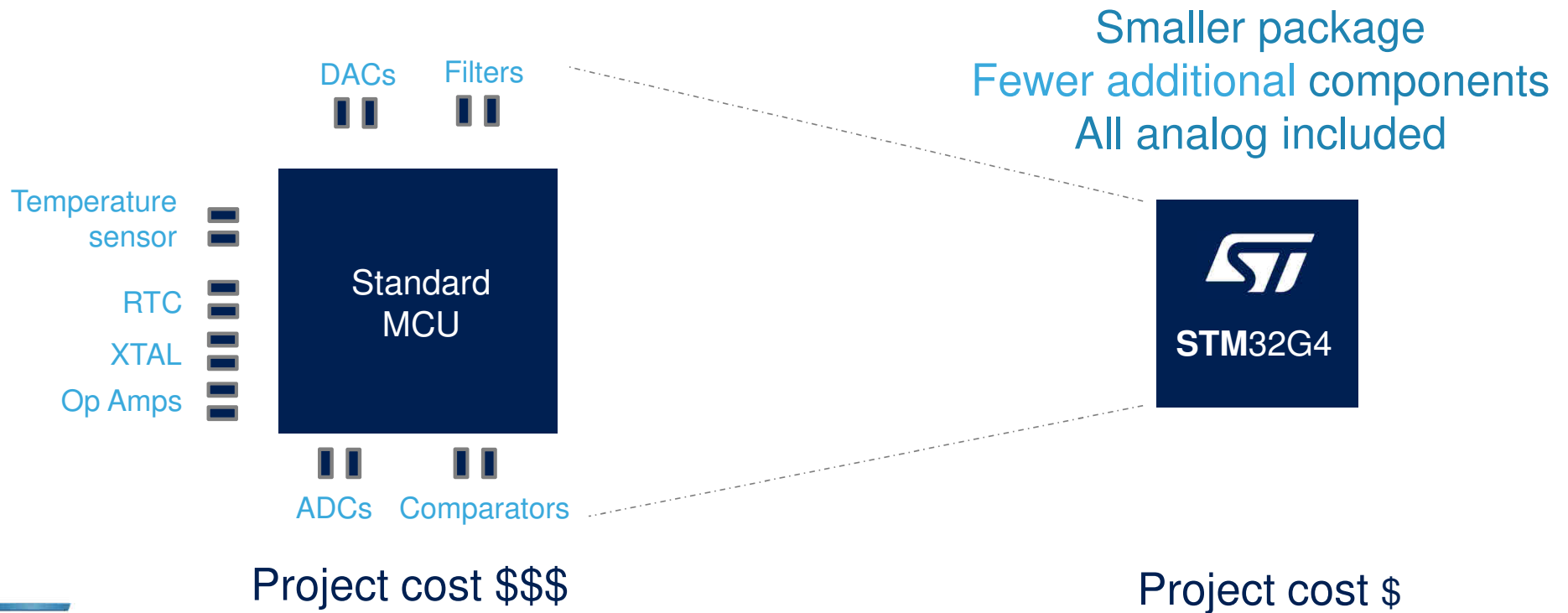




Reducing PCB Size and BOM Cost

29

System-on-Chip – All-in-one solution





Rich, Advanced Analog

30

Mixed-signal SoC for wide variety of applications

ADC (up to 5)	Values
Topology	SAR 12-bit + HW oversampling → 16-bit
Sampling rate	Up to 4 Msps
Input	Single-ended and differential
Offset and Gain compensation	Auto calibration to reduce gain and offset

DAC (up to 7)	Values
Sampling rate	15 Msps (internal) 1Msps (from buffered output)
Settling time	16ns

Op-Amp (up to 6)	Values
GBW	13 MHz
Slew rate	45 V/μs
Offset	3mV over full T° range 1.5mV @ 25°C
PGA Gain (accuracy)	2, 4, 8, 16, -1,-3,-7,-15 (1%) 32, 64, -31,-63 (2%)

Comparator (up to 7)	Values
Power supply	1.62 .. 3.6V
Propagation delay	16.7ns
Offset	-6 .. +2 mV
Hysteresis	8 steps: 0, 9, 18, 27, 36, 45, 54, 63 mV



Key Features for Targeted Applications

31

Motor Control

Home appliances, E-bikes, Air Conditioning

- Fast CPU 170MHz
- Mathematical accelerator (Cordic)
- Advanced Motor Control timers
- Fast comparators
- 4Msps ADC-12bit + HW oversampling
- Op-Amp with built-in gain (PGA)
- DAC-12bit
- 1% RC accuracy (UART communication w/o external Xtal)



High-End Consumer

Rechargeable devices, drones, toys

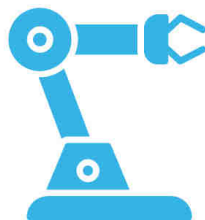
- Low-thickness, small form-factor
- Low consumption in run mode ~ 160µA/MHz
- Embedded analog
- SAI (Sound Audio Interface)
- USB type-C Power Delivery 3.0



Industrial devices Measurements

Industrial equipment

- Fast CPU 170MHz
- Mathematical accelerator (Cordic)
- High temperature 125°C
- CAN FD support
- SPI, USART, I²C
- Advanced timers
- Real Time Clock with backup registers
- Dual bank flash for **live** upgrade
- AES & security



Digital Power

Servers, Telecom, EV Charging station

- Fast CPU 170 MHz
- Mathematical accelerator (Filtering)
- 12ch High Resolution timer (184ps)
- 4Msps ADC-12bit + HW oversampling
- Fast comparators (17ns)
- Embedded analog
- Dual bank flash for **live** upgrade
- AES & security
- FMAC for 3p3z compensation

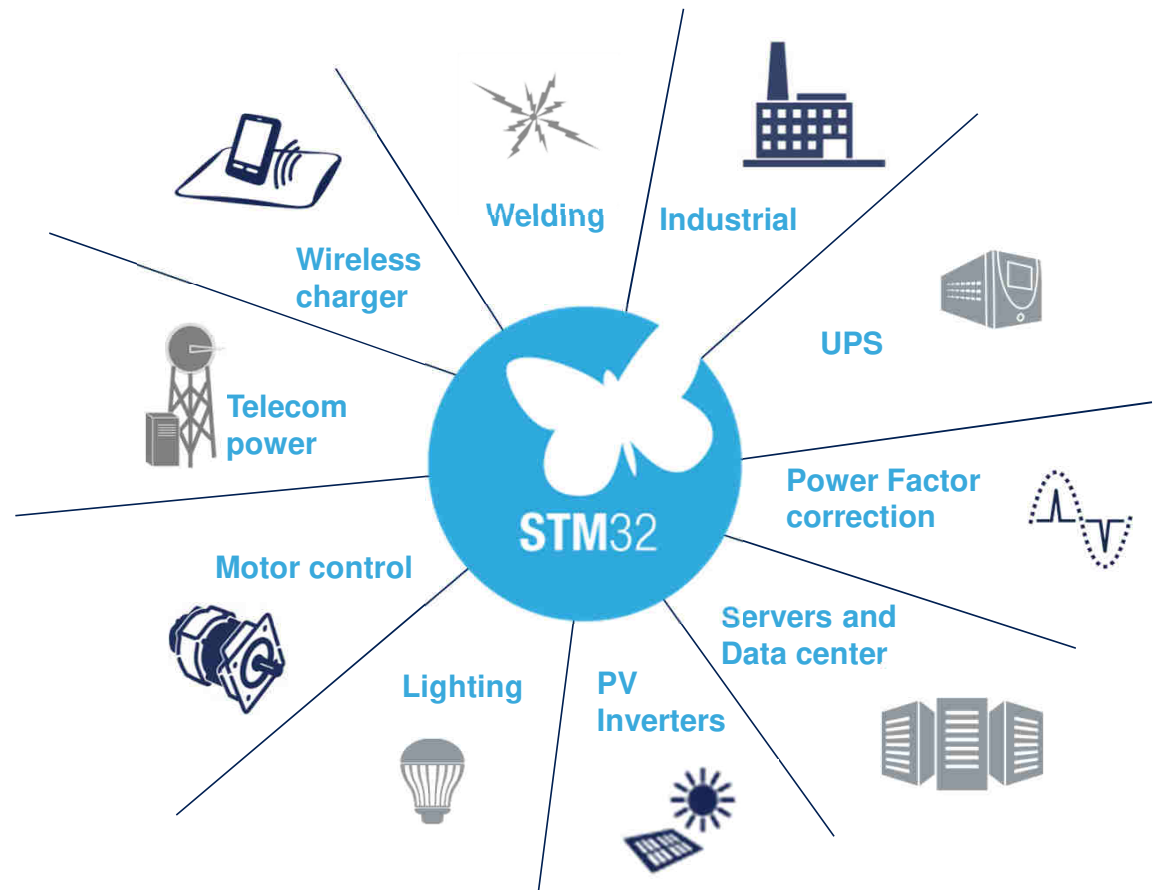




Ease Digital Power Conversion

32

Enhance your digital power solutions using the STM32G4's full features High Resolution Timer (HRTIM)





HRTimer – Not only High Resolution...

33

High resolution PWM

- 12 channels with 184ps resolution on frequency and duty cycle
- 184ps is equivalent to 5.4GHz timer clock

Flexible PWM generation

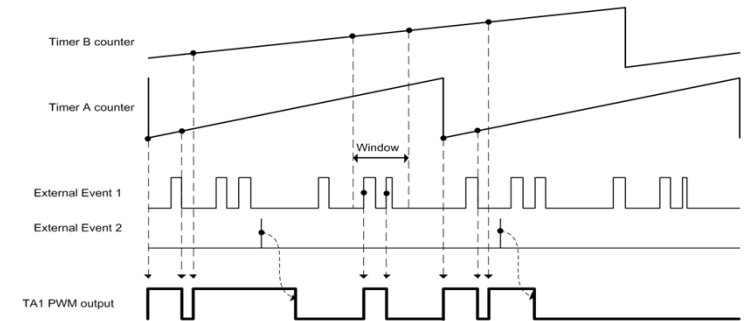
- 7x independent time base to create various shape of PWM
- 6x complementary pair PWM outputs
- Up to 32 set/reset transition per PWM period thx to the built-in crossbar
- Master/Slave configuration for multi phase converter

Multiple Event handler

- 6x Digital and Analog fault input
- 10x Events cycle to cycle current control or PWM restart (constant Ton/Toff)
- Blanking, windowing and digital filter

12 independent channels

- Any topology supported from 1x 12 PWM (triple interleaved LLC (servers application) up to 12x1 PWM (multiple independent buck converters (lighting))





Greater Security

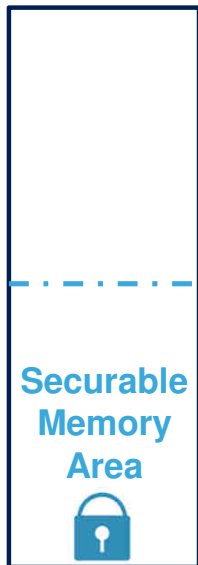
34

Integrated security features, ready for tomorrow's needs

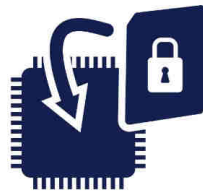
User Flash

Bank1

Bank2



STM32Trust: a complete toolset
an ecosystem for embedded security



<https://www.st.com/stm32trust>

Securable Memory Area:

- Configurable size
- Can be secured once exiting
- No more access nor debug possible
- Good fit to store critical data
 - Critical routines
 - Keys



Securable user memory AES TRNG PCROP MPU Readout protection CRC Write Protection

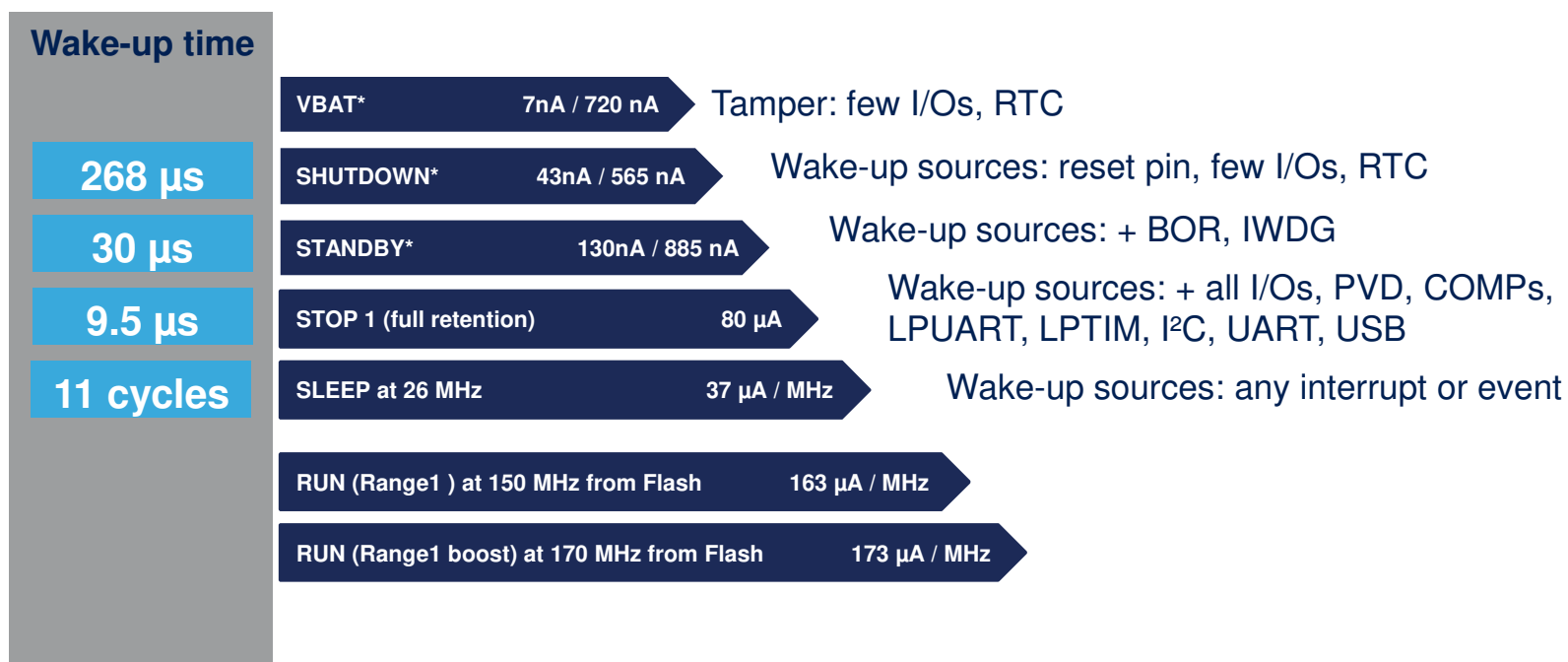
	Securable user memory	AES TRNG	PCROP	MPU	Readout protection	CRC	Write Protection
Secure firmware install (SFI)	●	●			●		
Secure Firmware upgrade (SFU)	●	●			●		●
Mutual Distrustful			●				
Firmware IP protection			●				
Secret key storage	●				●		
Secured communication		●				●	
Authentication	●	●			●		
Task cloisoning				●			



Dynamic Efficiency Modes

35

When Mainstream MCU Series meets low-power requirements



Conditions: 25°C, $V_{DD} = 3V$

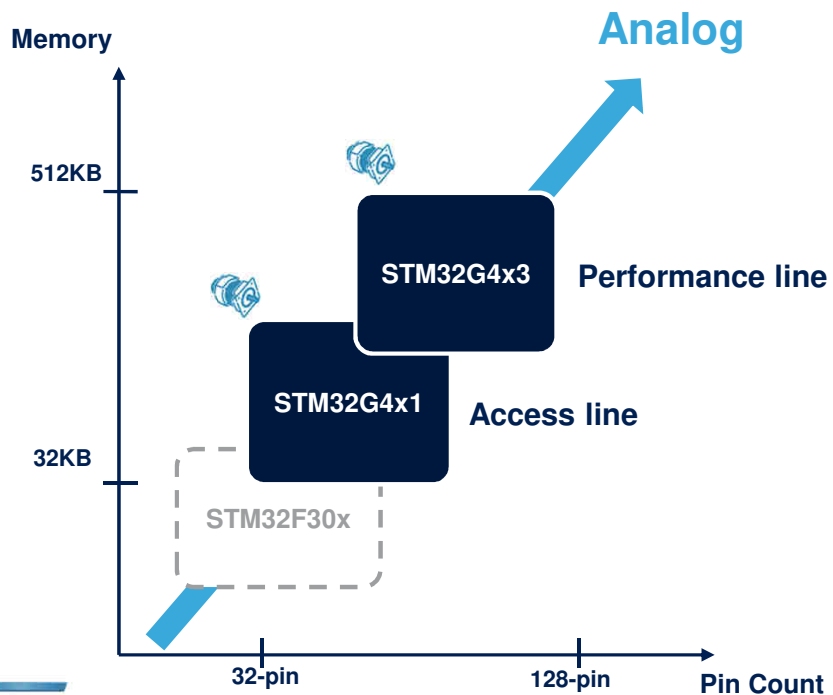
Note : * without RTC / with RTC



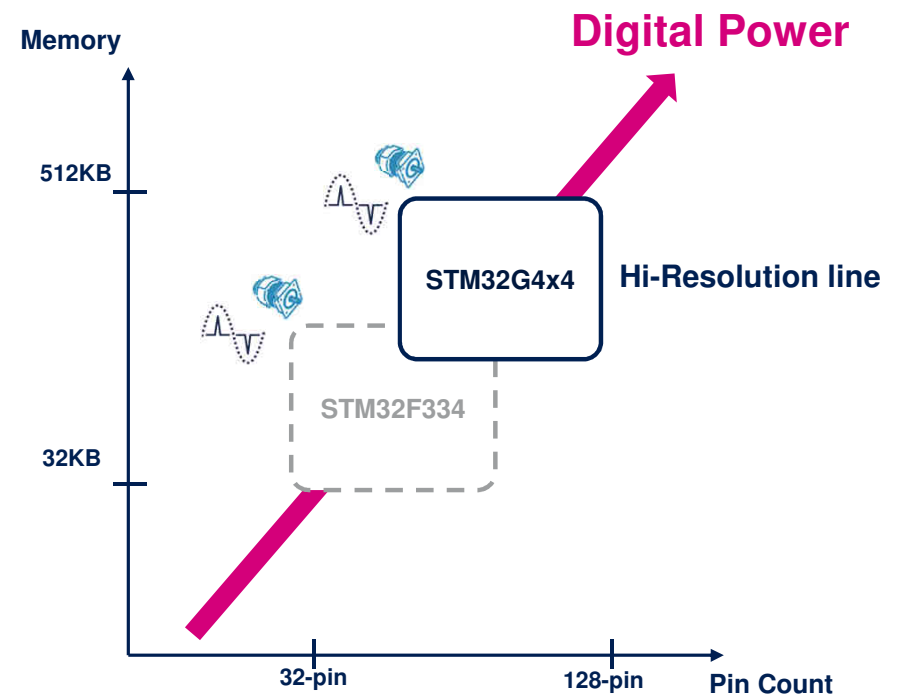
STM32G4 Products Lines

36

General Purpose



Applications Specific





Extensive & Innovative Peripheral Set

37

No compromise on what matters

Unit parameters	STM32G474 Hi-Resolution line	STM32G473 Performance line	STM32G431 Access line
Core, frequency	Arm Cortex-M4, 170 MHz		
Flash (max)	512 Kbytes (2x256KB dual bank)		128 Kbytes single bank
RAM (up to)	96 Kbytes		22 Kbytes
CCM –SRAM (code-SRAM)	32 Kbytes		10 Kbytes
12-bit ADC SAR	4x 12-bit 4 MSPS		2x 12-bit 4 MSPS
Comparator	7		4
Op amp with 4 built-in gain values with 1% accuracy	6		3
12-bit DAC	7		4
Motor Control timer	3x (170 MHz)		2x (170 MHz)
CAN-FD	3x		1x
12 channel Hi-resolution Timer	1x	-	-
Power supply	1.72 to 3.6 V		



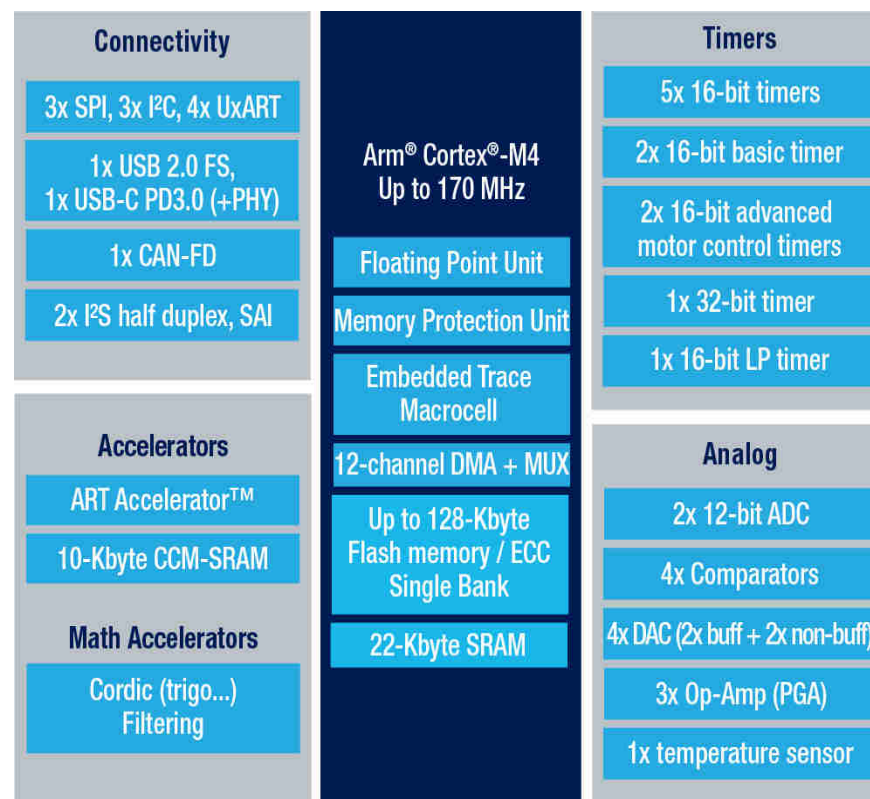
- | | | | |
|--|---|--|---|
| Connectivity | Arm® Cortex®-M4
Up to 170 MHz
213 DMIPS | | Timers |
| 4x SPI, 4x I²C, 6x UxART | | | 5x 16-bit timers |
| 1x USB 2.0 FS,
1x USB-C PD3.0 (+PHY) | | | 2x 16-bit basic timers |
| 3x CAN-FD | | | 3x 16-bit advanced
motor control timers |
| 2x I²S half duplex, SAI | | | 2x 32-bit timers |
| | | | 1x 16-bit LP timer |
| External interface | | | 1x HR timer (D-Power)
12-channel w/ 184ps
(A. delay line) |
| FSMC 8-/16-bit (TFT-LCD,
SRAM, NOR, NAND) | Floating Point Unit | | |
| Quad SPI | Memory Protection Unit | | |
| | Embedded Trace
Macrocell | | |
| | 16-channel DMA + MUX | | Analog |
| Accelerators | Up to 2x 256-Kbyte
Flash memory / ECC
Dual Bank | | 5x 12-bit ADC w/ HW overspl |
| ART Accelerator™ | 96-Kbyte SRAM | | 7x Comparators |
| 32-Kbyte CCM-SRAM | | | 7x DAC (3x buff + 4x non-buff) |
| Math Accelerators | | | 6x Op-Amp (PGA) |
| Cordic (trigo...)
Filtering | | | 1x temperature sensor |
| | | | Internal voltage reference |

- 



Access line [32KB .. 128KB] and up to 512KB in H1-2020 !

- **32-bit Arm Cortex-M4 core with FPU**
- **ART + CCM-SRAM + Mathematic Accelerators**
- **Single Bank Flash with ECC**
- **SRAM with Parity bit**
- **+/- 1% internal clock**
- **1.72 to 3.6V power supply**
- **Up to 125°C**

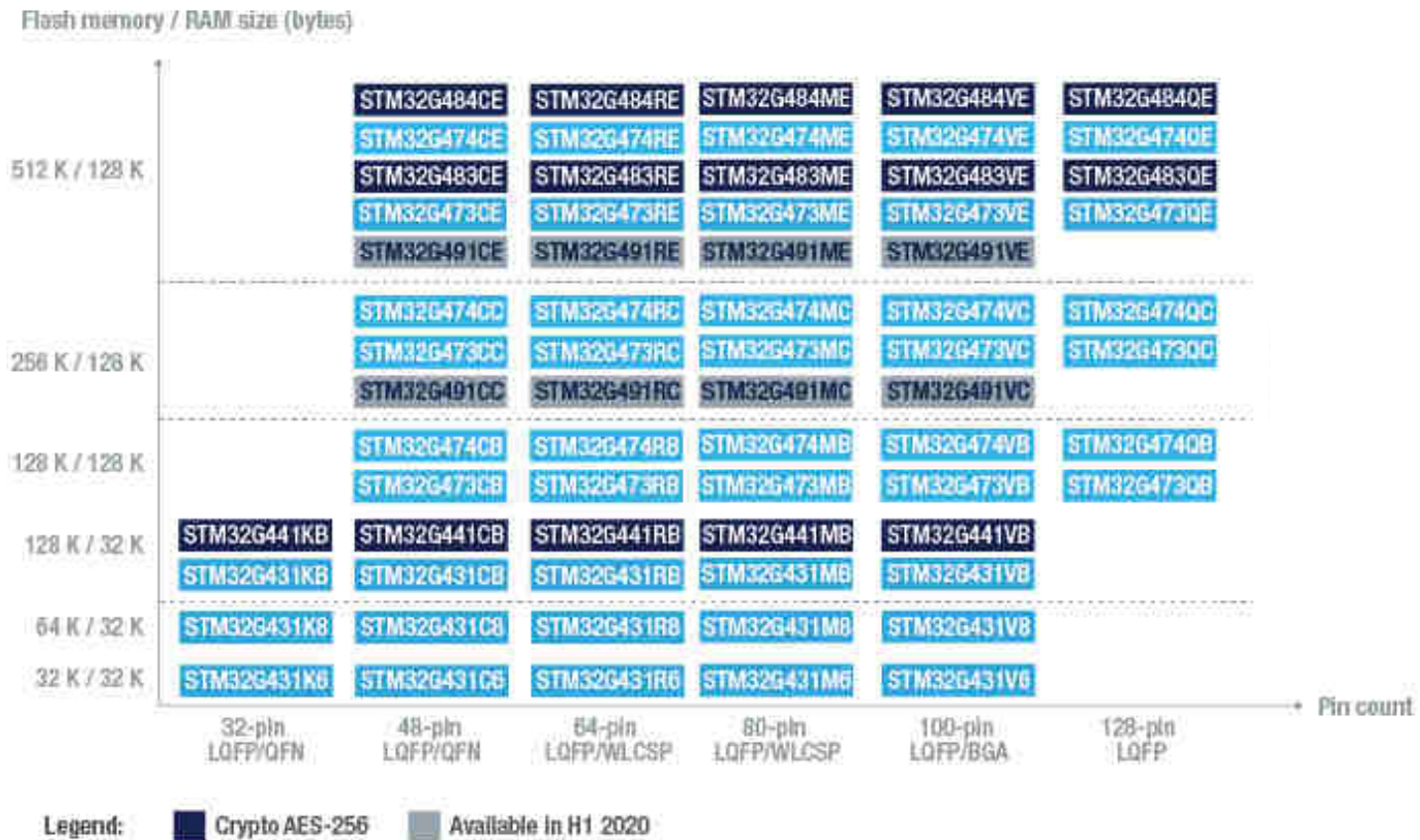


- **2x Advanced Motor Control timers**
- **Rich Advanced Analog**
- **CAN Flexible Data rate**
- **USB-C Power Delivery3.0**
- **Advanced Security and Safety features**
- **Robustness: highest level 5 / FTB/ESD - IEC 61000-4-4**



STM32G4 Portfolio

40





Broad Portfolio

41

Portfolio extended to support budget applications efficiently

More memory and pin counts

Flash memory (bytes)	32-pin LQFP QFN	48-pin LQFP QFN	64-pin LQFP BGA WLCSP	80-pin LQFP WLCSP	100-pin LQFP BGA	121-pin BGA	128-pin LQFP
512 K		✓	✓	✓	✓	✓	✓
256 K		✓	✓	✓	✓	✓	✓
128 K	✓	✓	✓	✓	✓	✓	✓
64 K	✓	✓	✓	✓	✓		
32 K	✓	✓	✓	✓	✓		

More packages



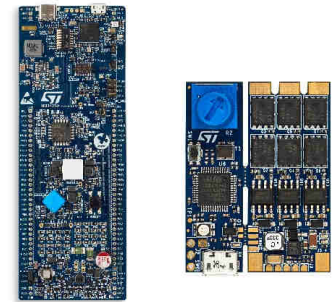
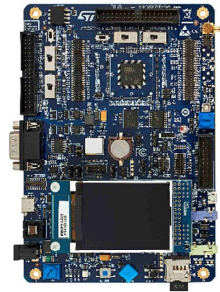
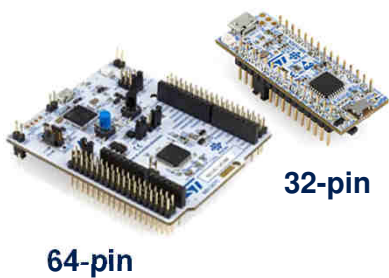
Note: new packages in STM32 portfolio



STM32G4 Hardware Solutions

42

Accelerate evaluation, prototyping and design



STM32 Nucleo

Flexible prototyping

- NUCLEO-G431RB
- NUCLEO-G474RE
- NUCLEO-G431KB

Evaluation boards

Full feature STM32G4 evaluation

- STM32G484E-EVAL
- STM32G474E-EVAL
- STM32G474E-EVAL1

Motor Control Pack

Full feature for Motor Control and Analog

- P-NUCLEO-IHM03

Discovery kits

Key feature prototyping

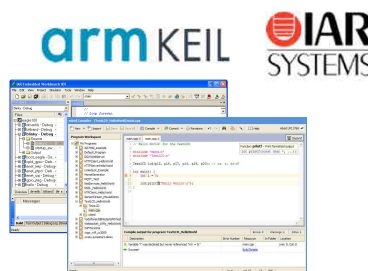
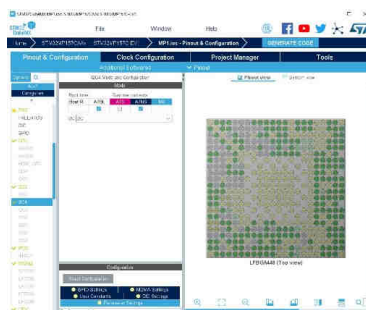
- B-G474E-DPOW1
- B-G431B-ESC1



STM32G4 Software Tools

43

Complete support of Arm Cortex-M ecosystem



All-in-one STM32 programming tool
Multi-mode, user-friendly



STM32CubeMX

STM32CubeMX

- Configure and generate Code
- Conflicts solver

IDEs Compile and Debug

Flexible Solutions

- Partners IDE, like IAR and Keil
- Free IDE based on Eclipse, like STM32CubeIDE*

STM32 Programming Tool

STM32CubeProgrammer

- Flash and/or system memory
- GUI or command line interface

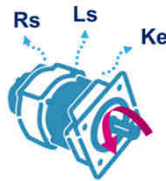


Dedicated Ecosystems

44

Motor Control

- **Complete ecosystem** (HW boards, SW Development Kit (SDK), docs and trainings)
 - **X-CUBE-MCSDK** (v5.4)
 - Motor Control FW library based on STM32Cube HAL and LL
 - Motor control workbench: Graphical configurator of the motor control library linked with STM32CubeMx
 - **P-NUCLEO-IHM03**: Motor Control Nucleo pack
 - NUCLEO-G431RB Nucleo-64
 - X-NUCLEO-IHM16M1 motor driver expansion board
 - Low Voltage motor
- **State of the art algorithms** (FOC, 6-step, sensorless...)
- **Motor Profiler**: Plug and spin your motor within less than one minute



Digital Power

- **Complete ecosystem** (HW boards, FW examples, SW tools, docs and trainings)
- **Dedicated HRTIM Cook Book - AN4539**: How to operate the Hi-Resolution timer in different topology
- **Digital Power training** (PSU and PFC) – based on STM32 G4 series – done in collaboration with Biricha (from Q4 2019)





STM32G4 Series – Take Away

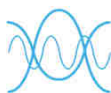
45

Analog-rich MCUs for mixed-signal applications



Performance

170MHz Cortex-M4 coupled with 3x accelerators

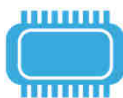


Rich and Advanced Integrated Analog

ADC, DAC, Op-Amp, Comp.



Safety and security focus



Large portfolio available from NOW!

32..512KB Flash memory

32..128-pin packages

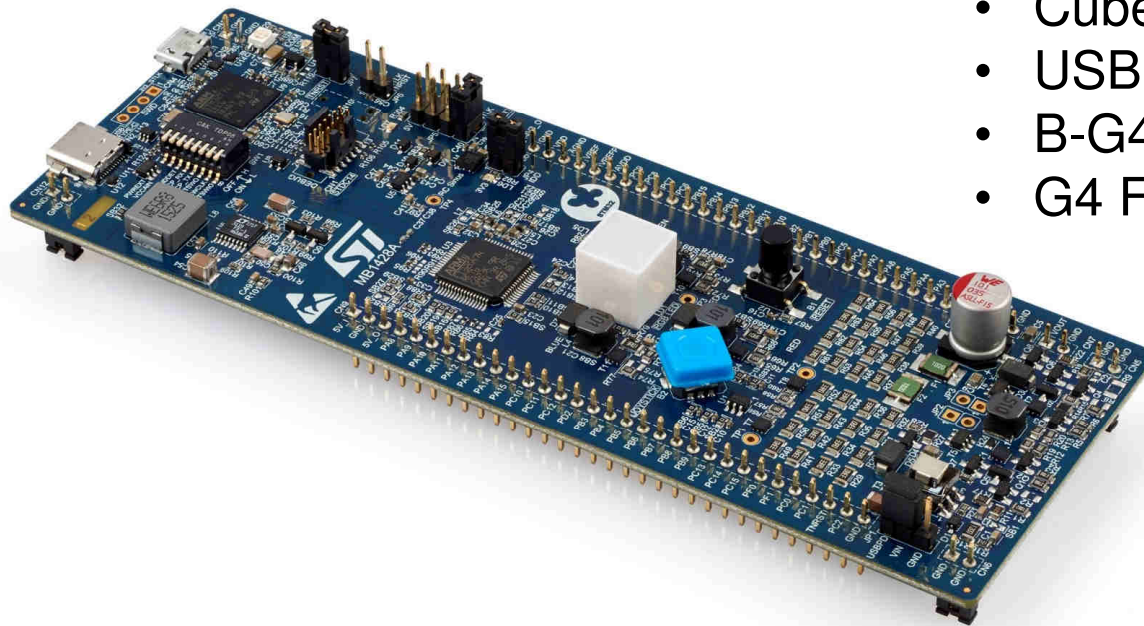
STM32G4 Mainstream Series Mixed Signal MCU

Demo Hands On





Systems Check



- Cube Programmer
- USB cable
- B-G474E-DPOW1 Discovery Board
- G4 Family Cube Library



B-G474E-DPOW1

48

Jumpers – 1 of 2

**Do not attach
USB cable here**

**All switches
Down**

CN2
USB Type-C™
receptacle

SW1
Octal mechanical
switch

U6
LTC3114-1

CN1
STDC14 debug
connector

LD7
5V LED

LD6
3V3 LED

CN3
USB micro-B
ST-LINK receptacle

LD9
COM LED

LD8
overcurrent LED

U9
STLINK-V3E MCU

JP7
TNRST

JP6
STLK NRST

JP5
5V_SELECTION

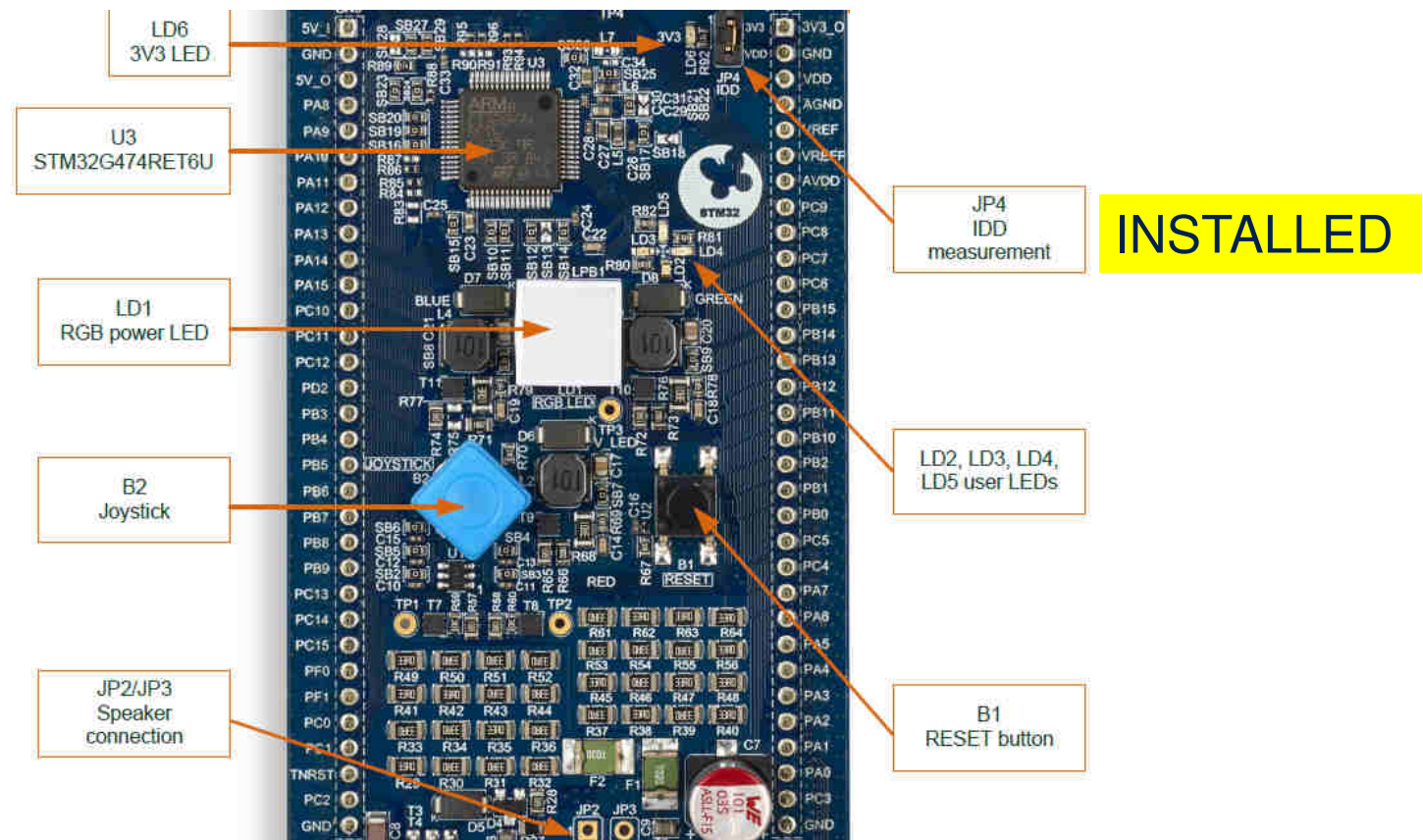
**Attach USB
cable here**

INSTALLED

OPEN

INSTALLED on STLK

Jumpers – 2 of 2

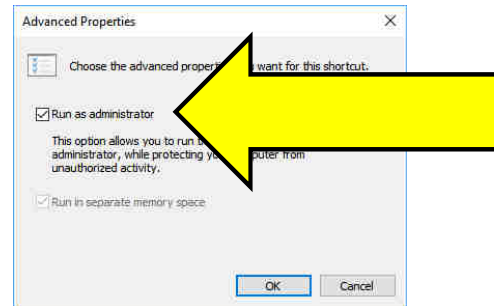
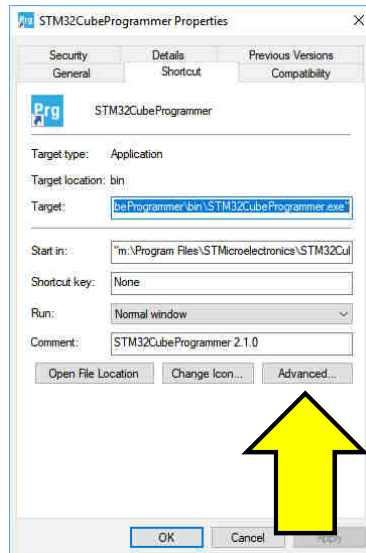




CubeProgrammer

50

Start up

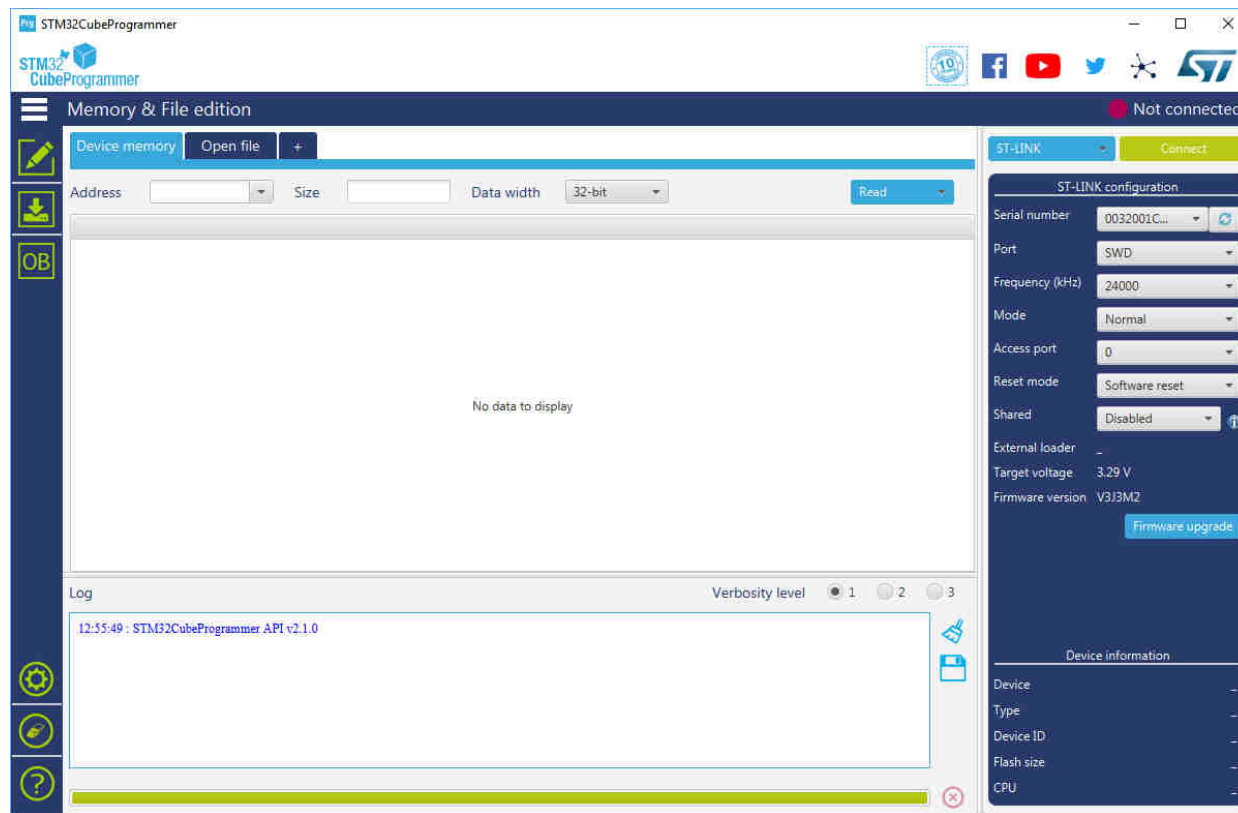




CubeProgrammer

51

Splash Page - Connect





CubeProgrammer

52

Splash Page – Open File

The screenshot shows the STM32CubeProgrammer application window. The 'Memory & File edition' tab is active, and the 'Open file' button is highlighted with a yellow arrow and the text 'Open File'. The main area displays a table of memory addresses and their corresponding data. The 'Log' panel at the bottom shows the progress of a data read operation.

Address	0	4	8	C	ASCII
0x08000000	20000560	080001ED	08002D6D	08002A31	... i...m...1*...
0x08000010	08002D69	0800052D	08003009	00000000	i-...0.....
0x08000020	00000000	00000000	00000000	08002DED	i-..
0x08000030	080005F3	00000000	08002D6F	08002DEF	6.....0-..i-..
0x08000040	08000207	08000207	08000207	08000207	
0x08000050	08000207	08000207	08000207	08000207	
0x08000060	08000207	08000207	08000207	08000531	1...
0x08000070	08000207	08000207	08000207	08000207	
0x08000080	08000207	08000207	08000207	08000207	
0x08000090	08000207	08000207	08000207	08000207	
0x080000A0	08000207	08000207	08000207	08000207	
0x080000B0	08000207	08000207	08000207	08000207	
0x080000C0	08000207	08000207	08000207	08000207	

Log

12:58:46 : Size: 16 Bytes
12:58:46 : Bank: 0x02
12:58:46 : Address: 0x0022070
12:58:46 : Size: 8 Bytes
12:58:46 : UPLOADING...
12:58:46 : Size: 1024 Bytes
12:58:46 : Address: 0x8000000
12:58:46 : Read progress:
12:58:46 : Data read successfully
12:58:46 : Time elapsed during the read operation is: 00:00:00.001

Verbosity level: 1 2 3

ST-LINK configuration

Serial number: 00000000
Port: COM1
Frequency (kHz): 24000
Mode: Normal
Access port: 0
Reset mode: Software reset
Shared: Disabled
External loader: ...
Target voltage: 3.29 V
Firmware version: V3.2M2
Firmware upgrade

Device information

Device: STM32G47x/G48x
Type: MCU
Device ID: 0x469
Flash size: 512 KB
CPU: Cortex-M4

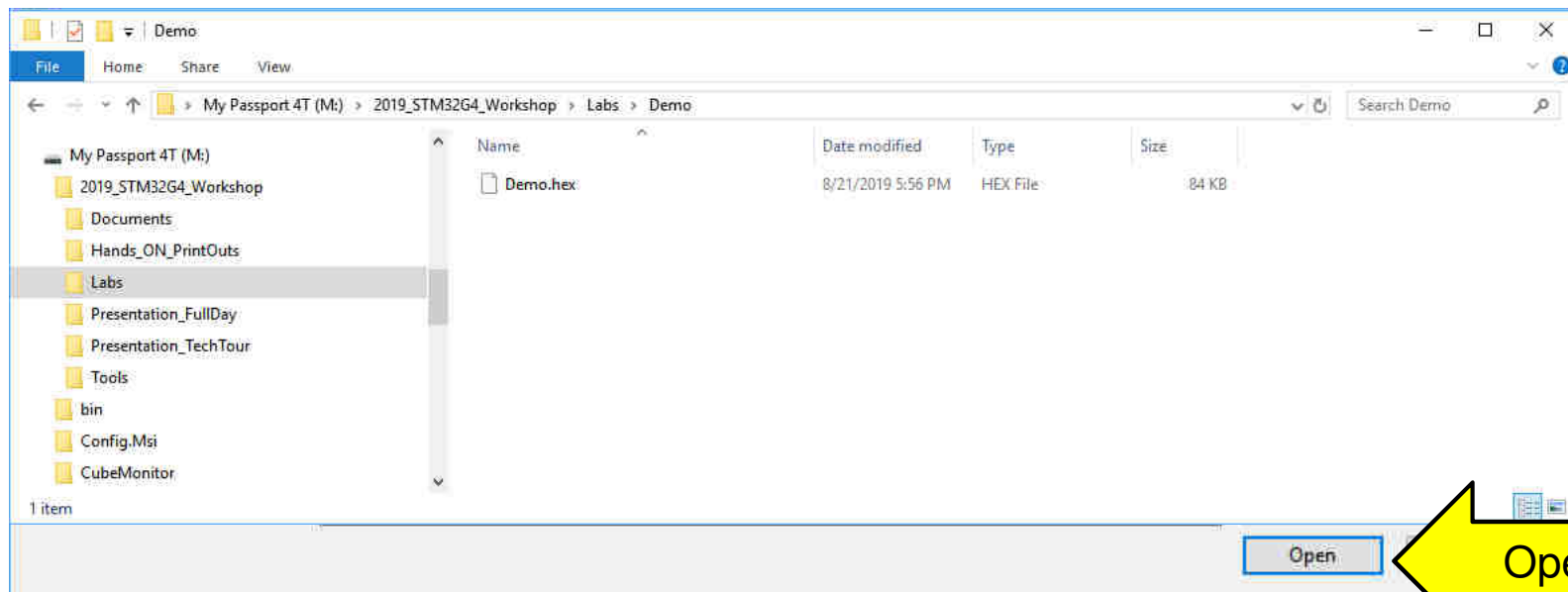




CubeProgrammer

54

File Folder





CubeProgrammer

55

Move Demo.hex from the PC to the Discovery board

The screenshot shows the STM32CubeProgrammer software interface. The 'Memory & File edition' window is open, displaying a table of memory addresses and their corresponding data. A yellow arrow points to the 'Download' button in the top right corner of the window.

Memory & File edition

Device memory: Demo.hex

Address: 0x8000000 Size: 0x400 Data width: 32-bit

Download

Address	0	4	8	C	ASCII
0x08000000	20000968	080074D9	08007319	08007318	h.. ùt...s...
0x08000010	0800731D	0800731F	08007321	00000000	.s...!s....
0x08000020	00000000	00000000	00000000	08007323	#s..
0x08000030	08007325	00000000	08007327	08007329	%s.....'s..)s..
0x08000040	080074F5	080074F9	080074FD	08007501	ôt..ùt..ýt...u..
0x08000050	08007505	08007509	0800750D	08007511	.u...u...u...u...
0x08000060	0800732D	08007515	08007333	08007519	-s...u...3s...u...
0x08000070	0800751D	08007521	08007525	08007529	.u...!u...%u...)u...
0x08000080	0800752D	08007531	08007535	08007539	-u...!u...5u...9u...
0x08000090	0800753D	08007541	08007545	08007339	=u...Au...Eu...9s...
0x080000A0	08007549	0800754D	08007551	08007555	Iu...Mu...Qu...Uu...
0x080000B0	08007559	0800755D	08007561	08007565	Yu...Ju...au...eu...
0x080000C0	08007569	0800756D	08007571	08007575	Iu...mu...qu...uu...

Log

Verbosity level: 1 2 3

12:58:46: Address: 0x40022070
12:58:46: Size: 8 Bytes
12:58:46: UPLOADING...
12:58:46: Size: 1024 Bytes
12:58:46: Address: 0x8000000
12:58:46: Read progress:
12:58:46: Data read successfully
12:58:46: Time elapsed during the read operation is: 00:00:00.001
13:51:18: Read File: M:\2019_STM32G4_Workshop\HandsOn_Labs\Demo\Demo.hex
13:51:18: Number of segments: 1
13:51:18: ...segment[0]: address=0x8000000, size=0x76D9

Port: COM1
Frequency (KHz): 4000
Mode: Normal
Access point: JTAG
Reset mode: Software reset
Shared: Connected
External loader: ...
Target voltage: 3.29 V
Firmware version: V3.13M2
Firmware upgrade

Device information

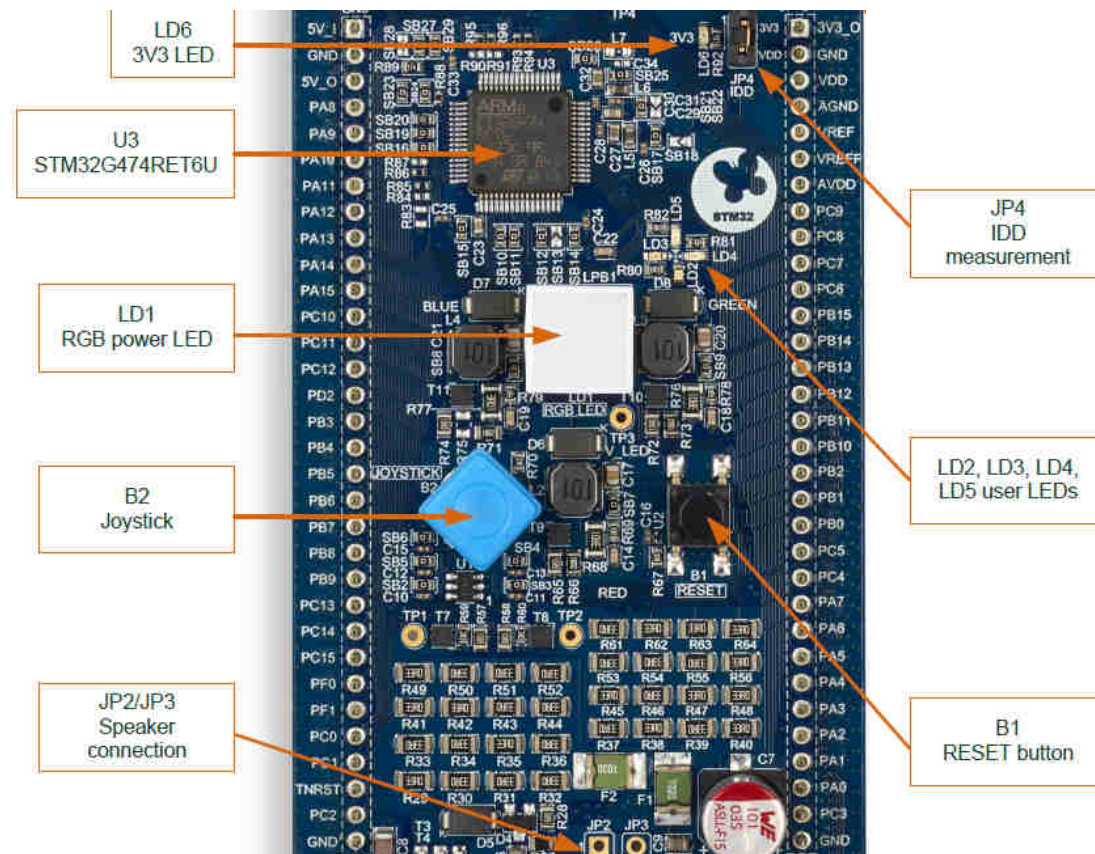
Device: STM32G47x/G48x
Type: MCU
Device ID: 0x469
Flash size: 512 KB
CPU: Cortex-M4



B-G474E-DPOW1

56

Press RESET (Black Button) – LED's cycle



LED's Cycle

PRESS Quickly

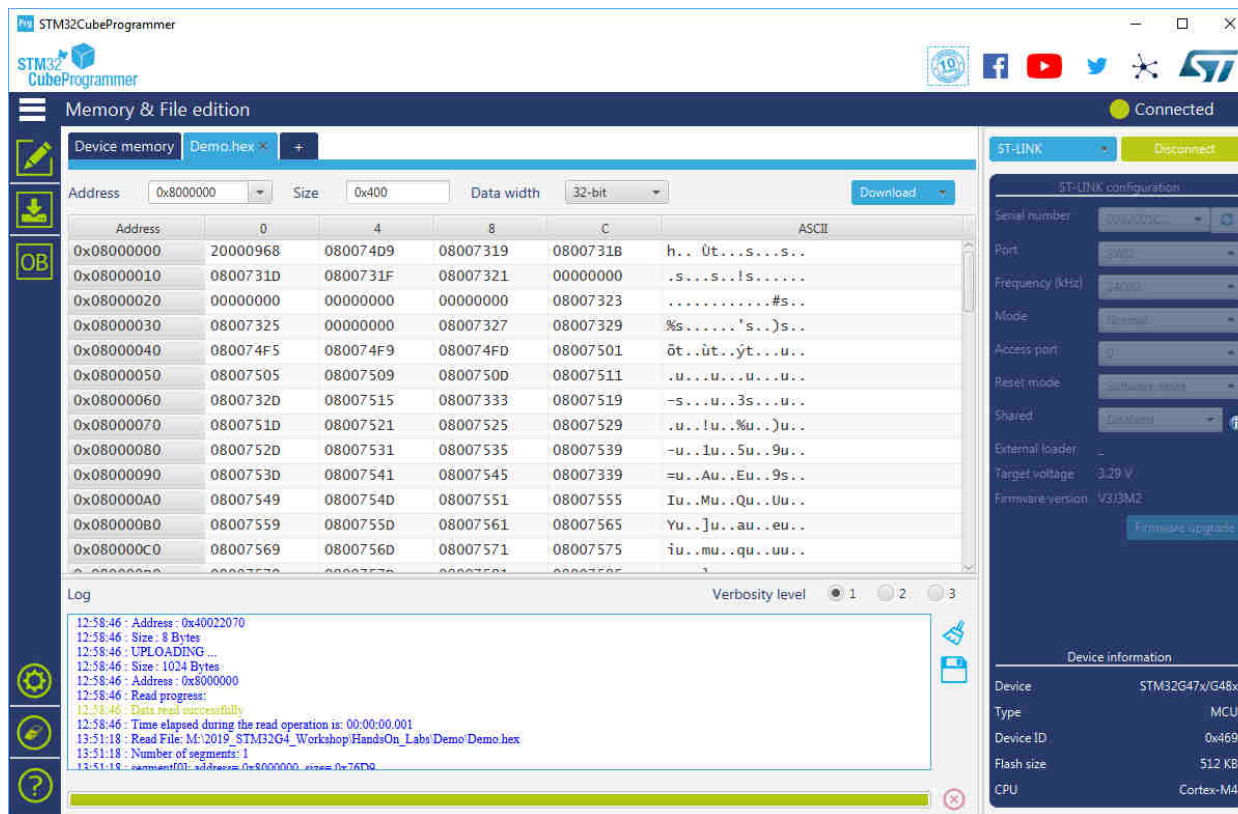


CubeProgrammer

57

Shut down the CubeProgrammer to release the VCOM port. ...

Exit





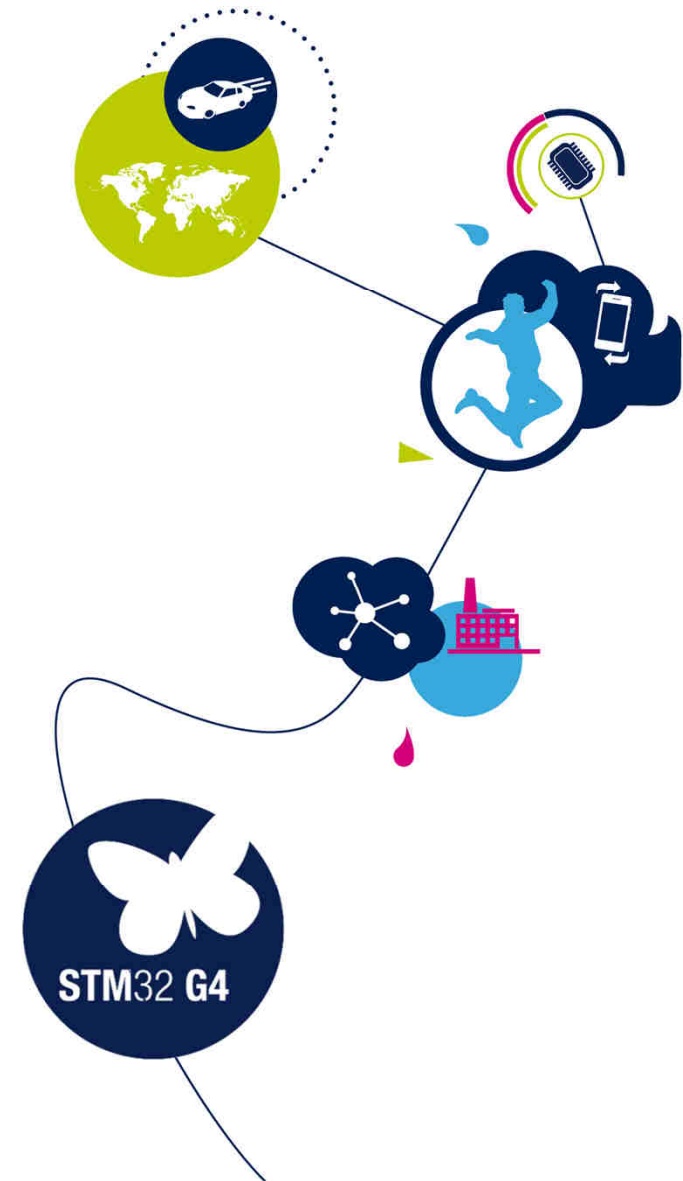
Getting back home – “CompletedLab”

For all of today’s Hands On Labs:

1. The *.hex file is available to you as a final product of the lab if you get lost along the way and still want to see the result of the lab.
2. The *.uvprojx file is available to you as a final result in case you want to pull it up in KEIL MDK-ARM and just walk through the files you see there.
3. The final STM32CubeMx *.ioc file for each lab is made available to you if you want to see the finished product of the lab or if you want to extend the lab with your own design ideas.

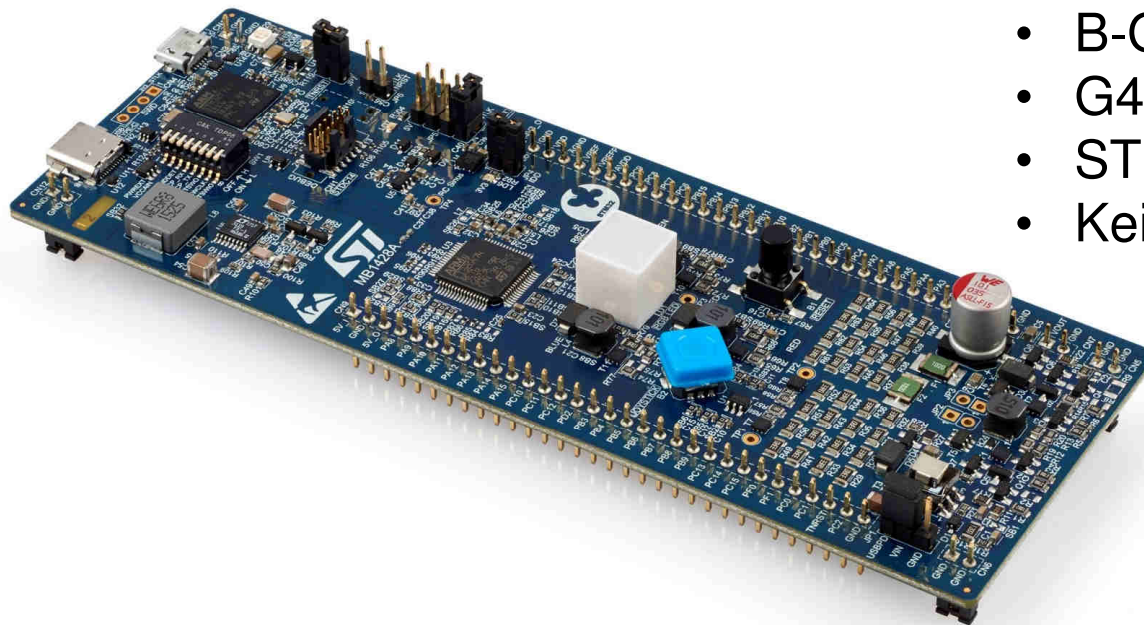
STM32G4 mixed-signal MCU workshop

*Hands On:
DAC Signal Generation*





Systems Check



- USB cable
- B-G474E-DPOW1 Discovery Board
- G4 Family Cube Library
- STM32CubeMx
- Keil MDK-ARM



Rich, Advanced Analog

61

Mixed-signal SoC for wide variety of applications

ADC (up to 5)	Values
Topology	SAR 12-bit + HW oversampling → 16-bit
Sampling rate	Up to 4 Msps
Input	Single-ended and differential
Offset and Gain compensation	Auto calibration to reduce gain and offset

DAC (up to 7)	Values
Sampling rate	15 Msps (internal) 1Msps (from buffered output)
Settling time	16ns

Op-Amp (up to 6)	Values
GBW	13 MHz
Slew rate	45 V/μs
Offset	3mV over full T° range 1.5mV @ 25°C
PGA Gain (accuracy)	2, 4, 8, 16, -1,-3,-7,-15 (1%) 32, 64, -31,-63 (2%)

Comparator (up to 7)	Values
Power supply	1.62 .. 3.6V
Propagation delay	16.7ns
Offset	-6 .. +2 mV
Hysteresis	8 steps: 0, 9, 18, 27, 36, 45, 54, 63 mV



Digital to Analog Converter

62

G4 DAC

- 3 external outputs:
 - DAC1-OUT1 and OUT2, 1Msps
- 4 internal outputs:
 - DAC3 and DAC4
 - OUT1 and OUT2, 15Msps
- Double data DMA
- Sample & Hold
- Left or Right data alignment
- Signed or unsigned data input format {Q1.15, Q1.11, Q1.7}

Internally connected to:

- Voltage reference
- External Triggers
- Comparator inputs
- Op Amp inputs
- Multiple Timers (+ HRTimer)
- DMA



Digital to Analog Converter

63

G4 DAC Sample and Hold

DAC conversion during the “sample and hold” mode has three phases:

1. Sampling phase: the “sample and hold” element is charged with the desired voltage.
2. Holding phase: the DAC output is tri-stated (High-Z) to maintain the “sample and hold” element’s stored electrical charge.
3. Refresh phase: due to leakage coming from several sources, a refresh phase is essential to maintain the output voltage at the desired value (+/-LSB).



Digital to Analog Converter

64

Dual Conversion from DMA

We will use the DAC dual channel mode with DMA to generate signals on both DAC channels at the same time.

DAC conversions are made with a timer trigger and data fed from DMA.

The timer is configured to trigger the DAC channels at about 1kHz.

DMA is feeding both channels at the time.

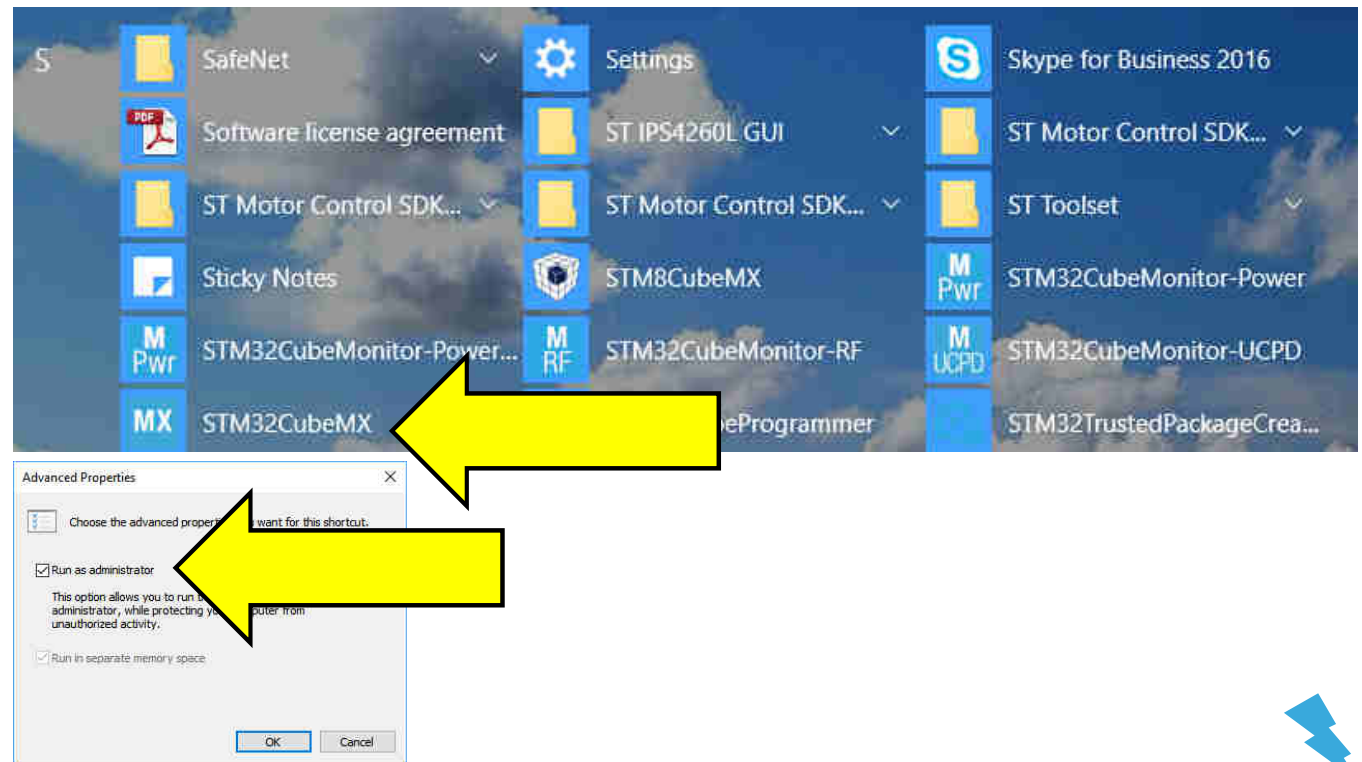
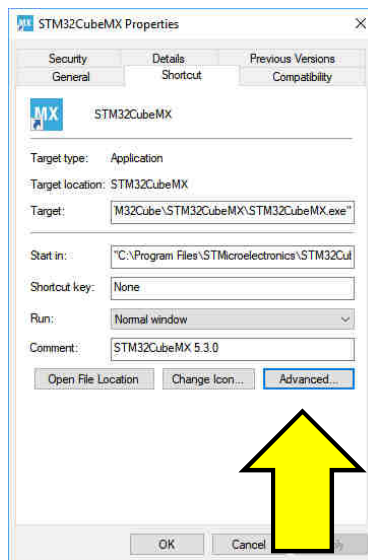
Channel 1 pattern is $\sim 0V / \sim 1.65V / \sim 3.3V / \sim 1.65V$

Channel 2 pattern is $\sim 1.65V / \sim 3.3V / \sim 1.65V / \sim 0V$

We will use the built in Logic Analyzer today instead of an o'scope on the DACs



Start up





Splash Page – G4 Library Installed?

The screenshot displays the STM32CubeMX software interface. On the left, the 'Embedded Software Packages Manager' window is open, showing a table of installed and available software packages. A yellow arrow points to the 'STM32G4' section of the table. The main splash page on the right features a 'Manage software installations' section with a 'CHECK FOR UPDATES' button and an 'INSTALL / REMOVE' button. A large yellow arrow points from the 'INSTALL / REMOVE' button to the right, with the text 'Install/Remove' written inside it. Below the splash page, there is a promotional banner for the 'New multicore STM32MP1 Series for Industrial and IoT applications' featuring the STM32MP1 logo and the OpenSTLinux Distribution logo.

Description	Installation	Available Version
STM32F7		
STM32G0		
STM32G4		
STM32Cube MCU Package for STM32G4 Series	1.1.0	1.1.0
STM32Cube MCU Package for STM32G4 Series	1.0.1	1.0.1

Details

From Local From Url Refresh Install Now Remove Now Close

Manage software installations

Check for STM32CubeMX and embedded software packages updates

CHECK FOR UPDATES

Install or remove embedded software packages

INSTALL / REMOVE

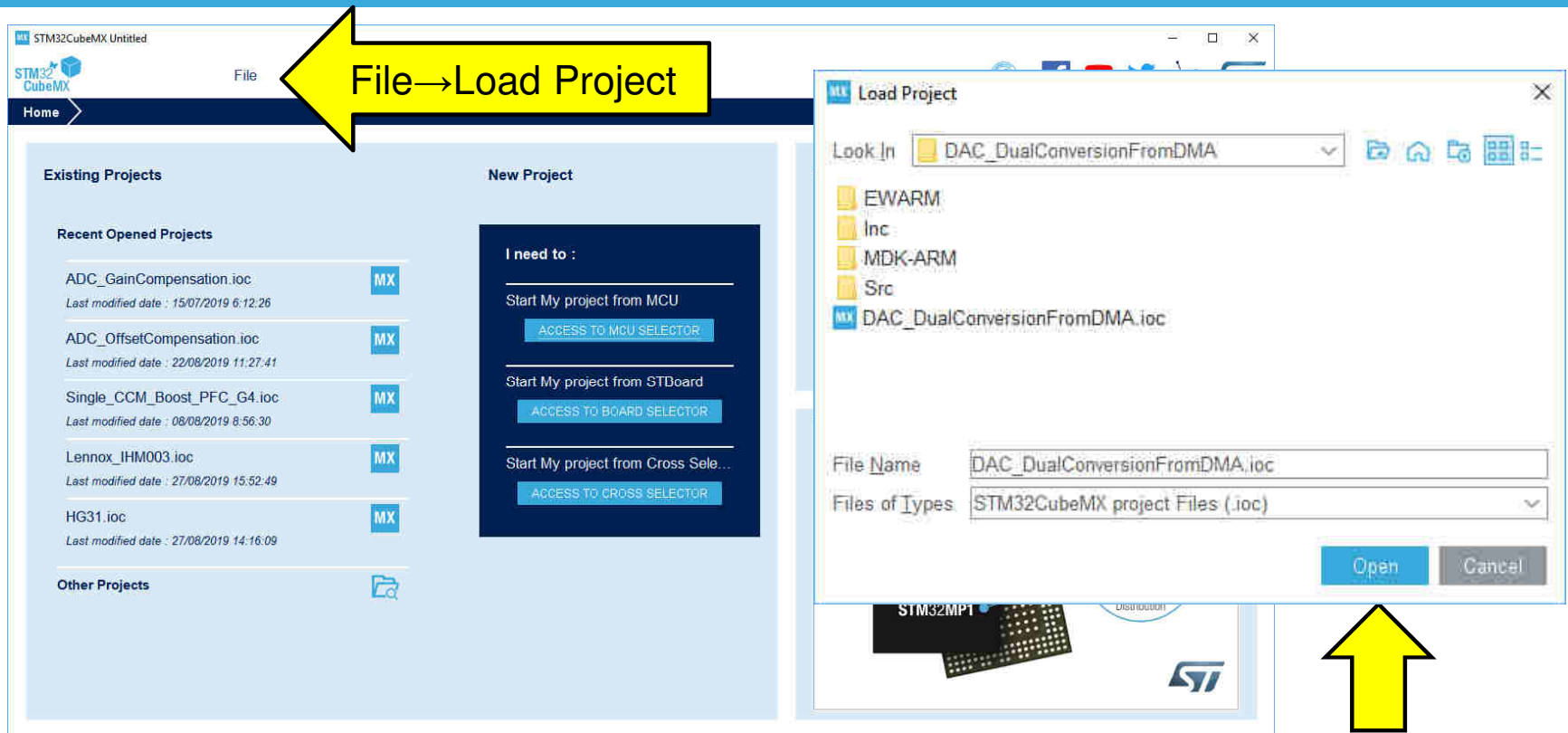
New multicore STM32MP1 Series for Industrial and IoT applications

STM32MP1

OpenSTLinux Distribution



Splash Page





Graphic Design Representation





DAC1, Timer 2

The screenshot shows the STM32CubeMX DAC1 configuration interface. The left sidebar lists various components, with DAC1 highlighted. The main panel shows the DAC1 Mode and Configuration settings. The 'Mode' section has 'OUT1 mode' and 'OUT2 mode' both set to 'Connected to external pin only', and 'External Trigger' is unchecked. The 'Configuration' section has tabs for 'Parameter Settings', 'User Constants', 'NVIC Settings', 'DMA Settings', and 'GPIO Settings'. The 'Parameter Settings' tab is active, showing 'DAC Out1 Settings' and 'DAC Out2 Settings'. The 'DAC Out1 Settings' section has 'Output Buffer' set to 'Enable', 'DAC High Frequency' set to 'Mode Automatic', 'DMA Double Data' set to 'Disable', 'Signed Format' set to 'Disable', 'Trigger' set to 'Timer 2 Trigger Out event', 'Trigger2' set to 'None', 'Wave generation mode' set to 'Disabled', 'User Trimming' set to 'Factory trimming', and 'Sample And Hold' set to 'Sampleandhold Disable'. The 'DAC Out2 Settings' section has 'Output Buffer' set to 'Enable', 'DAC High Frequency' set to 'Mode Automatic', 'DMA Double Data' set to 'Disable', 'Signed Format' set to 'Disable', 'Trigger' set to 'Timer 2 Trigger Out event', 'Trigger2' set to 'None', 'Wave generation mode' set to 'Disabled', 'User Trimming' set to 'Factory trimming', and 'Sample And Hold' set to 'Sampleandhold Disable'. The right panel shows the 'Pinout view' of the STM32G474RETx microcontroller. The DAC1_OUT1 pin is connected to PA1, and the DAC1_OUT2 pin is connected to PA2. The PA1 and PA2 pins are highlighted in green. Two yellow arrows point from the text 'DAC1' to the DAC1 configuration section. Two yellow arrows point from the text 'Timer 2 Trigger Out' to the 'Trigger' and 'Trigger2' settings in the DAC Out1 and DAC Out2 sections.

DAC1

Timer 2 Trigger Out

Timer 2 Trigger Out



Project Manager – Keil MDK-ARM V5

STM32CubeMX DAC_DualConversionFromDMA.ioc: STM32G474RETx

File Window Help

Home > STM32G474RETx > DAC_DualConversionFromDMA.ioc - Project Manager > GENERATE CODE

Pinout & Configuration Clock Configuration **Project Manager** Tools

Project

Project Settings

Project Name
DAC_DualConversionFromDMA

Project Location
\\jeffery blausen\STM32Cube\Repository\STM32Cube_FW_G4_V1.1.0\Projects\B-G474E-DPOW1\Examples\DAC_DualConversionFromDMA

Application Structure
Basic ☐ Do not generate the main()

Code Generator

Toolchain Folder Location
\\jeffery blausen\STM32Cube\Repository\STM32Cube_FW_G4_V1.1.0\Projects\B-G474E-DPOW1\Examples\DAC_DualConversionFromDMA

Toolchain / IDE
MDK-ARM V5

Advanced Settings

Linker Settings

Minimum Heap Size 0x200

Minimum Stack Size 0x400

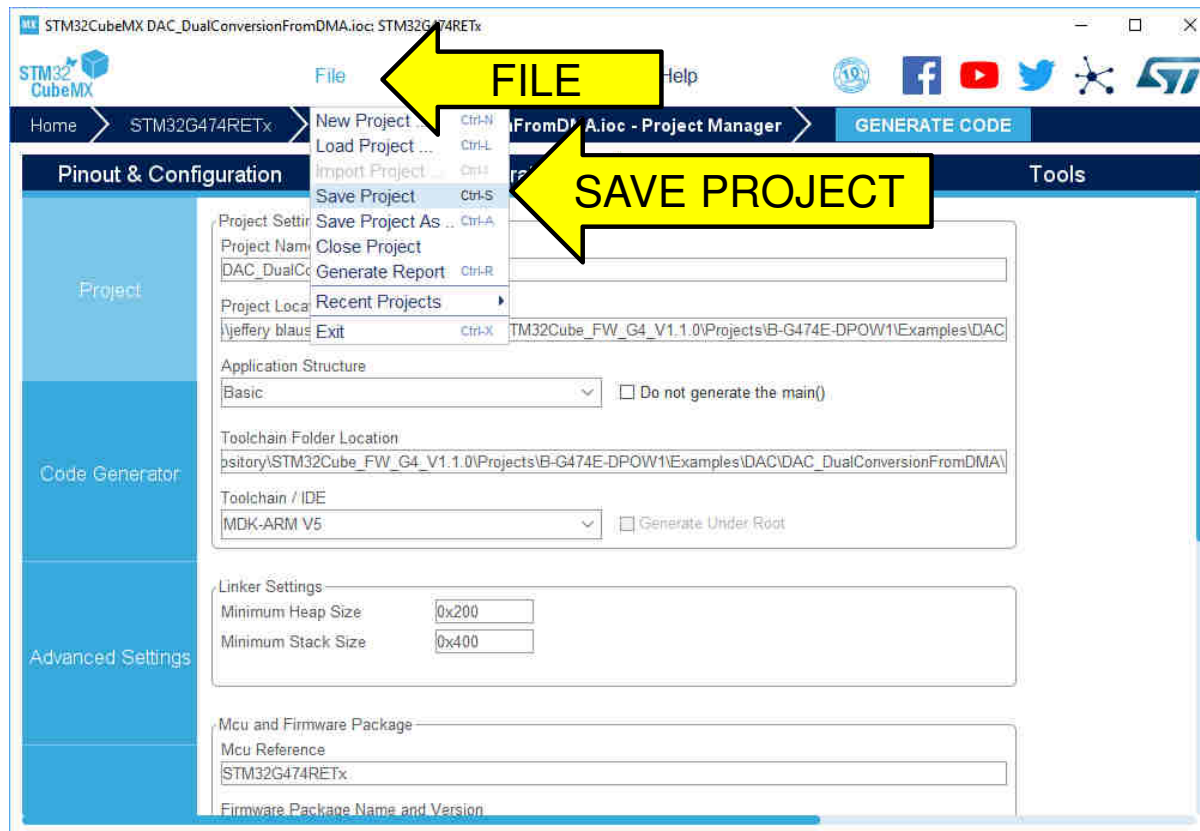
Mcu and Firmware Package

Mcu Reference
STM32G474RETx

Firmware Package Name and Version

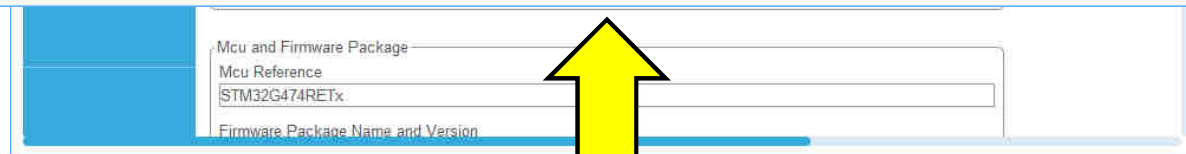
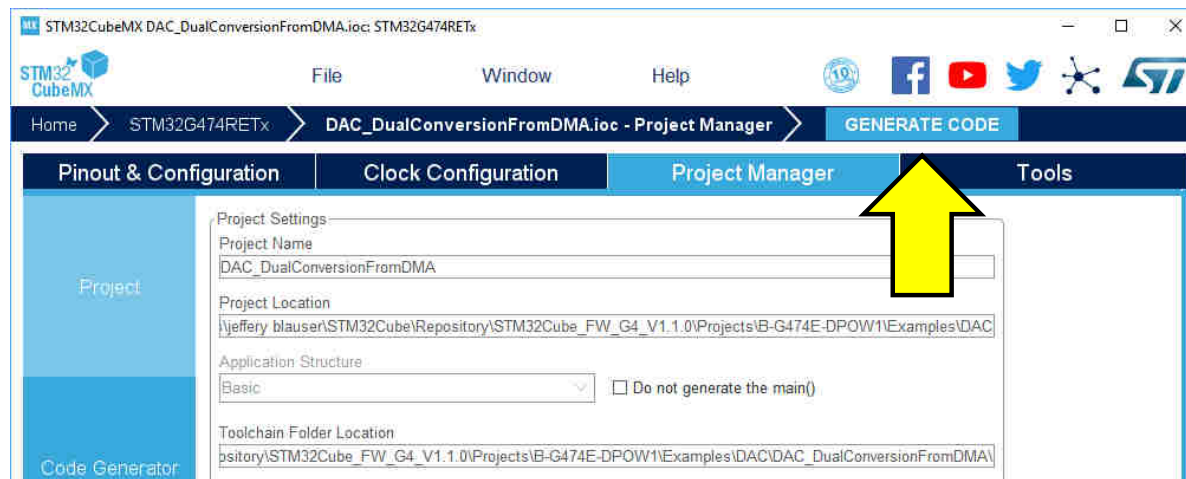


FILE→SAVE PROJECT





Generate Code





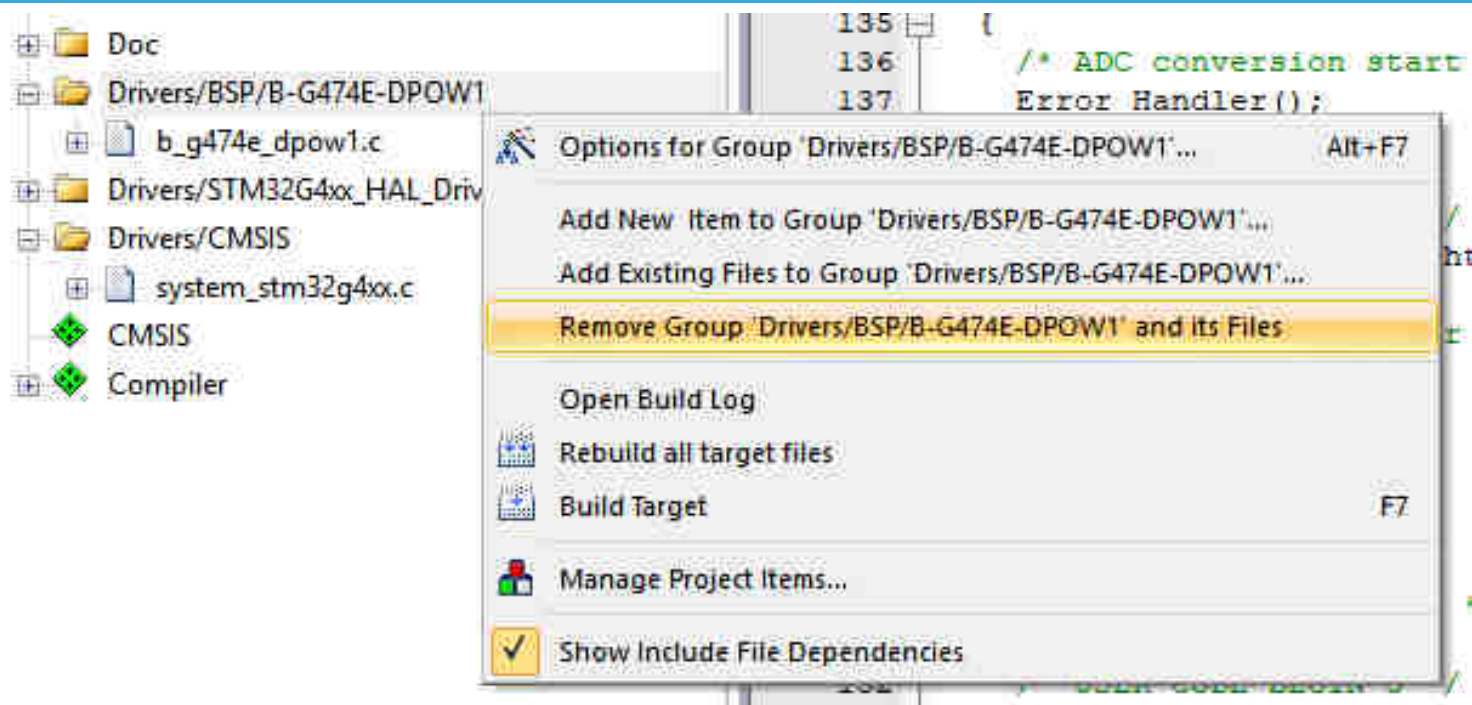
Keil Start up page

```
128 if (HAL_DACEx_DualStart_DMA(&hdac1, DAC_CHANNEL_1, wave, sizeof(wave))
129 {
130     /* DAC conversion start error */
131     Error_Handler();
132 }
133
134 BSP_LED_On(LED4);
135 while (1)
136 {
137     /* USER CODE END WHILE */
138
139     /* USER CODE BEGIN 3 */
140 }
141 /* USER CODE END 3 */
142
143
144 /**
145  * @brief System Clock Configuration
146  * @retval None
147  */
148 void SystemClock_Config(void)
149 {
150     RCC_OscInitTypeDef RCC_OscInitStruct = {0};
151     RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
152
153     /* Configure the main internal regulator output voltage
154     */
155     HAL_PWREx_ControlVoltageScaling(PWR_REGULATOR_VOLTAGE_SCALE1);
156     /* Initializes the CPU, AHB and APB busses clocks
157     */
158     RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
159     RCC_OscInitStruct.HSIState = RCC_HSI_ON;
160     RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
161     RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
162     RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSI;
163     RCC_OscInitStruct.PLL.PLLM = RCC_PLLM_DIV4;
164     RCC_OscInitStruct.PLL.PLLN = 75;
165     RCC_OscInitStruct.PLL.PLLP = RCC_PLLP_DIV2;
166     RCC_OscInitStruct.PLL.PLLQ = RCC_PLLQ_DIV2;
167     RCC_OscInitStruct.PLL.PLLR = RCC_PLLR_DIV2;
168     if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
169     {
170         Error_Handler();
171     }
172     /* Initializes the CPU, AHB and APB busses clocks
```





Delete Drivers Folder

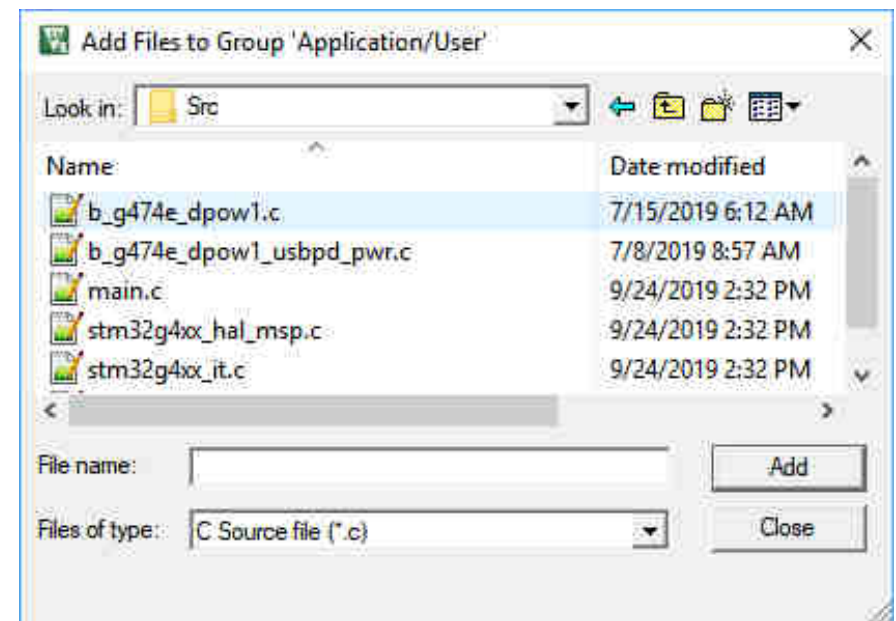
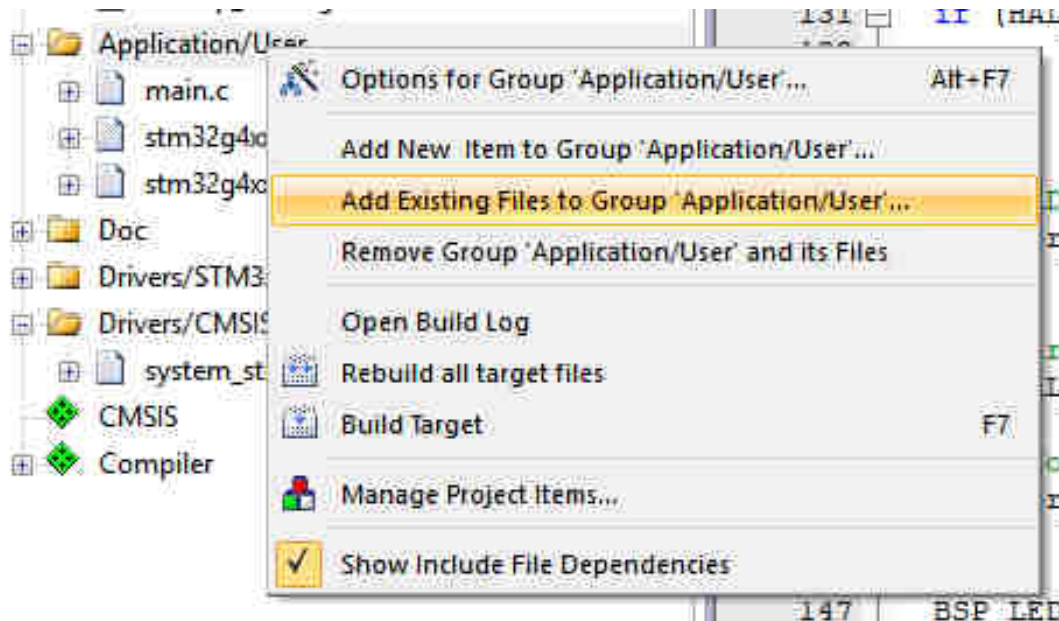




Keil MDK-ARM

75

Application/User folder add b_g474e_dpow1.c from SRC folder - CLOSE

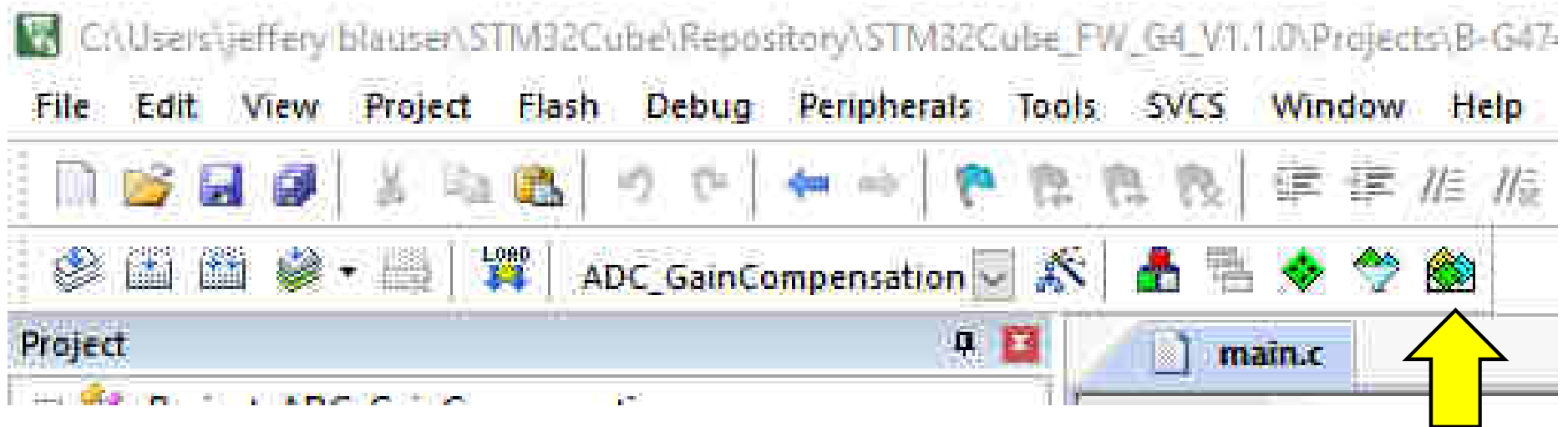




Keil MDK-ARM

76

Check the Device Pack





Keil MDK-ARM

77

G4 Device Pack

The screenshot shows the Keil MDK-ARM Pack Installer window. The 'Devices' tab is active, displaying a list of devices. The 'STM32G474' device is highlighted with a yellow arrow. The 'Packs' tab is also visible, showing a list of packs. The 'Keil:STM32G4xx_DFP' pack is highlighted with a yellow arrow. A yellow arrow points to the 'Exit' button in the top right corner. The output window at the bottom shows error messages related to downloading pack descriptions.

Device: STM32G474

Search:

Device

- Silicon Labs (906 Devices)
- Sinowealth (1 Device)
- SONIX (50 Devices)
- STMicroelectronics (1284 Devices)
 - STBlueNRG Series (1 Device)
 - STBlueNRG-1 Series (1 Device)
 - STBlueNRG-2 Series (1 Device)
 - STM32F0 Series (111 Devices)
 - STM32F1 Series (95 Devices)
 - STM32F2 Series (46 Devices)
 - STM32F3 Series (90 Devices)
 - STM32F4 Series (205 Devices)
 - STM32F7 Series (115 Devices)
 - STM32G0 Series (34 Devices)
 - STM32G4 Series (118 Devices)
 - STM32G431 (24 Devices)
 - STM32G441 (8 Devices)
 - STM32G471 (17 Devices)
 - STM32G473 (25 Devices)
 - STM32G474 (25 Devices)
 - STM32G483 (9 Devices)
 - STM32G484 (9 Devices)
 - STM32GBK1 (1 Device)
 - STM32H7 Series (22 Devices)
 - STM32L0 Series (152 Devices)
 - STM32L1 Series (81 Devices)

Summary

Pack

Pack	Action	Description
Device Specific	1 Pack	STM32G4xx_DFP
Generic	40 Packs	
Alibaba:AliOSThings	Install	AliOS Things software pack
ARM:AMP	Install	Software components for inter processor communication (Asymmetric Multi
ARM:CMSIS	Up to date	CMSIS (Cortex Microcontroller Software Interface Standard)
ARM:CMSIS-Driver	Up to date	CMSIS Drivers for external devices
ARM:CMSIS-Driver_Val...	Up to date	CMSIS-Driver Validation
ARM:CMSIS-FreeRTOS	Up to date	Bundle of FreeRTOS for Cortex-M and Cortex-A
ARM:CMSIS-RTOS_Val...	Up to date	CMSIS-RTOS Validation
ARM:mbedClient	Install	ARM mbed Client for Cortex-M devices
ARM:mbedTLS	Install	ARM mbed Cryptographic and SSL/TLS library for Cortex-M devices
ARM:minar	Install	mbed OS Scheduler for Cortex-M devices
ARM:TFM	Install	Trusted Firmware-M (TF-M) is the reference implementation of Arm's Platfor
EmbeddedOffice:Flexi...	Install	Flexible Safety RTOS
Keil:ARM_Compiler	Up to date	Keil ARM Compiler extensions for ARM Compiler 5 and ARM Compiler 6
Keil:IMXRT105x_MWP	Install+	NXN i.MX RT 105x MDK-Middleware examples and CMSIS-Drivers
Keil:Jansson	Install	Jansson is a C library for encoding, decoding and manipulating JSON data
Keil:MDK-Middleware	Up to date	Middleware for Keil MDK-Professional and MDK-Plus
lwIP:lwIP	Install	lwIP is a light-weight implementation of the TCP/IP protocol suite
MDK-Packs:AWS_IoT_...	Install+	SDK for connecting to AWS IoT from a device using embedded C
MDK-Packs:Azure_IoT	Install+	Microsoft Azure IoT SDKs and Libraries
MDK-Packs:cJSON	Install	Ultralightweight JSON parser in ANSI C
MDK-Packs:Google_Io...	Install+	Google Cloud IoT Device Connector
MDK-Packs:IoT_Socket	Install	Simple IP Socket (BSD like)
MDK-Packs:Jsmin	Install	Minimalistic JSON parser/tokenizer in C
MDK-Pack:PaHo-MQ...	Install+	Embedded MQTT C/C++ Client Libraries

Output

Cannot download file http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54018M_DFP.pdsc: Illegal MIME type 'text/html'

Cannot download file http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54018_DFP.pdsc: Illegal MIME type 'text/html'

Cannot download file http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54101_DFP.pdsc: Illegal MIME type 'text/html'

Action (2 left): Update Pack descriptions, download http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54102_DFP.pdsc

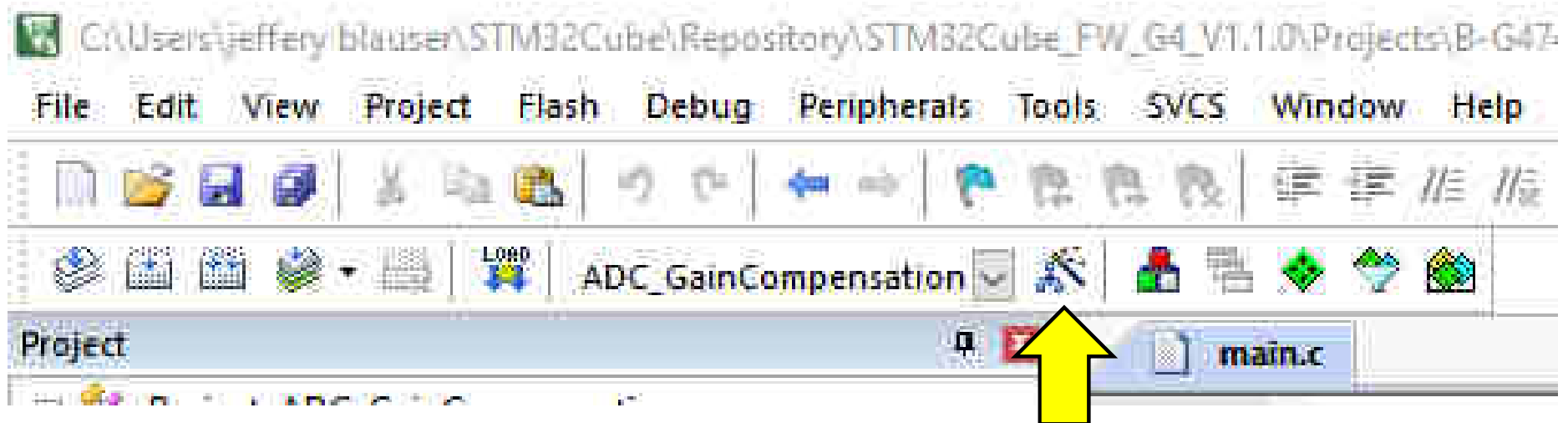
0% ONLINE



Keil MDK-ARM

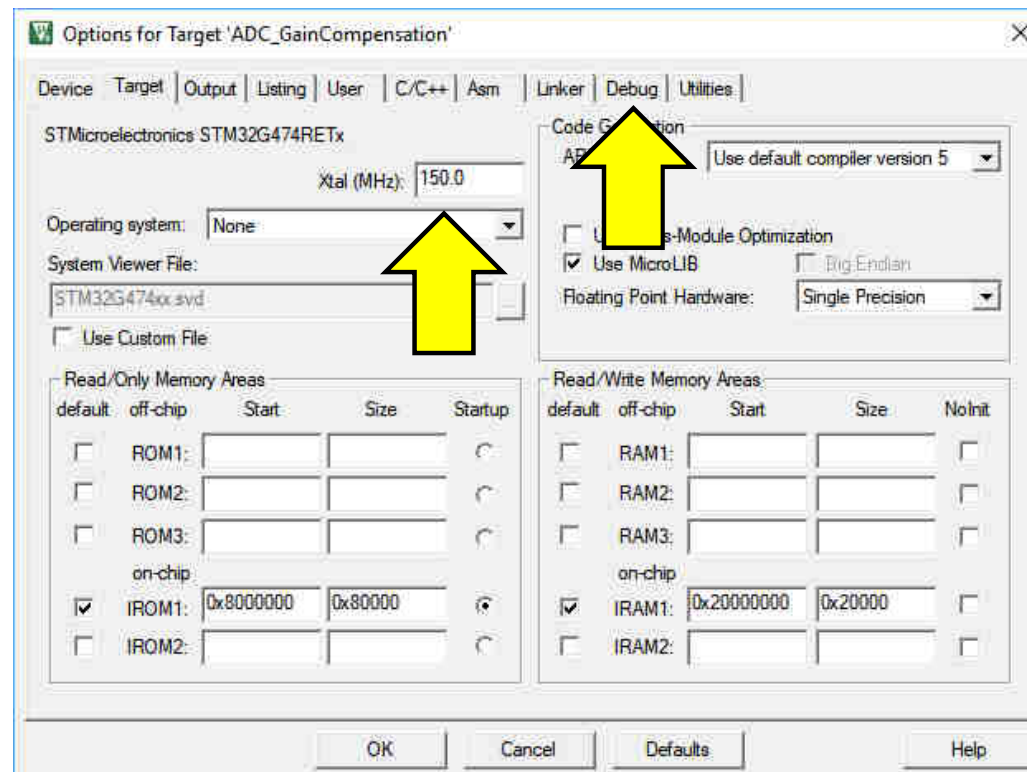
78

Check the Options



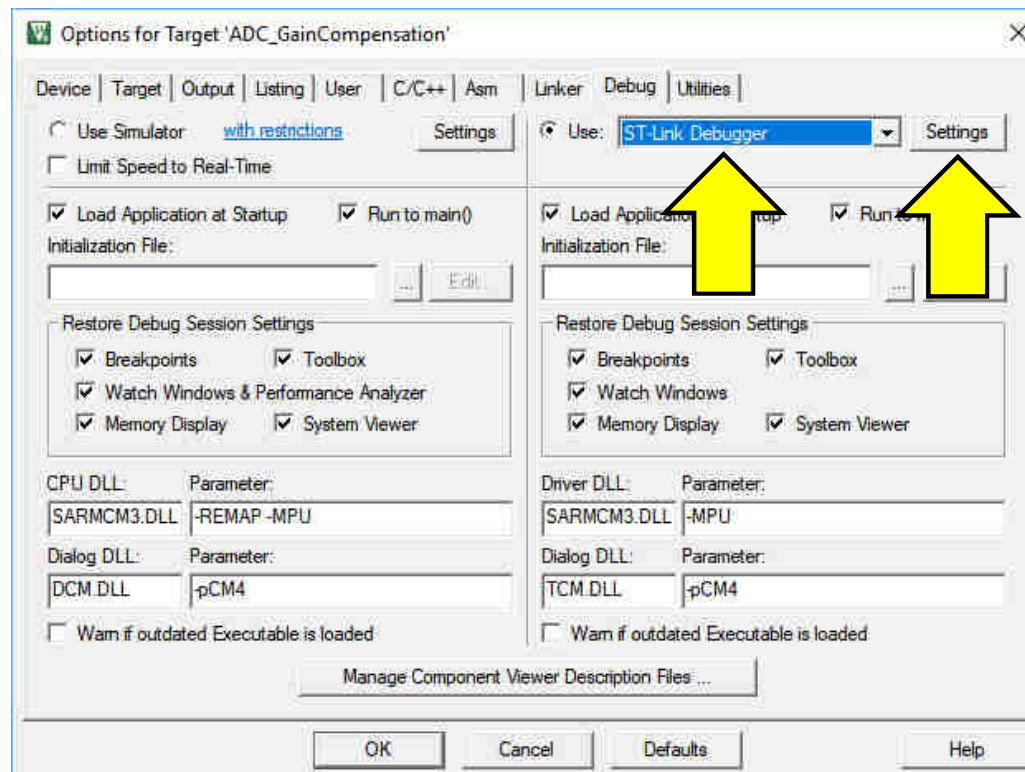


Clock speed and Debug



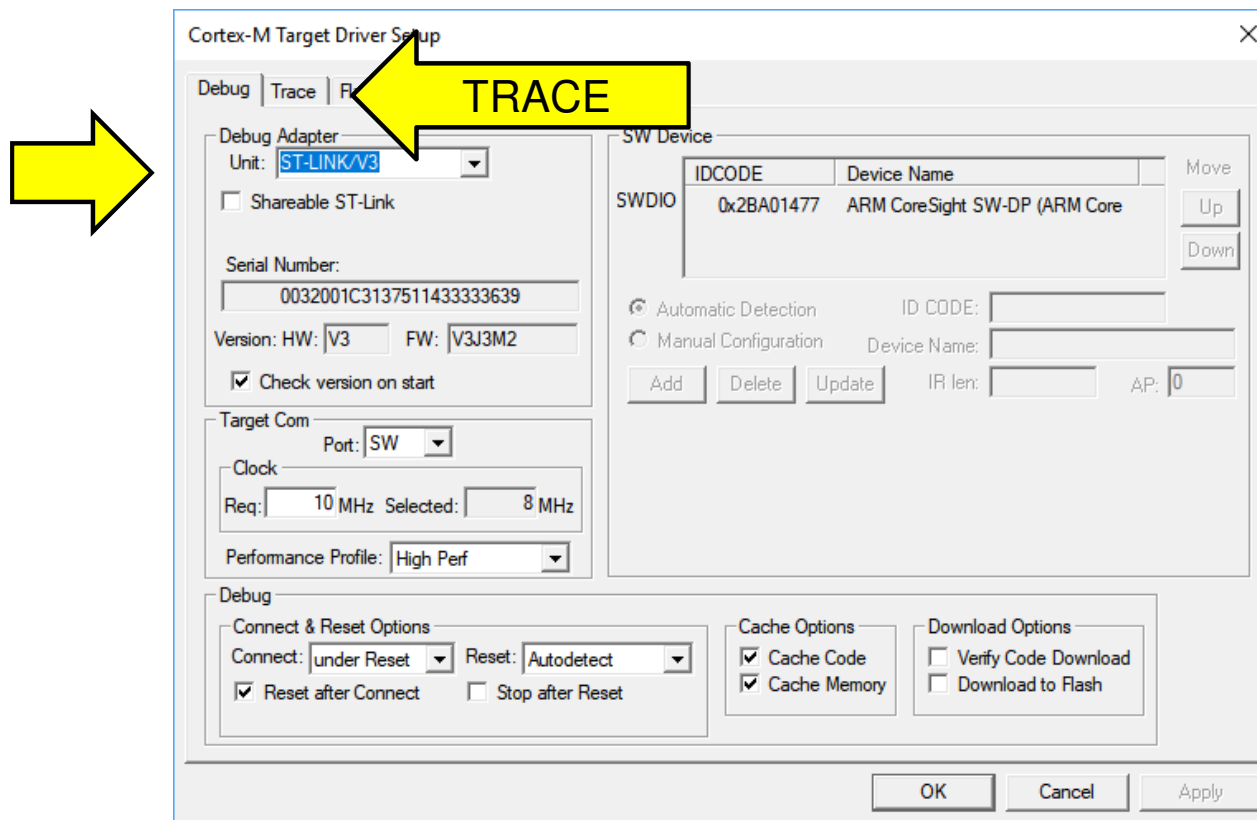


ST-Link Debugger - Settings





ST-Link/V3 & TRACE





Core Clock & TRACE Enable

The screenshot shows the 'Cortex-M Target Driver Setup' dialog box with the 'Trace' tab selected. The following settings are highlighted with yellow arrows and text boxes:

- Core Clock:** 150.000000 MHz (Arrow labeled '150')
- Trace Enable:** ☒ (Arrow labeled 'TRACE Enable')
- Trace Port:** Serial Wire Output - UART/NRZ
- SWO Clock Prescaler:** 13
- Autodetect:** ☒
- SWO Clock:** 11.538461 MHz
- Timestamps:** ☒ Enable, Prescaler: 1
- PC Sampling:** Prescaler: 1024*16, ☐ Periodic, Period: <Disabled>, ☐ on Data R/W Sample
- Trace Events:** ☐ CPI: Cycles per Instruction, ☐ EXC: Exception overhead, ☐ SLEEP: Sleep Cycles, ☐ LSU: Load Store Unit Cycles, ☐ FOLD: Folded Instructions, ☒ EXCTRC: Exception Tracing (Arrow labeled 'Uncheck EXCTRC')
- ITM Stimulus Ports:** Enable: 0xFFFFFFFF, Privilege: 0x00000008. The table below shows the status of various ports:

Port	31	24	23	16	15	8	7	0
Enable	☒	☒	☒	☒	☒	☒	☒	☒
Privilege	☒	☒	☒	☒	☒	☒	☒	☒

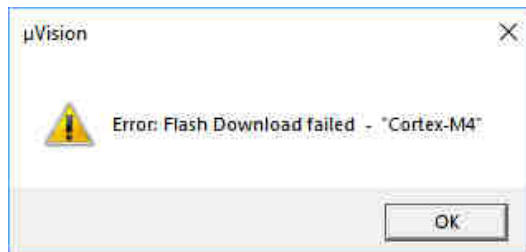
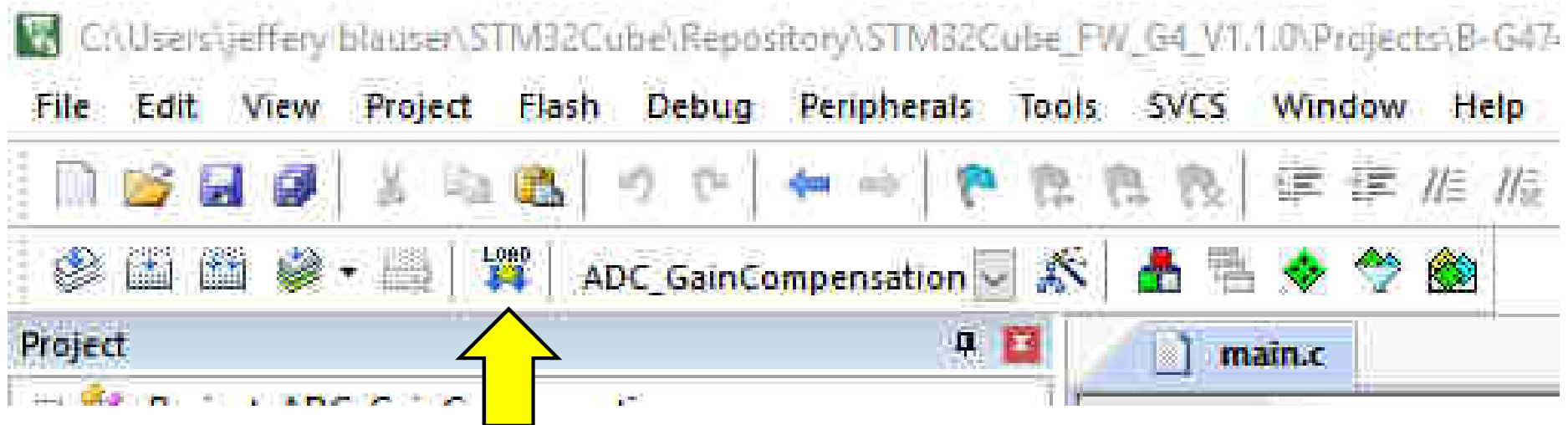
Arrows labeled 'Uncheck all but 15' point to the 'Port 15..8' and 'Port 7..0' columns. The 'OK' button is highlighted with a yellow arrow labeled 'OK'.

Advanced settings:

- ☐ Ignore packets with no SYNC
- ☐ Overwrite CYCCNT



Load



Build Output

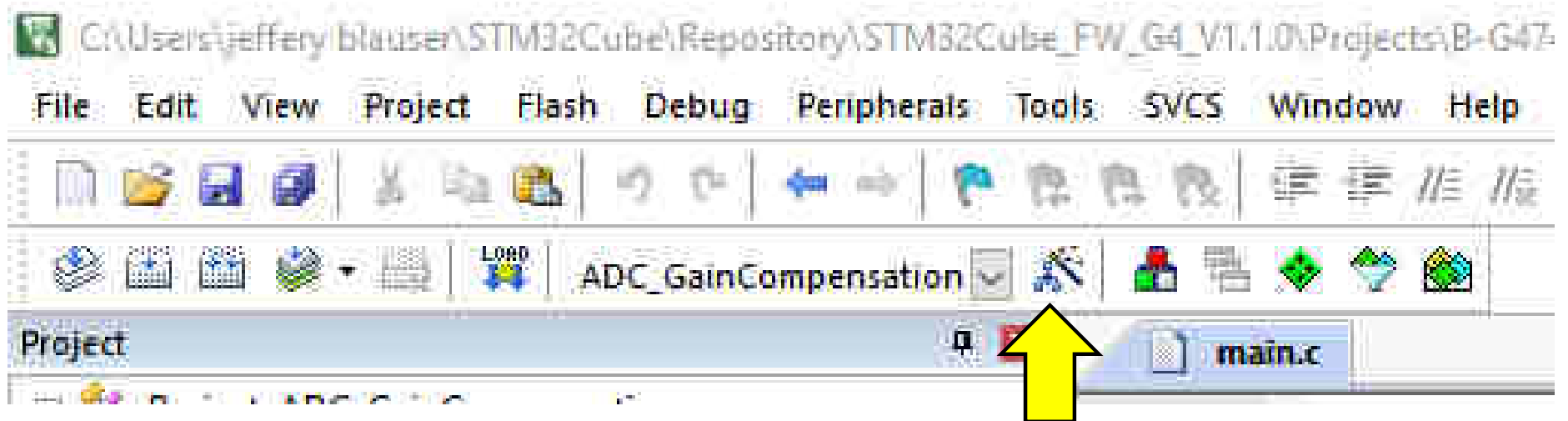
```
Contents mismatch at: 08001860H (Flash=38H Required=C5H) !
Contents mismatch at: 08001861H (Flash=46H Required=F8H) !
Contents mismatch at: 08001862H (Flash=FEH Required=88H) !
Contents mismatch at: 08001863H (Flash=F7H Required=00H) !
Contents mismatch at: 08001864H (Flash=B9H Required=60H) !
Contents mismatch at: 08001865H (Flash=FCH Required=6CH) !
Too many errors to display !
Error: Flash Download failed - "Cortex-M4"
```



Keil MDK-ARM

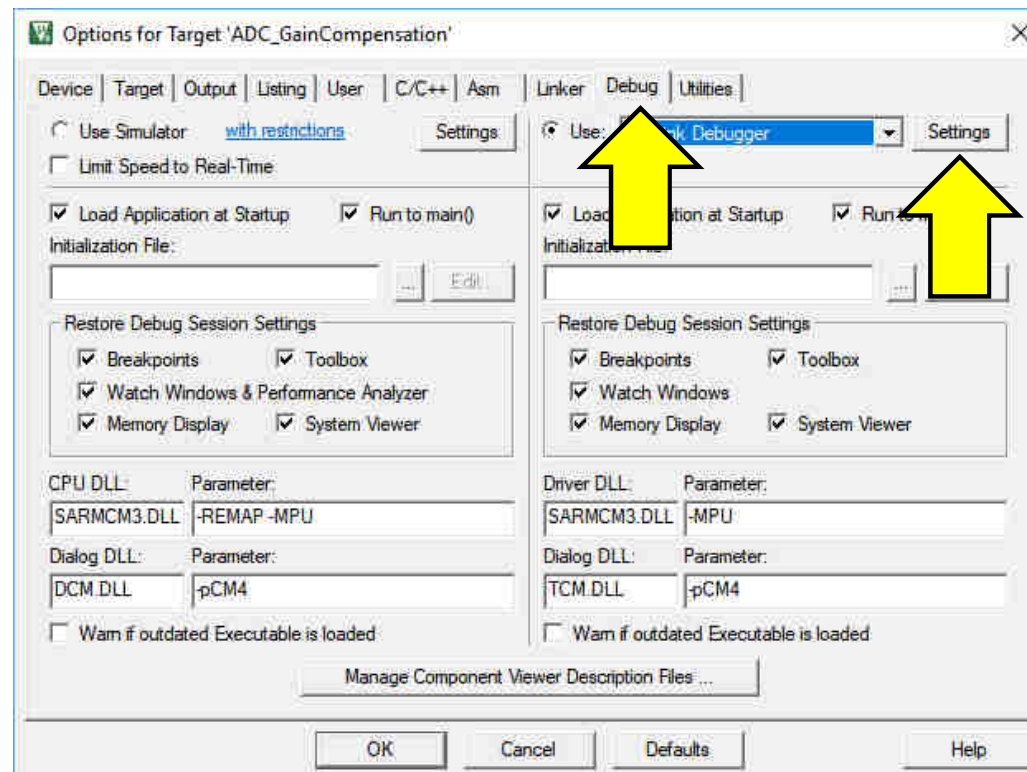
84

Options



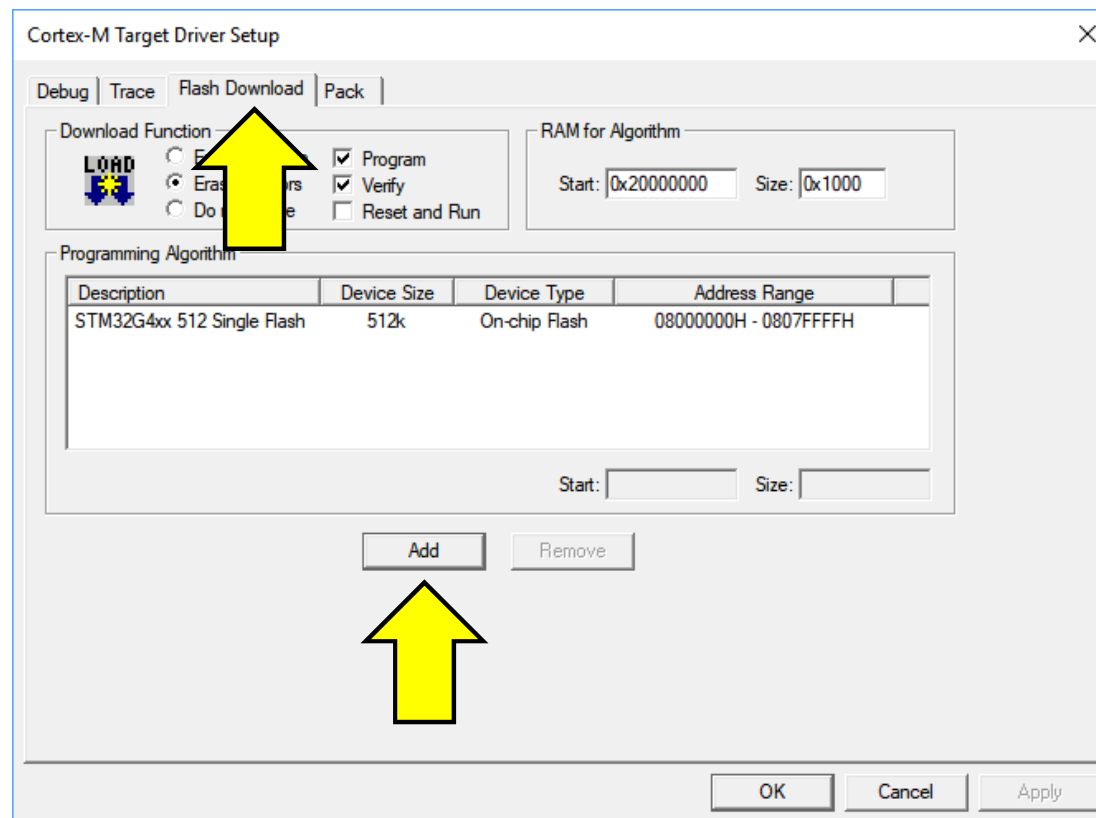


ST-Link Debugger - Settings



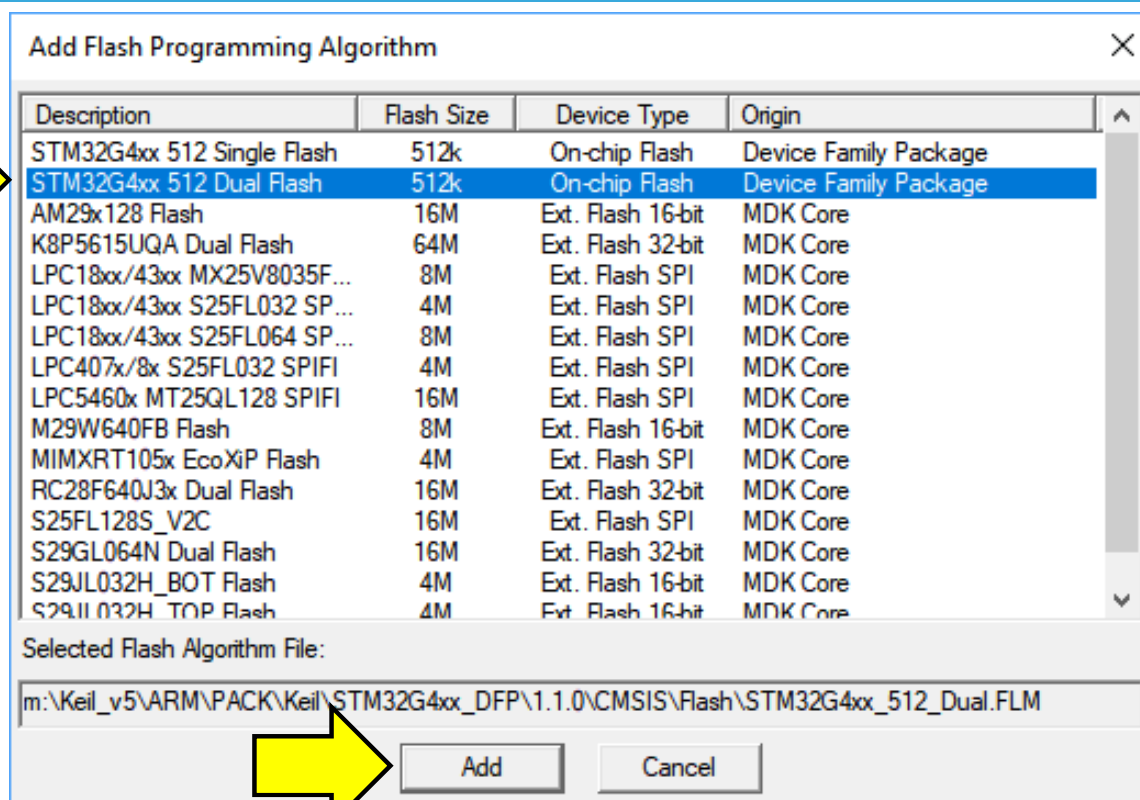
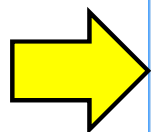


Flash Download → Add

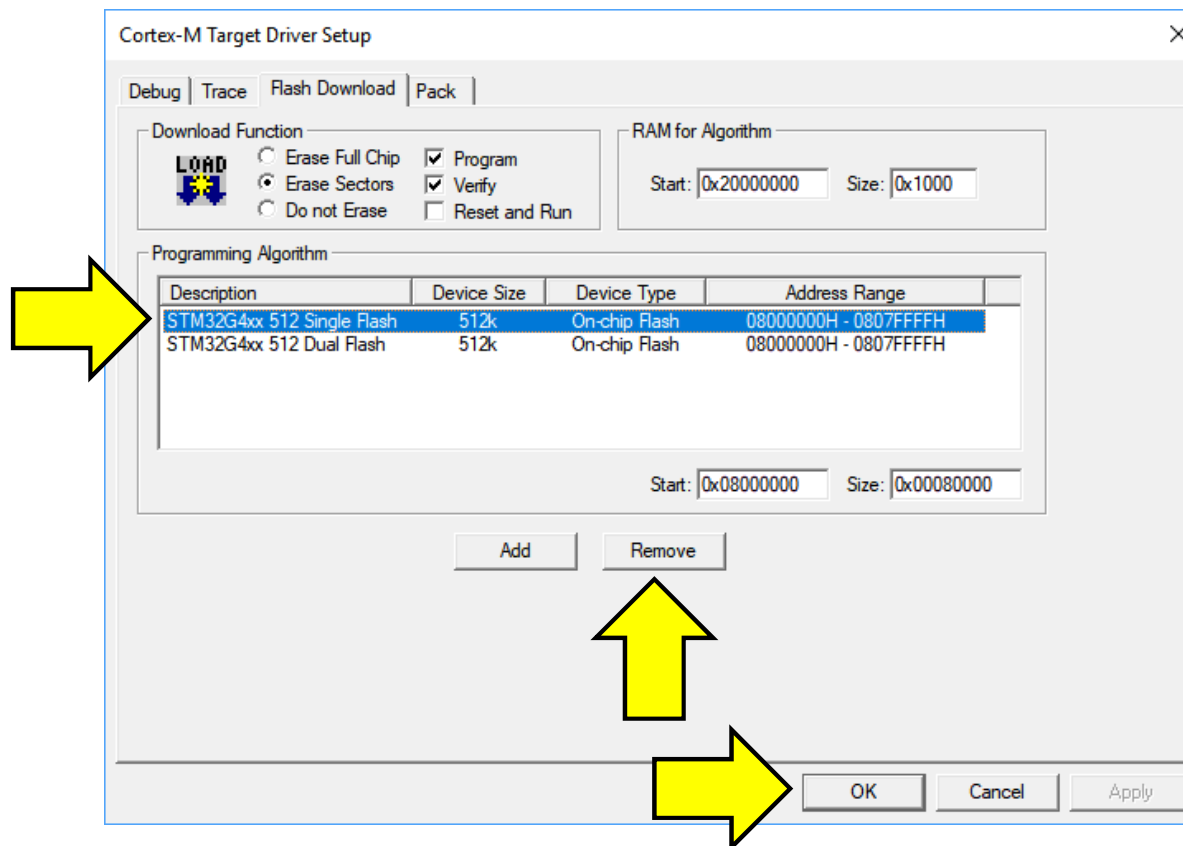




STM32G4xx 512 Dual Flash



Remove Single Flash





Place 2 new variables for the Analyzer – main.c about line 56

```
51  /* USER CODE BEGIN PV */
52  /* Private variables -----
53
54  uint32_t wave[4] = {0x07FF0
55
56  uint16_t Display_DAC1DOR1;
57  uint16_t Display_DAC1DOR2;
58
59  /* USER CODE END PV */
```

See c:\2019_STM32G4_Workshop\Labs\DAC
For the file CodeLinesForMain_DAC.txt for cut and paste
Or open the file from the Doc folder in the Editor



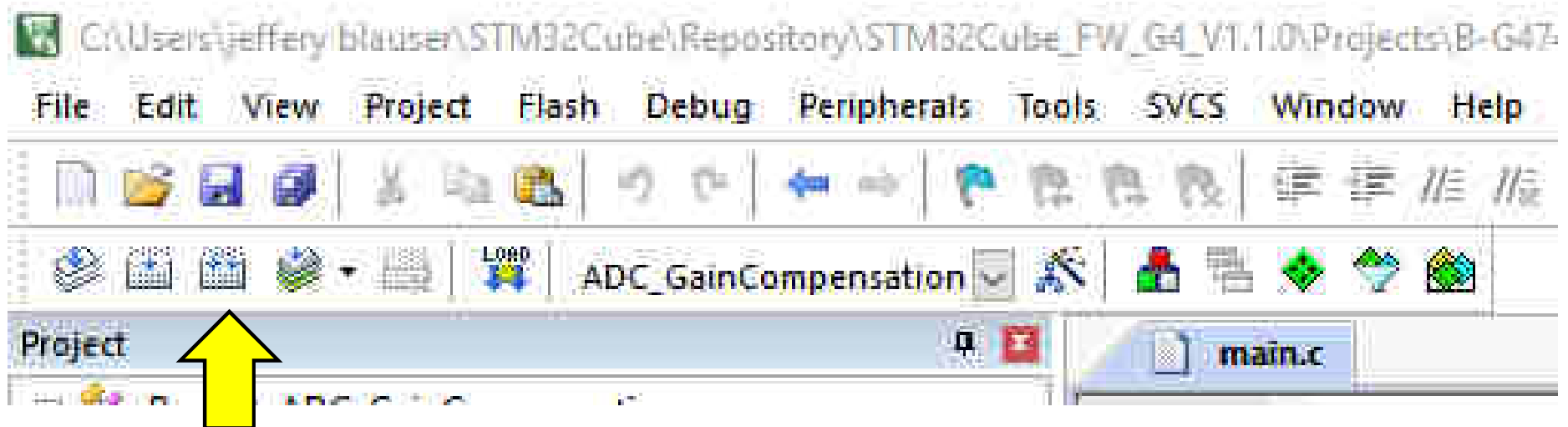
Execute 3 new lines for the Analyzer – main.c about line 143

```
138 while (1)
139 {
140     /* USER CODE END WHILE */
141
142     /* USER CODE BEGIN 3 */
143     Display_DAC1DOR1 = DAC1->DOR1;
144     Display_DAC1DOR2 = DAC1->DOR2;
145     HAL_Delay(25);
146 }
```

See c:\2019_STM32G4_Workshop\Labs\DAC
For the file CodeLinesForMain_DAC.txt for cut and paste.
Or open the file from the Doc folder in the Editor



Build

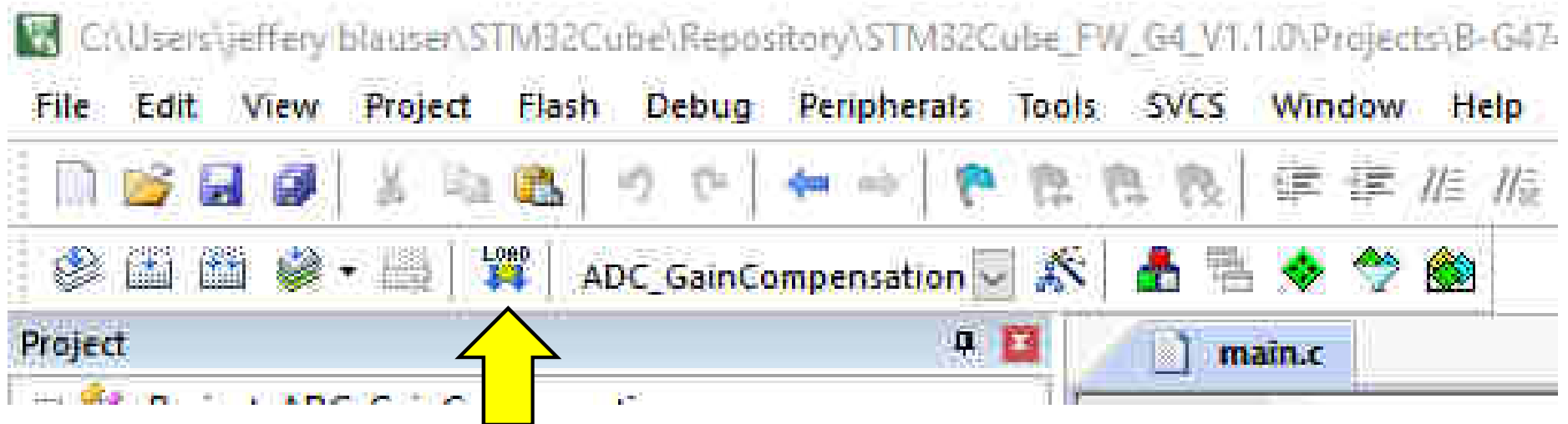


Build Output

```
Program Size: Code=6572 RO-data=576 RW-data=92 ZI-data=1204
FromELF: creating hex file...
DAC_DualConversionFromDMA\DAC_DualConversionFromDMA.hex: Warning: Q9931W: Your license for feature Keil will expire in 30 days
Finished: 0 information, 1 warning and 0 error messages.
"DAC_DualConversionFromDMA\DAC_DualConversionFromDMA.axf" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:01:38
```



Load

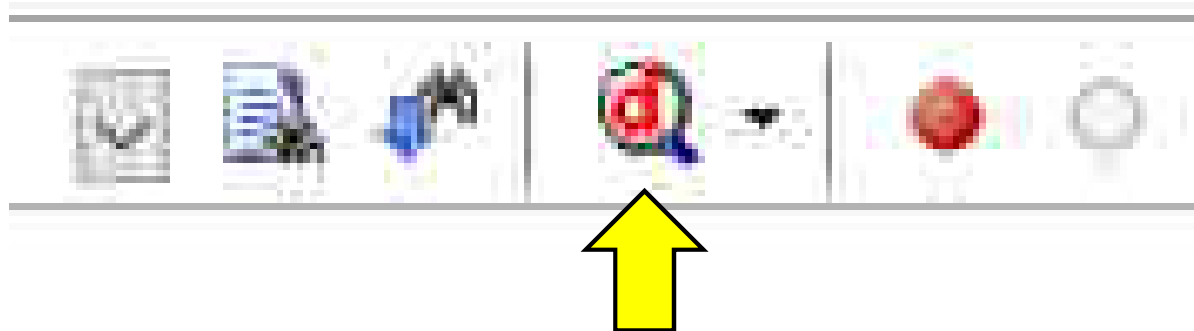
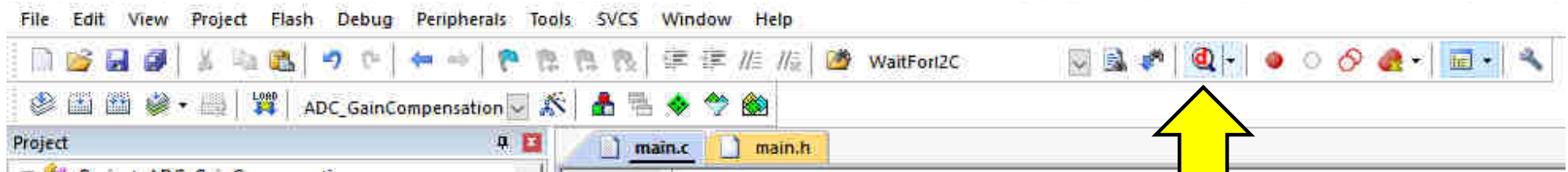




Keil MDK-ARM

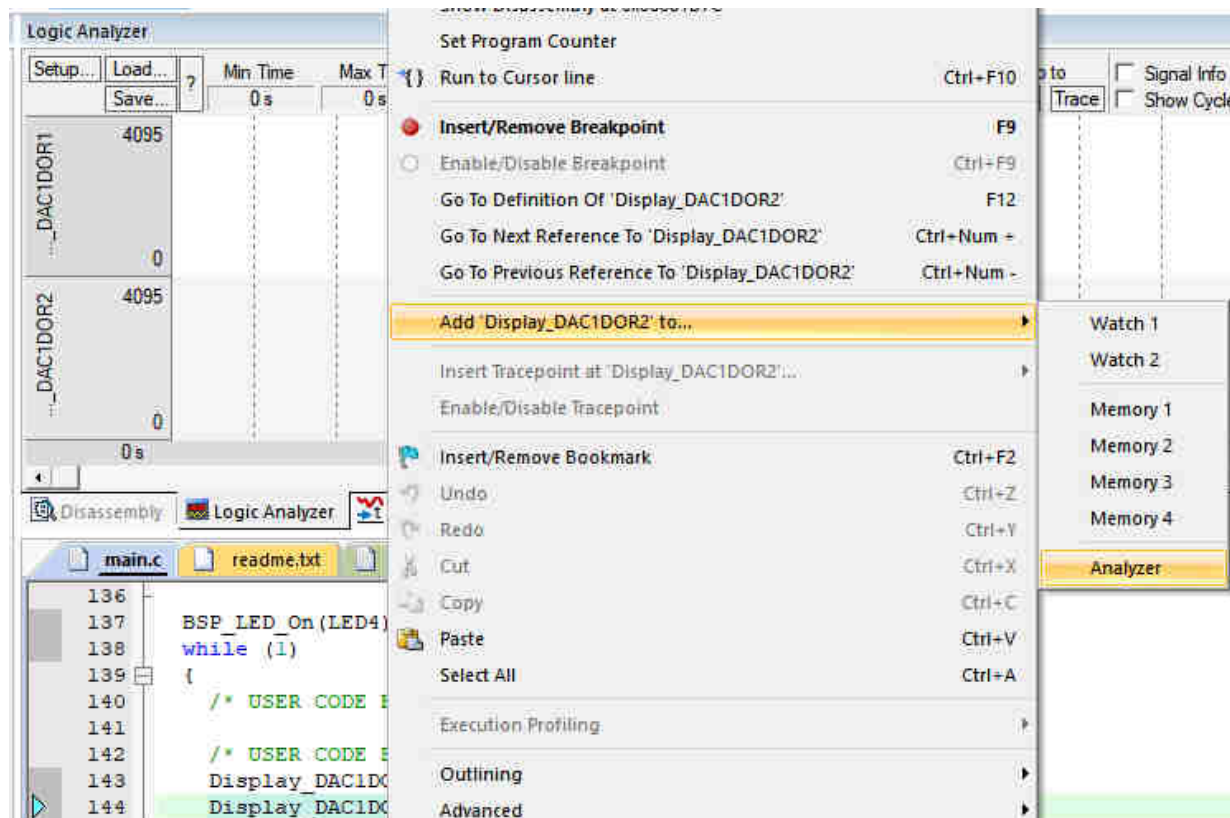
93

Debugger



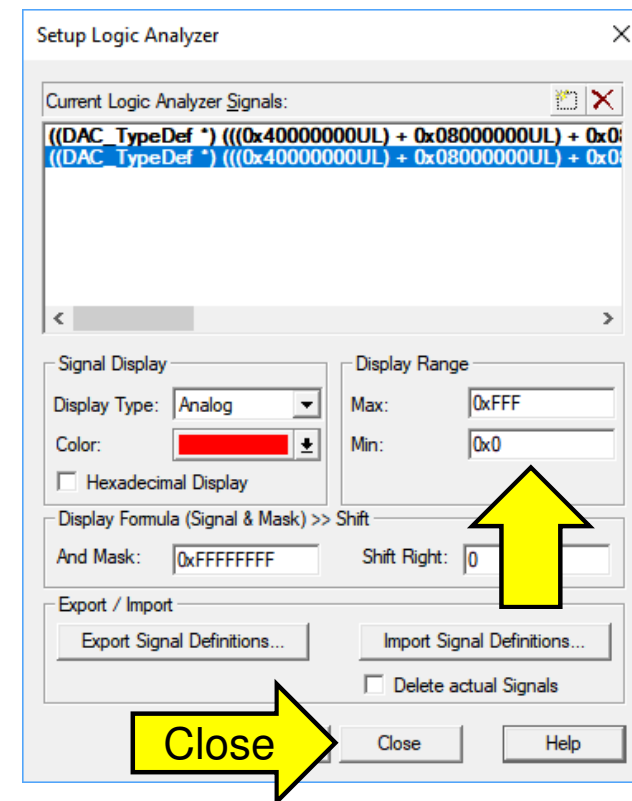
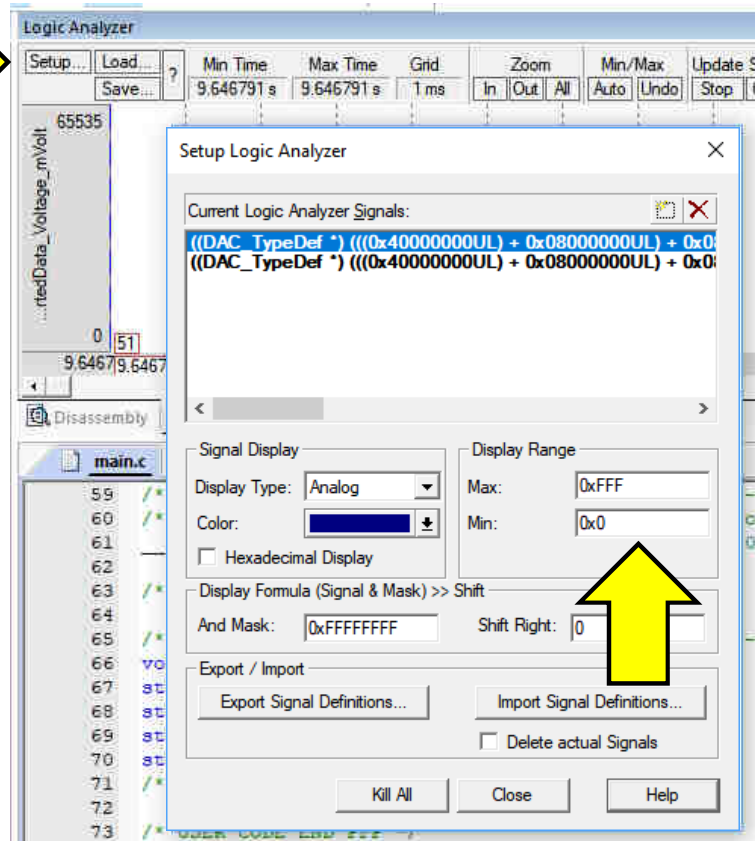


Add to Analyzer Display_DAC1DOR1 & 2



Setup – Min=0x0, Max = 0xFFFF

Setup

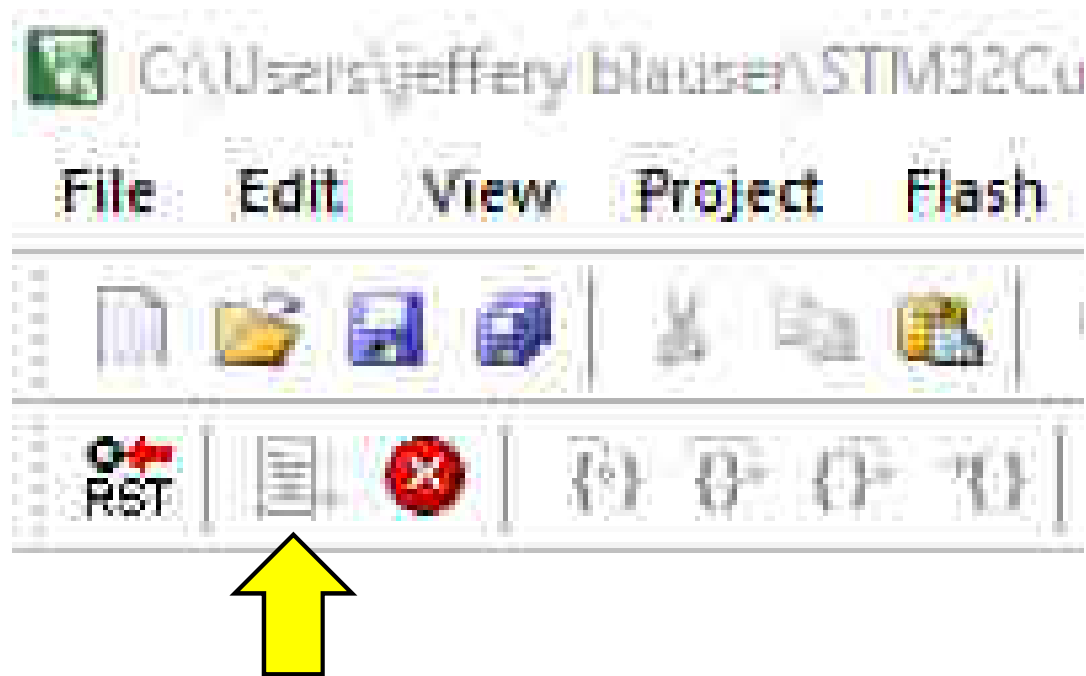




Keil MDK-ARM

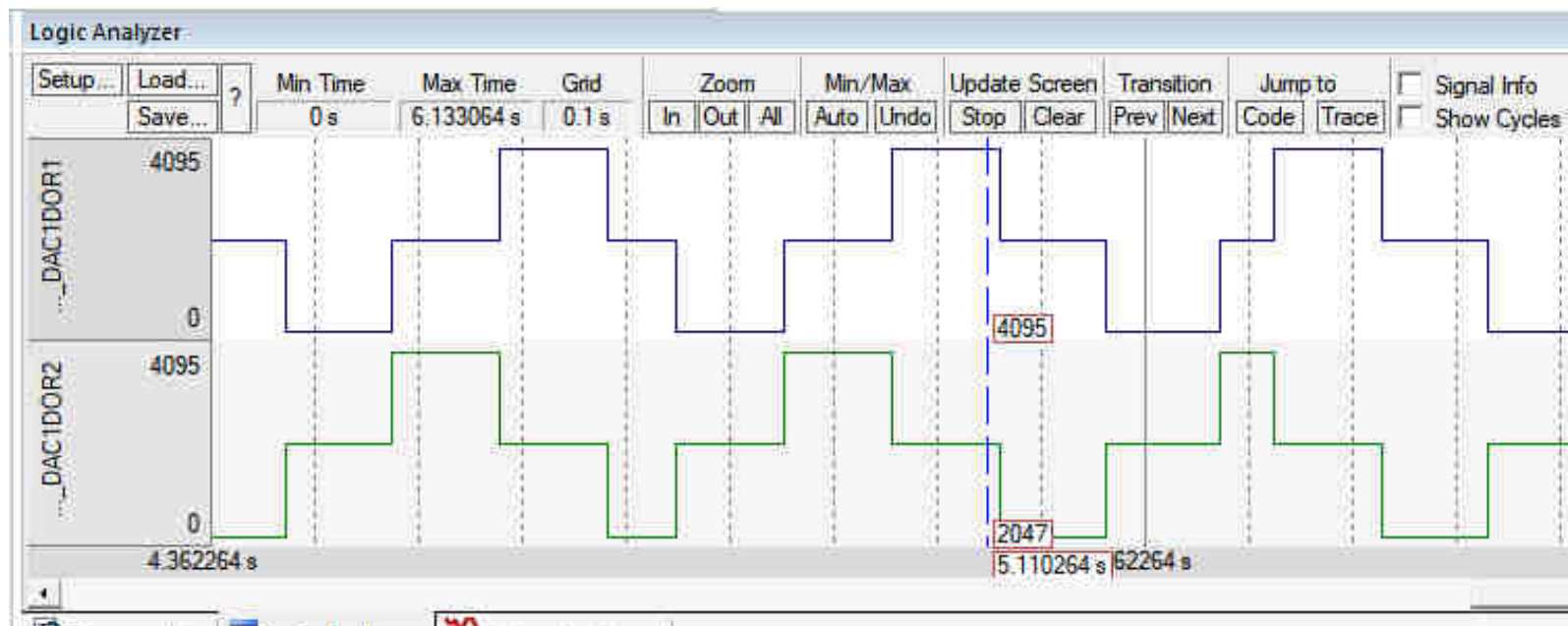
96

RUN



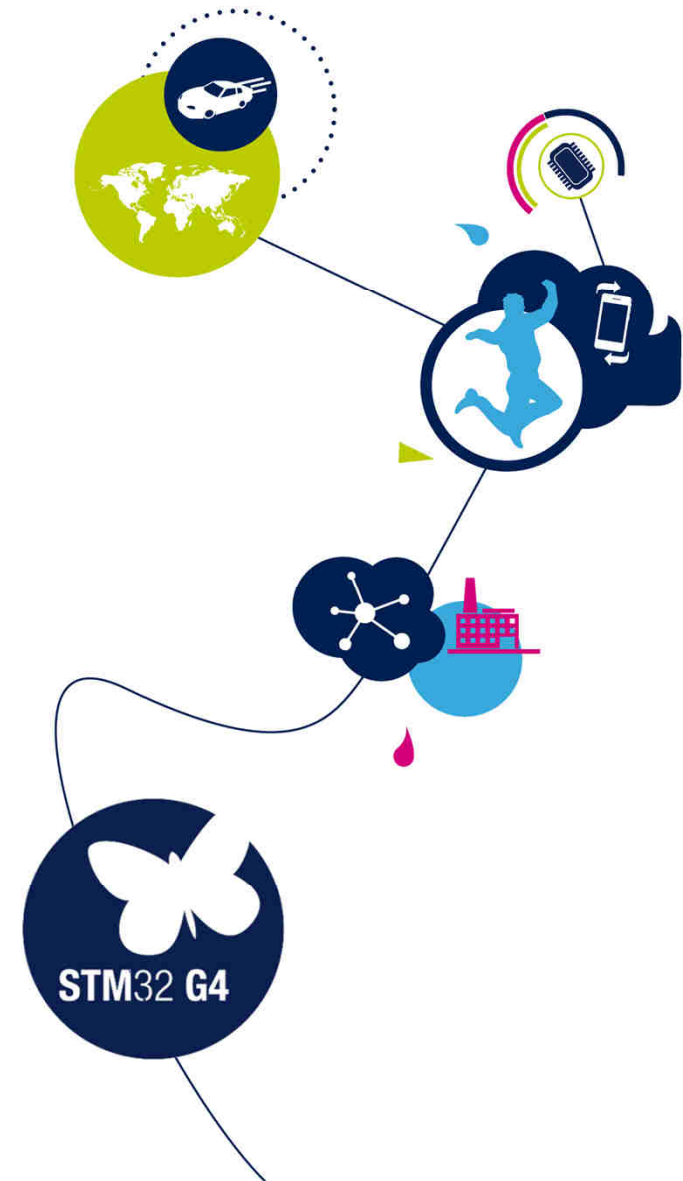


Logic Analyzer Output



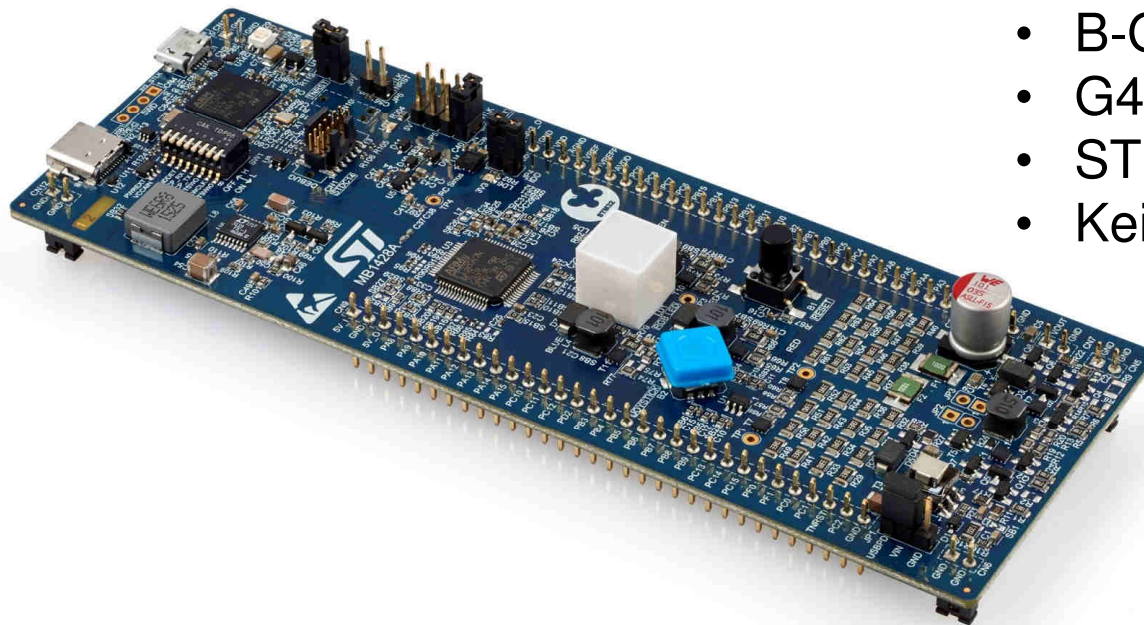
STM32G4 mixed-signal MCU workshop

*Hands On:
Programmable Gain Amplifier*





Systems Check



- USB cable
- B-G474E-DPOW1 Discovery Board
- G4 Family Cube Library
- STM32CubeMx
- Keil MDK-ARM



Rich, Advanced Analog

100

Mixed-signal SoC for wide variety of applications

ADC (up to 5)	Values
Topology	SAR 12-bit + HW oversampling → 16-bit
Sampling rate	Up to 4 Msps
Input	Single-ended and differential
Offset and Gain compensation	Auto calibration to reduce gain and offset

DAC (up to 7)	Values
Sampling rate	15 Msps (internal) 1Msps (from buffered output)
Settling time	16ns

Op-Amp (up to 6)	Values
GBW	13 MHz
Slew rate	45 V/μs
Offset	3mV over full T° range 1.5mV @ 25°C
PGA Gain (accuracy)	2, 4, 8, 16, -1,-3,-7,-15 (1%) 32, 64, -31,-63 (2%)

Comparator (up to 7)	Values
Power supply	1.62 .. 3.6V
Propagation delay	16.7ns
Offset	-6 .. +2 mV
Hysteresis	8 steps: 0, 9, 18, 27, 36, 45, 54, 63 mV



Programmable Gain Amplifiers

101

G4 PGA

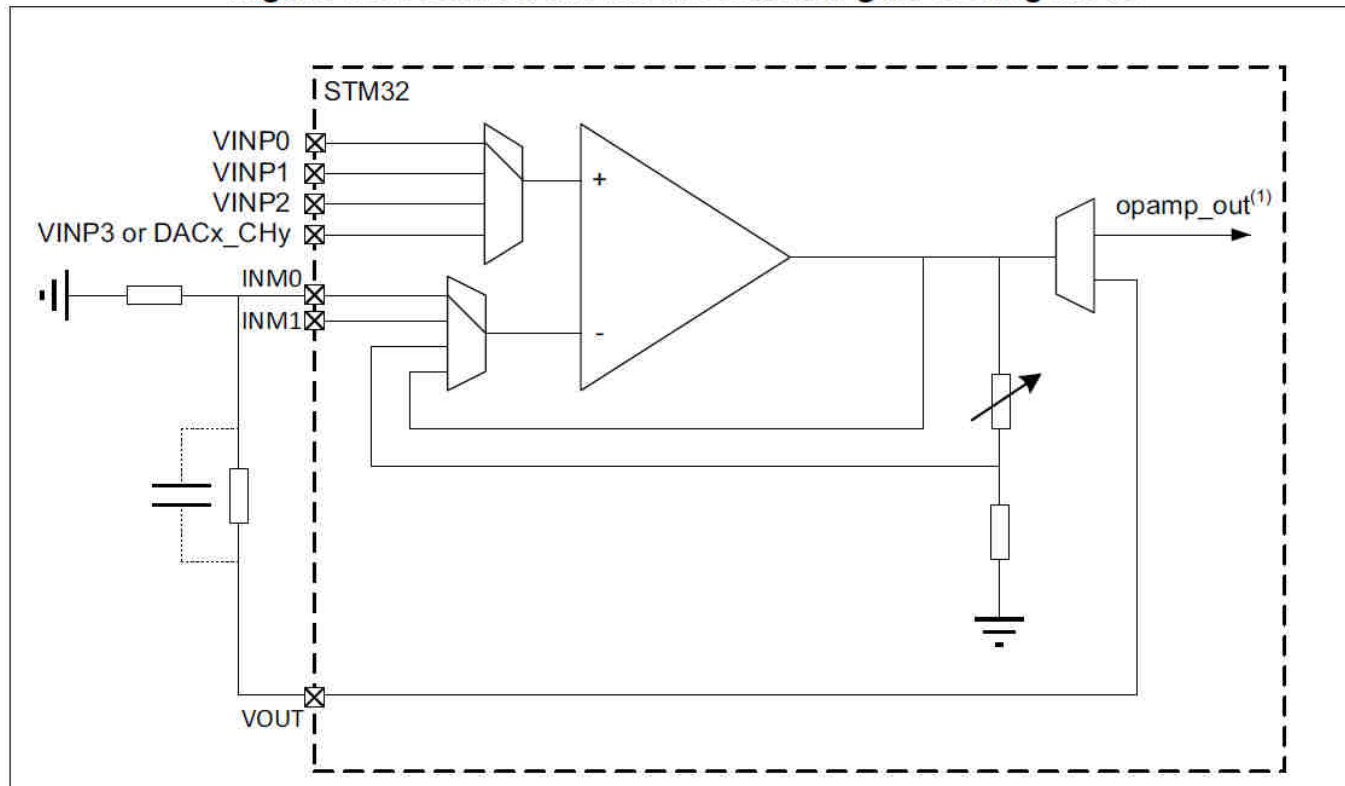
- 6 PGA's available
- Non inverting gain 2/4/8/16/32/64
- Inverting gain -1/-3/-7/-15/-31/-63
- Rail-to-Rail input and output
- Low input bias current
- Low input offset voltage
- 15MHz GBW product
- High speed mode for better slew rate

Internally connected to:

- ADC inputs
- External pins, both inputs and outputs
- Internal resistors
- DAC3 & DAC4 outputs

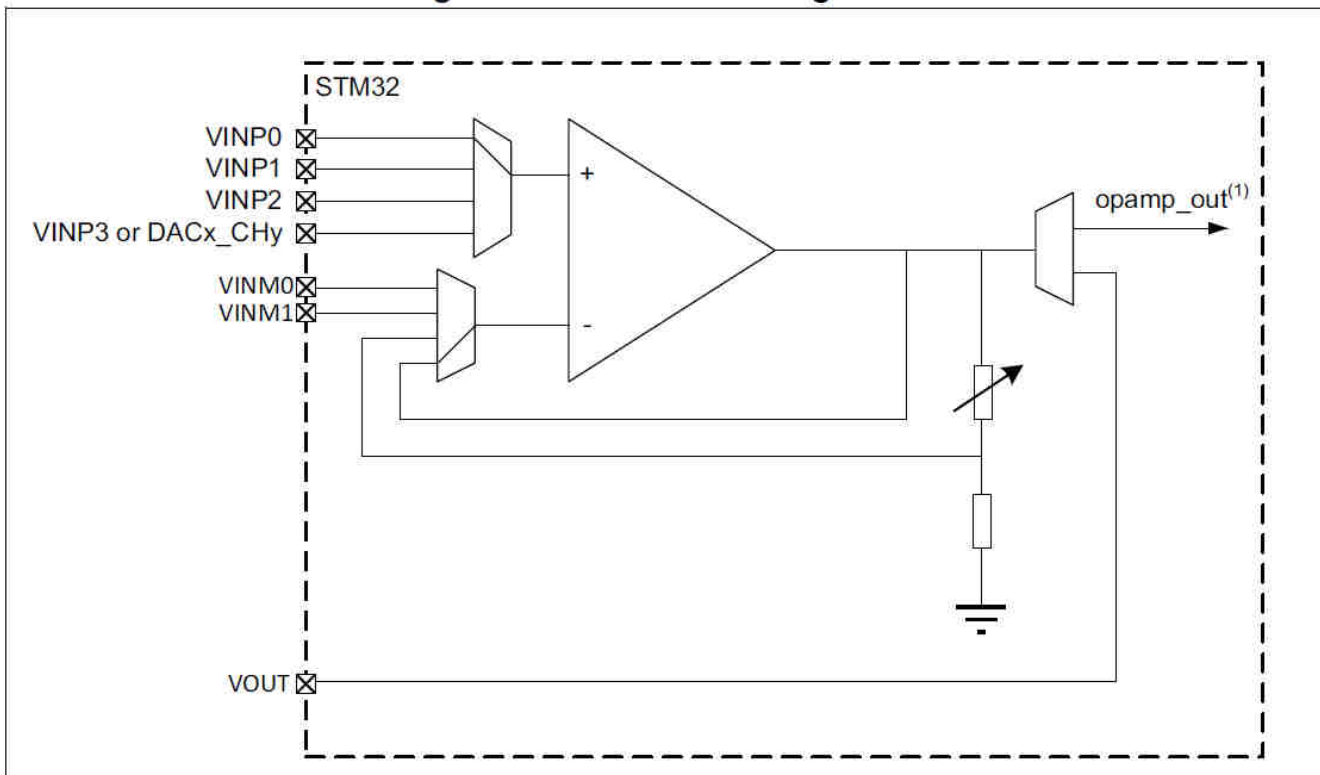
G4 Op Amp Stand Alone Mode

Figure 168. Standalone mode: external gain setting mode



G4 Op Amp Follower Mode

Figure 169. Follower configuration



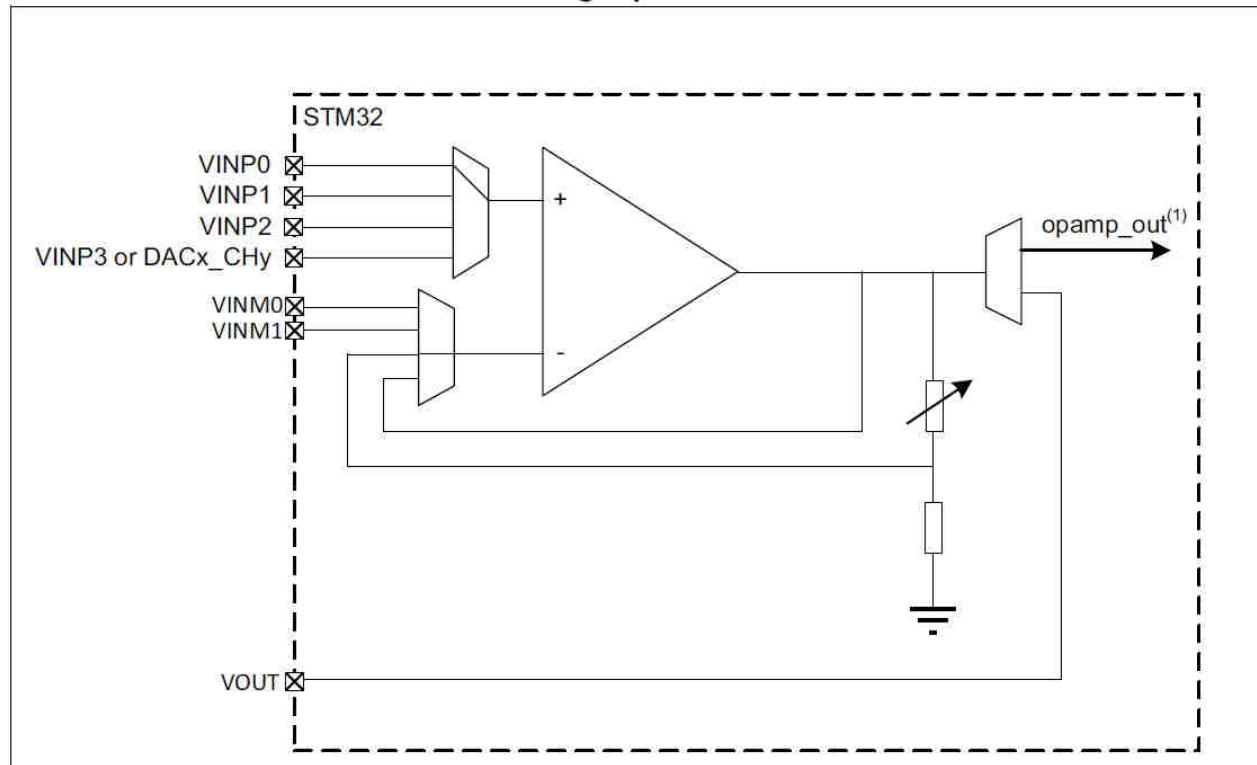


Programmable Gain Amplifiers

104

G4 Op Amp PGA Mode – This is what we will use today

Figure 170. PGA mode, internal gain setting (x2/x4/x8/x16/x32/x64),
inverting input not used





Lab steps 1 of 2

- The DMA provides samples (sine wave) to the DAC.
 - The DAC peripheral generates a sine wave signal on DAC1_OUT1(PA4) which is amplified by the OPAMP3. (Later DAC3_OUT2 as well)
 - The OPAMP3 amplified output is on PB1 with gain of either 2 or 4. The OPAMP gain is changed on the fly while the OPAMP remains enabled.
- The test steps are:
 - Step 1:
 - DAC: normal power mode
 - OPAMP: normal power mode with gain = 2
 - DMA: circular mode - DMA half transfer IT handled by CPU



Lab steps 2 of 2

- Step 2:

DAC: normal power mode

OPAMP: normal power mode

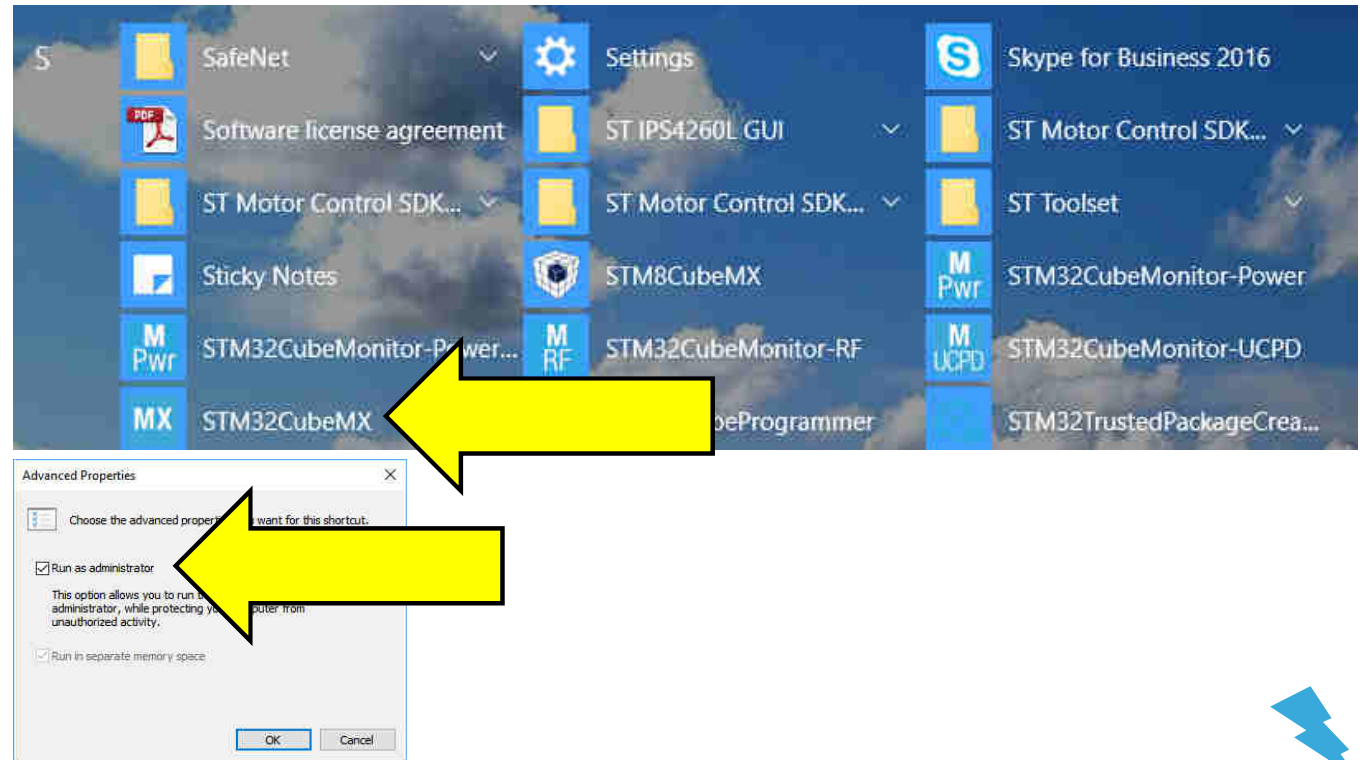
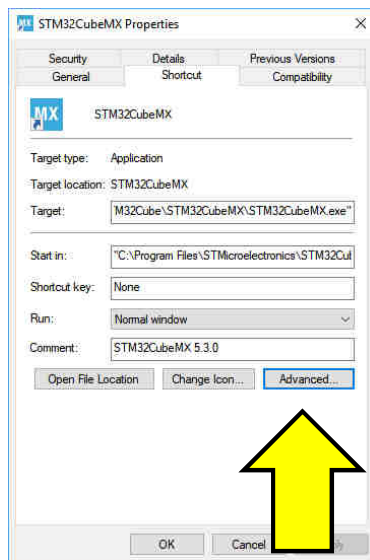
- gain is changed from 2 to 4 or from 4 to 2

DMA: circular mode - DMA half transfer IT handled by CPU

We will use the built in Logic Analyzer today instead of an o'scope on the DACs and Op Amps.



Start up





Splash Page – G4 Library Installed?

The screenshot displays the STM32CubeMX software interface. In the foreground, the 'Embedded Software Packages Manager' window is open, showing a list of packages. A yellow arrow points to the 'STM32G4' section, which lists two packages: 'STM32Cube MCU Package for STM32G4 Series' with versions 1.1.0 and 1.0.1. The background shows the main splash page with a 'Manage software installations' section. A yellow arrow points to the 'INSTALL / REMOVE' button in this section, with the text 'Install/Remove' next to it.

Embedded Software Packages Manager

STM32Cube MCU Packages and embedded software packs releases

Releases Information was last refreshed less than one hour ago.

Description	Installation	Available Version
STM32F7		
STM32G0		
STM32G4		
STM32Cube MCU Package for STM32G4 Series	1.1.0	1.1.0
STM32Cube MCU Package for STM32G4 Series	1.0.1	1.0.1

Details

From Local From Url Refresh Install Now Remove Now Close

Manage software installations

Check for STM32CubeMX and embedded software packa...

CHECK FOR UPDATES

Install or remove embedded software packa...

INSTALL / REMOVE

Install/Remove

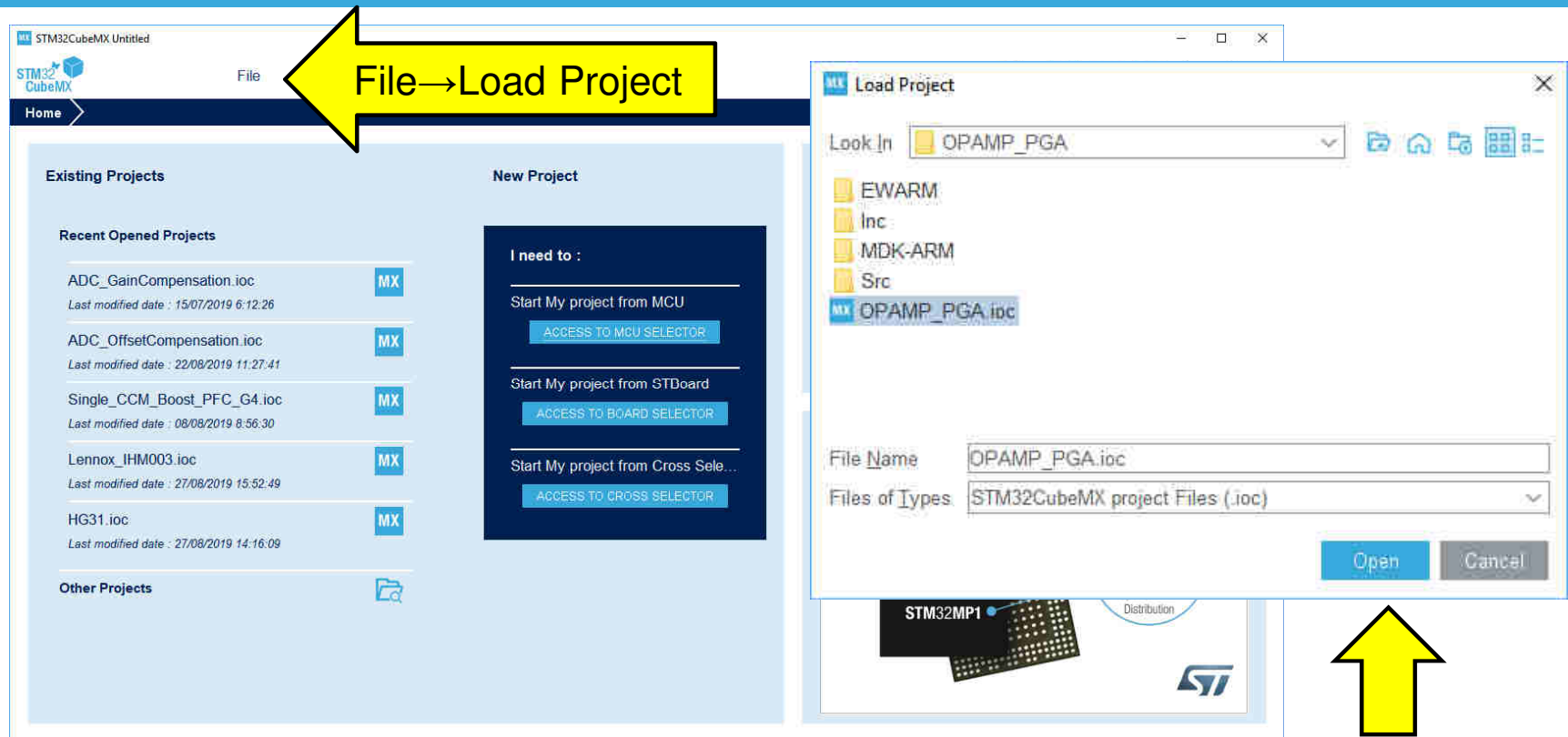
New multicore STM32MP1 Series for Industrial and IoT applications

STM32MP1

OpenSTLinux Distribution



Splash Page

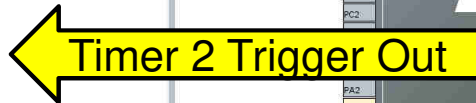
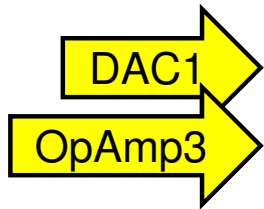
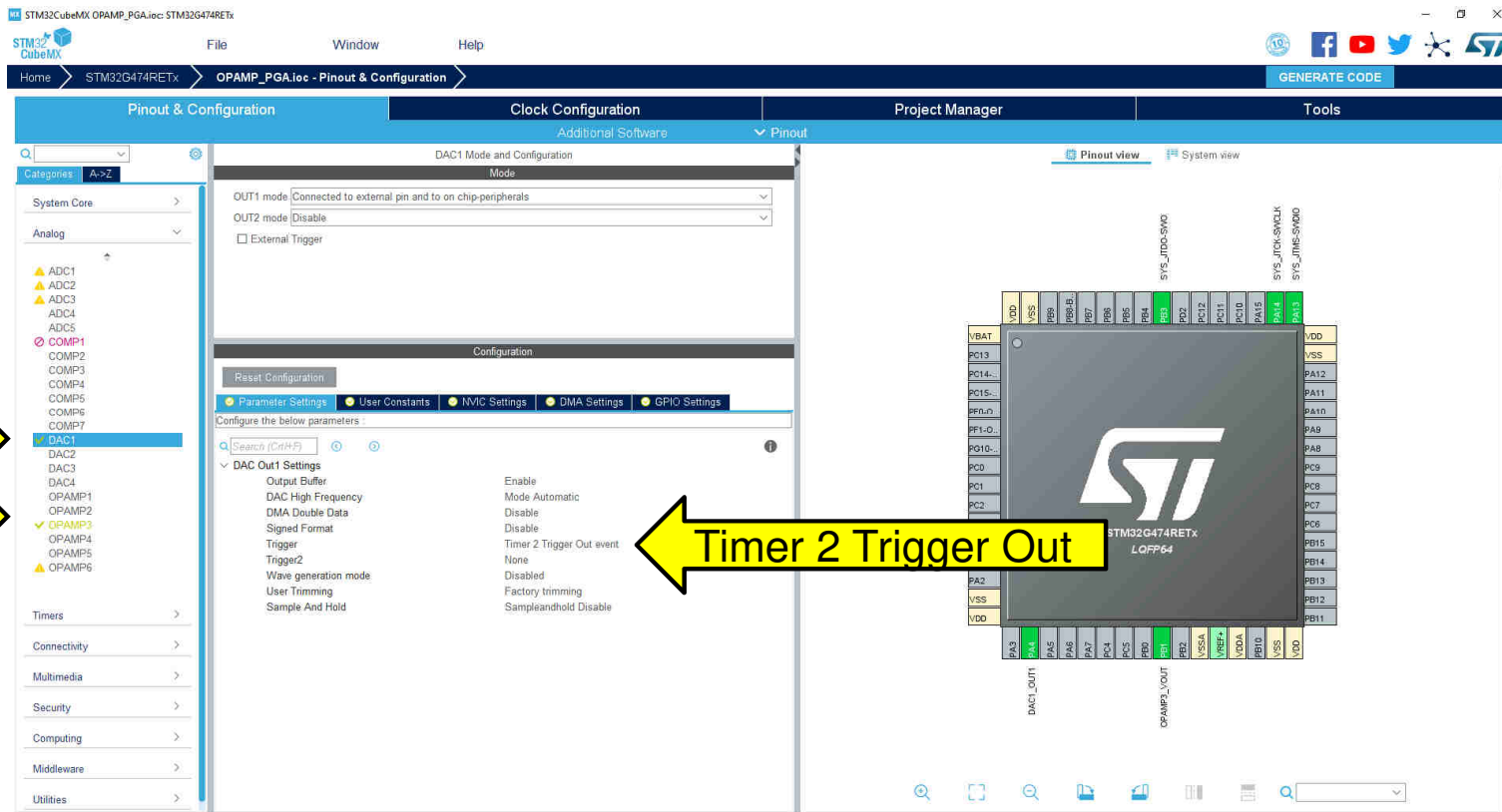


C:\2019_STM32G4_Workshop\Labs\OpAmp

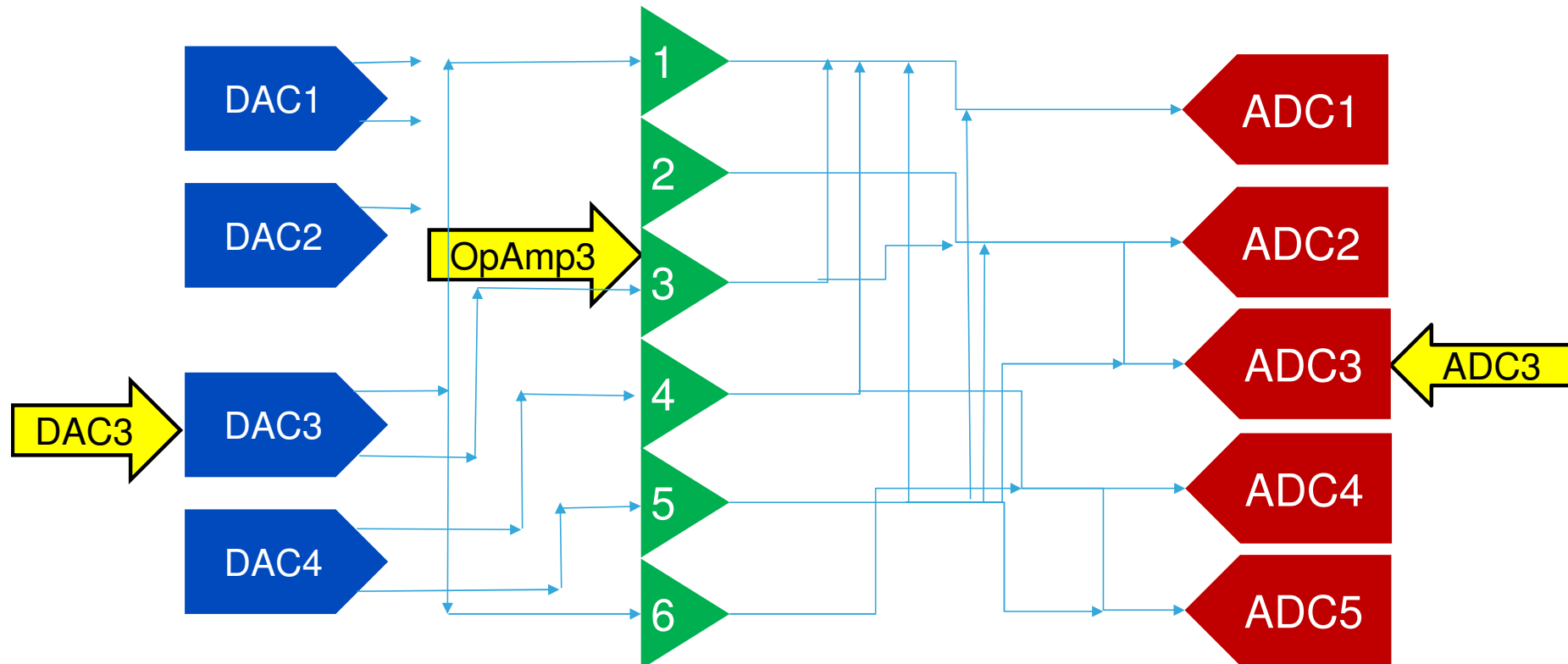




DAC1 (PA4), Timer 2, Op Amp 3 (PB1)



DAC3_OUT2 - Op Amp 3 – ADC3





DAC3_OUT2 - Op Amp 3 – ADC3





DAC3 – Output, Trigger



The screenshot shows the STM32CubeMx configuration interface for DAC3. On the left, a sidebar lists various components under the 'Analog' category. DAC3 is selected and highlighted in blue. On the right, the 'DAC3 Mode and Configuration' window is open. It has two tabs: 'Mode' and 'Configuration'. The 'Mode' tab is active, showing three options: 'OUT1 Connected to on chip-peripherals only' (unchecked), 'OUT2 Connected to on chip-peripherals only' (checked), and 'External Trigger' (unchecked). A yellow arrow points from this text to the 'OUT2' option. Below the 'Mode' tab is the 'Configuration' tab, which is currently inactive. It contains a 'Reset Configuration' button and four sub-tabs: 'Parameter Settings' (active), 'User Constants', 'NVIC Settings', and 'DMA Settings'. Under 'Parameter Settings', there is a search bar and a list of parameters for 'DAC Out2 Settings'. The parameters and their values are: Output Buffer (Disable), DAC High Frequency (Mode Automatic), DMA Double Data (Disable), Signed Format (Disable), Trigger (Timer 2 Trigger Out event), Trigger2 (None), Wave generation mode (Disabled), User Trimming (Factory trimming), and Sample And Hold (Sampleandhold Disable).

Mode

- ☐ OUT1 Connected to on chip-peripherals only
- ☒ OUT2 Connected to on chip-peripherals only
- ☐ External Trigger

Configuration

Reset Configuration

Parameter Settings | User Constants | NVIC Settings | DMA Settings

Configure the below parameters :

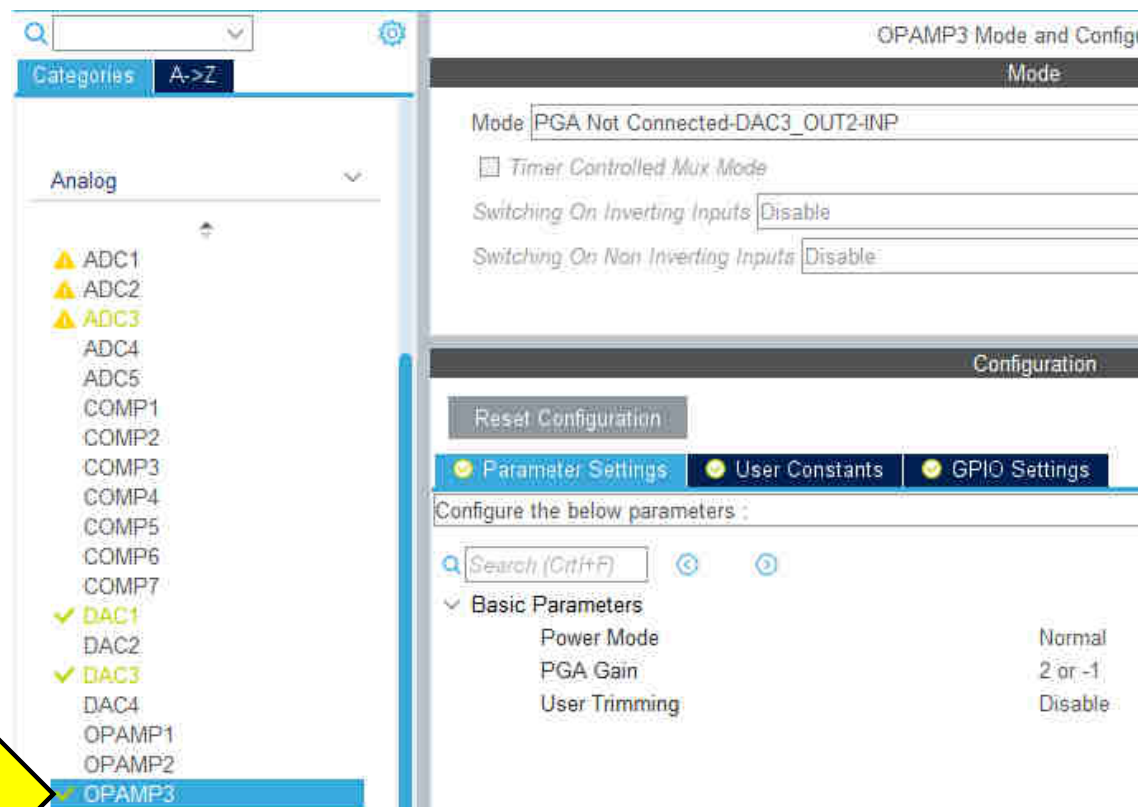
Search (Ctrl+F)

▼ DAC Out2 Settings

Output Buffer	Disable
DAC High Frequency	Mode Automatic
DMA Double Data	Disable
Signed Format	Disable
Trigger	Timer 2 Trigger Out event
Trigger2	None
Wave generation mode	Disabled
User Trimming	Factory trimming
Sample And Hold	Sampleandhold Disable

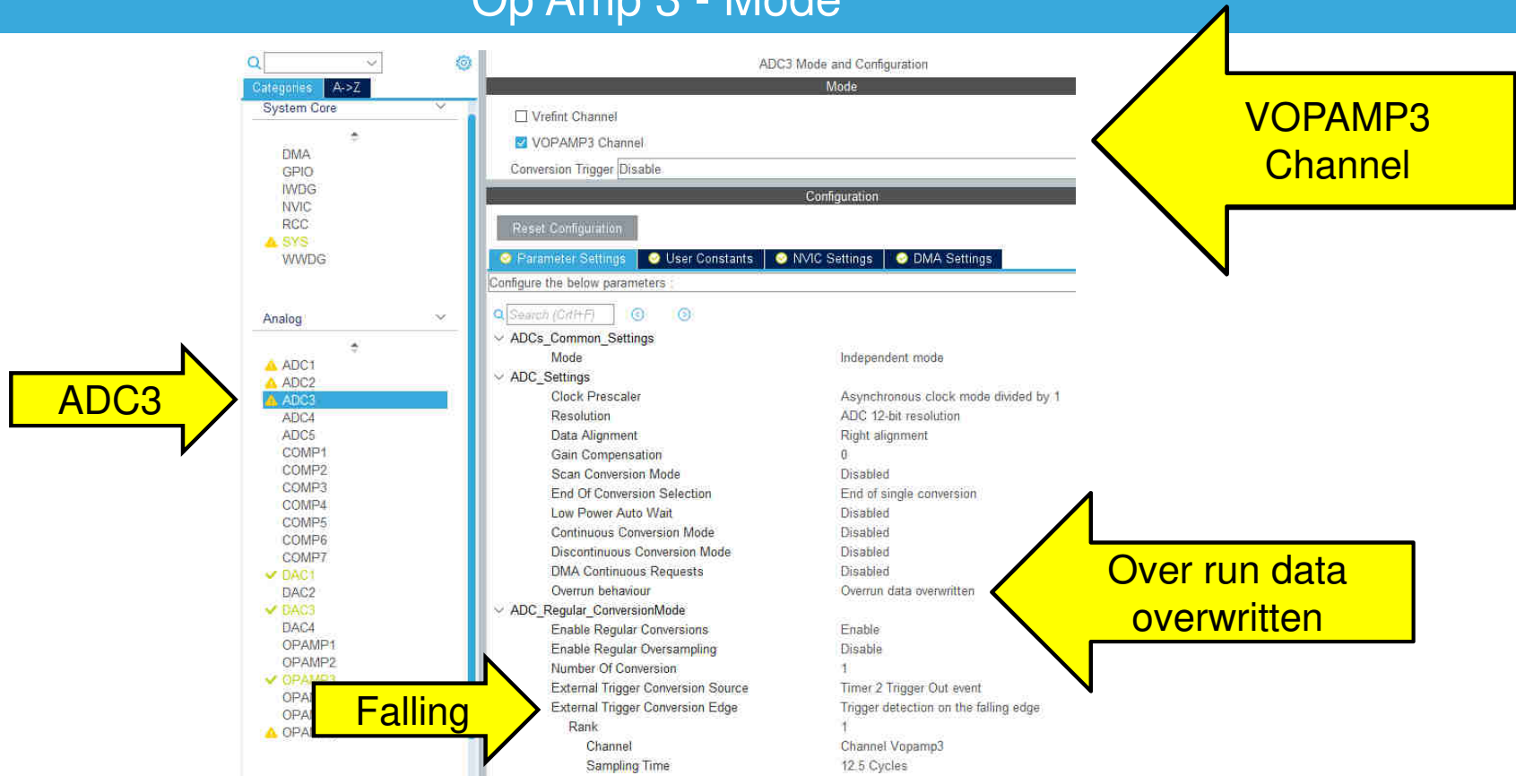


Op Amp 3 - Mode



PGA Not
Connected –
DAC3_OUT2-INP

Op Amp 3 - Mode



ADC3

VOPAMP3 Channel

Falling

Over run data overwritten

ADC3 Mode and Configuration

Mode

☐ Vrefint Channel

☒ VOPAMP3 Channel

Conversion Trigger: Disable

Configuration

Reset Configuration

Parameter Settings | User Constants | NVIC Settings | DMA Settings

Configure the below parameters:

Search (Ctrl+F)

ADCs_Common_Settings

Mode: Independent mode

ADC_Settings

Clock Prescaler: Asynchronous clock mode divided by 1

Resolution: ADC 12-bit resolution

Data Alignment: Right alignment

Gain Compensation: 0

Scan Conversion Mode: Disabled

End Of Conversion Selection: End of single conversion

Low Power Auto Wait: Disabled

Continuous Conversion Mode: Disabled

Discontinuous Conversion Mode: Disabled

DMA Continuous Requests: Disabled

Overrun behaviour: Overrun data overwritten

ADC_Regular_ConversionMode

Enable Regular Conversions: Enable

Enable Regular Oversampling: Disable

Number Of Conversion: 1

External Trigger Conversion Source: Timer 2 Trigger Out event

External Trigger Conversion Edge: Trigger detection on the falling edge

Rank: 1

Channel: Channel Vopamp3

Sampling Time: 12.5 Cycles



Project Manager – Keil MDK-ARM V5

STM32CubeMX OPAMP_PGA.ioc: STM32G474RETx

File Window Help

Home > STM32G474RETx > OPAMP_PGA.ioc - Project Manager > GENERATE CODE

Pinout & Configuration Clock Configuration Project Manager Tools

Project Settings

Project Name
OPAMP_PGA

Project Location
C:\Users\blauer\STM32Cube\Repository\STM32Cube_FW_G4_V1.1.0\Projects\B-G474E-DPOW1\Examples\OPAMP

Application Structure
Basic ☐ Do not generate the main()

Toolchain Folder Location
C:\Users\blauer\STM32Cube\Repository\STM32Cube_FW_G4_V1.1.0\Projects\B-G474E-DPOW1\Examples\OPAMP\OPAMP_PGA\

Toolchain / IDE
MDK-ARM V5

Linker Settings

Minimum Heap Size 0x200

Minimum Stack Size 0x400

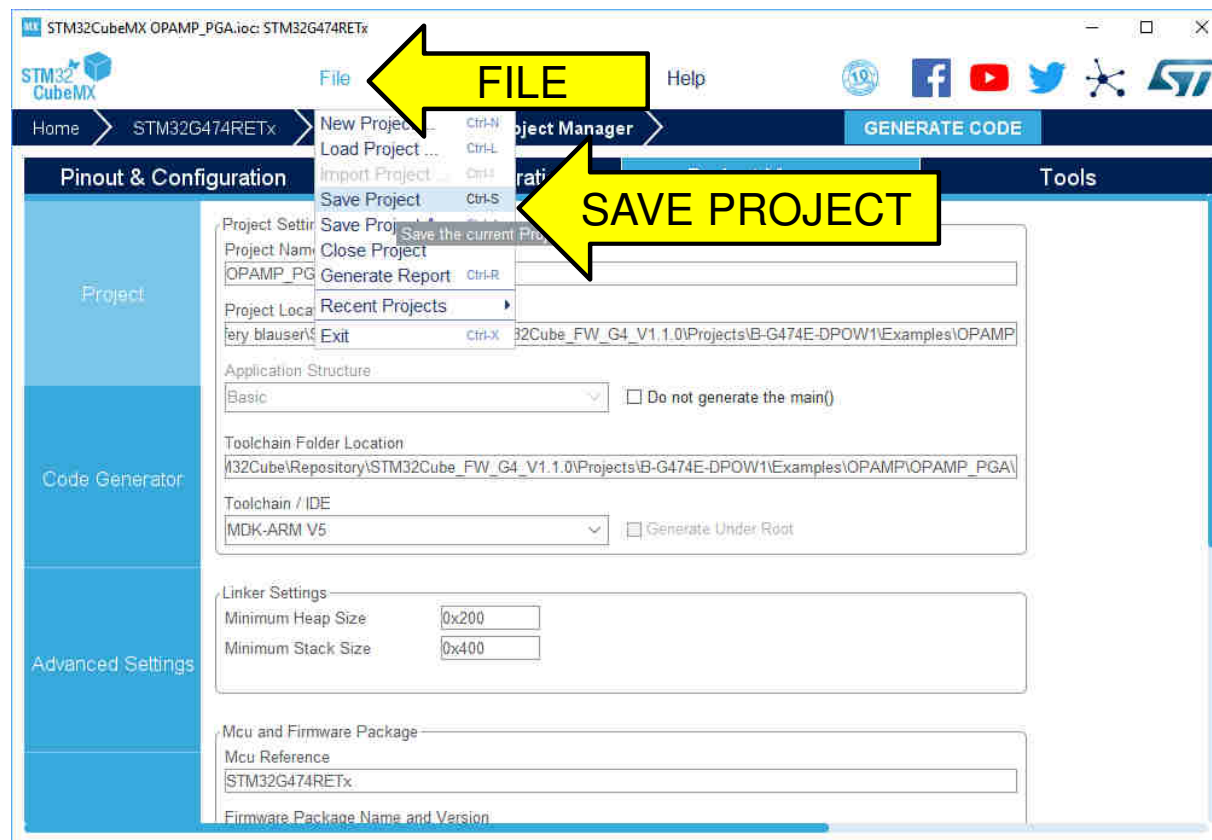
Mcu and Firmware Package

Mcu Reference
STM32G474RETx

Firmware Package Name and Version



FILE→SAVE PROJECT





Generate Code→Open Project

The screenshot displays the STM32CubeMX interface. The top menu bar includes 'File', 'Window', and 'Help'. The breadcrumb navigation shows 'Home > STM32G474RETx > OPAMP_PGA.ioc - Project Manager'. The 'GENERATE CODE' button is highlighted with a yellow arrow. Below the breadcrumb, the 'Project Manager' tab is active, showing 'Project Settings' with fields for 'Project Name' (OPAMP_PGA) and 'Project Location'. The 'Code Generation' dialog box is open, displaying a message: 'The Code is successfully generated under C:/Users/jeffery blausen/STM32Cube/Repository/STM32Cube_FW_G4_V1.1.0/Projects/B-G474E-DPOW1/Examples/OPAMP/OPAMP_PGA'. The dialog has three buttons: 'Open Folder', 'Open Project' (highlighted with a yellow arrow), and 'Close'. The background shows the 'Advanced Settings' tab with fields for 'MDK-ARM V5', 'Linker Settings' (Minimum Heap Size: 0x200, Minimum Stack Size: 0x400), and 'Mcu and Firmware Package' (Mcu Reference: STM32G474RETx).



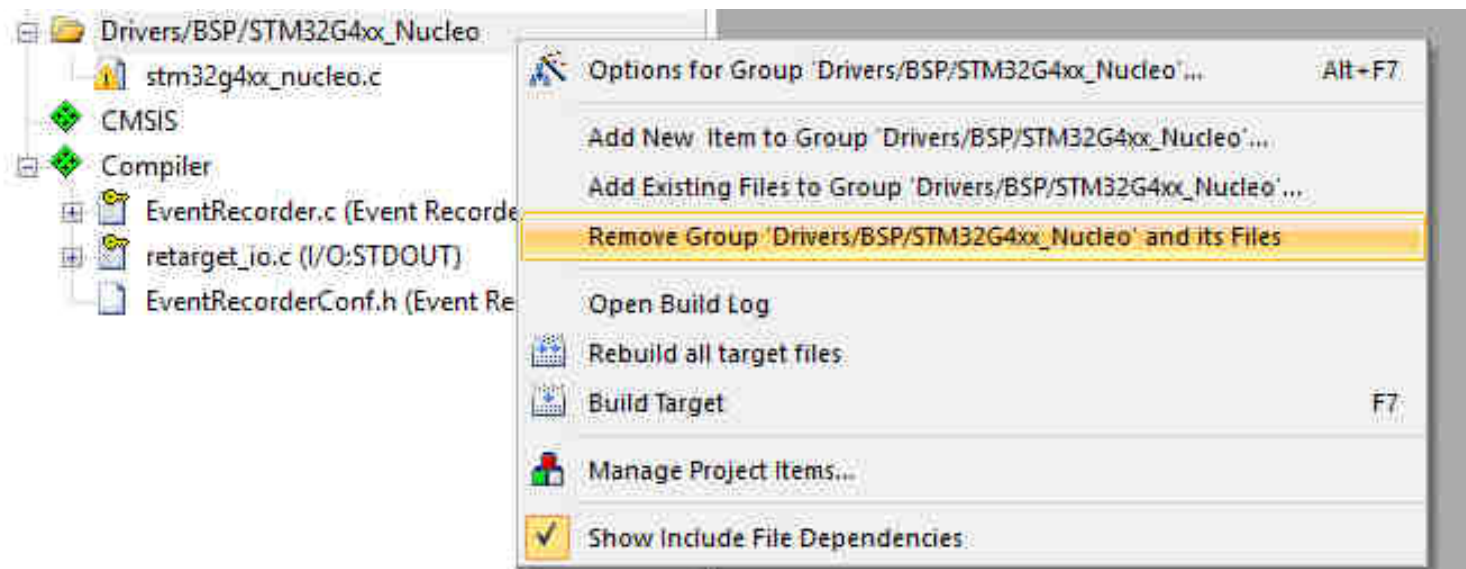
Keil Start up page

```
143 if (UserButtonStatus == 1)
144 {
145     HAL_Delay(200);
146     UserButtonStatus = 0;
147     /* Change the gain */
148     if (hopamp3.Init.PgaGain == OPAMP_PGA_GAIN_2_OR_MINUS_1)
149     {
150         hopamp3.Init.PgaGain = OPAMP_PGA_GAIN_4_OR_MINUS_3;
151     }
152     else
153     {
154         hopamp3.Init.PgaGain = OPAMP_PGA_GAIN_2_OR_MINUS_1;
155     }
156     /* Update OPAMP config */
157     /* Gain is changed on the fly */
158     if (HAL_OK != HAL_OPAMP_Init(&hopamp3))
159     {
160         Error_Handler();
161     }
162 }
163 }
164 /* USER CODE END WHILE */
165
166 /* USER CODE BEGIN 3 */
167 /* USER CODE END 3 */
168 }
169
170 /**
171  * @brief System Clock Configuration
172  * @retval None
173  */
174 void SystemClock_Config(void)
175 {
176     RCC_OscInitTypeDef RCC_OscInitStruct = {0};
177     RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
178
179     /* Configure the main internal regulator output voltage
180     */
181     HAL_PWREX_ControlVoltageScaling(FWR_REGULATOR_VOLTAGE_SCALE1_BOOST);
182     /* Initializes the CPU, AHB and APB buses clocks
183     */
184     RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
185     RCC_OscInitStruct.HSISource = RCC_HSI_ON;
186     RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
187     RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
```





Delete Drivers Folder

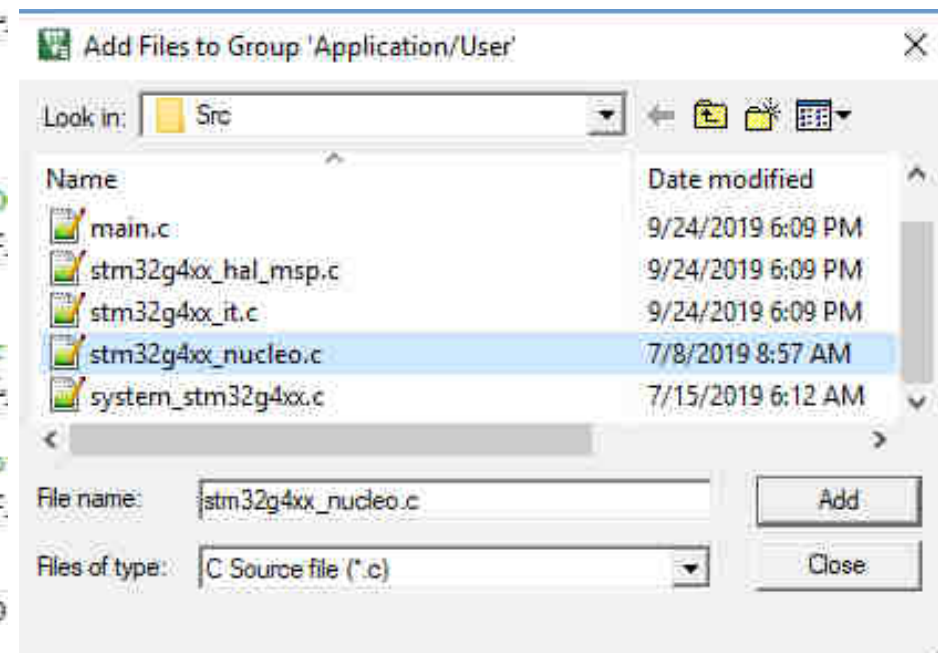
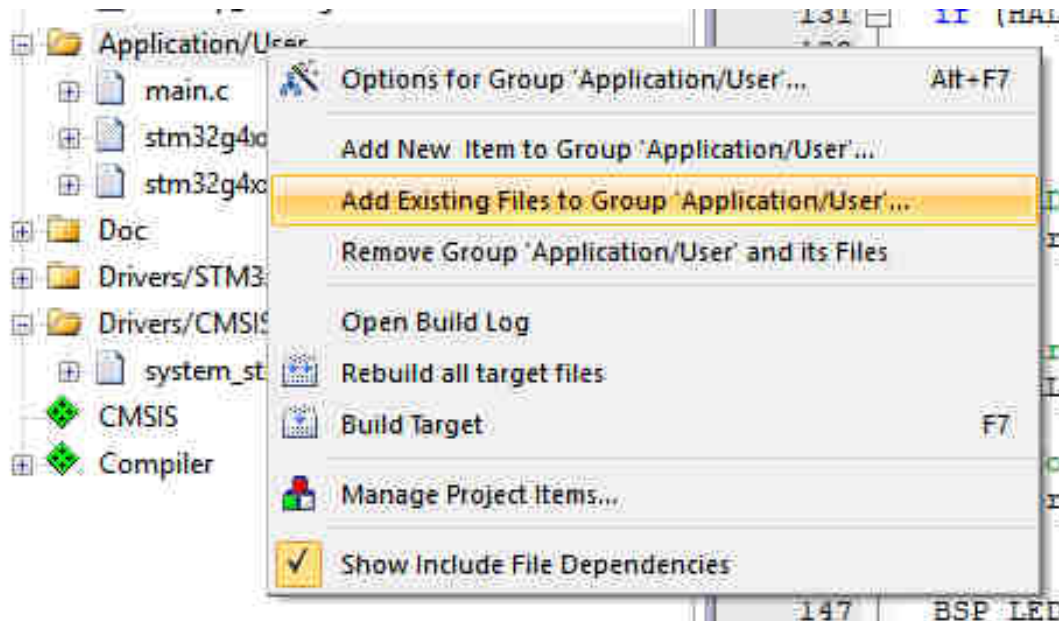




Keil MDK-ARM

122

Application/User folder stm32g4xx_nucleo.c from SRC folder - CLOSE

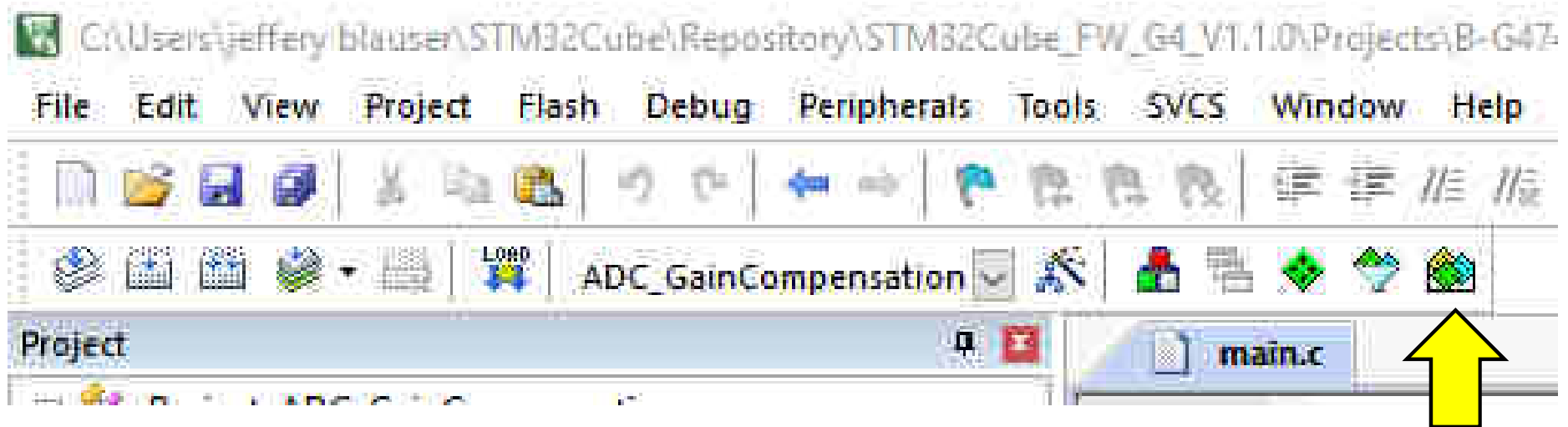




Keil MDK-ARM

123

Check the Device Pack





G4 Device Pack

The screenshot shows the Keil MDK-ARM Pack Installer window. The left pane displays a tree view of devices, with the STM32G474 device selected under the STM32G4 Series. The right pane shows a list of packs, with the Keil:STM32G4xx_DFP pack highlighted. A yellow arrow points to the 'Exit' button in the top right corner. Another yellow arrow points to the 'Keil:STM32G4xx_DFP' pack in the right pane. A third yellow arrow points to the 'STM32G474' device in the left pane. The bottom pane shows the output log with error messages about downloading files from the mcuxpresso.nxp.com repository.

Device: STM32G474

Search: / Summary

Device

- Silicon Labs: 906 Devices
- Sinowealth: 1 Device
- SONIX: 50 Devices
- STMicroelectronics: 1284 Devices
 - STBlueNRG Series: 1 Device
 - STBlueNRG-1 Series: 1 Device
 - STBlueNRG-2 Series: 1 Device
 - STM32F0 Series: 111 Devices
 - STM32F1 Series: 95 Devices
 - STM32F2 Series: 46 Devices
 - STM32F3 Series: 90 Devices
 - STM32F4 Series: 205 Devices
 - STM32F7 Series: 115 Devices
 - STM32G0 Series: 34 Devices
 - STM32G4 Series: 118 Devices
 - STM32G431: 24 Devices
 - STM32G441: 8 Devices
 - STM32G471: 17 Devices
 - STM32G473: 25 Devices
 - STM32G474: 25 Devices
 - STM32G483: 9 Devices
 - STM32G484: 9 Devices
 - STM32GBK1: 1 Device
 - STM32H7 Series: 22 Devices
 - STM32L0 Series: 152 Devices
 - STM32L1 Series: 81 Devices

Packs

Pack	Action	Description
Device Specific	1 Pack	STM32G4
Keil:STM32G4xx_DFP	Up to date	STM32G4xx Device Family Pack
Generic	40 Packs	
Alibaba:AliOSThings	Install	AliOS Things software pack
ARM:AMP	Install	Software components for inter processor communication (Asymmetric Multi
ARM:CMSIS	Up to date	CMSIS (Cortex Microcontroller Software Interface Standard)
ARM:CMSIS-Driver	Up to date	CMSIS Drivers for external devices
ARM:CMSIS-Driver_Val...	Up to date	CMSIS-Driver Validation
ARM:CMSIS-FreeRTOS	Up to date	Bundle of FreeRTOS for Cortex-M and Cortex-A
ARM:CMSIS-RTOS_Val...	Up to date	CMSIS-RTOS Validation
ARM:mbedClient	Install	ARM mbed Client for Cortex-M devices
ARM:mbedTLS	Install	ARM mbed Cryptographic and SSL/TLS library for Cortex-M devices
ARM:minar	Install	mbed OS Scheduler for Cortex-M devices
ARM:TFM	Install	Trusted Firmware-M (TF-M) is the reference implementation of Arm's Platfor
EmbeddedOffice:Flexi...	Install	Flexible Safety RTOS
Keil:ARM_Compiler	Up to date	Keil ARM Compiler extensions for ARM Compiler 5 and ARM Compiler 6
Keil:IMXRT105x_MWP	Install+	NXP i.MX RT 105x MDK-Middleware examples and CMSIS-Drivers
Keil:Jansson	Install	Jansson is a C library for encoding, decoding and manipulating JSON data
Keil:MDK-Middleware	Up to date	Middleware for Keil MDK-Professional and MDK-Plus
lwIP:lwIP	Install	lwIP is a light-weight implementation of the TCP/IP protocol suite
MDK-Packs:AWS_IoT_...	Install+	SDK for connecting to AWS IoT from a device using embedded C
MDK-Packs:Azure_IoT	Install+	Microsoft Azure IoT SDKs and Libraries
MDK-Packs:JSON	Install	Ultralightweight JSON parser in ANSI C
MDK-Packs:Google_Io...	Install+	Google Cloud IoT Device Connector
MDK-Packs:IoT_Socket	Install	Simple IP Socket (BSD like)
MDK-Packs:Jsmin	Install	Minimalistic JSON parser/tokenizer in C
MDK-Pack:PaHo-MQ...	Install+	Embedded MQTT C/C++ Client Libraries

Output

Cannot download file http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54018M_DFP.pdsc: Illegal MIME type 'text/html'

Cannot download file http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54018_DFP.pdsc: Illegal MIME type 'text/html'

Cannot download file http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54101_DFP.pdsc: Illegal MIME type 'text/html'

Action [2 left]: Update Pack descriptions, download http://mcuxpresso.nxp.com/cmsis_pack/repo/NXP.LPC54102_DFP.pdsc

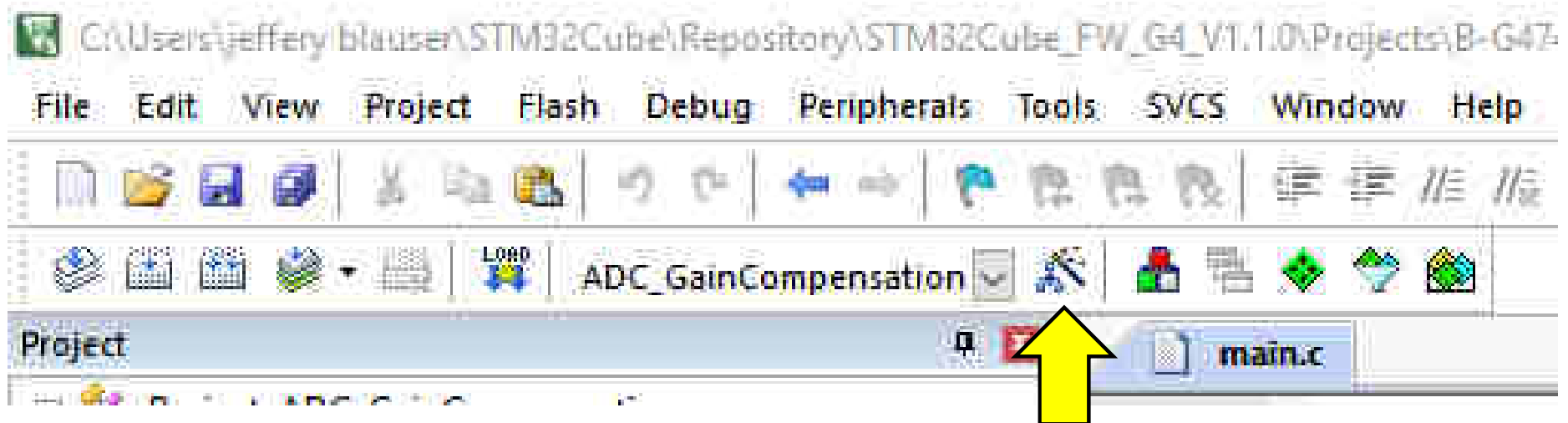
0% ONLINE



Keil MDK-ARM

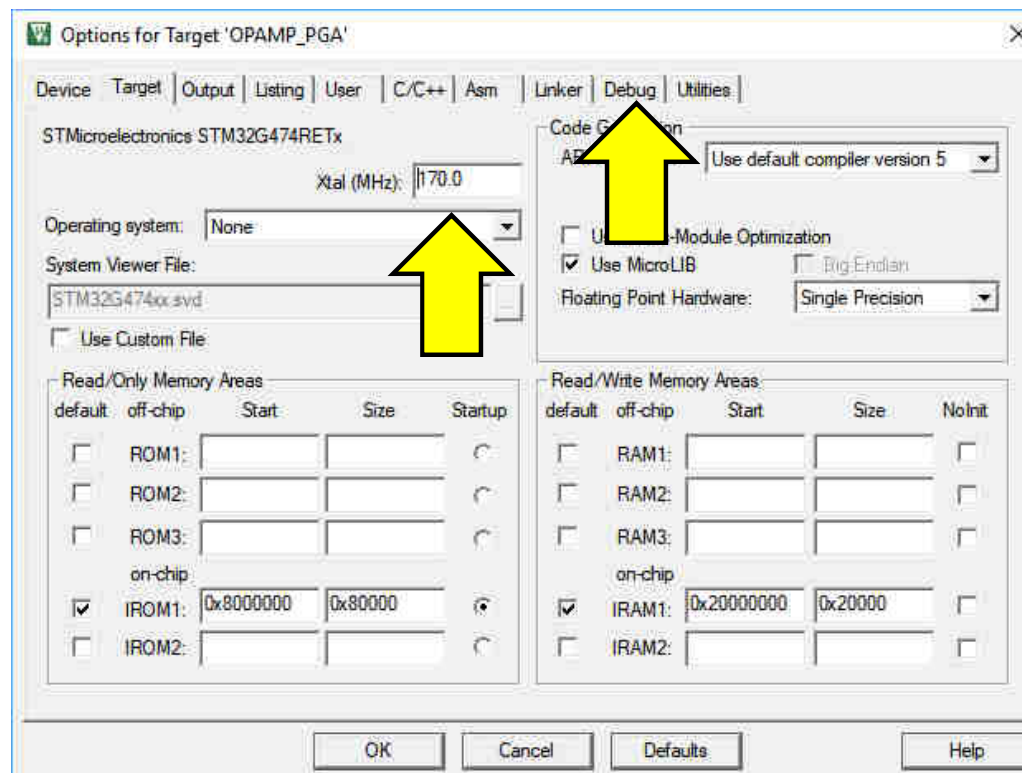
125

Check the Options



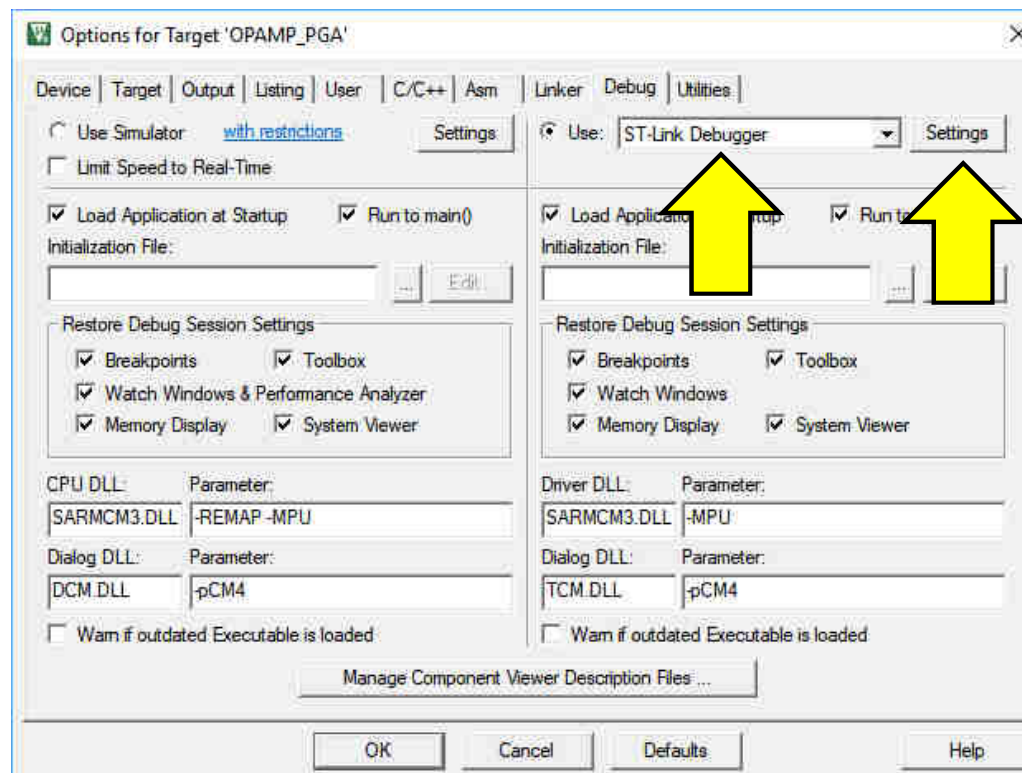


Clock speed and Debug



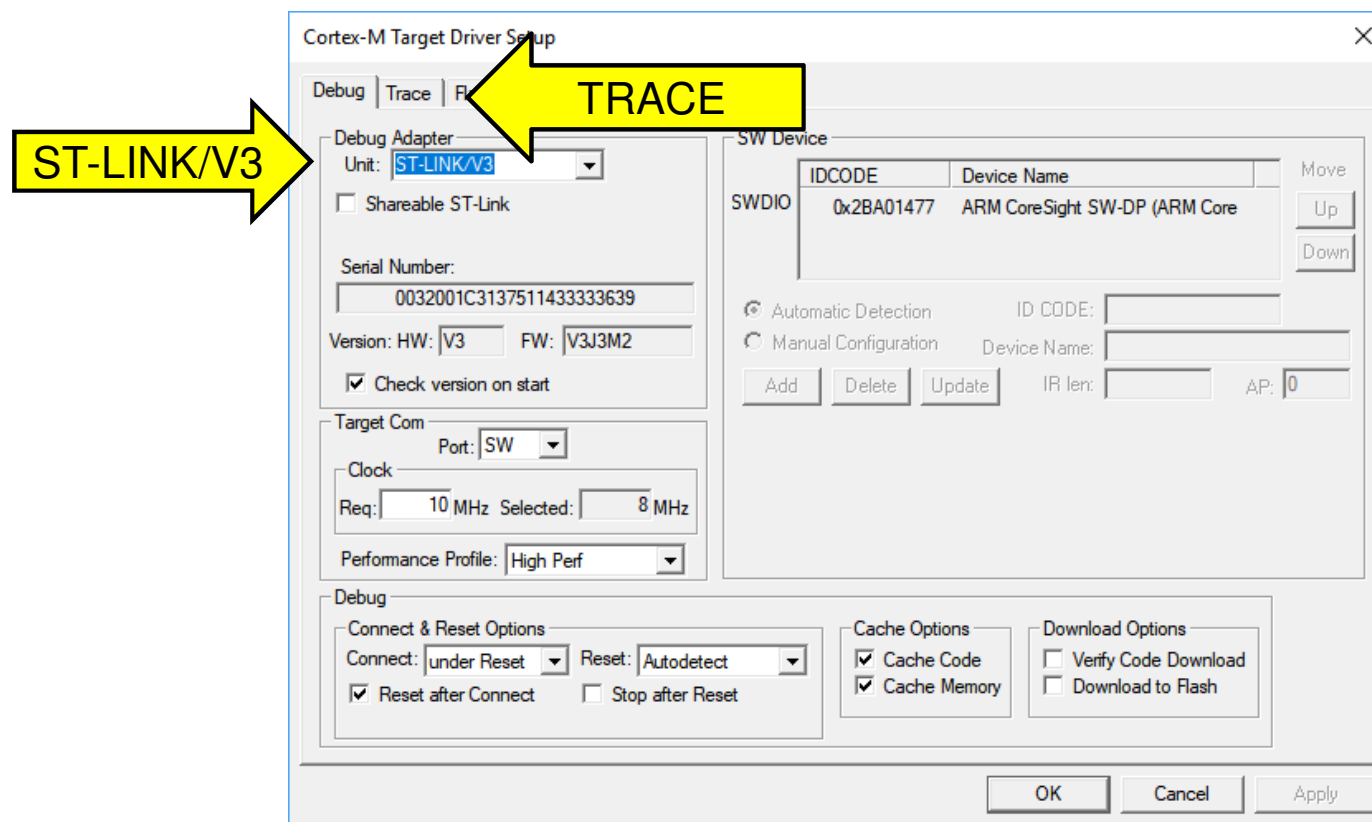


ST-Link Debugger - Settings





ST-Link/V3 & TRACE



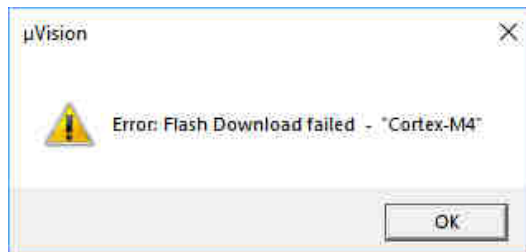
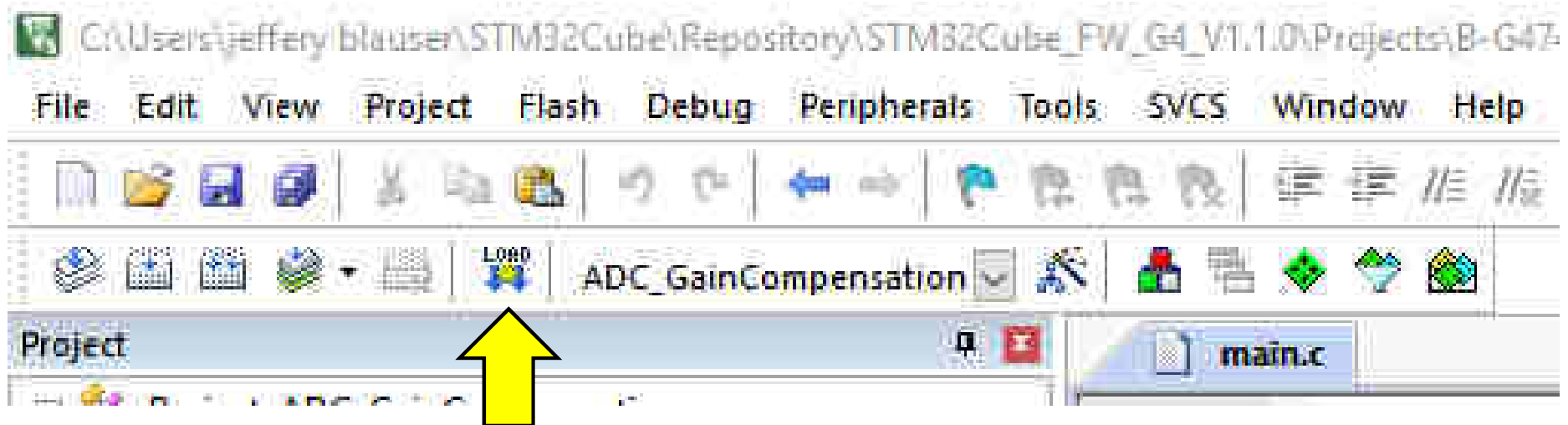


Core Clock & TRACE Enable

The screenshot shows the 'Cortex-M Target Driver Setup' dialog box with the 'Trace' tab selected. The following settings are highlighted with yellow arrows and text boxes:

- Core Clock:** 170.000000 MHz (Arrow labeled '170')
- Trace Enable:** ☒ (Arrow labeled 'TRACE Enable')
- Trace Port:** Serial Wire Output - UART/NRZ
- SWO Clock Prescaler:** 15
- Autodetect:** ☒
- SWO Clock:** 11.333333 MHz
- Timestamps:** ☒ Enable, Prescaler: 1
- PC Sampling:** Prescaler: 1024*16, ☐ Periodic, Period: <Disabled>, ☐ on Data R/W Sample
- Trace Events:** ☐ CPI: Cycles per Instruction, ☐ EXC: Exception overhead, ☐ SLEEP: Sleep Cycles, ☐ LSU: Load Store Unit Cycles, ☐ FOLD: Folded Instructions, ☐ EXCTRC: Exception Tracing (Arrow labeled 'Uncheck EXCTRC')
- ITM Stimulus Ports:** Enable: 0x00008000, Privilege: 0x00000008. The 'Port 31..24' checkbox is checked, while others are unchecked (Arrow labeled 'Uncheck all but 31 & 0').
- Advanced settings:** ☐ Ignore packets with no SYNC, ☐ Overwrite CYCCNT
- Buttons:** OK, Cancel, Apply (Arrow labeled 'OK' points to the OK button)

Load



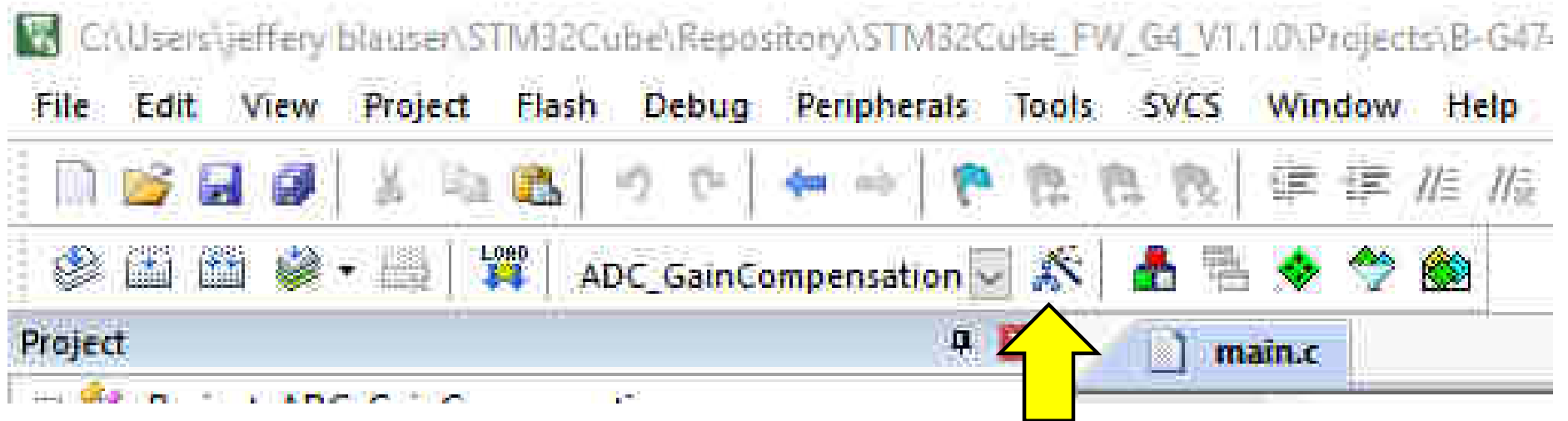
```
Build Output
Contents mismatch at: 08001860H (Flash=38H Required=C5H) !
Contents mismatch at: 08001861H (Flash=46H Required=F8H) !
Contents mismatch at: 08001862H (Flash=FEH Required=88H) !
Contents mismatch at: 08001863H (Flash=F7H Required=00H) !
Contents mismatch at: 08001864H (Flash=B9H Required=60H) !
Contents mismatch at: 08001865H (Flash=FCH Required=6CH) !
Too many errors to display !
Error: Flash Download failed - "Cortex-M4"
```



Keil MDK-ARM

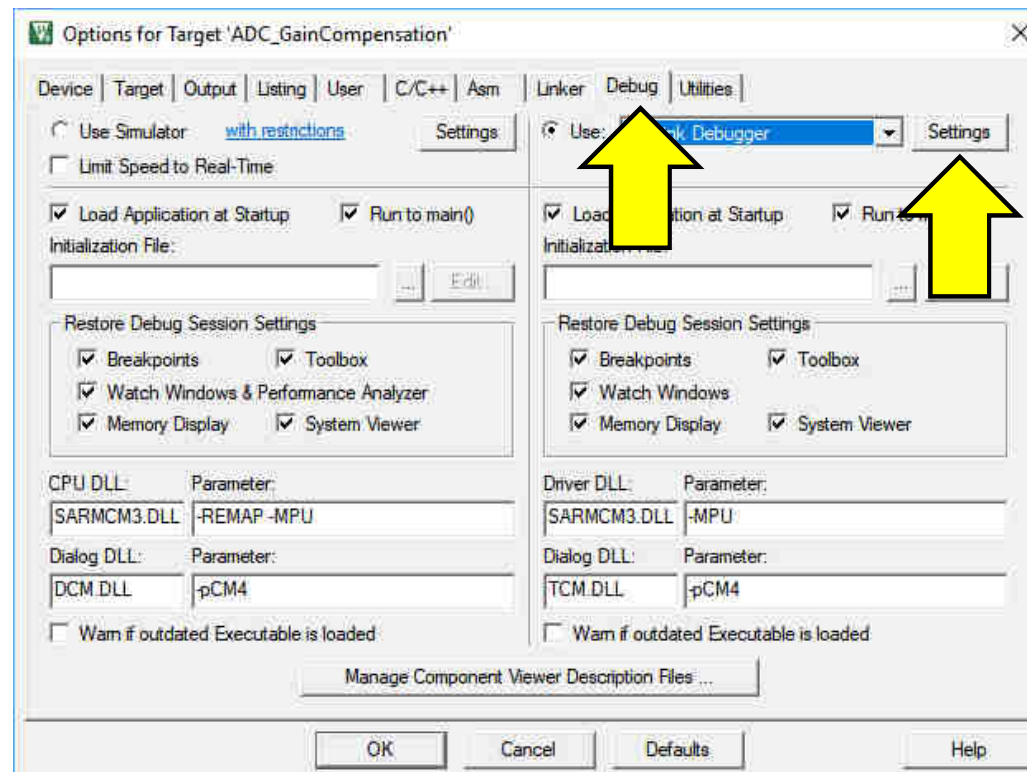
131

Options



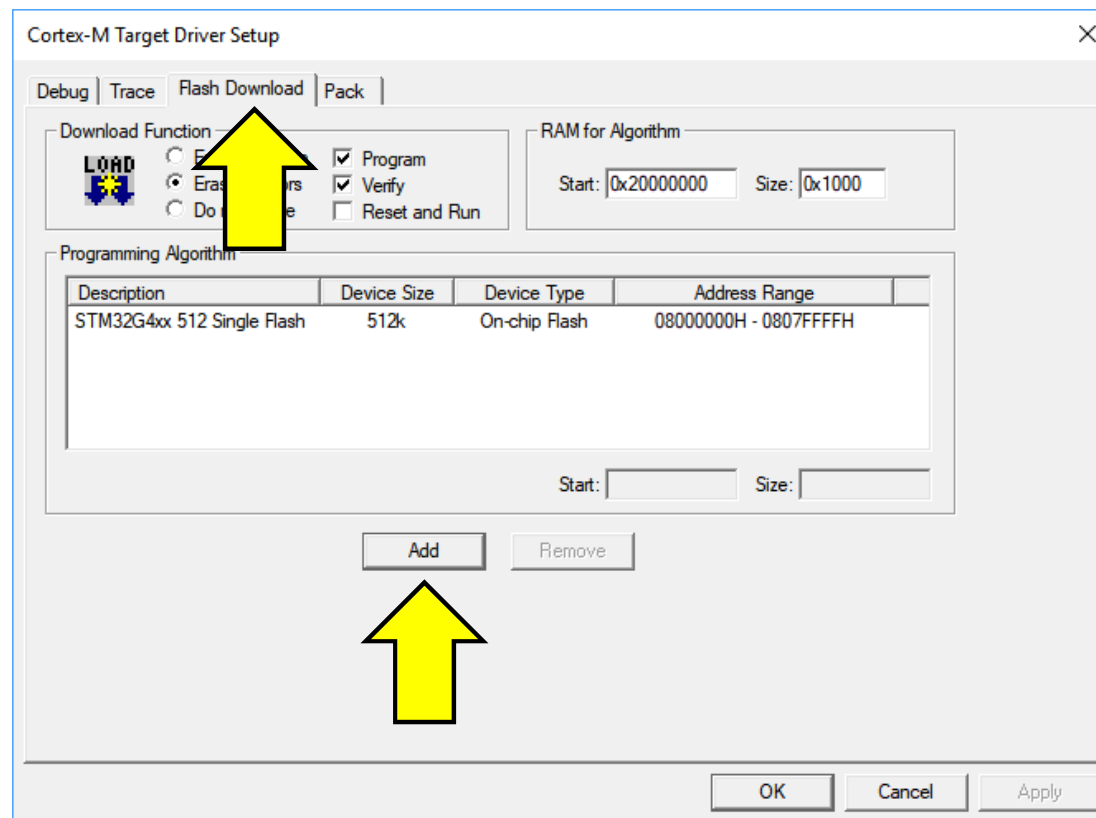


ST-Link Debugger - Settings



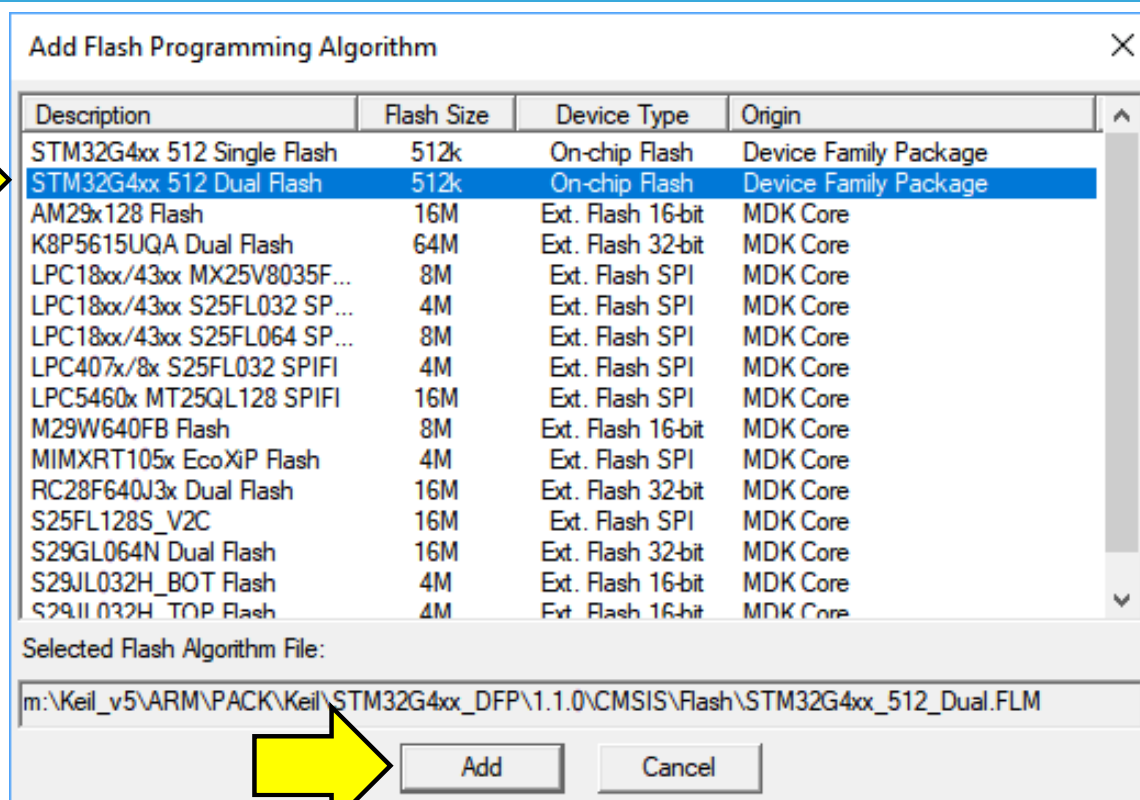
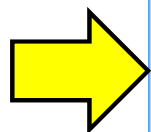


Flash Download → Add

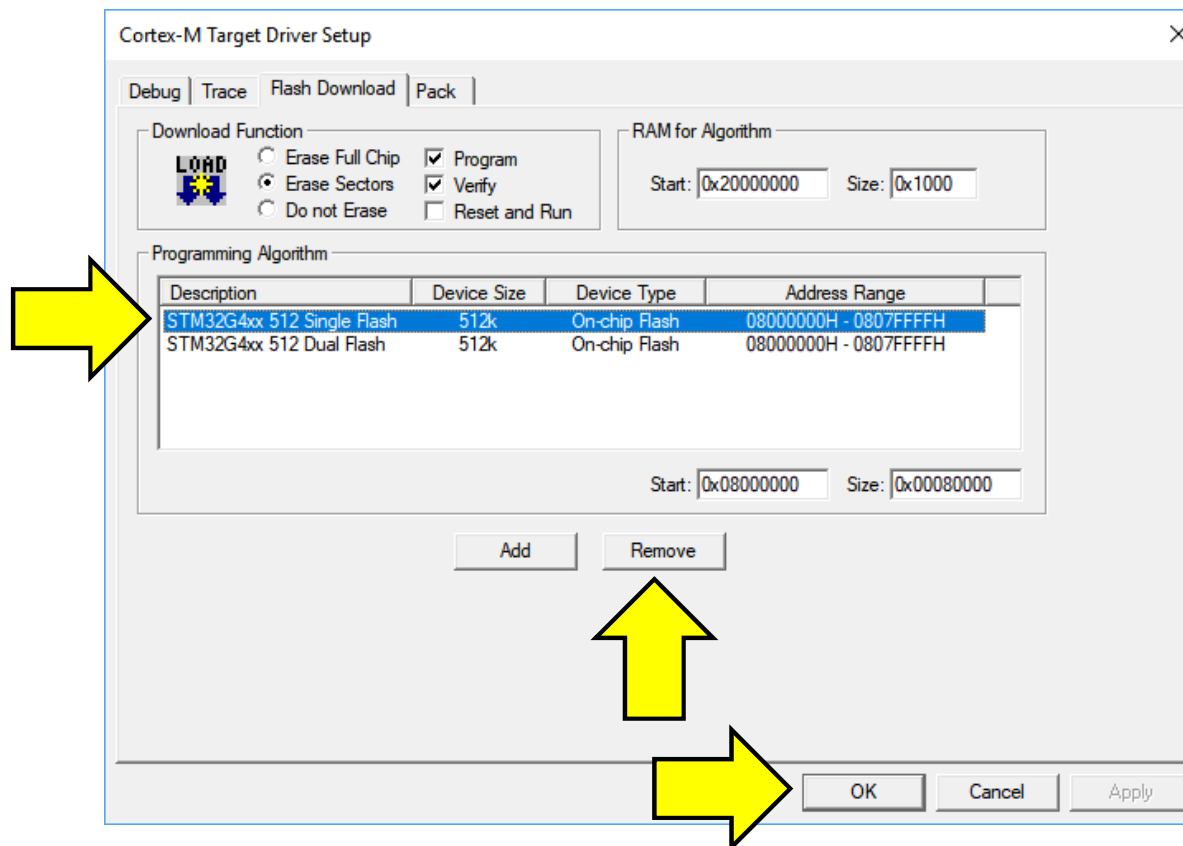




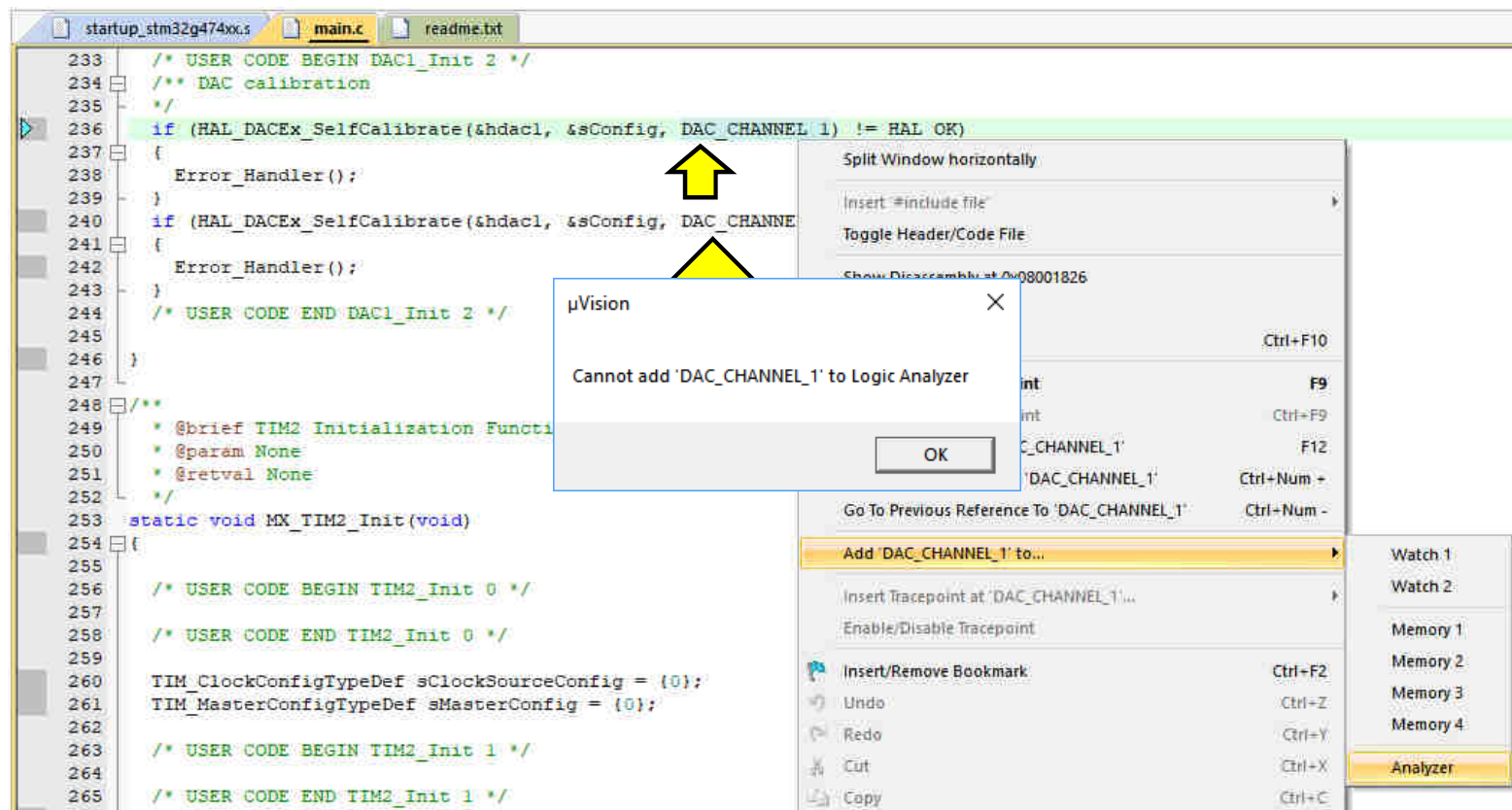
STM32G4xx 512 Dual Flash



Remove Single Flash



Place variable in the Analyzer – main.c about lines 235 & 240



```

233  /* USER CODE BEGIN DAC1_Init 2 */
234  /** DAC calibration
235  */
236  if (HAL_DACEx_SelfCalibrate(&hdac1, &sConfig, DAC_CHANNEL_1) != HAL_OK)
237  {
238      Error_Handler();
239  }
240  if (HAL_DACEx_SelfCalibrate(&hdac1, &sConfig, DAC_CHANNEL_1) != HAL_OK)
241  {
242      Error_Handler();
243  }
244  /* USER CODE END DAC1_Init 2 */
245
246  }
247
248  /**
249   * @brief TIM2 Initialization Function
250   * @param None
251   * @retval None
252   */
253  static void MX_TIM2_Init(void)
254  {
255
256      /* USER CODE BEGIN TIM2_Init 0 */
257
258      /* USER CODE END TIM2_Init 0 */
259
260      TIM_ClockConfigTypeDef sClockSourceConfig = {0};
261      TIM_MasterConfigTypeDef sMasterConfig = {0};
262
263      /* USER CODE BEGIN TIM2_Init 1 */
264
265      /* USER CODE END TIM2_Init 1 */

```

µVision

Cannot add 'DAC_CHANNEL_1' to Logic Analyzer

OK

Split Window horizontally

Insert #include file

Toggle Header/Code File

Show Disassembly at 0x08001826

Go To Previous Reference To 'DAC_CHANNEL_1'

Ctrl+F10

int

F9

Ctrl+F9

C_CHANNEL_1'

F12

'DAC_CHANNEL_1'

Ctrl+Num +

Ctrl+Num -

Add 'DAC_CHANNEL_1' to...

Insert Tracepoint at 'DAC_CHANNEL_1'...

Enable/Disable Tracepoint

Insert/Remove Bookmark

Ctrl+F2

Undo

Ctrl+Z

Redo

Ctrl+Y

Cut

Ctrl+X

Copy

Ctrl+C

Watch 1

Watch 2

Memory 1

Memory 2

Memory 3

Memory 4

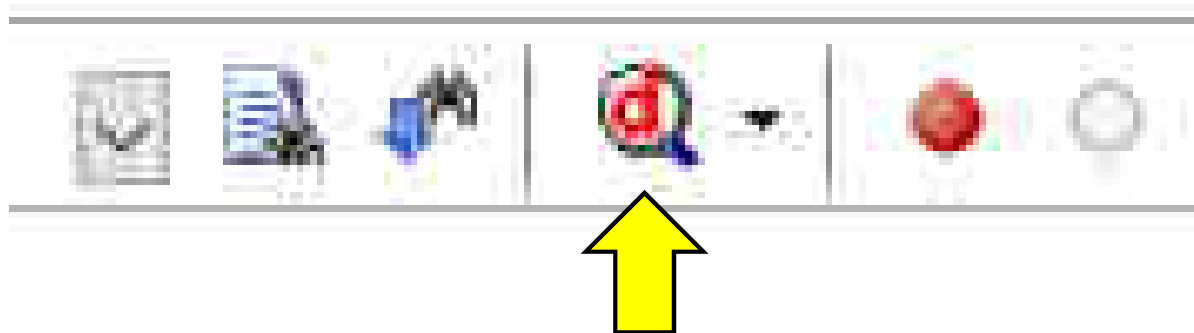
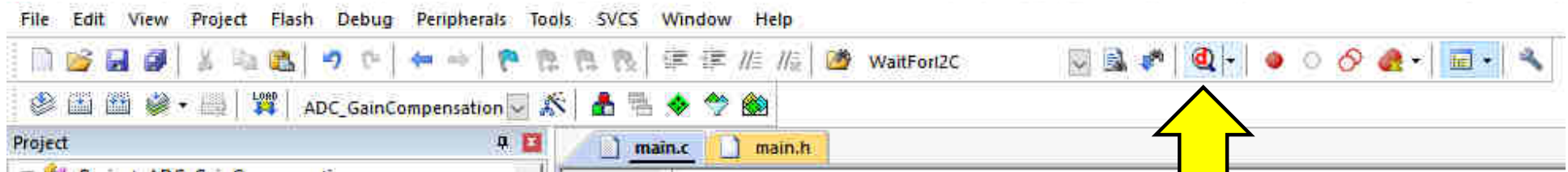
Analyzer



Keil MDK-ARM

137

Debugger – Exit back to the Editor



Add 3 new variables in main.c about line 59

```
54  /* USER CODE BEGIN PV */
55  /* Private variables -----
56  /* set to 1 after User push-button interrupt */
57  __IO uint32_t UserButtonStatus = 0;
58
59  uint16_t  Display_DAC1DOR1;
60  uint16_t  Display_DAC3DOR2;
61  uint16_t  Display_ADC3IN1;
```

See C:\2019_STM32G4_Workshop\Labs\OPAMP
CodeLinesForOpAmp.txt if you prefer to copy and paste
Or open the file from the Doc folder in the Editor



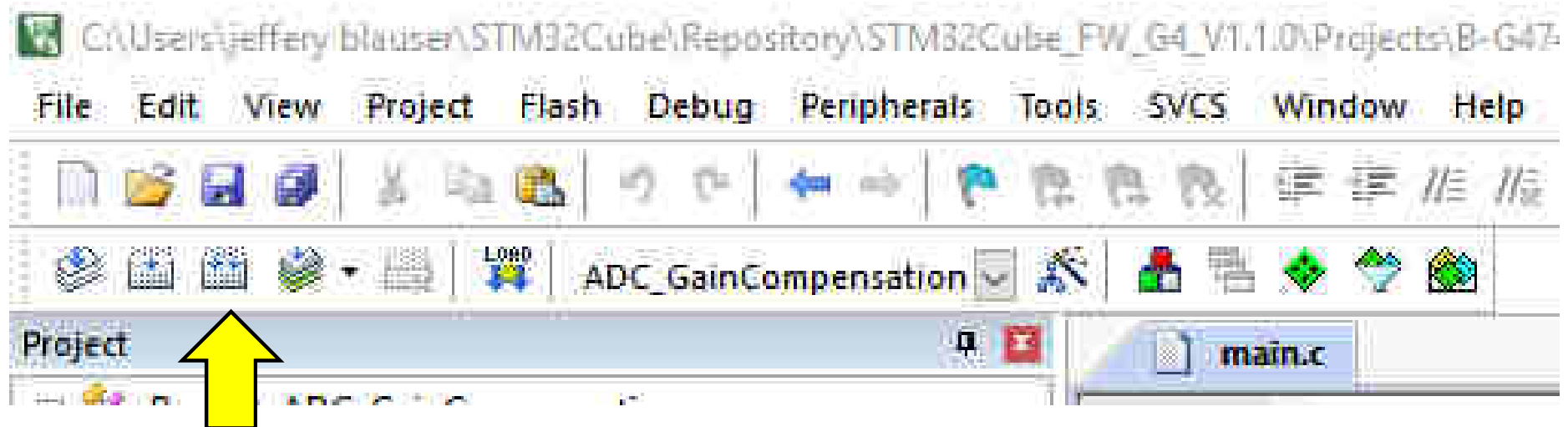
Add new lines in main.c about line 177

```
174         Error_Handler();
175     }
176 }
177 /* Display the global variables in the Logic Analyzer */
178 Display_DAC1DOR1 = DAC1->DOR1;
179 HAL_Delay(1);    //1 millisecond delay to transfer data to the PC
180 Display_DAC3DOR2 = DAC3->DOR1;
181 HAL_Delay(1);    //1 millisecond delay to transfer data to the PC
182 Display_ADC3IN1 = ADC3->DR;
183 HAL_Delay(1);    //1 millisecond delay to transfer data to the PC
184 }
185 /* USER CODE END WHILE */
```

See C:\2019_STM32G4_Workshop\Labs\OPAMP
CodeLinesForOpAmp.txt if you prefer to copy and paste
Or open the file from the Doc folder in the Editor



Build

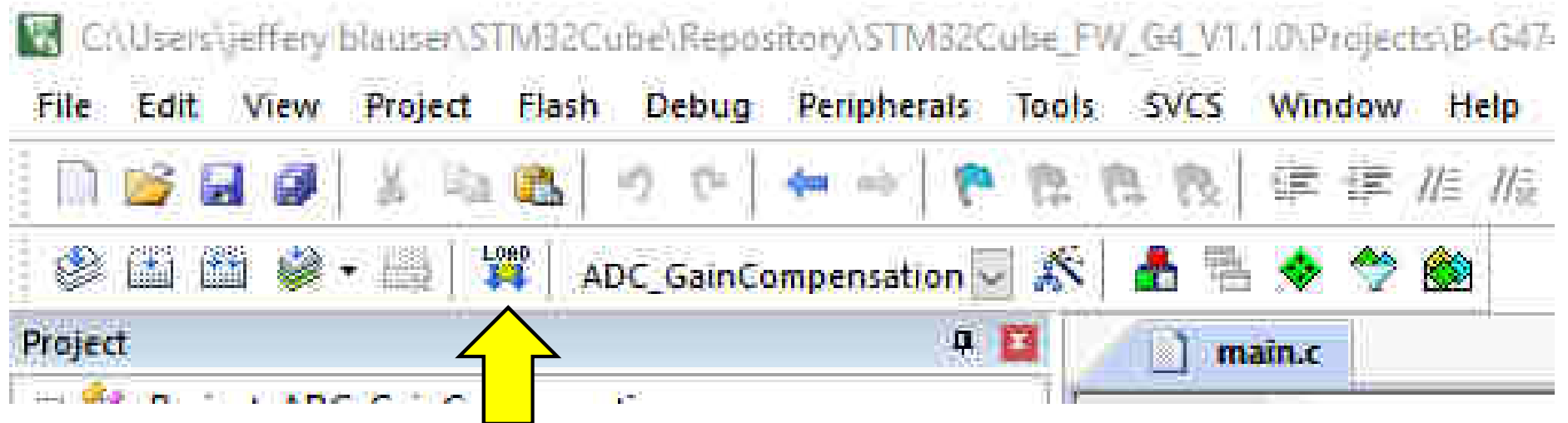


Build Output

```
Program Size: Code=6516 RO-data=596 RW-data=44 ZI-data=1268
FromELF: creating hex file...
OPAMP_PGA\OPAMP_PGA.hex: Warning: Q9931W: Your license for feature Keil will expire in 30 days
Finished: 0 information, 1 warning and 0 error messages.
"OPAMP_PGA\OPAMP_PGA.axf" - 0 Error(s), 0 Warning(s).
Build Time Elapsed: 00:01:22
```



Load



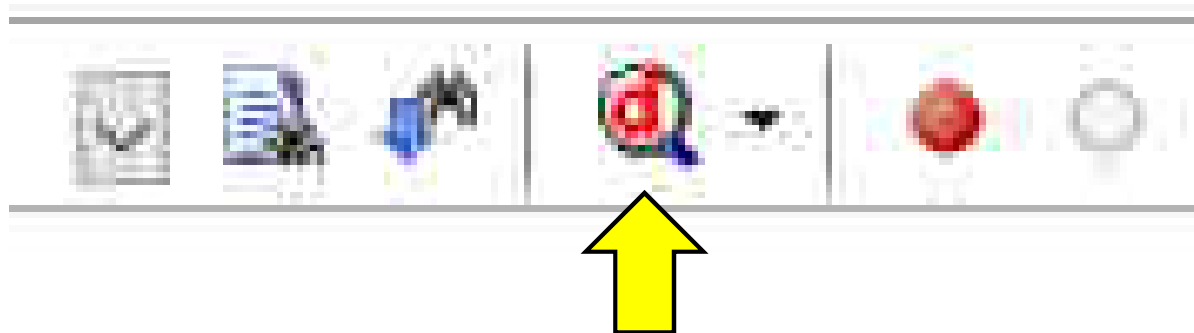
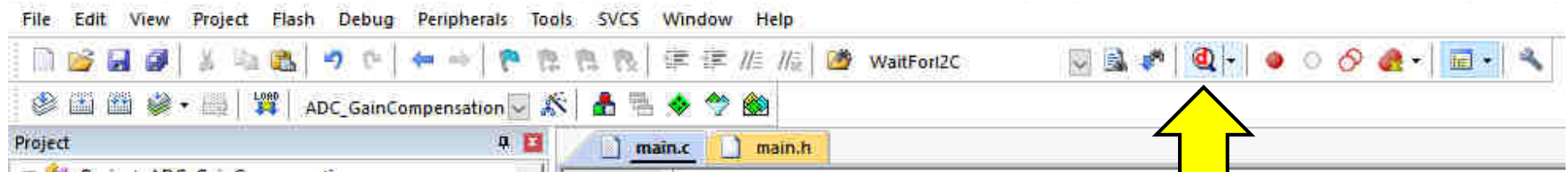
```
Build Output
Build Time Elapsed: 00:01:22
Load "OPAMP_PGA\\OPAMP_PGA.axf"
Erase Done.
Programming Done.
Verify OK.
Flash Load finished at 17:34:57
```



Keil MDK-ARM

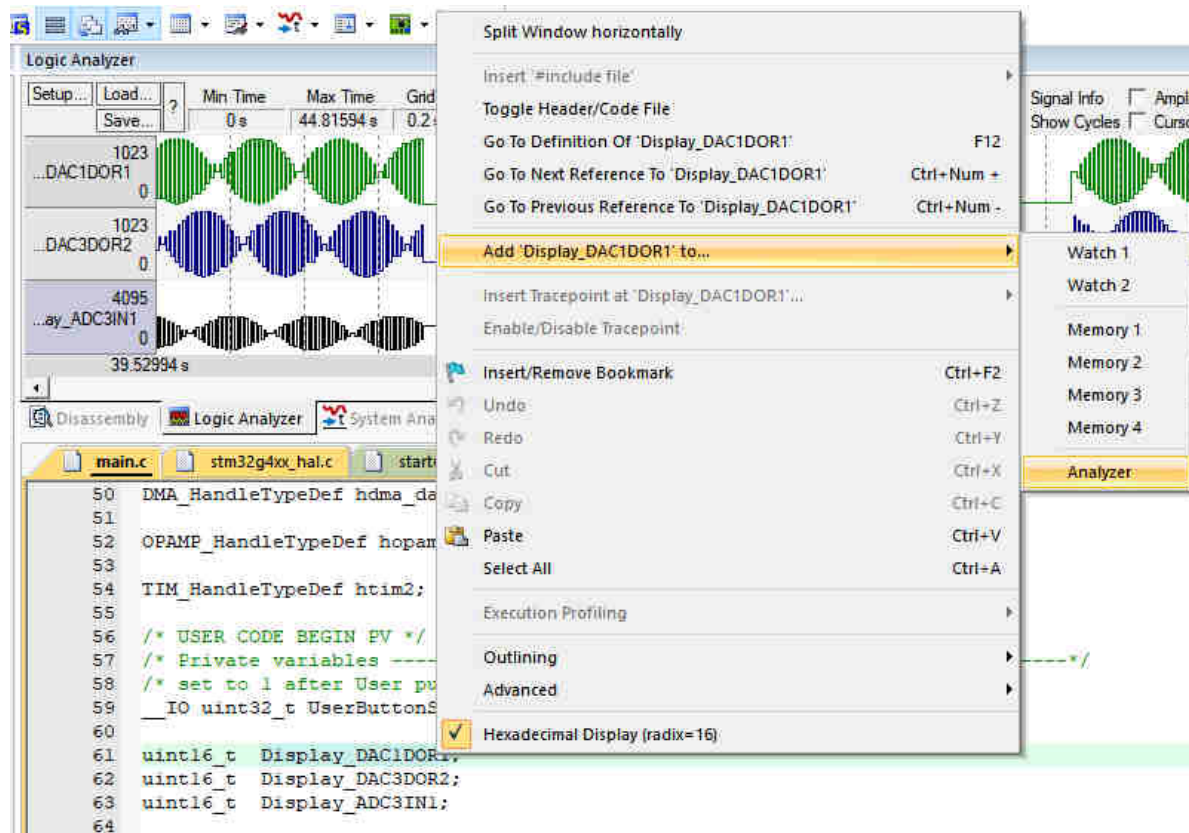
142

Debugger



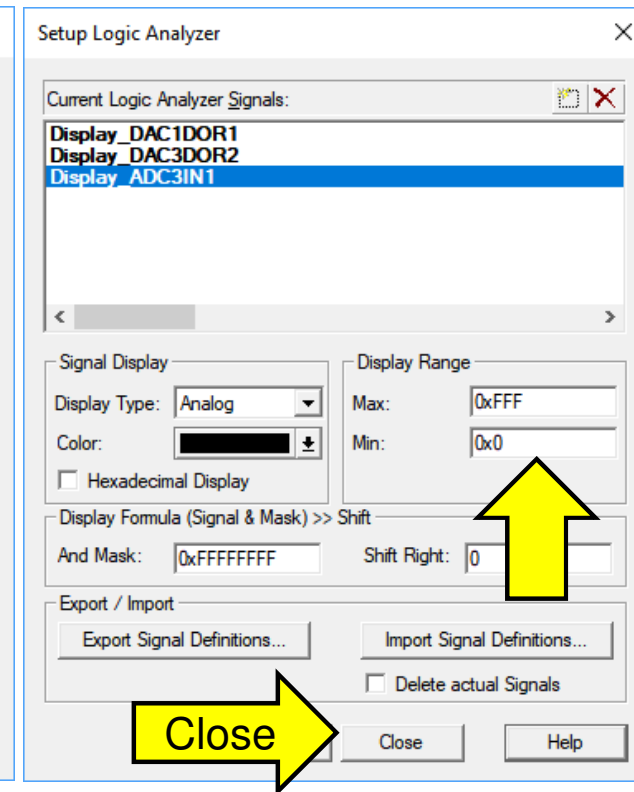
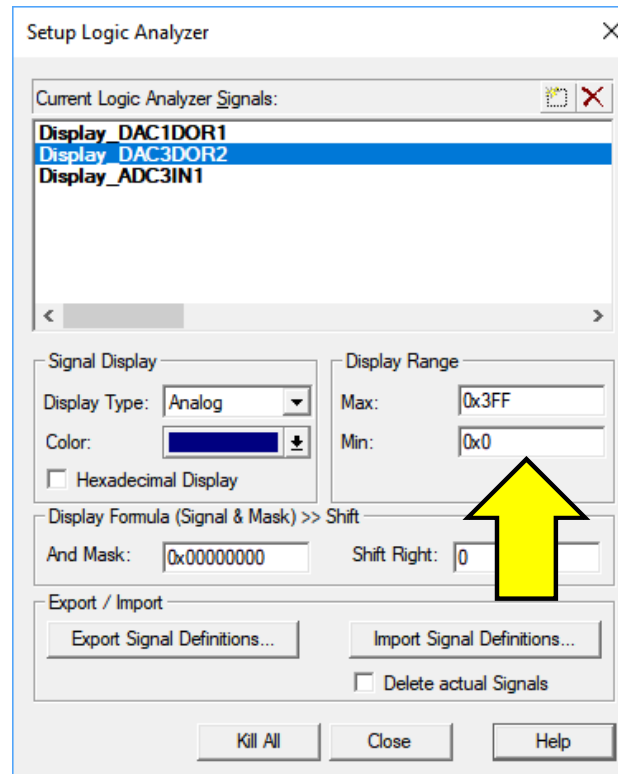
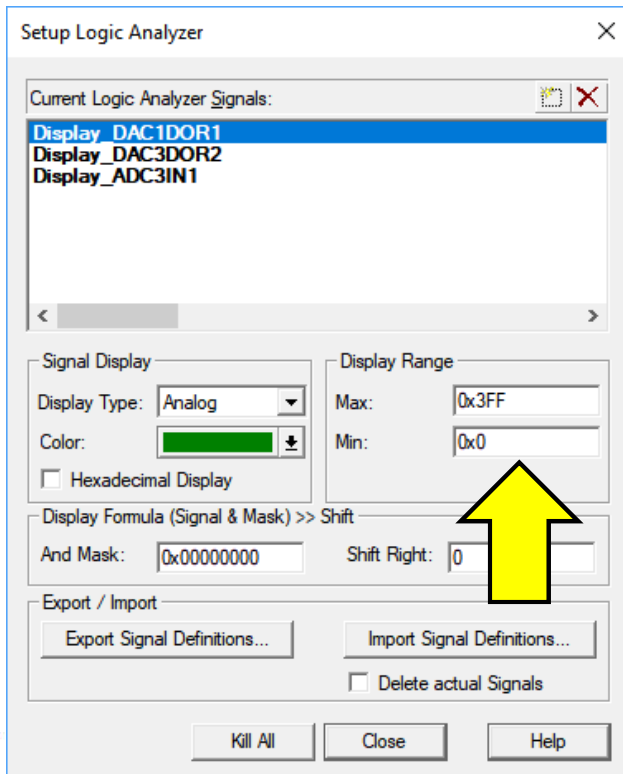


Find the 3 Display_xxx about main.c line 61 – right click on each – add to Analyzer





Setup – Min=0x0, Max = 0x3FF for DACs, then Min=0x0, Max = 0xFFF for ADC - Close

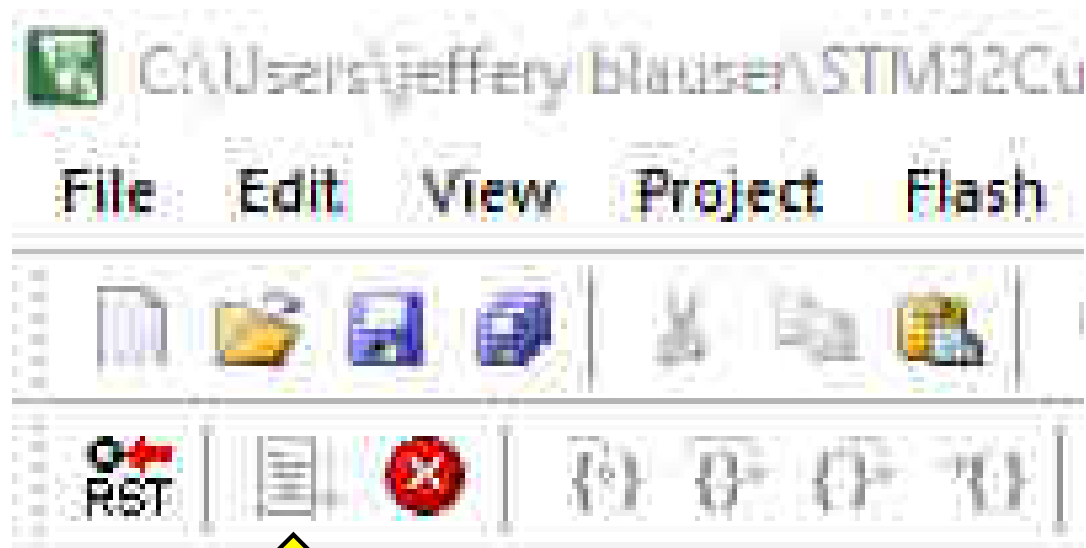




Keil MDK-ARM

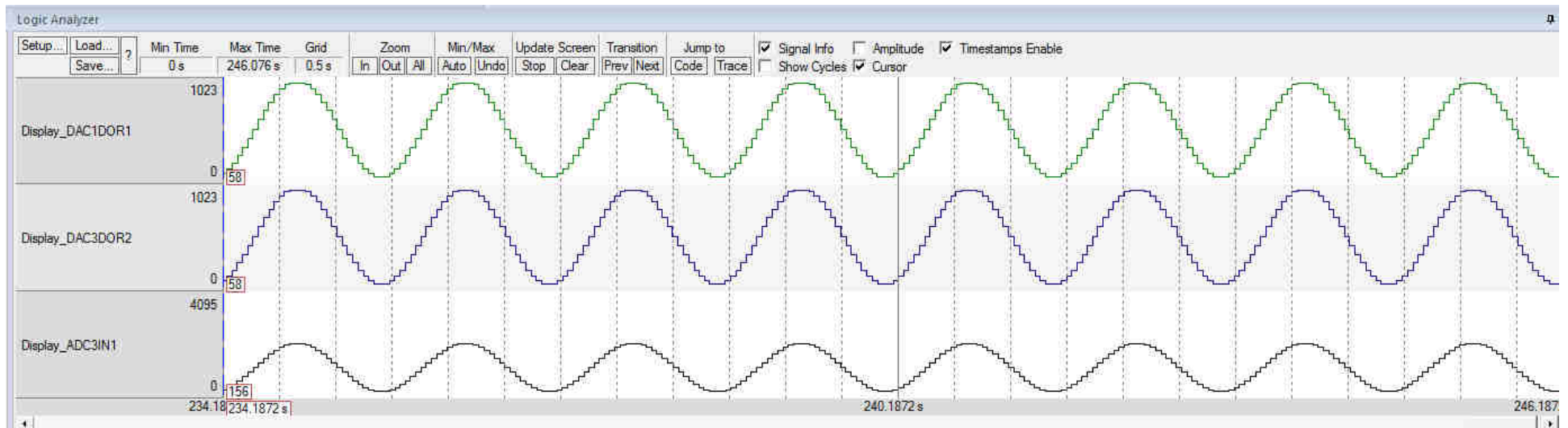
145

RUN



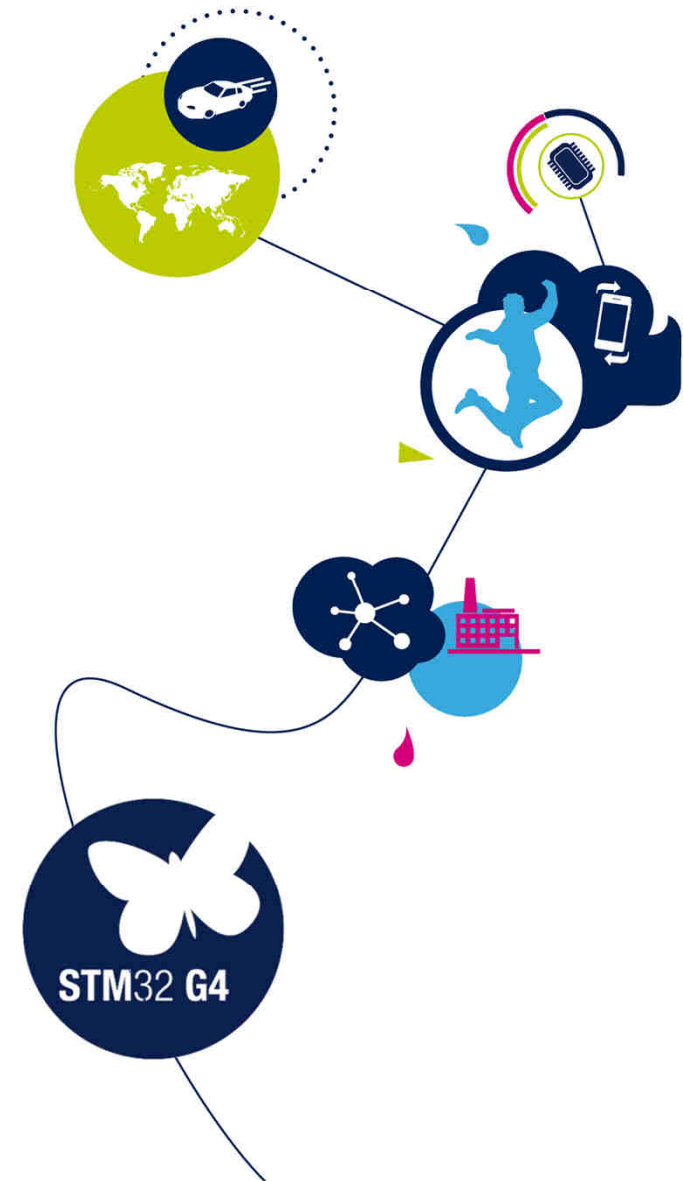


RUN



STM32G4 mixed-signal MCU workshop

*Hands On:
CORDIC Hardware Accelerator*





CORDIC: Main features

148

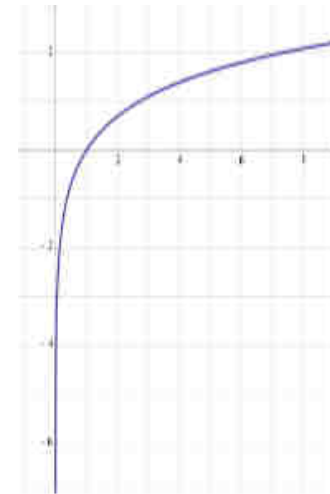
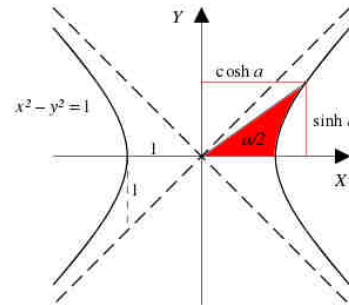
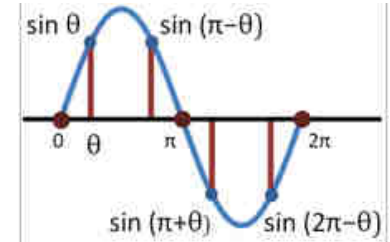
- 24-bit CORDIC rotation engine
 - Trigonometric and Hyperbolic functions
- Circular and Hyperbolic modes
- Rotation and Vectoring modes
- Programmable precision
- Low latency AHB slave interface
- DMA read and write channels



Cordic: Main features

149

- Supported functions:
 - Sine, Cosine: Vector rotation (polar to rectangular)
 - Arctangent, Modulus: Vector translation (rectangular to polar)
 - Hyperbolic cosine
 - Hyperbolic sine
 - Hyperbolic tangent
 - Natural logarithm
 - Square root
- Fixed point signed input and output
 - Q1.31 or Q1.15 format
- Performance improvement relative to software
 - Sine and Cosine functions are typically 5x faster than software execution





CORDIC - Precision

150

- Precision is dependent on number of iterations

Function	Number of iteration (precision)	Number of cycles	Max residual error	
			q1.31 format	q1.15. format
Sin, Cos, Phase, Mod, Atan	4	1	2^{-3}	2^{-3}
	8	2	2^{-7}	2^{-7}
	12	3	2^{-11}	2^{-11}
	16	4	2^{-15}	2^{-15}
	20	5	2^{-18}	2^{-16}
	24	6	2^{-19}	2^{-16}



CORDIC - Precision

151

Function	Number of iteration (precision)	Number of cycles	Max residual error	
			q1.31 format	q1.15. format
Sinh, Cosh, Atanh, Ln	4	1	2^{-2}	2^{-2}
	8	2	2^{-6}	2^{-6}
	12	3	2^{-10}	2^{-10}
	16	4	2^{-13}	2^{-13}
	20	5	2^{-17}	2^{-15}
	24	6	2^{-18}	2^{-15}
Sqrt	4	1	2^{-7}	2^{-7}
	8	2	2^{-14}	2^{-14}
	12	3	2^{-19}	2^{-15}



CORDIC accelerator

152

Objective:

- To show how to configure the CORDIC accelerator.
 - Calculate the SIN and COS of angles with 6 cycles of precision.
- Enable the Event Recorder to measure the time for the CORDIC and ARM Math Library functions.
- Display CORDIC results in the Logic Analyzer.



CORDIC configuration

153

- Open CubeMX project “CORDIC_SinCos_DMA_Perf.ioc” file
 - \labs\Cordic\CORDIC_SinCos_DMA_Perf

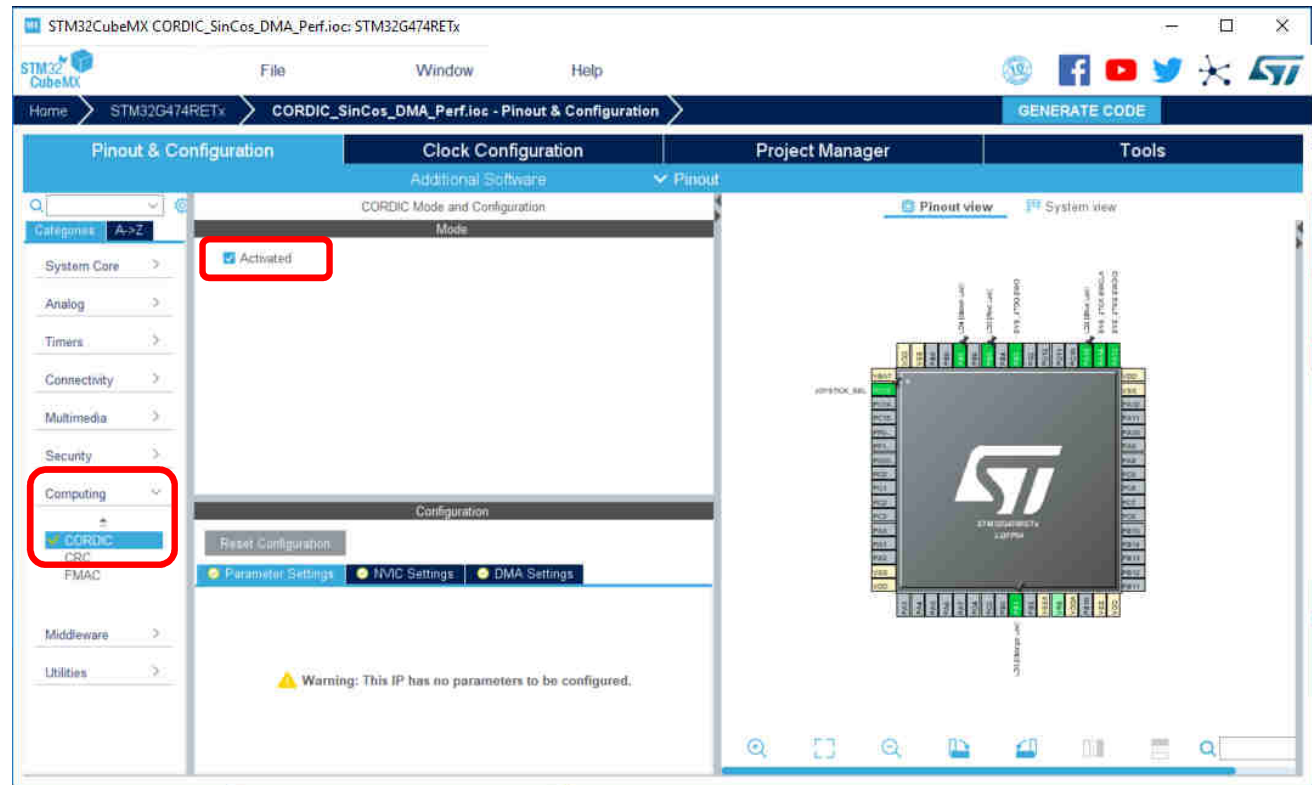
Name	Date modified	Type	Size
Drivers	8/28/2019 4:57 PM	File folder	
Inc	8/28/2019 8:22 PM	File folder	
MDK-ARM	8/29/2019 2:53 AM	File folder	
Src	8/28/2019 8:22 PM	File folder	
.mxproject	8/28/2019 8:22 PM	MXPROJECT File	7 KB
MX CORDIC_SinCos_DMA_Perf.ioc	8/29/2019 3:32 AM	IOC File	9 KB



CORDIC configuration

154

- Expand “Computing” Category
- Select “CORDIC”
 - CHECK “Activated”





CORDIC configuration

155

- DMA Request is enabled to transfer data in and out of CORDIC
- 32-bit Data Width (Word)

Configuration

Reset Configuration

Parameter Settings NVIC Settings **DMA Settings**

DMA Request	Channel	Direction	Priority
CORDIC_READ	DMA1 Channel 1	Peripheral To Memory	Low
CORDIC_WRITE	DMA1 Channel 2	Memory To Peripheral	Low

Add Delete

DMA Request Settings

Mode: Normal

Increment Address: ☐

Data Width: Word

DMA Request Synchronization Settings

Enable synchronization: ☐

Synchronization signal:

Signal polarity:

Enable event: ☐

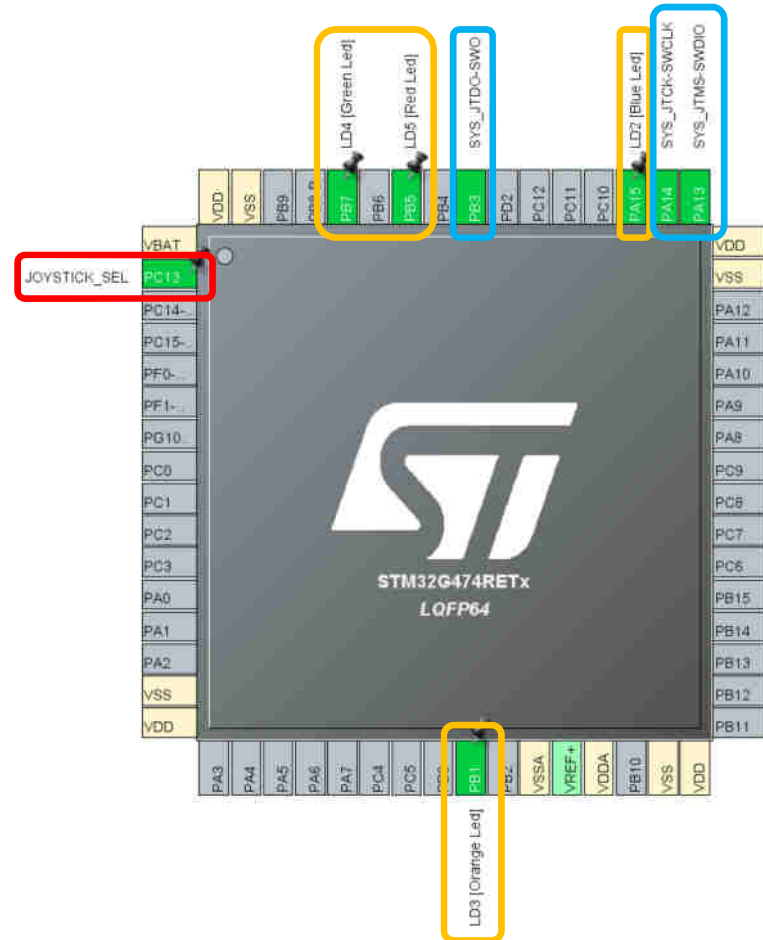
Request number:



Additional Peripheral configuration

156

- **JOYSTICK_SEL**: EXTI13
 - PC13
- **LD2-5**: GPIO_Output
 - PA15, PB1, PB5 and PB7
- **Serial Wire Debug with Trace (SWO)**:
 - PA13, PA14 and PB3

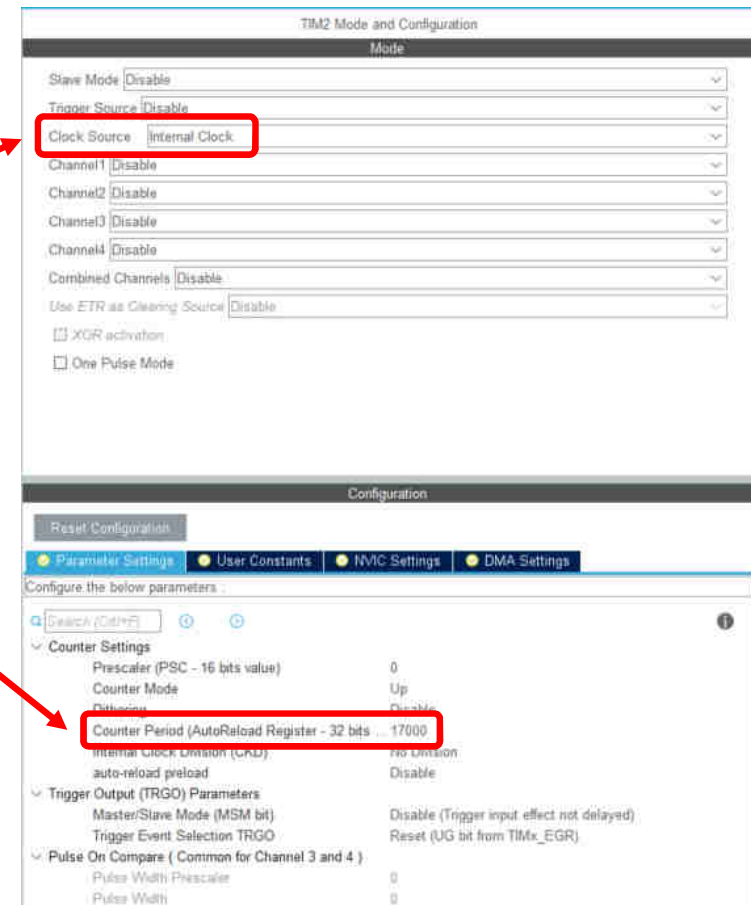
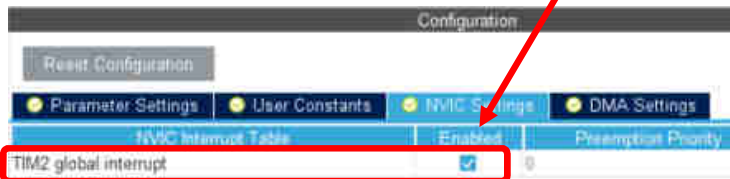




Additional Peripheral configuration

157

- **Timer 2:** Debug Logic Analyzer
- Internal Clock Source = 170 MHz
- Counter Period = 17000
 - 10 kHz timebase interrupt
- TIM2 global interrupt Enabled

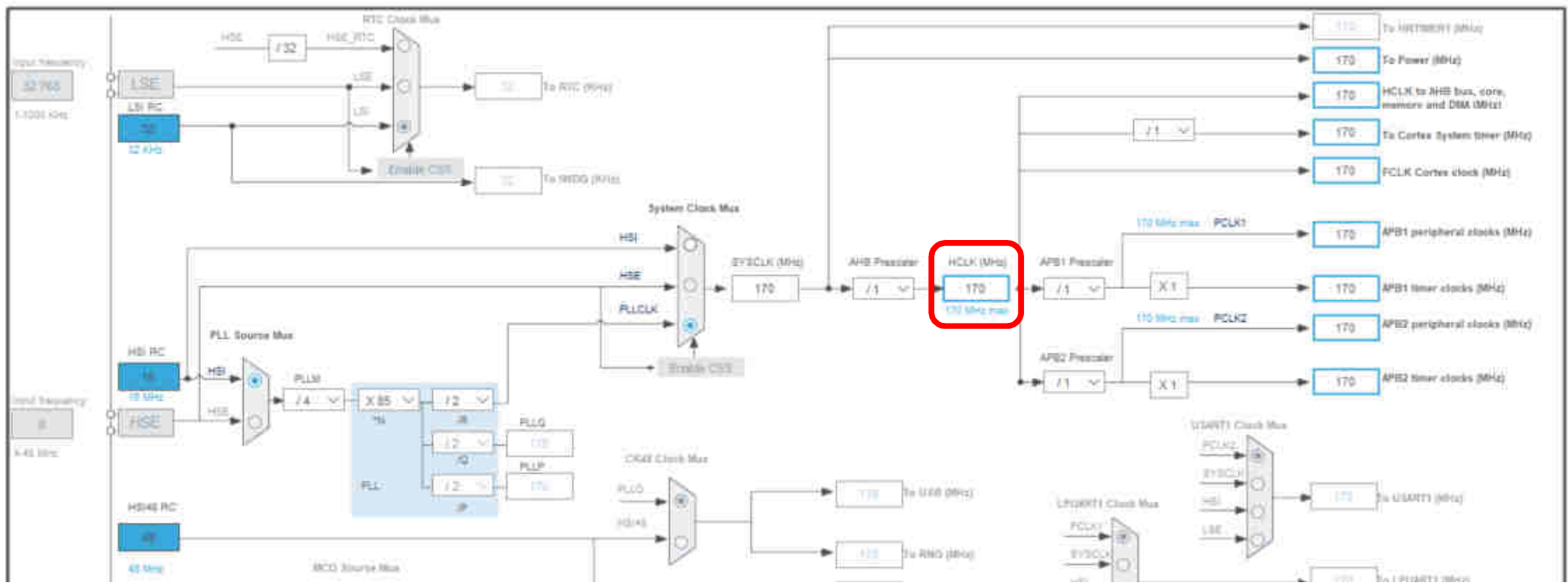




Hands-On: CORDIC

158

- Validate “HCLK (MHz)” is set to 170

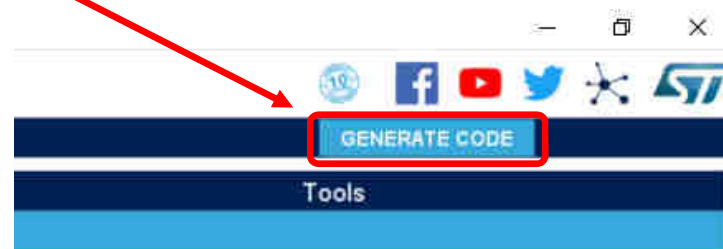




Hands-On: CORDIC

159

- Generate Code



- Open Project in MDK-ARM





Hands-On: CORDIC

160

- aAnglesCordic - Array of angles for CORDIC Q1.31 format (Line 57)

```
56 /* Array of angles for CORDIC Q1.31 format, regularly incremented from 0 to 2*pi */
57 static int32_t aAnglesCordic[ARRAY_SIZE] =
58 {
59     0x00000000, 0x04000000, 0x08000000, 0x0C000000,
```

- aAnglesLib – Array of angles for ARM Math library (Line 78)

```
77 /* Array of angles for arm math library Q1.31 format, regularly incremented from 0 to 2*pi */
78 static int32_t aAnglesLib[ARRAY_SIZE] =
79 {
80     0x00000000, 0x02000000, 0x04000000, 0x06000000,
```

- aRefSin & aRefCos – Array of Sin & Cos result reference (Line 99 & 120)

```
98 /* Array of reference sines in Q1.31 format */
99 static int32_t aRefSin[ARRAY_SIZE] =
100 {
101     0x00000000, 0x0C8BD35E, 0x18F8B83C, 0x25280C5D,
```

```
119 /* Array of reference cosines in Q1.31 format */
120 static int32_t aRefCos[ARRAY_SIZE] =
121 {
122     0x80000000, 0x7F62368F, 0x7D8A5F3F, 0x7A7D055B,
```

- aResults – Array of calculation results (Line 142)

```
140 /* Array of calculation results in Q1.31 format.
141    Will contain alternatively Sine and Cosine of input angles */
142 static int32_t aResults[2 * ARRAY_SIZE];
```



CORDIC configuration

161

- Sine calculation function (Line 219)
- 6 cycles of precision (Line 220)
- No Scaling (Line 221)
- One input angle data (Line 222)
 - Q1.31 format for input data (Line 224)
- Two output data (Sine and Cosine) (Line 223)
 - Q1.31 format for input data (Line 225)

```
217  /* USER CODE BEGIN 2 */
218  /*##-2- Configure the CORDIC peripheral #####
219  sCordicConfig.Function      = CORDIC_FUNCTION_SINE;
220  sCordicConfig.Precision     = CORDIC_PRECISION_6CYCLES;
221  sCordicConfig.Scale         = CORDIC_SCALE_0;
222  sCordicConfig.NbWrite       = CORDIC_NBWRITE_1;
223  sCordicConfig.NbRead        = CORDIC_NBREAD_2;
224  sCordicConfig.InSize        = CORDIC_INSIZE_32BITS;
225  sCordicConfig.OutSize       = CORDIC_OUTSIZE_32BITS;
```



CORDIC configuration

162

- Step = 0

- Calculation made with CORDIC
- Simple call to
HAL_CORDIC_Calculate_DMA

```
246  /*##### Calculation using CORDIC #####*/
247  case 0:
248      for (uint32_t i = 0; i < LOOP_NB; i++)
249      {
250
251          /* Start calculation of sines in DMA mode */
252          if (HAL_CORDIC_Calculate_DMA(&hcordic, aAnglesCordic, aResults,
253                                     ARRAY_SIZE, CORDIC_DMA_DIR_IN_OUT) != HAL_OK)
254          {
255              /* Processing Error */
256              Error_Handler();
257          }
258      }
```

- Step = 1

- Calculation made with ARM Math library
- Lines 277 through 283 are called to
perform same CORDIC operation

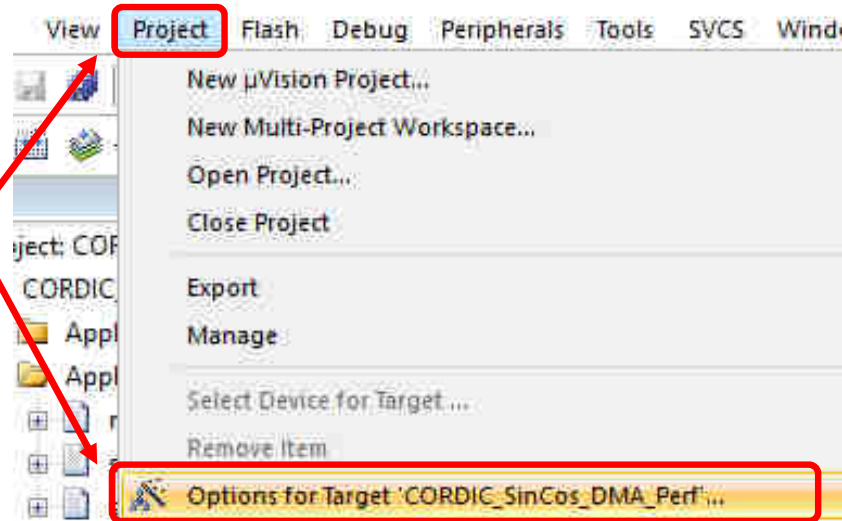
```
272  /*##### Calculation using arm math library #####*/
273  case 1:
274      for (uint32_t i = 0; i < LOOP_NB; i++)
275      {
276
277          for (uint32_t j = 0; j < ARRAY_SIZE; j++)
278          {
279              /* Calculate sine */
280              aResults[2*j] = arm_sin_q31(aAnglesLib[j]);
281
282              /* Calculate cosine */
283              aResults[(2*j) + 1] = arm_cos_q31(aAnglesLib[j]);
284          }
285      }
```



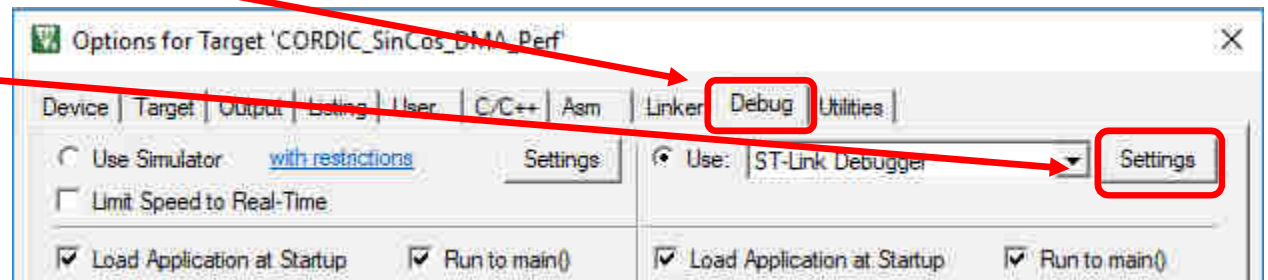
Enable Debug Trace

163

- Open “Options for Target ‘CORDIC_SinCos_DMA_Perf’ ...” window.



- Switch to “Debug” tab
- Click “Settings” button

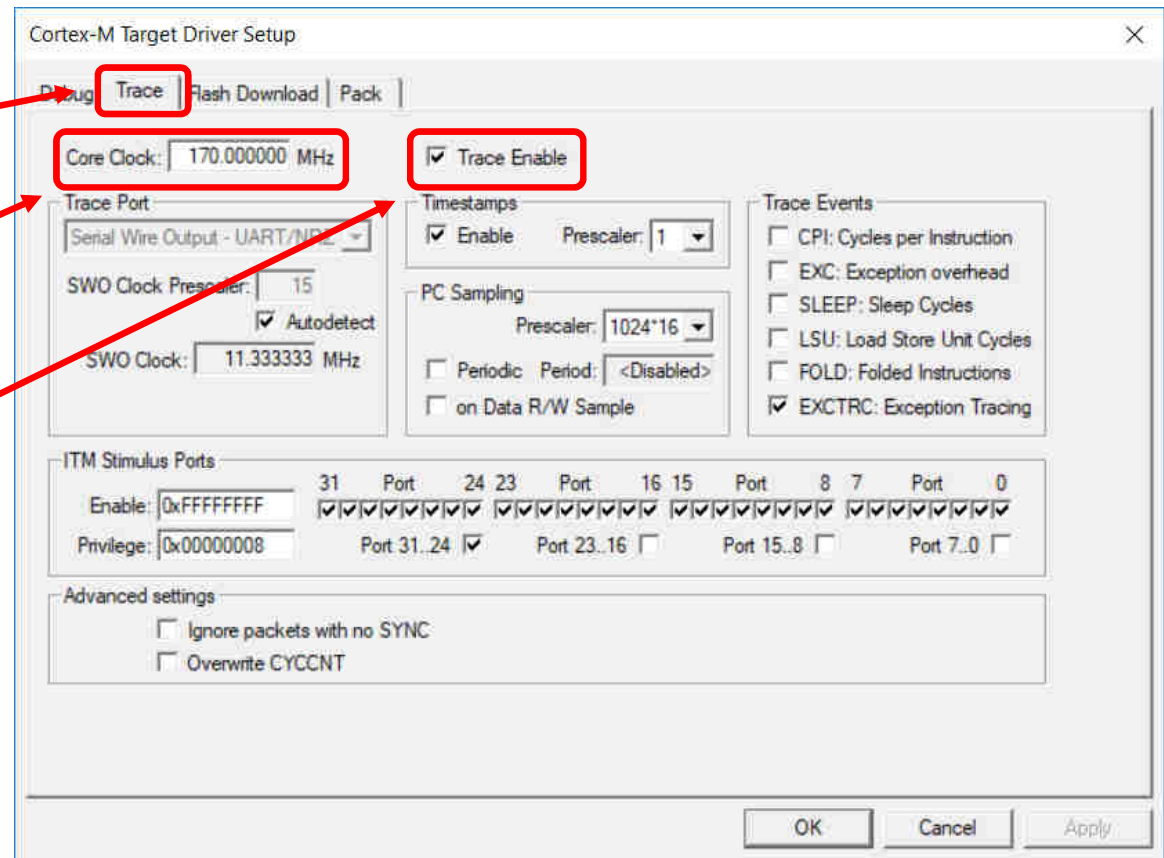




Enable Debug Trace

164

- Switch to “Trace” tab
- Set “Core Clock” = 170 MHz
- Click “Trace Enable” check box



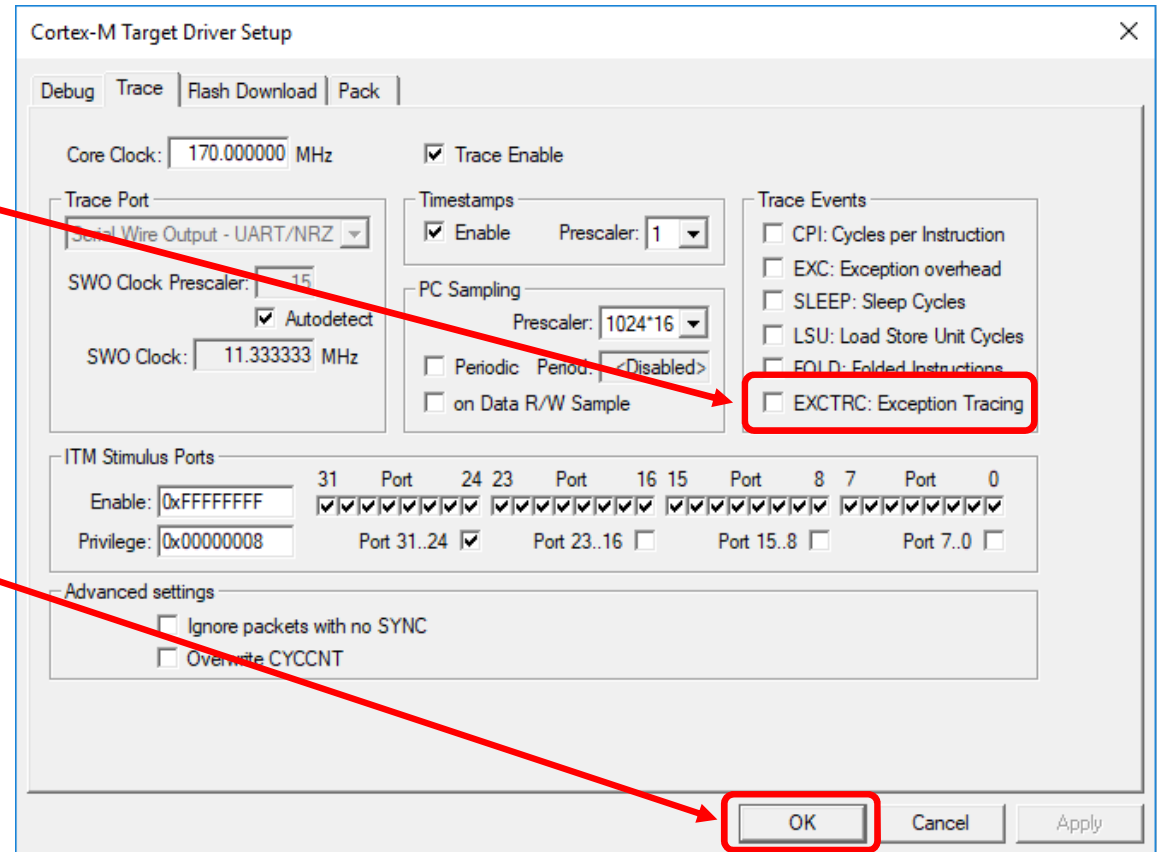


Enable Debug Trace

165

- Disable “Exception Tracing”
 - Uncheck Box

- Click “OK” to close setup window

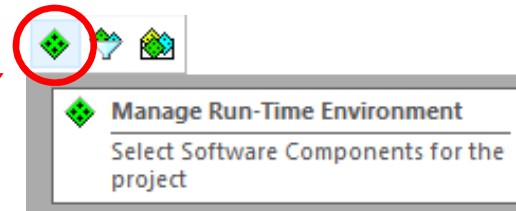




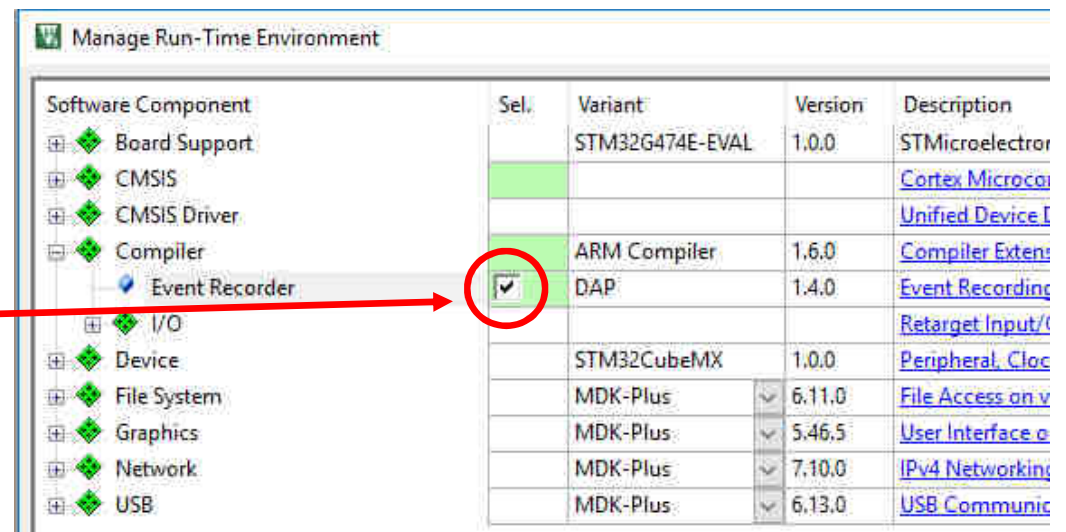
Enable Event Recorder

166

- Click on Manage Run-Time Environment



- Expand "Compiler"
- Enable "Event Recorder"
 - Check Box
- Click "OK"





Enable Event Recorder

167

- Open “main.c” under Application/User folder



- Line 27: Add (uncomment)

#include “EventRecorder.h”

```
24  /* Private includes -----
25  /* USER CODE BEGIN Includes */
26  #include <arm_math.h>
27  #include "EventRecorder.h"
28  /* USER CODE END Includes */
```

- Line 191 & 192: Add (uncomment)

EventRecorderInitialize(EventRecordAll, 1);
EventRecorderStart();

```
178 int main(void)
179 {
180     /* USER CODE BEGIN 1 */
181     /* STM32G4xx HAL library initialization:
182        - Configure the Flash prefetch
183        - SysTick timer is configured by default as source of time base, but user
184          can eventually implement his proper time base source (a general purpose
185          timer for example or other time source), keeping in mind that Time base
186          duration should be kept 1ms since PPP_TIMEOUT_VALUES are defined and
187          handled in milliseconds basis.
188        - Set NVIC Group Priority to 4
189        - Low Level Initialization
190     */
191     EventRecorderInitialize(EventRecordAll, 1);
192     EventRecorderStart();
193     /* USER CODE END 1 */
```



Enable Event Recorder

168

- Add (uncomment)

Line 250: EventRecord2(0,i,1);

Line 258: EventRecord2(0,i,2);

```
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
/*##### Calculation using CORDIC #####*/  
case 0:  
    for (uint32_t i = 0; i < LOOP_NB; i++)  
    {  
        EventRecord2(0,i,1);  
        /* Start calculation of sines in DMA mode */  
        if (HAL_CORDIC_Calculate_DMA(&hcordic, aAnglesCordic, aResults,  
                                     ARRAY_SIZE, CORDIC_DMA_DIR_IN_OUT) != HAL_OK)  
        {  
            /* Processing Error */  
            Error_Handler();  
        }  
        EventRecord2(0,i,2);  
    }
```

- Add (uncomment)

Line 276: EventRecord2(1,i,1);

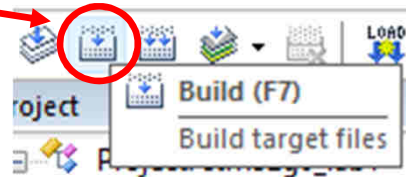
Line 285: EventRecord2(1,i,2);

```
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
/*##### Calculation using arm math library #####*/  
case 1:  
    for (uint32_t i = 0; i < LOOP_NB; i++)  
    {  
        EventRecord2(1,i,1);  
        for (uint32_t j = 0; j < ARRAY_SIZE; j++)  
        {  
            /* Calculate sine */  
            aResults[2*j] = arm_sin_q31(aAnglesLib[j]);  
  
            /* Calculate cosine */  
            aResults[(2*j) + 1] = arm_cos_q31(aAnglesLib[j]);  
        }  
        EventRecord2(1,i,2);  
    }
```

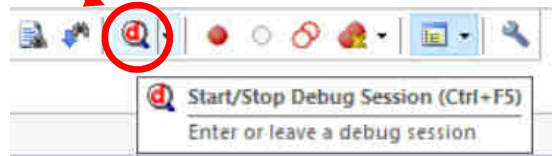
Hands-On: CORDIC

169

- Click the “Build”



- “Start Debug” Session



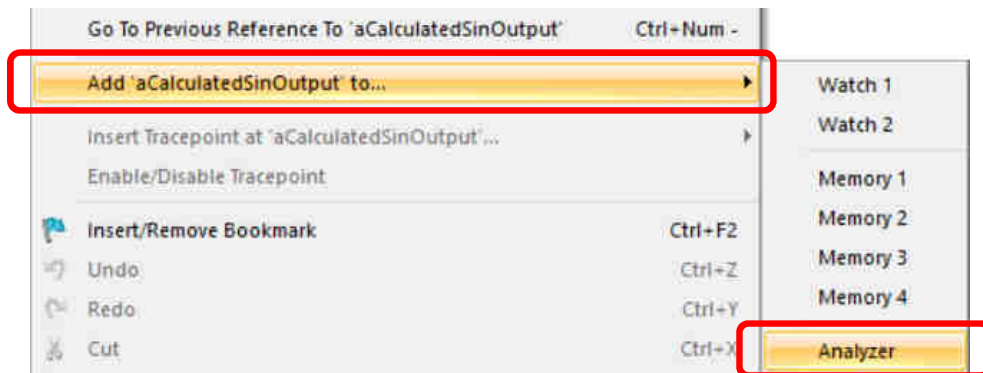
Enable Logic Analyzer

170

- On line 152, Right mouse click on “aCalculatedSinOutput” variable

```
151  /* Logic Analyzer variables */  
152  static int32_t aCalculatedSinOutput = 0;  
153  static int32_t aCalculatedCosOutput = 0;  
154  int32_t aResults_ptr = 0;
```

- Move mouse over “Add ‘aCalculatedSinOutput’ to...”
- Click on “Analyzer”



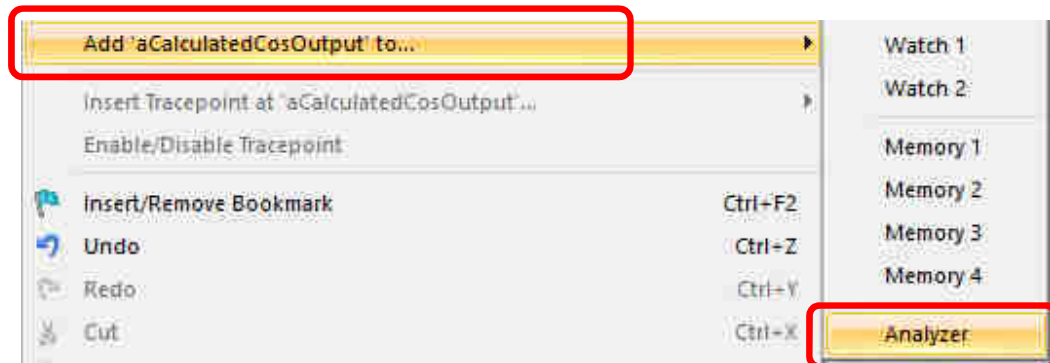
Enable Logic Analyzer

171

- On line 153, Right mouse click on “aCalculatedCosOutput” variable

```
151  /* Logic Analyzer variables */  
152  static int32 t aCalculatedSinOutput = 0;  
153  static int32_t aCalculatedCosOutput = 0;  
154  int32_t aResults_ptr = 0;
```

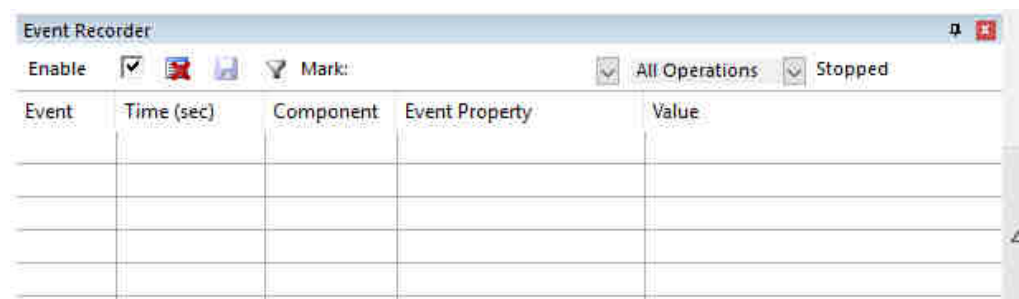
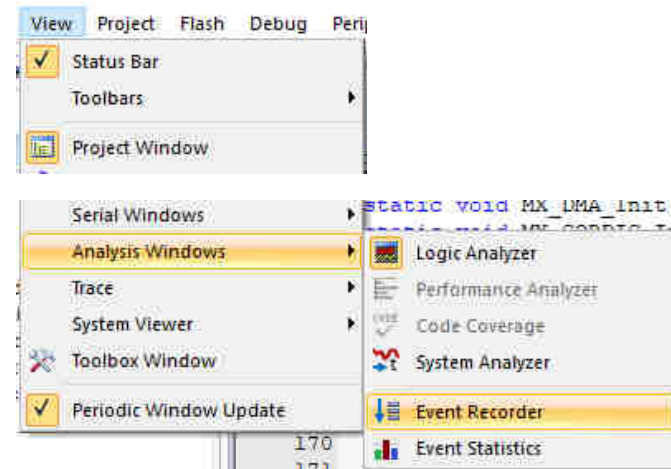
- Move mouse over “Add ‘aCalculatedCosOutput’ to...”
- Click on “Analyzer”



View Event Recorder

172

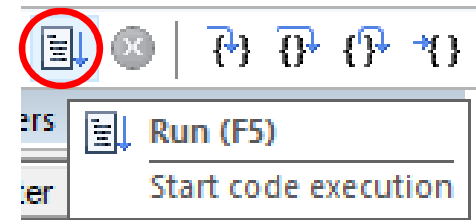
- Go to “View” menu
- Select “Analysis Windows”
- Select “Event Recorder”



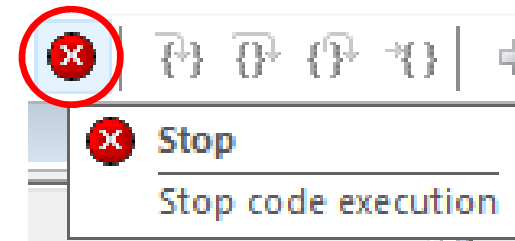
Run the Project

173

- “Run” the project
 - Capture trace data
 - Press “JOYSTICK_SEL” triggers “CORDIC” or “ARM Math library”
 - Green Led blinking
 - Fast blink (CORDIC used)
 - Slow blink (ARM Math Lib. Used)



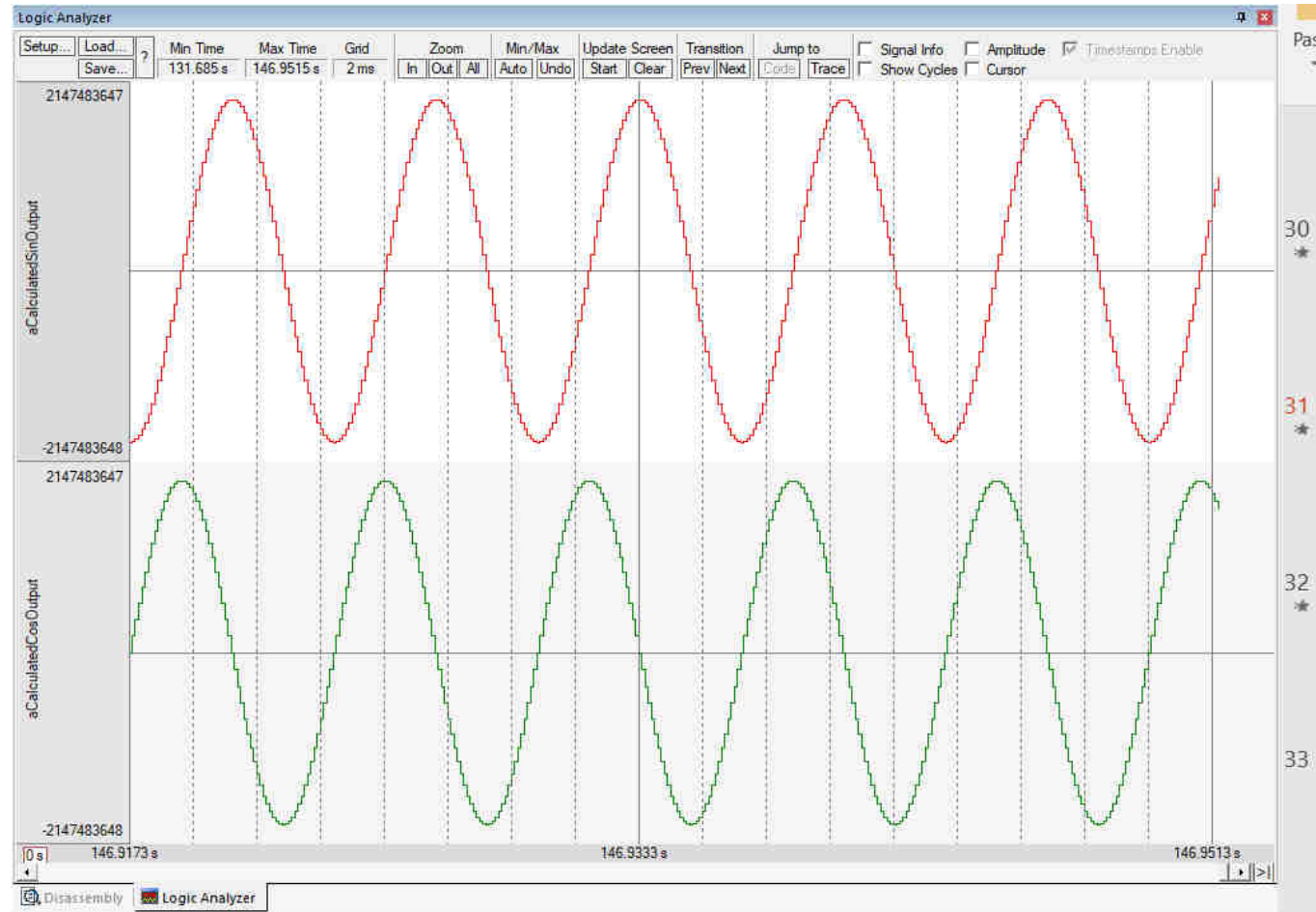
- “Stop” the project execution



Logic Analyzer

174

- CORDIC results are shown in Logic Analyzer



Event Recorder

175

- Events Captured

- Event Number
- Event Time Stamp
- Event Property
 - Id = 0 (CORDIC)
 - Id = 1 (ARM Math Library)

- Event Values

- 1st: Loop Number
- 2nd: Start(0x01) and End(0x02) of array calculation

Event	Time (sec)	Component	Event Property	Value
41819	1.44499562		id=0x0000	0x000031AF,0x00000002
41820	1.44500605		id=0x0000	0x000031B0,0x00000001
41821	1.44501337		id=0x0000	0x000031B0,0x00000002
41822	1.44502308		id=0x0000	0x000031B1,0x00000001
41823	1.44502995		id=0x0000	0x000031B1,0x00000002
41824	1.44503989		id=0x0000	0x000031B2,0x00000001
41825	1.44504672		id=0x0000	0x000031B2,0x00000002
41826	1.44505666		id=0x0000	0x000031B3,0x00000001
41827	1.44506349		id=0x0000	0x000031B3,0x00000002
41828	1.44507343		id=0x0000	0x000031B4,0x00000001
41829	1.44508026		id=0x0000	0x000031B4,0x00000002
41830	1.44509020		id=0x0000	0x000031B5,0x00000001
41831	1.44510062		id=0x0000	0x000031B5,0x00000002
41832	1.55492708		id=0x0000	0x0000057F,0x00000001
41833	1.55493392		id=0x0000	0x0000057F,0x00000002



	Execution Time (measured values)
CORDIC	6.84 μ S
ARM Math Library	42.24 μ S

- Theoretical Calculation:

- 64 Array Size
- 2 functions output: sin and cos
- 24bit precision: 6cycles
- Expected time: 4.52us

$$=(1/170\text{MHz})*64*2*6$$



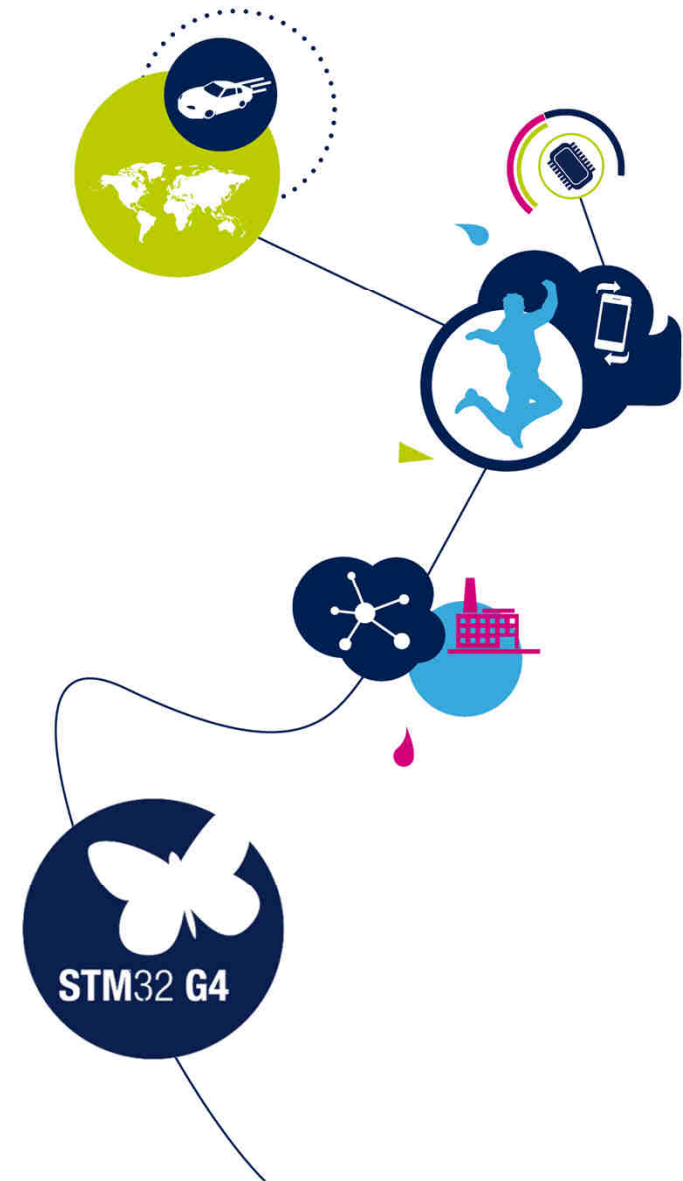
CORDIC: Additional Notes

176

- Use Cube Low Level driver to reduce HAL overhead
- AN5325: Getting started with the CORDIC accelerator using the STM32CubeG4 MCU Package
- Keil Application Note 321: Event Recorder Debugging with STM32G Cortex-M0/M0+

STM32G4 mixed-signal MCU workshop

Filter Math Accelerator (FMAC) Overview





Filter Math Accelerator

178

G4 FMAC

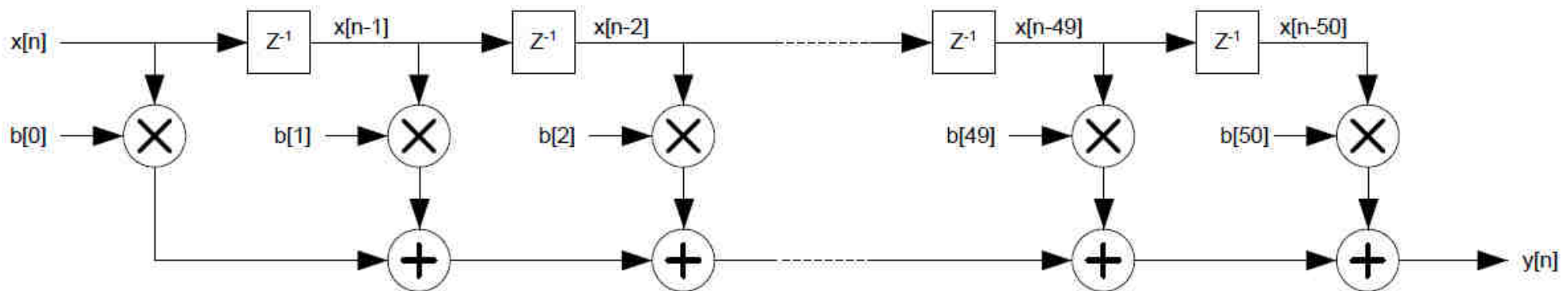
- Multiplier (16x16)
- Accumulator (24bit + 2bit)
- Address generation logic
- Input and Output Circular Buffers
- FIR & IIR filters possible
- Dot product, convolution, correlation
- Does not stall the CPU
- 256 x 16bit local memory
- Built-in sequencer
- Runs at the System Clock speed

Internally connected to:

- DMA
- AHB slave interface
- NVIC

G4 Adaptive FIR Filter

Figure 7. 51-tap FIR filter



- FIR filter is a convolution of the signal with filter coefficients
- The number of coefficients determines the minimum notch size
- Calculating N coefficients requires $N*N$ operations



G4 Adaptive FIR Filter - Quantities

- Filter Coefficients will probably be in floating point format
- Coefficient conversion to 16 bit fixed point is necessary
- Fixed point format is Q1.15
 - Bits [0..14] represent fractional digits
 - Bit 15 is a sign bit
 - Range = -1.0 (0x8000) to +0.999969482421875 (0x7FFF)
- Proper design will keep the absolute value coefficients ≤ 1
 - If not, all coefficients will need to be divided by the greatest coefficient
- To convert floating to fixed point, the Floating Point Unit is used
 - FPU takes 8 cycles per conversion
 - 51 coefficients take ~408 clock cycles



Filter Math Accelerator

181

Lab steps 1 of 6

This Hands On Lab illustrates noise cancellation using an adaptive FIR filter.

The experiment will be run on two input frames:

- a gaussian noise with two tone interferers.
- a gaussian noise with three tone interferers.

Create individual coefficient sets.

The result is a "whitened" noise.

Figure 1. Gaussian noise with two-tone interferer

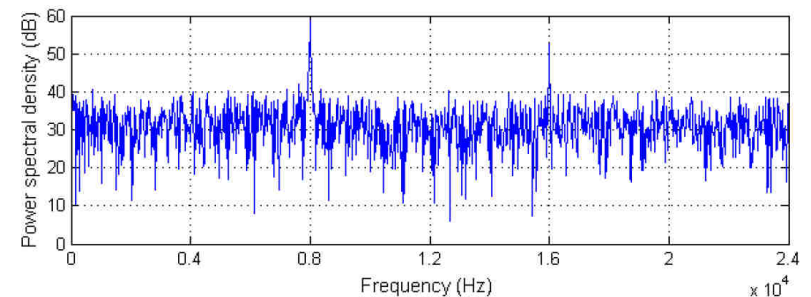
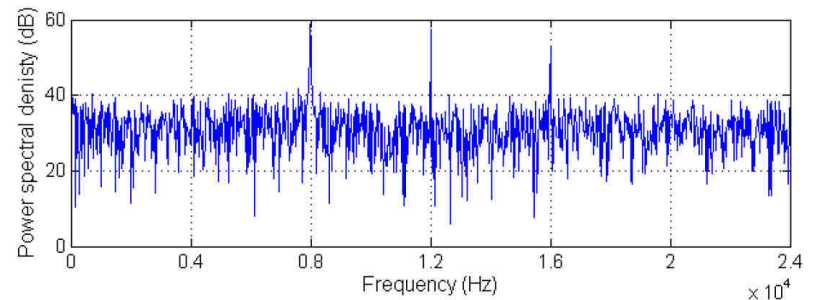


Figure 4. Gaussian noise with three-tone interferer





Filter Math Accelerator

182

Lab steps 2 of 6

- The FMAC peripheral will be configured to perform an adaptive FIR filter.
- There will be 51 feed-forward taps.
- The gain is equal to 0.
- The input and output thresholds are set to 1.
- The clipping feature can be enabled or disabled thanks to the compilation

switch CLIP_ENABLED in main.h.



Filter Math Accelerator

183

Lab steps 3 of 6

The FMAC peripheral is making the best use of its 256 x 16-bit internal memory.

The local memory is divided into three buffers:

- 0 to 50: coefficient buffer;
- 51 to 150: input buffer;
- 151 to 250: output buffer;
- 251 to 255: unused.



Filter Math Accelerator

184

Lab steps 4 of 6

- DMA mode is used to transfer input data from memory TO the FMAC peripheral.
- DMA mode is used to transfer output data FROM FMAC peripheral to memory.
- The coefficient buffer contains the vector of feed-forward taps
- This is named the B buffer
 - (size comprised in [2:127]; here, size $N = 51$ elements).
- This buffer is initialized during the configuration step in DMA mode.



Filter Math Accelerator

185

Lab steps 5 of 6

The size of the input buffer is the sum of:

- the size of the coefficient vector B ($N = 51$ elements);
- the size of the additional space D1 needed for throughput improvement (minimum: 0; here, $D1 = 49$ elements).

The input buffer is filled during the preload step in DMA mode; new values are added in DMA mode once the preloaded ones have been consumed.

The size of the output buffer is the one of the additional space D2 needed for a better throughput (minimum: 1; here, $D2 = 100$ elements).

The output buffer is read in DMA mode.



Filter Math Accelerator

186

Lab steps 6 of 6

STM32 board LEDs are used to monitor the example status:

- LED4 is ON when correct FMAC FIR results have been calculated.
- LED5 is blinking (1 second period) if an error is detected in the FIR filter results or if there is an initialization or configuration error.

The filtered output is displayed in the Terminal I/O if the compilation switch `PRINT_OUTPUT` is enabled in `main.h`.

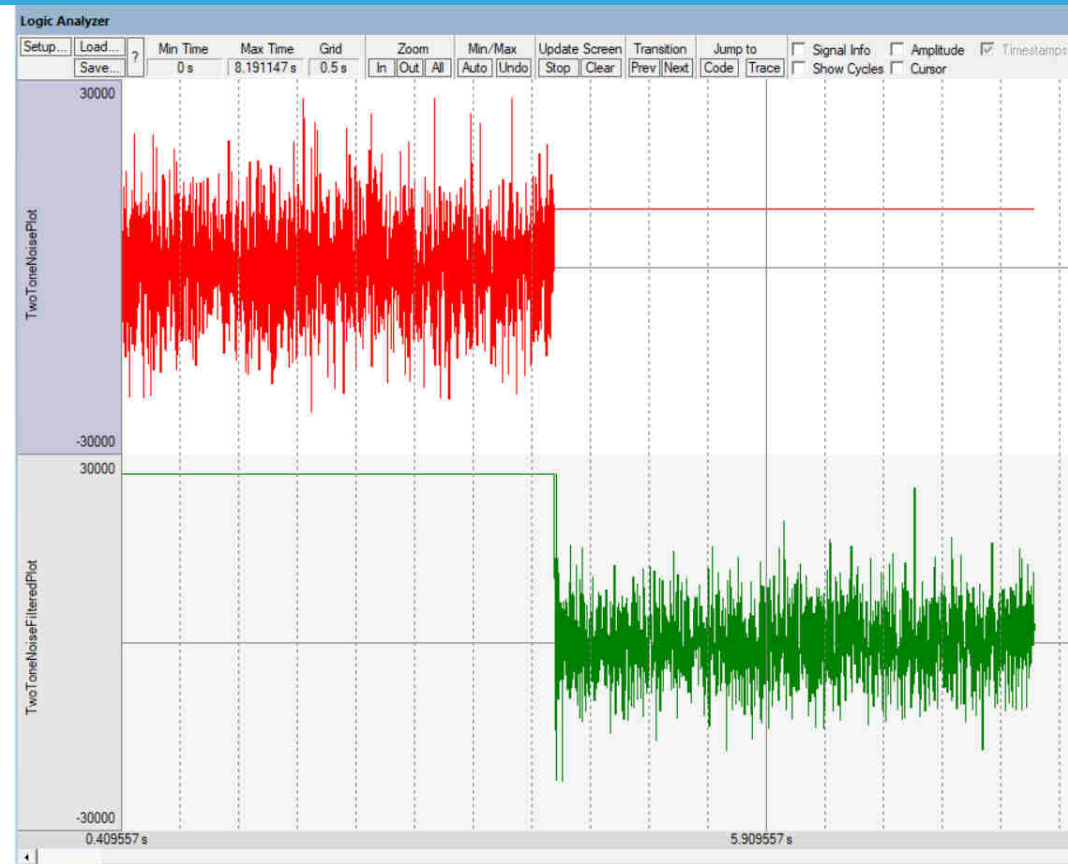


Filter Math Accelerator

187

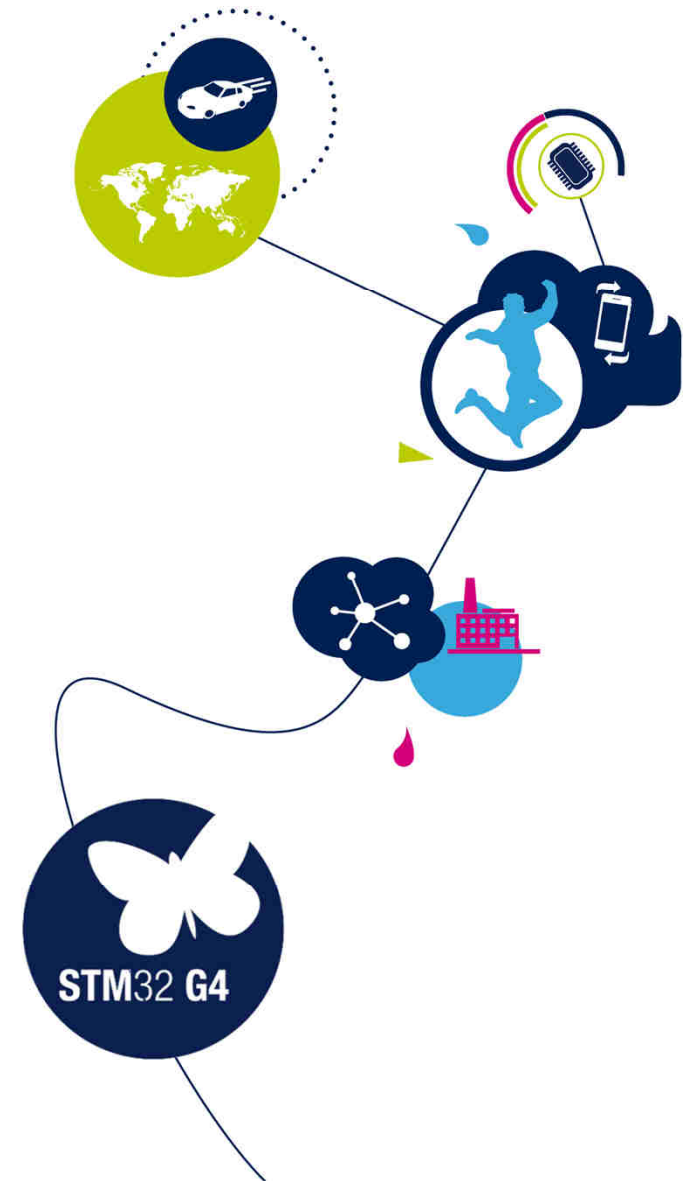
How to view it

- Red shows 2 tone noise
- Green the filtered signal



STM32G4 mixed-signal MCU workshop

*Hands On:
Dual Bank Flash
Live Firmware Upgrade*





Dual Bank flash

189

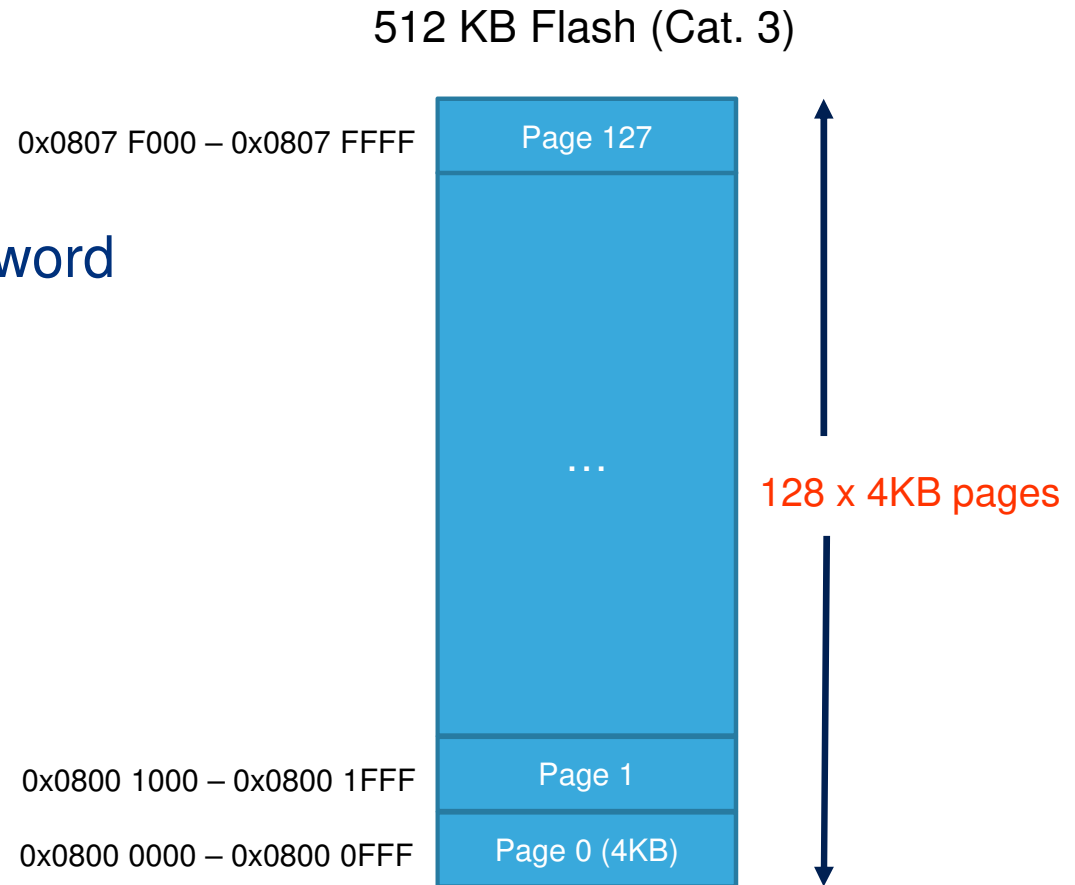
Objective:

- To show the application running from one bank and writing to the second bank(Read-While-Write). Demonstrate the live upgrade possibility.
- Using a Terminal (TeraTerm) to send updated application binary file via the Virtual Comm Port.

512 KB die Flash Organization

190

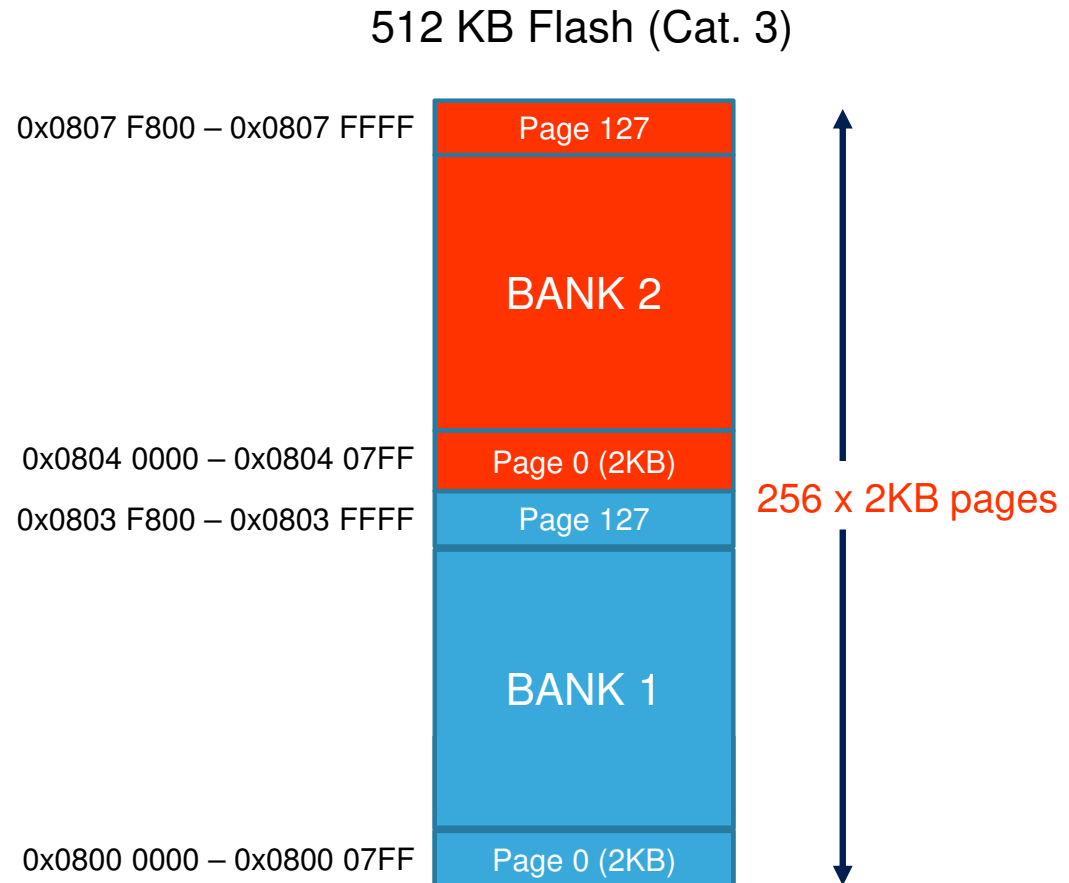
- DBANK = 0 (Single Bank)
 - 128 bits read access
- ECC: 8 bits per 64 bit double-word
- 4 Write protection areas
- Page and Mass erase



512 KB die Flash Organization

191

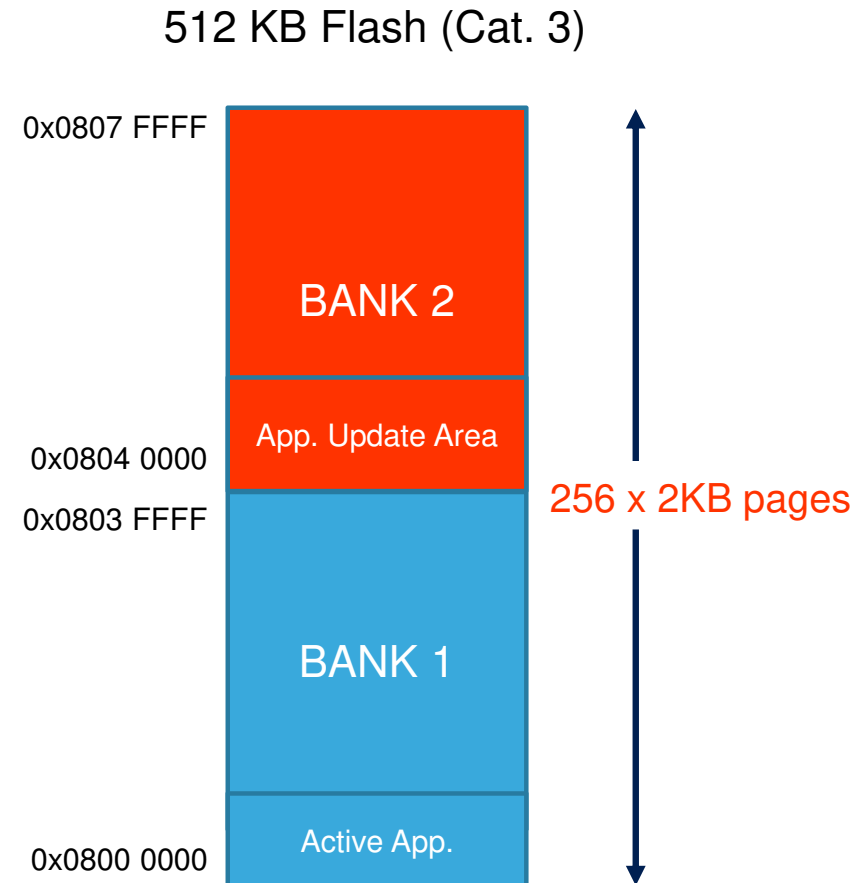
- DBANK = 1 (Dual Bank - Default)
 - 64 bits read access
- Read-While-Write (RWW)
- ECC: 8 bits per 64 bit double-word
- 2 Write protection areas per bank
- Page, Bank and Mass erase



512 KB die Flash Organization

192

- Active application executing from Bank 1
- Using serial port to receive update
 - Temporary storage in RAM
- Write to BANK 2 area
- Switch application execution to BANK 2





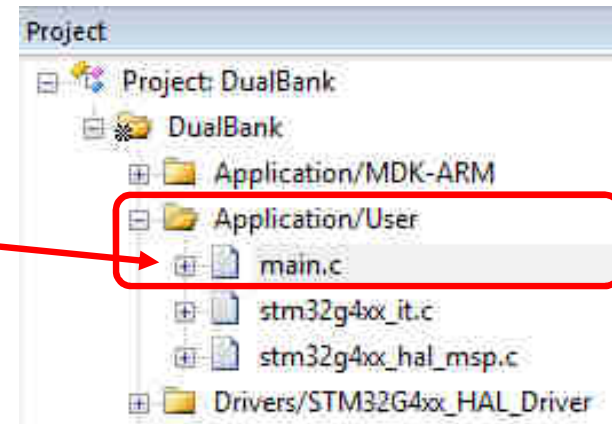
Hands-On: Dual Bank flash

193

- Open the DualBank project in Keil MDK-ARM folder
 - DualBank\MDK-ARM\DualBank.uvprojx



- Expand “Application/User” folder
- Open “main.c” file
 - Initialize the Clocks, GPIO and Peripherals
 - Main while loop (State Machine)
 - Handles BANK 2 page erase
 - Handles BANK 2 page write
 - Jump and Execute in Bank 2





- Timer 2 IRQ Callback(Line 481)
 - 10 Hz period
 - LD_ptr: Tracks LED to toggle
- GPIO EXTI Callback (Line 509)
 - Detects rising edge of JOYSTICK_SEL button
- UART Rx complete Callback (Line 524)
 - Signals that the USART Rx DMA has completed

IRQ Callback functions

194

```
481 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
482 {
483     switch ( LD_ptr )
484     {
```

```
509 void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
510 {
511     if ( ( GPIO_Pin == JOYSTICK_SEL_Pin ) && ( UpdateState == 0 ) )
```

```
524 void HAL_UART_RxCpltCallback(UART_HandleTypeDef *huart)
525 {
526     rx_complete = 1;
527 }
```



Peripheral configuration

195

- **JOYSTICK_SEL**: EXTI13
 - PC13
- **LD2-5**: GPIO_Output
 - PA15, PB1, PB5 and PB7
- **ST-LINK VCP**: USART3
 - PC10 and PC13
- **Serial Wire Debug with Trace (SWO)**:
 - PA13, PA14 and PB3

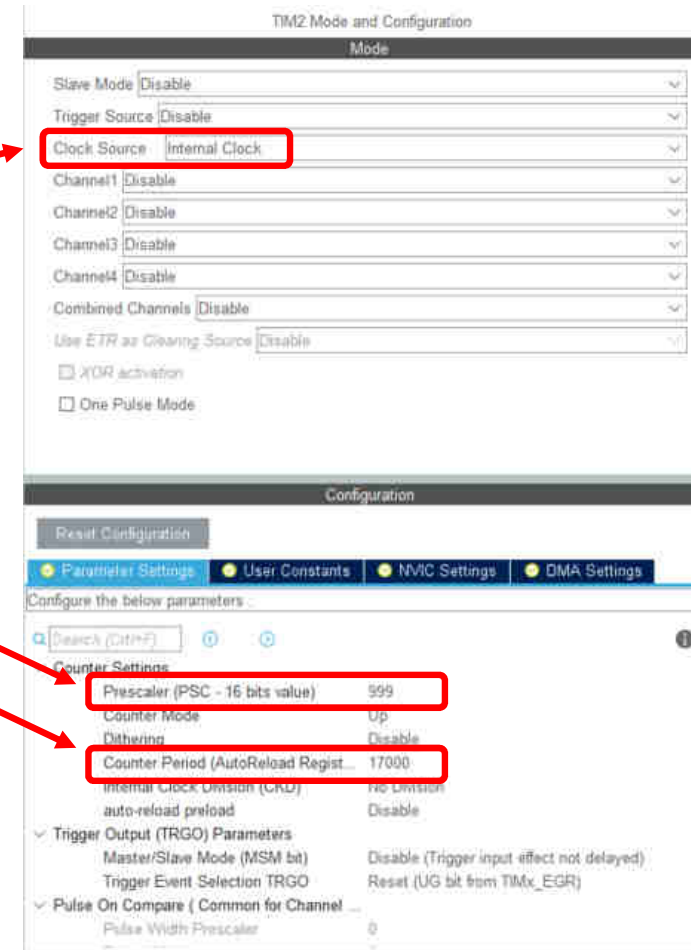
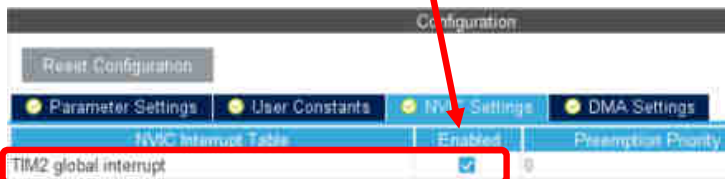




Peripheral configuration

196

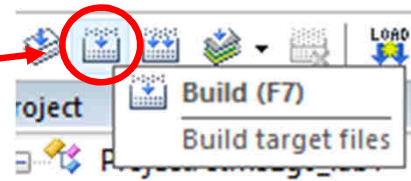
- **Timer 2:** LED blink rate timer
- Internal Clock Source = 170 MHz
- Prescaler = 999
- Counter Period = 17000
 - 10 Hz timebase interrupt
- TIM2 global interrupt Enabled



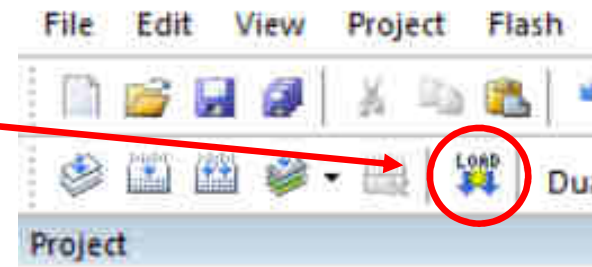
Build the Project

197

- “Build” project



- “LOAD” project to flash



- Press the “RESET” button on the Discovery
 - 4 LEDs blinking in rotating pattern.

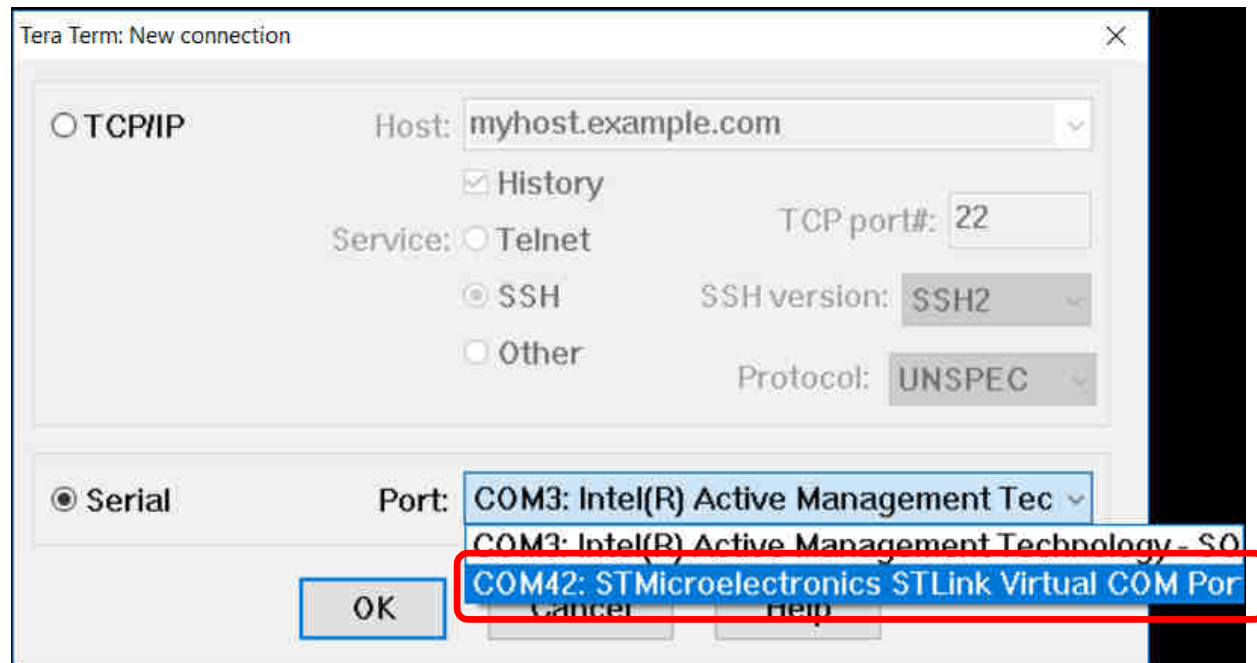
Hands On: Dual Bank

198

- Launch “TeraTerm”



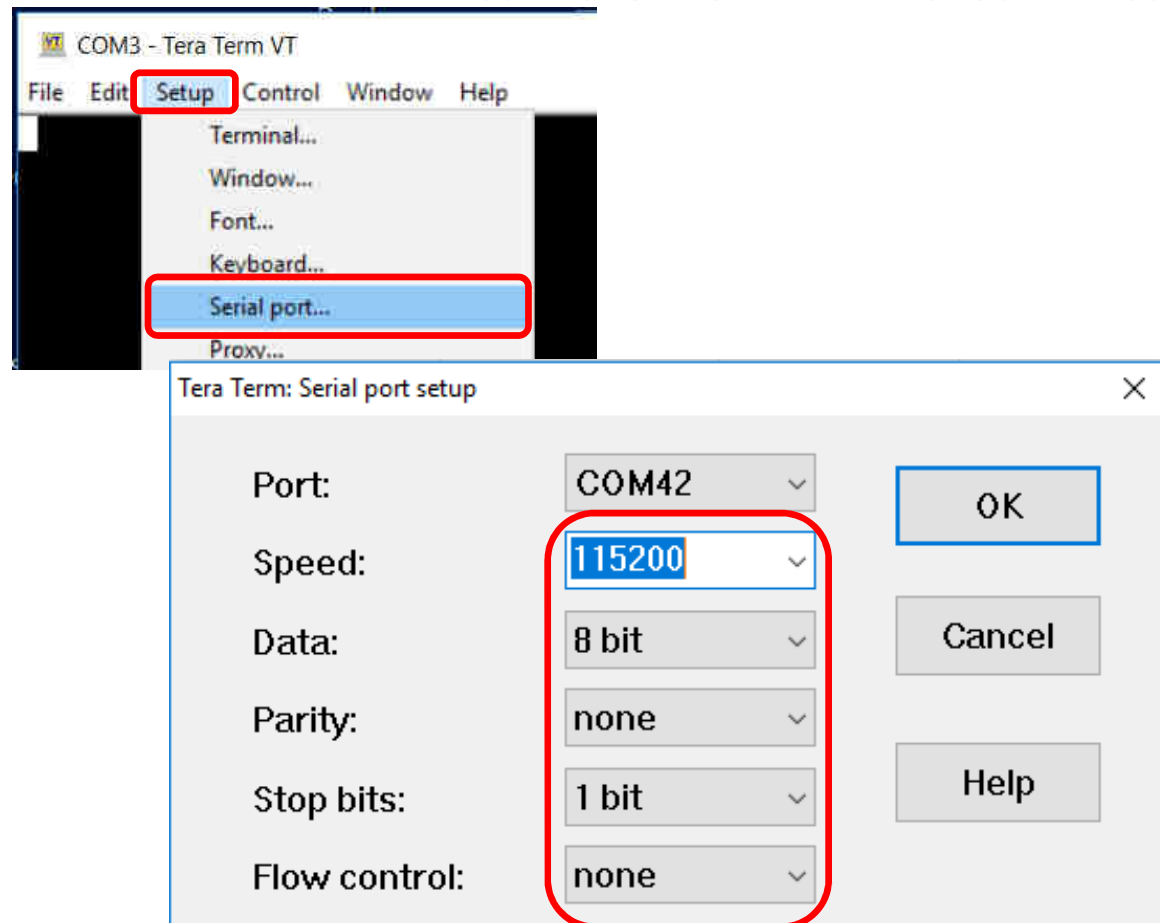
- Open “New Connection”
 - Click “Serial” button
 - Change Port to “STLink Virtual COM Port”
 - Click “OK”



Hands On: Dual Bank

199

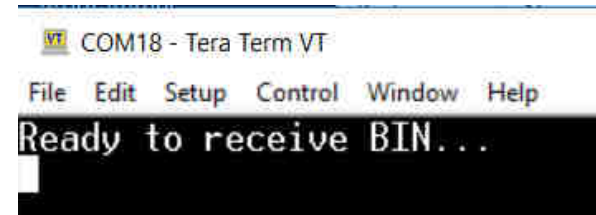
- Verify “Serial port..”
 - Baud = 115200
 - Data = 8 bit
 - Parity = none
 - Stop = 1 bit
 - Flow control = none



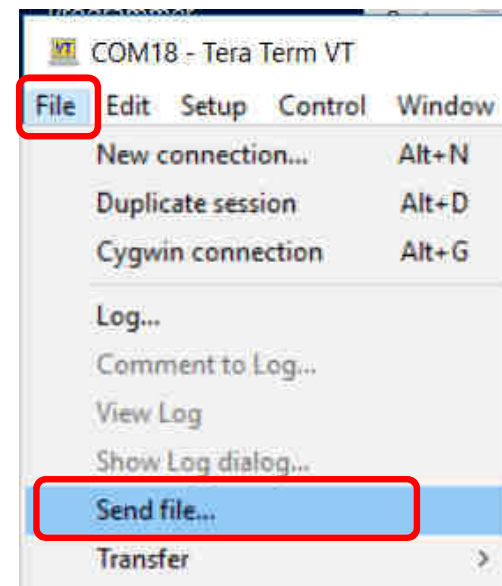
Hands On: Dual Bank

200

- Press “JOYSTICK_SEL” on Discovery



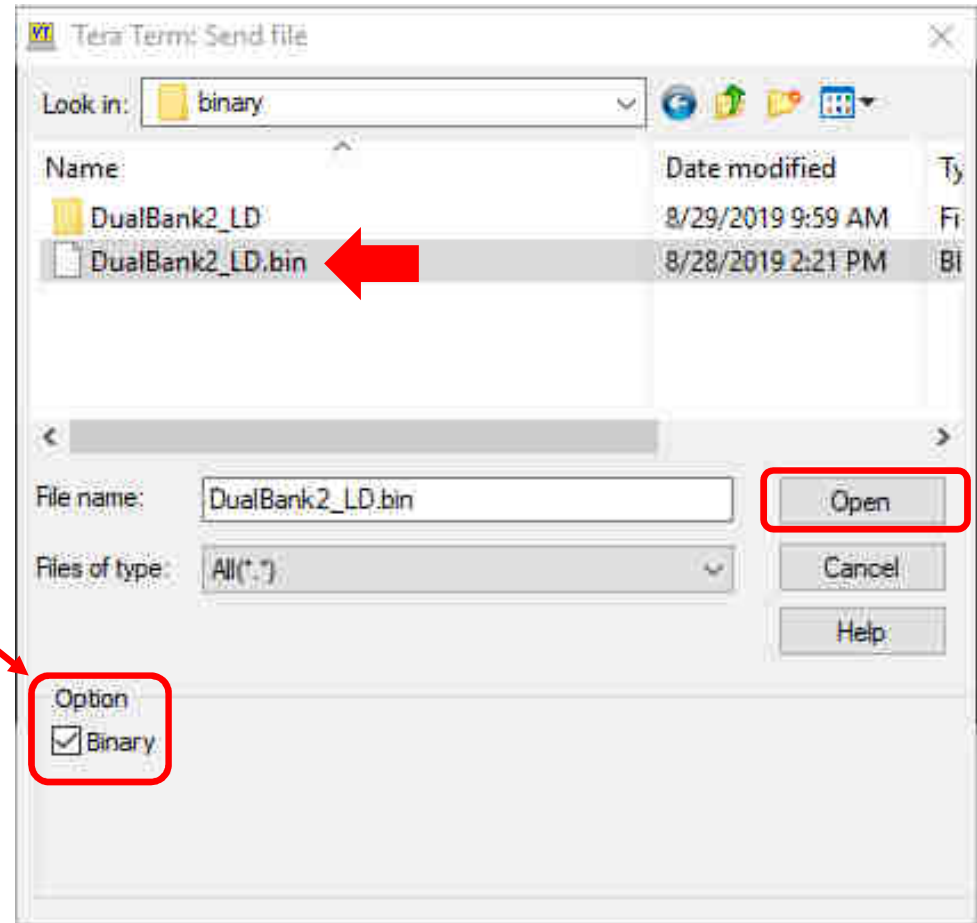
- Send binary file in TeraTerm
 - File > Send file...



Hands On: Dual Bank

201

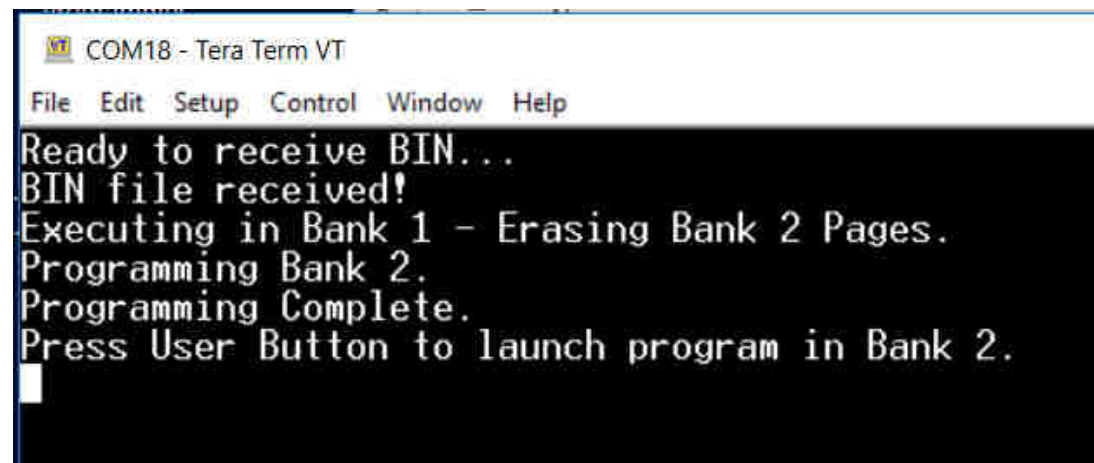
- Select “DualBank2_LD.bin” file
 - DualBank\binary folder
- Check “Binary” Option
- Click “Open”



Hands On: Dual Bank

202

- TeraTerm will display programming status



A screenshot of a TeraTerm VT window titled 'COM18 - Tera Term VT'. The window has a menu bar with 'File', 'Edit', 'Setup', 'Control', 'Window', and 'Help'. The main text area shows the following status messages: 'Ready to receive BIN...', 'BIN file received!', 'Executing in Bank 1 - Erasing Bank 2 Pages.', 'Programming Bank 2.', 'Programming Complete.', and 'Press User Button to launch program in Bank 2.' A small white cursor is visible at the end of the last line.

```
COM18 - Tera Term VT
File Edit Setup Control Window Help
Ready to receive BIN...
BIN file received!
Executing in Bank 1 - Erasing Bank 2 Pages.
Programming Bank 2.
Programming Complete.
Press User Button to launch program in Bank 2.
█
```

- Press “User Button” to launch program in Bank 2
 - LEDs blink pattern has changed.



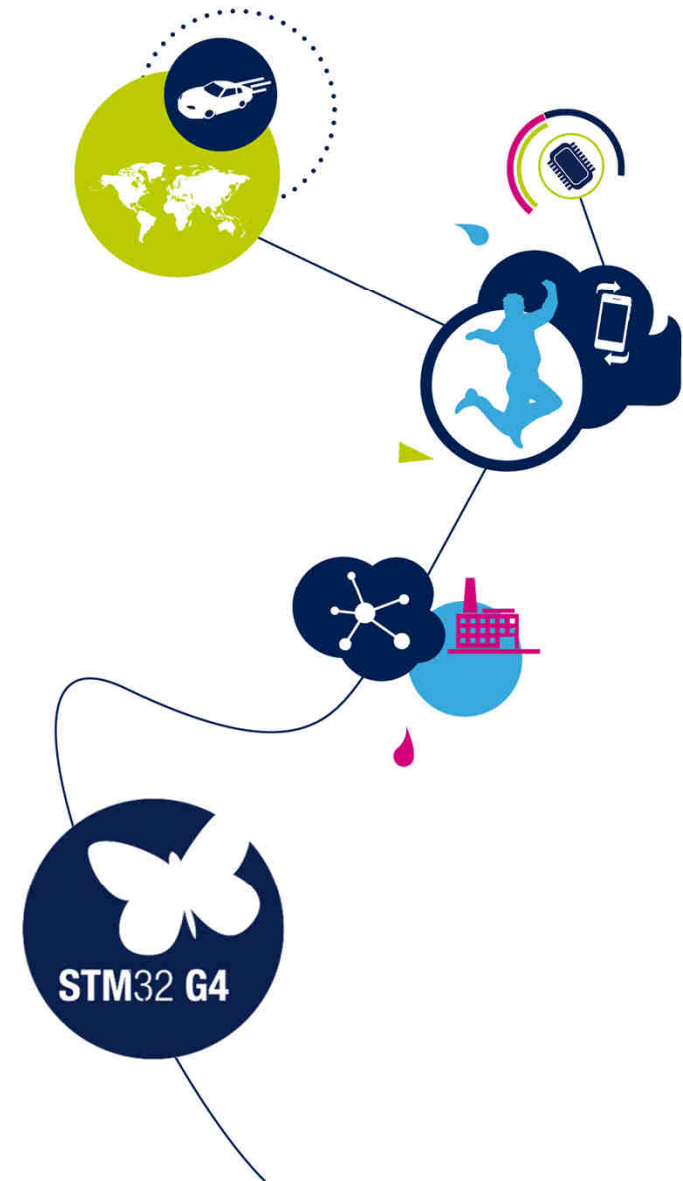
Dual Bank: Additional Notes

203

- Push board “RESET” button to switch execution back to Bank 1
 - To select boot from Bank 2, set the BFB2 bit in the user option bytes
- Reference Manual RM0440: STM32G4 Series advanced ARM-based 32-bit MCUs
 - Section 3: Embedded Flash memory for category 3 devices
- AN2606: STM32 microcontroller system memory boot mode

STM32G4 mixed-signal MCU workshop

Wrap Up



STM32G4 Workshop Related Survey

205

Use your phone to scan the QR code or type the link into your browser.



<https://www.surveymonkey.com/r/PC5DHBM>



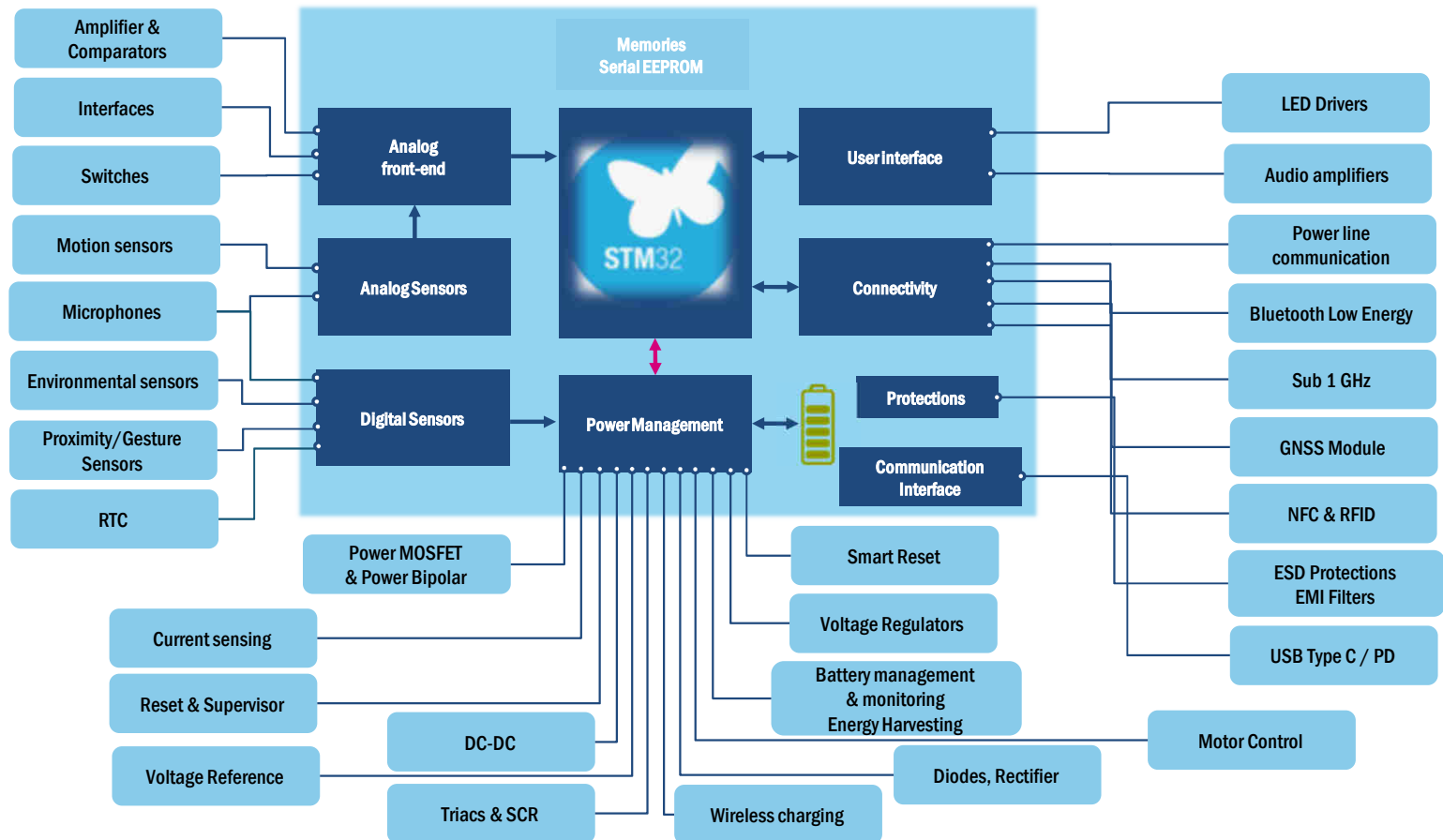
We Greatly Value Your Feedback - Thank you!

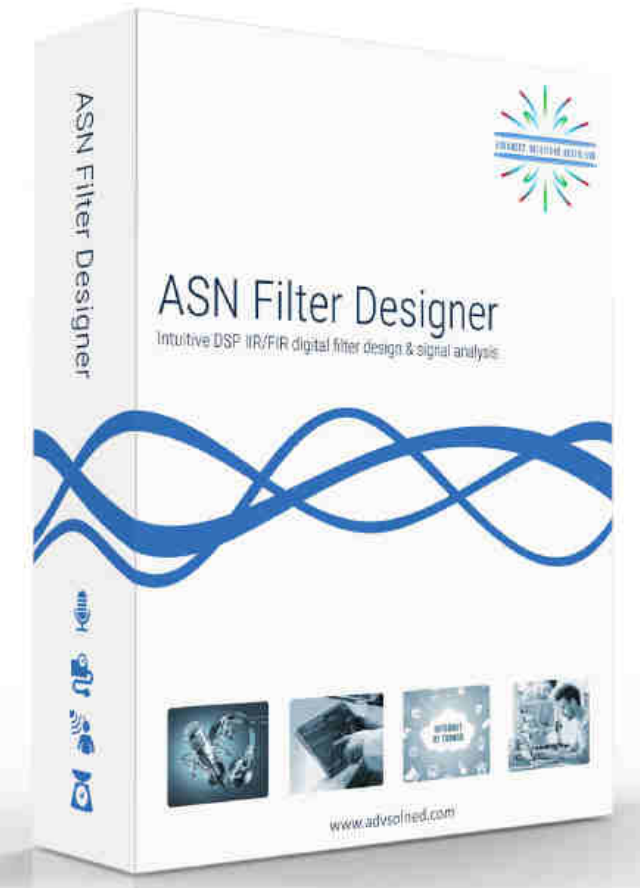
STMicroelectronics: one of the broadest Standard Product Portfolios

The largest offerings of companion products for every **STM32** Microcontrollers



- Protection devices
- Power bipolar transistors
- Power MOSFETs
- Amplifiers and Comparators
- Linear voltage regulators
- EEPROM
- Rectifiers
- SCRs and TRIACs
- Interface ICs
- Voltage references
- Reset and supervisor ICs





ASN Filter Designer

207

Powerful DSP experimentation platform

Design and fine-tune your digital filter within minutes, and then test it in real-time with a variety of test signals (including your own datasets) via the tool's built-in signal analyser.

Real-time design feedback

No need to explicitly define technical specifications, just sketch your requirements and let the tool automatically fill in the exact specifications for you!

Live math scripting

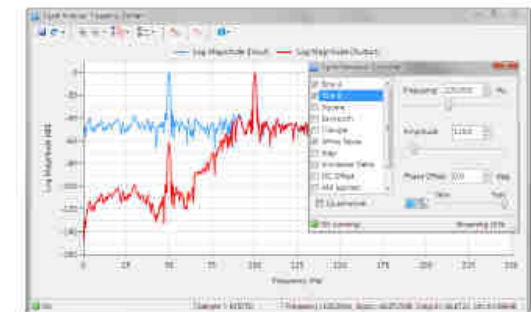
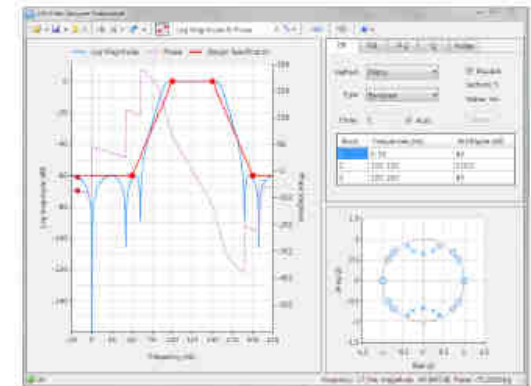
Interactively experiment with specialised transfer functions, pole-zero pairs and other mathematical functions with ASN FilterScript.

Automatic documentation

Get a detailed specification report of all designed filter blocks for official project documentation, which can be pasted directly into Microsoft Word/Powerpoint.

Deploy to Arm CMSIS-DSP, C/C++, Python, Matlab and more

Easily integrate your designed filter blocks with your other algorithms in Arm CMSIS-DSP, C/C++, Python, Matlab, Octave, Scilab or Labview using provided royalty free software development frameworks.



Download a free demo

http://www.advsolned.com/asn_filter_designer/



Biricha Digital Power Design Tool

208



Digital Power workshops with BIRICHA

351

How to implement Digital Power Supplies (PSU) and Power Factor Correction (PFC)

STM32 PSU/PFC design
Step-in Digital Power technology



Biricha Digital Power:
World leading expertise and training in Digital Power

Workshop based on STM32F334 and STM32G474
Nucleo and its dedicated Digital Power expansion
board

STMicroelectronics brings industry leading ST MCUs
together with Biricha's tools and training





Appendix – Reference Documents

209

Datasheet and Reference Manual Links

Datasheets

<u>STM32G431xx</u>	STM32G431x6 STM32G431x8 STM32G431xB
<u>STM32G441xB</u>	STM32G441xB
<u>STM32G473xx</u>	STM32G473xB STM32G473xC STM32G473xE
<u>STM32G474xx</u>	STM32G474xB STM32G474xC STM32G474xE
<u>STM32G483xE</u>	STM32G483xE
<u>STM32G484xE</u>	STM32G484xE

Reference Manual

<u>RM0440</u>	STM32G4 Series Reference Manual
-------------------------------	---------------------------------



Appendix – Reference Documents

210

Application Note Links

<u>AN2834</u>	How to get the best ADC accuracy
<u>AN4013</u>	STM32 Cross series timer overview
<u>AN4232</u>	Getting started with analog comparators
<u>AN4296</u>	Using CCM SRAM with IAR, Keil, and GNU based tool chains
<u>AN4539</u>	HRTIM Cookbook
<u>AN4899</u>	STM32 GPIO configuration for hardware settings and low power consumption
<u>AN4989</u>	STM32 debug toolbox
<u>AN5305</u>	Digital filter implementation with the FMAC
<u>AN5306</u>	Operational Amplifier usage in STM32G4
<u>AN5310</u>	Guideline for using analog features of the G4 vs. the F3
<u>AN5315</u>	STM32Cube firmware examples for STM32G4
<u>AN5325</u>	Getting started with the CORDIC accelerator
<u>AN5345</u>	High-brightness RGB LED control using the B-G474E-DPOW1 Discovery kit
<u>AN5346</u>	STM32G4 use tips and recommendations



Appendix – Reference Documents

211

User Manual Links

<u>UM1721</u>	Developing applications on STM32Cube with FatFS
<u>UM1722</u>	Developing applications on STM32Cube with RTOS
<u>UM2454</u>	STM32G4 Series Safety Manual
<u>UM2492</u>	Getting started with STM32CubeG4 for STM32G4 Series
<u>UM2552</u>	USB Power Delivery
<u>UM2570</u>	Description of STM32G4 HAL and low-layer drivers
<u>UM2573</u>	STM32CubeG4 Nucleo demonstration firmware
<u>UM2577</u>	Discovery kit with STM32G474RE MCU



STM32G4 Series – Take Away

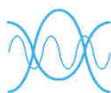
212

Analog-rich MCUs for mixed-signal applications



Performance

170MHz Cortex-M4 coupled with 3x accelerators

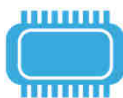


Rich and Advanced Integrated Analog

ADC, DAC, Op-Amp, Comp.



Safety and security focus



Large portfolio available from NOW!

32..512KB Flash memory

32..128-pin packages

Releasing Your Creativity

213

