XMC1100 Boot Kit Getting Started





	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	- Boot Modes
5	- DAVE TM
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	- Boot Modes
5	- DAVE TM
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



1	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	- Boot Modes
5	
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



1	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	– Boot Modes
5	- DAVE TM
6	Getting Started
7	- Example - Blinky based on XMC Lib
8	 – Example – Blinky based on DAVE[™] APPs



1	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	– Boot Modes
5	- DAVE TM
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



1	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	- Boot Modes
5	- DAVE TM
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



1	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	– Boot Modes
5	- DAVE TM
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



Kit Overview
Hardware Overview
Tooling Overview
- Boot Modes
- DAVE TM
Getting Started
 Example – Blinky based on XMC Lib
 Example – Blinky based on DAVE[™] APPs



	Kit Overview
2	Hardware Overview
3	Tooling Overview
4	- Boot Modes
5	- DAVE TM
6	Getting Started
7	 Example – Blinky based on XMC Lib
8	 Example – Blinky based on DAVE[™] APPs



Kit Overview (1/2)

- > XMC1100 Boot Kit
 - Consists of an XMC1100 CPU Card
 - Compatible with Arduino[™] Shields (*http://shieldlist.org*)





Arduino[™] Shield



Kit Overview (2/2)

> XMC1100 CPU Card for Arduino[™]



Home

Hardware Overview



- Connect XMC1100 CPU Card to PC via USB cable >
- CPU Card is powered up (as indicated by LED on the card) >





Note: Supported Application Card may be additionally connected to > the CPU card

Home

Tooling Overview – Boot Modes



- > Boot Modes available
 - UART Bootstrap-Loader Mode
 - User Mode (Halt After Reset)
 - User Mode (Debug) Default Mode of device on Boot Kit
 - User Mode (Productive)
- > Boot Modes can be configured via:
 - DAVE[™]
 - Download DAVE[™]
 - DAVE[™] v4.1.2 download
 - MemTool
 - Download MemTool

MemTool v4.65.exe download

 For more information on how to configure the BMI value, please refer to the XMC1000 Tooling Guide.



Tooling Overview – DAVETM (1/5)



> DAVE[™] download package is available at:

http://infineon-community.com/LP=400



Please register to download DAVE™ version 4 and DAVE™ SDK version 4.

DAVE TM version 4 and DAVE SDK version 4 is now available as productive version. The current versions are: DAVE TM v4.1.2 and DAVE TM SDK v4.1.2.

After registration you will receive a confirmation email with a link to the download-page. With a click on the link you can download a zip file that contains a setup exe-file and a PDF-file with installation instructions. Please check the JUNK or SPAM folder of your mail server if you don't receive a confirmation email.

First Name*		
Last Name*		
Email Address*		
Country*	Please select	~
Company*		
Business Phone		
Target Application	please select	~



Tooling Overview – DAVE[™] (2/5)

- > After registration, download and unzip the installer package
- Run DAVE-4.1.2-Setup.exe to install
 DAVE[™] IDE and SEGGER J-Link drivers
- > Open DAVE[™]



- > Update DAVE[™] and DAVE[™] libraries
 - Help \rightarrow Check for Updates
 - Help \rightarrow Check for DAVE APP Updates







Tooling Overview – DAVE[™] (3/5)



- > Install DAVETM APPs libraries and Device Description
 - Help → Install DAVE APP/Example/Device Library

Window	Help		
(÷ • •		Welcome	
	?	Help Contents	
	82	Search	
		Dynamic Help	
		Key Assist	Ctrl+Shift+L
		Tips and Tricks	
		DAVE [™] Forum	
		DAVE [™] News	
		Send Feedback Mail	
		Cheat Sheets	
	Install DAVE APP/Example/Device Library]
		Check for DAVE APP Updates	
		Uninstall DAVE APP/Example/Device Library	
	ay -	Check for Updates	
	₽.	Install New Software	
	۲	Installation Details	
	-	About DAVE™	

 Note: You may skip the above step if you are not using DAVE[™] APPs

Tooling Overview – DAVE[™] (4/5)



> Select DAVE Apps Library Manager in the drop-down menu

😺 Library Manager Wizard		
Download Libraries Page This wizard page helps in downloading the libraries o	of type example projects or apps library	
Dave Site Work with : DAVE Apps Library Manager	Find more library by working with the <u>Library Update S</u>	▼ Add) ites preferences
Libraries Enter the keywords to filter :		
Name DEVICE_PACK	Version	Path
Library_DAVEDeviceDescriptions DAVE APPs		
•		4
Select All Deselect All		
Description		
Filters		
Show only latest version		
?	< Back Next > Fini	sh Cancel

- Select DEVICE_PACK, Library_DAVEDeviceDescriptions (XMC1100 Device) and DAVE APPs
 - DEVICE_PACK
 - V Library_DAVEDeviceDescriptions
 - DAVE APPs

Tooling Overview – DAVE[™] (5/5)



> Accept terms of the license agreements and click Finish



> DAVE[™] APPs libraries and Device Description are installed

Home

Getting Started – Example – Blinky based on XMC Lib (1/6)



1. Open DAVE[™]



- Create a new "Simple Main" project:
 - File →New →DAVE Project
 - Enter project name e.g. "XMC1100_Blinky"
 - Select "Simple Main Project" as Project Type



3. Select the device accordingly

😺 New DAVE Project	
Microcontroller Selection Page	
Select the microcontroller for which the project has to be created	
▲ 🖉 XMC1100 Series	
XMC1100-Q040x0064	
XMC1100-Q040x0032	
XMC1100-Q040x0016	E
XMC1100-T038x0064	
XMC1100-T038x0032	
XMC1100-T038x0016	
XMC1100-Q024x0064	
XMC1100-Q024x0032	
XMC1100-Q024x0016	
XMC1100-Q024x0008	-
Microcontroller Features	
Package= TSSOP38	
ROM= 64 KB Flash	â
RAM= 16 KB RAM	=
InOut= 34 digital I/O	
ADC= 12 ADC Channels, 12-bit, Analog-to-Digital Converter	-
Linker Option	
Remove unused sections	
Runtime Library	
Library Newlib-nano 👻	
Add floating point support for printf	
Add floating point support for scanf	
? < Back Next > Finish	Cancel
C Dack INEXL Plinish	Cancer

Getting Started – Example – Blinky based on XMC Lib (2/6)



- > For this project, we will use
 - System clock frequency of 8MHz
 - LED on Port pin 0.5
 - System timer, SysTick, as the time base for interrupt
- > Next, we will show you how to
 - 1. Set up the System or Main Clock (MCLK)
 - 2. Configure Port pin
 - 3. Configure SysTick and define its exception service routine

Getting Started – Example – Blinky based on XMC Lib (3/6)



- 1. Set up System or Main Clock (MCLK)
 - Include the header files required for MCLK and GPIO configuration

```
#include "xmc_gpio.h"
#include "xmc_scu.h"
```

 MCLK configured via **IDIV** and **FDIV** bit fields in XMC_SCU_CLOCK_CONFIG data structure

```
XMC_SCU_CLOCK_CONFIG_t clock_config =
{
    .pclk_src = XMC_SCU_CLOCK_PCLKSRC_DOUBLE_MCLK, /*PCLK = 2*MCLK*/
    .rtc_src = XMC_SCU_CLOCK_RTCCLKSRC_DCO2,
    .fdiv = 0, /**< Fractional divider */
    .idiv = 4, /**MCLK = 8MHz */
};</pre>
```

- Initializes clock generators and clock tree in Main.c

```
int main(void)
{
   /* Ensure clock frequency is set at 8MHz (MCLK) */
    XMC_SCU_CLOCK_Init(&clock_config);
```

Getting Started – Example – Blinky based on XMC Lib (4/6)



- 2. Configure Port pin
 - GPIO to toggle the LED is configured via mode and output_level of XMC_GPIOC_CONFIG structure.

```
XMC_GPIO_CONFIG_t gpio_output_config =
{
    .mode = XMC_GPIO_MODE_OUTPUT_PUSH_PULL,
    .output_level = XMC_GPIO_OUTPUT_LEVEL_HIGH,
};
```

- Initializes port pin P0.5 as general purpose output pin in Main.c

```
/* Initialise P0.0 as an output pin */
    XMC_GPI0_Init(LED1, &gpio_output_config);
```

Getting Started – Example – Blinky based on XMC Lib (5/6)



- 3. Configure SysTick and define its exception service routine
 - SysTick exception handler is defined in **startup_XMC1100.s**

```
/* ----- */
.globl SysTick_Veneer
SysTick_Veneer:
LDR R0, =SysTick_Handler
MOV PC,R0
/* ----- */
```

- Initialize the SysTick in Main.c

```
/* System timer configuration */
   SysTick_Config(SystemCoreClock / TICKS_PER_SECOND);
```

Define the SysTick exception handler routine in Main.c
 void SysTick_Handler(void)

```
{
  static uint32_t ticks = 0;
  ticks++;
  if (ticks == TICKS_WAIT)
  {
    XMC_GPI0_ToggleOutput(LED1);
    ticks = 0;
  }
}
```

Getting Started – Example – Blinky based on XMC Lib (6/6)



- > Build project
 - 1. Click 🔁

- 'Invoking: ARM-GCC Print Size'
 "C:\DAVEv4\DAVE-4.1.2\eclipse\ARM-GCC-49/bin/arm-none-eabitext data bss dec hex filename
 2204 20 1044 3268 cc4 XMC1100_Blinky.elf
 'Finished building: XMC1100 Blinky.siz'
- 2. Wait for Build to finish
- > Download code
 - 1. Click 🏇
 - 2. Switch to Debug perspective E DAVE IDE S DAVE CE S PinMapping The Debug
 - 3. Click 🕩 to run code
- > LED blinks every 0.2s



Home

Getting Started – Example – Blinky based on DAVETM APPs (1/7)



1. Open DAVE[™]



- Create a new "DAVE CE" project:
 - File →New →DAVE Project
 - Enter project name e.g.
 "XMC1100_Blinky_withApps"
 - Select "DAVE CE Project" as Project Type



3. Select the device accordingly

윷 New DAVE Project	
Microcontroller Selection Page	
Select the microcontroller for which the project has to be created	
▲ 🛛 XMC1100 Series	*
XMC1100-Q040x0064	
XMC1100-Q040x0032	
XMC1100-Q040x0016	E
XMC1100-T038x0064	
XMC1100-T038x0032	
XMC1100-T038x0016	
XMC1100-Q024x0064	
XMC1100-Q024x0032	
XMC1100-Q024x0016	
XMC1100-Q024x0008	
XMC1100-T016x0064	T
Microcontroller Features	
Package= TSSOP38 ROM= 64 KB Flash RAM= 16 KB RAM InOut= 34 digital I/O ADC= 12 ADC Channels, 12-bit, Analog-to-Digital Converter	
Linker Option	
Remove unused sections	
Runtime Library	
Library Newlib-nano 👻	
Add floating point support for printf	
Add floating point support for scanf	
- Add housing point support for scalin	
? < Back Next > Finish	Cancel

Getting Started – Example – Blinky based on DAVETM APPs (2/7)



- > For this project, we will use
 - System clock frequency of 8MHz
 - LED on Port pin 0.5
 - System timer as the time base for interrupt
 - Time base of 0.2s
- > Next, we will show you how to
 - 1. Set up the System or Main Clock (MCLK)
 - 2. Configure Port pin
 - Configure System Timer and define its exception service routine

Getting Started – Example – Blinky based on DAVETM APPs (3/7)



- 1. Set up System or Main Clock (MCLK)
 - Click to add new APP
 - Double-click CLOCK_XMC1 APP and close window
 - Open APP configuration editor
 - In APP Dependency view, double-click CLOCK_XMC1

NPP Dependency 🛛	🔒 HW Signal Connectivity
CLOCK_XMC1 CLOCK_XMC1_0	

- Configure APP instance
 - In APP configuration window, set Main clock (MCLK) to 8MHz

💺 Add New APP	
Show hidden categories	Search filter
 System ANALOG JO (4.0.4) CLOCK_XMC1 (4.0.6) CMSIS_DOSP (4.0.2) CMSIS_RTOS_RTX (4) CPU_CTRL_XMC1 (4) DIGITAL_JO (4.0.6) EVENT_DETECTOR [- 	4.0.2] 1.0.2]
Double-Click on the APP to add it Show latest versions only Hide beta versions	t to the active project.



Getting Started – Example – Blinky based on DAVETM APPs (4/7)



- 2. Configure Port pin
 - Click to add new APP
 - Double-click **DIGITAL_IO** APP and close window
 - Open APP configuration editor
 - In APP Dependency view, doubleclick DIGITAL_IO
 - Configure APP instance
 - In APP configuration window, set **Pin** direction to Input/Output and set Initial output level to High

eneral Settin	gs		
in direction:	Input	/Output 👻	
Input Setting	gs		
Mode:	Tristate 👻		
Hysteresis:	Standard 👻		
Output Setti	ngs		
Mode:		Push Pull	-



- Assign pin to P0.5
 - Click To open Manual Pin
 Allocator
 - Set Pin Number (Port) to #22
 (P0.5)
 - Solve and Save

Wanual Pin Allocator			×
APP Instance Name	APP Pin Name	Pin Number (Port)	
▲ DIGITAL_IO_0			
	pin	#22 (P0.5)	Ŧ
?		Save Reset	Close

Getting Started – Example – Blinky based on DAVETM APPs (5/7)



- **3.** Configure System Timer and define its exception service routine
 - Click to add new APP
 - Double-click SYSTIMER APP and close window
 - Open APP configuration editor
 - In **APP Dependency** view, double-click SYSTIMER
 - Configure APP instance
 - In APP configuration window, under General Settings tab, set System timer tick interval to 100000us (0.1s)

SYSTIMER_0	x	
General Settings	Interrupt Settin	igs
SysTick timer per	iod [us]:	100000
Number of softw	are timers:	1





Getting Started – Example – Blinky based on DAVETM APPs (6/7)



- Create software timer using SYSTIMER Apps.

```
TimerId = (uint32_t)SYSTIMER_CreateTimer(Point2SEC,SYSTIMER_MODE_PERIODIC,(void*)LED_Toggle_EveryPoint2Sec,NULL);
if (TimerId != 0U)
{
    //timer is created successfully, now start/run software timer
    status = SYSTIMER_StartTimer(TimerId);
}
```

- Define exception handler routine in Main.c
 - Define the toggle interval (in usec)

```
#define Point2SEC 200000U
void LED_Toggle_EveryPoint2Sec(void)
{
    DIGITAL_I0_ToggleOutput(&DIGITAL_I0_0); //toggles : 1 -> 0 (if initial output level is logic 1)
}
```

Getting Started – Example – Blinky based on DAVETM APPs (7/7)



- > Generate code
 - 1. Click 📝

'Invoking: ARM-GCC Print Size'
"C:\DAVEv4\DAVE-4.1.2\eclipse\ARM-GCC-49/bin/arm-none-eabi-size" - text data bss dec hex filename
 3908 20 1100 5028 13a4 XMC1100_Blinky_withApps.elf
'Finished building: XMC1100_Blinky_withApps.siz'

- > Build project
 - 1. Click 🔀
 - 2. Wait for Build to finish
- > Download code
 - 1. Click 🎋
 - 2. Switch to Debug perspective
 - 3. Click 🕩 to run code
- > LED blinks every 0.2s



Copyright © Infineon Technologies AG 2015. All rights reserved.

Home



General Information (1/2)

- > Where to buy kit:
 - <u>http://ehitex.com/starter-kits/for-xmc1000</u>
 - Order Number: KIT_XMC11_BOOT_001
- > Infineon parts utilized on kit:

Infineon Parts	Order Number
XMC1100 Microcontroller	XMC1100-T038F0064
XMC4200 Microcontroller	XMC4200-Q48F256
5V regulator	IFX25001TFV50
3V3 regulator	IFX25001MEV33



General Information (2/2)

- Kit documentation:
 - Boot Kit XMC1100
- Video Series: XMC1000 Boot Kit Getting Started
 - Introduction
 - <u>DAVE™ Setup</u>
 - Boot Mode Index Configuration via DAVE or MemTool
 - XMC1100 Hardware Setup
 - Simple Blinky Example
 - Blinky Example based on DAVE[™] Apps
 - Example Projects Download

Home

References – Where to find XMC Lib documentation?



1. Go to DAVE[™] Version 4 website 4. Click on **Modules** or **Files**

http://www.infineon.com/dave/v4

2. Download XMC Lib and unzip file

XMC [™] Lib	Ready to use APIs for peripherals which are tested for GNU-, ARM-, IAR- und TASKING- compiler, and released for Altium, ARM/KEIL, Atollic, IAR Systems und Rowley compiler IDEs. Low level driver libraries for XMC [™] peripherals (APIs), CMSIS / MISRA 2004 compliant including documentation. XMC [™] Lib – <u>Release Note</u>					
	System:	Timer/PWM:	Analog- mixed Signal:	Communication:	Application specific:	Examples:
	 DMA ERU FCE FLASH GPIO MATH PAU PRNG 	 CCU4 CCU8 HRPWM POSIF 	ACMPADCDACDSD	CAN I2C SPI UART USB USIC Ethernet	 BCCU LEDTS MATH POSIF HRPWM 	 Examples for all periperhals drivers and ARM, GCC, IAR, and Tasking

3. Open XMC1_Peripheral_Library





The XMC Peripheral Library (XMC Lib) consists of low-level drivers for the XMC product family peripherals

Built on top the Cortex Microcontroller Software Interface Standard (CMSIS) and MISRA-C 2004 compliant, it provides access to all peripheral features

Main Page Mo	fules Files	Main Page Modules Files
Modules		File List Globals
Here is a list of all mo	dules:	File List
		Here is a list of all documented files with brief description
* XMC Peripheral Li	ibrary	🗟 xmc1 flash.h
ACMP	Analog Comparator(ACMP) low level driver for XMC family of microcontrollers.	main and a second secon
BCCU	Brightness and Color Control Unit (BCCU) driver for the XMC1 microcontroller family	
CCU4	Capture Compare Unit 4 (CCU4) low level driver for XMC family of microcontrollers	xmc1_rtc.h
CCU8	Capture Compare Unit 8 (CCU8) low level driver for XMC family of microcontrollers	xmc1_scu.h
ERU	Event Request Unit (ERU) driver for the XMC microcontroller family	xmc acmp.h
FLASH	Flash driver for XMC microcontroller family	m xmc bccu.h
GPIO	General Purpose Input Output (GPIO) driver for the XMC microcontroller family	
12C	Inter Integrated Circuit(IIC) driver for the XMC microcontroller family	xmc_ccu4.h
LEDTS	LED and Touch-Sense control(LEDTS) driver for the XMC controller family	xmc_ccu8.h
MATH	MATH Coprocessor (MATH) driver for the XMC1302 microcontroller family	xmc eru.h
PAU	Peripheral Access Unit (PAU) driver for the XMC1000 microcontroller family	
POSIF	Position Interface Unit (POSIF) driver for the XMC microcontroller family	xmc_flash.h
PRNG	Pseudo Random Number Generator (PRNG) driver for XMC1000 microcontroller family	xmc_gpio.h
SCU	RTC driver for XMC microcontroller family	xmc_i2c.h
SPI	System Control Unit(SCU) driver for XMC microcontroller family Synchronous serial channel driver for SPI-like communication	xmc ledts.h
UART	Universal Asynchronous Receiver/Transmitter (UART) driver for XMC microcontroller family	
USIC	Universal Asynchronous Receiver/Transmitter (UART) driver for XMC microcontroller family Universal Serial Interface Channel(USIC) driver for serial communication	xmc_math.h
VADC	Versatile Analog to Digital Converter (VADC) driver for XMC microcontroller family	xmc_pau.h
WDT	Watchdog driver for the XMC microcontroller family	xmc_posif.h

Home

References – Where to find DAVETM APP documentation?



1. In DAVETM, go to Help \rightarrow Help Contents

Help	
	Welcome
?	Help Contents
X	Search
	Dynamic Help

2. Expand DAVE Apps → Click on **CLOCK_XMC1** → **Overview**





References – Where to download DAVE[™] Example Projects?



- **1.** Example Project library within $DAVE^{TM}$
- 2. DAVE[™] website
- 3. Example from XMC Lib package

37

References – How to load Example Project in DAVETM? (1/4)

- Example Project library within DAVE[™]
 - Help → Install DAVE APP/Example/Device Library

Help	•	
	Welcome	
?	Help Contents	
%	Search	
	Dynamic Help	
	Key Assist	Ctrl+Shift+L
	Tips and Tricks	
	DAVE [™] Forum	
	DAVE [™] News	
	Send Feedback Mail	
	Cheat Sheets	
è	Install DAVE APP/Example/Device Library	
	Check for DAVE APP Updates	
	Uninstall DAVE APP/Example/Device Library	

2. Select DAVE Project Library Manager

🤹 Library Mana	ger Wizard	
Download Lil	oraries Page	
This wizard pa library	ge helps in downloading the libraries of type example projects or apps	
Dave Site		
Work with :	DAVE Project Library Manager	▼ Add
	Find more library by working with the Library Update Sites p	references

3. Select Examples in the Libraries window \rightarrow Click Next

This wizard page helps in downloading the libraries of type example projects or apps ibrary Dave Site Work with : DAVE Project Library Manager	
Work with : DAVE Project Library Manager	
	▼ Add
Find more library by working with the Library Update Sites pr	eferences
Enter the keywords to filter :	
Name Version	n ^
	u v
< III.	F.
Select All Deselect All	

 Accept terms of the license agreements →Click Finish



> DAVE[™] website

DAVE™

EXAMPLES

 Go to DAVE[™] Version 4 website

http://www.infineon.com/dave/v4

XMC Lib (Low Level Driver for XMC MCUs) and DAVE™ APPs composed to application examples

2. Download DAVE[™] EXAMPLES

3. In DAVETM, go to File \rightarrow Import

4. Select DAVE Project \rightarrow Next



 Select Archive File → Browse to downloaded project zip file



6. Click Finish

(i)

Ŀ



- Example from XMC Lib package
 - Go to DAVE[™] Version 4 website

http://www.infineon.com/dave/v4

2. Download XMC Lib and unzip file

XMC Lib	Low level driver libraries for XMC peripherals (APIs), CMSIS / MISRA 2004 compliant including documentation						
	System:	Timer/PWM:	Analog- mixed Signal:	Communication:	tion: Application Examples: specific:		
	 DMA ERU FCE FLASH GPIO MATH PAU PRNG 	CCU4 CCU8 HRPWM POSIF	 ACMP ADC DAC 	CAN 12C SPI UART USB USIC	 BCCU LEDTS MATH POSIF HRPWM 	 Examples for all periperhals drivers and ARM, GCC, IAR, and Tasking 	

3. Example code (main.c) can be found within XMC Lib package



 Create new "Simple Main Project" in DAVE[™]

😜 New DAVE Project		
DAVE Project		
Create a new C/C++ project for Infined	n tool chains	
Project Name: GPIO_TOGGLE_Examp	le	
Use default location		
Location: C:/DAVE_Workspace		Browse
Project Type:	Tool Chain:	
a 🗁 Infineon Projects	ARM-GCC Applica	tion
ARM-GCC Application		
Easy Start Project	=	
Simple Main Project		
DAVE CE Project		
ARM-GCC Library	-	

Infineon



References – How to load Example Project in DAVE[™]? (4/4)



Select target device of selected main.c example

rget Selection Page	
lect the controller for which the project has to be created	
V ProcessorsInfo	
XMC4000	E
XMC4500 Series	-
XMC4400 Series	
XMC4200 Series	
XMC4100 Series	
XMC1000	
XMC1100 Series	

- 7. Click 💦 to Build project
- Click by to download and run project on target board

- 6. Delete main.c in the newly created DAVE project
- Copy main.c from XMC Lib example into DAVE project



Home



Part of your life. Part of tomorrow.

