QMX-32 Capella Data Sheet and User Manual

Quorb Inc. A Laboratory everywhere.

Table of Contents

Introduction

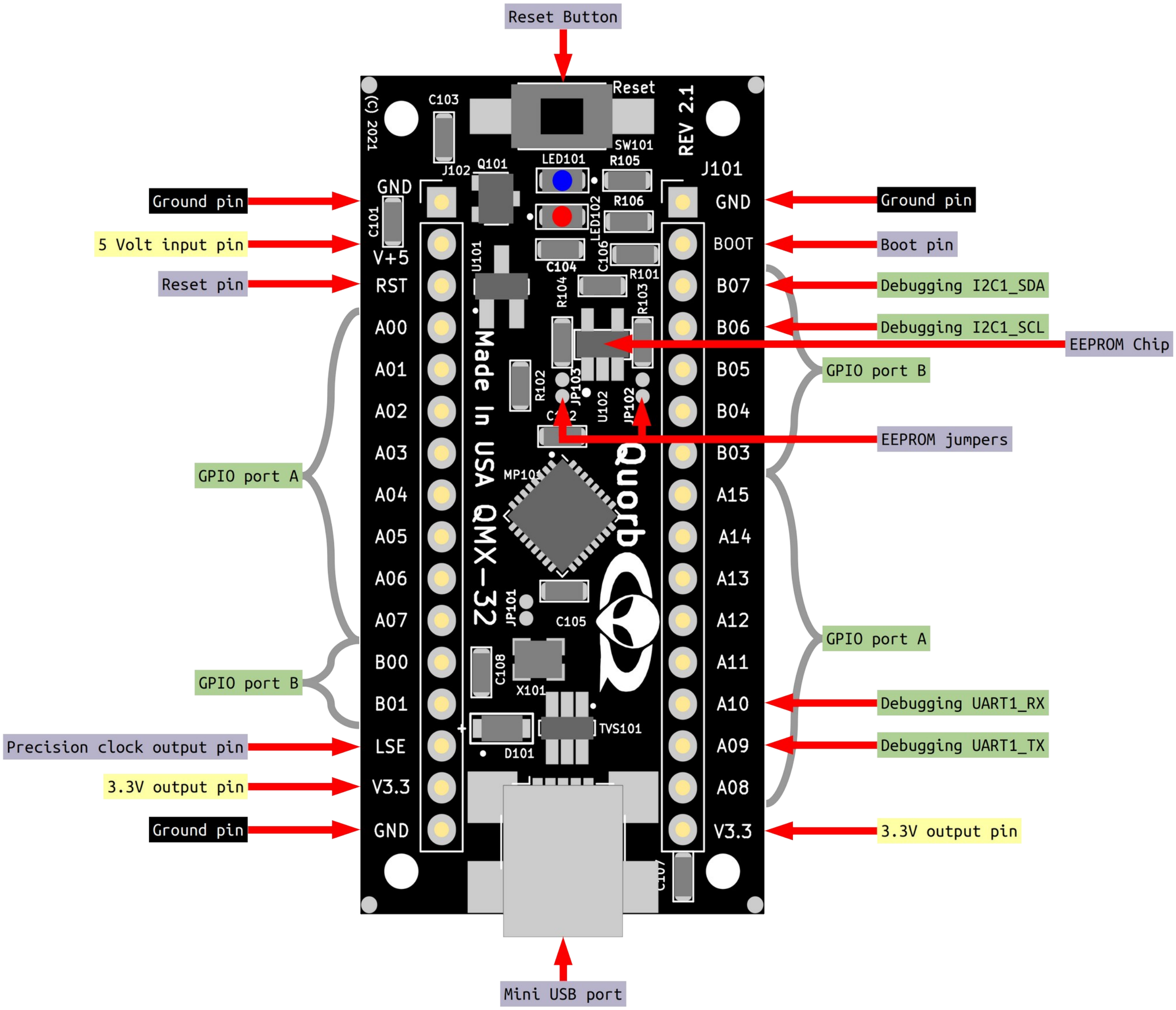
The QMX-32 is a 32 pin experimentation board which makes use of the STM32L432 micro-controller. The board has a built in EEPROM and 8Mhz oscillator. The pins on the board stick out over both the top and bottom of the board in order to allow for breadboard compatibility as well as easy testing and interfacing on both surfaces. The chip itself contains the following.

- An analog to digital converter (ADC) and a digital to analog converter (DAC)
- A controller area network (CAN 2.0)
- Two comparators (COMP)
- A programmable Operational amplifier (OPAMP)
- Two I Squared C channels (I2C)
- A Quad SPI Memory interface (QUADSPI)
- A serial audio interface (SAI)
- Two Serial Peripheral interfaces (SPI)
- A single wire protocol master interface (SWPMI)
- Four timers (TIM) and two low power timers (LPTIM)
- A touch sensor controller (TSC)
- Three universal synchronous/asynchronous receiver transmitter (USART) and a low power universal asynchronous receiver transmitter (LPUART)
- A Universal Serial Bus (USB)
- Two Wake up pins (WKUP)

Back to TOC

Board layout

FIG 1



Back to TOC

Commonly used inputs and outputs

There is a vast array of peripherals packed into the chip on the QMX-32 module, but some are used more often than others. Here are the most commonly used ones on this module.

Serial Communications

This section only deals with the basic communications to get the user up and running. In that light, there are several serial communications ports available on the QMX-32 module. The most commonly used ports are listed in this section. Additional serial communications peripherals can be understood by referring to the STM32L432 Reference Manual.

USART-1

USART-1 Is a programmable Universal Synchronous / Asynchronous communications peripheral built into the QMX-32 Micro-Controller module. This peripheral uses module pins: A09 for TX Data and A10 for RX Data (See Fig: 2.1) for the location of these module pins. This is a full functioning serial interface using the TTL level serial protocol. If the user wishes to use a standard RS232 interface with this peripheral, then the user should obtain a TTL to RS232 serial port adapter available on line from many vendors including Quorb Inc.

USART-2

USART-2 Is another programmable Universal Synchronous / Asynchronous communications peripheral built into the QMX-32 Micro-Controller module. This peripheral uses module pins: A02 for TX Data and A03 for RX Data (See Fig: 2.1) for the location of these module pins.

I2C-1

I2C-1 Is a serial peripheral controller that allows communications with I2C compatible devices at speeds of up to 1 Mbit/Second. It should be mentioned here that the QMX-32 module has an on-board 64K bit I2C EEPROM memory which is normally connected. This device may be disconnected from the I2C bus by cutting the jumpers JP102 and JP103 on the PCB. This will isolate the EEPROM from the module pins. The I2C-1 (SDA) port pin is attached to the B06 module pin and the I2C-1 (SCL) port pin is connected to the B07 module pin the module. (See Fig: 2.0) for the location of these module pins. There are 2.49K ohm pull-up resistors connected to the I2C-1 bus. This gives sufficient current to meet the 1 Mbit speed requirement for up to 5 external I2C devices.

SPI-1

SPI-1 Is a multi pin serial communications peripheral. This allows access to cameras, SD Memory Cards, and many single

CAN 2.0

The Micro-controller for the QMX-32 contains a Controller Area Network Serial Port. This communications protocol allows for error resistant serial communications similar to I2C in behavior but far more robust in operation.

USB 2.0 Full speed

An on-board USB port for USB communications as well as power and DFU programming.

Timers, ADC, Sound and Other Peripherals:

Timers:

There are many on chip timers available within the QMX-32 Module. Because of the complexity in using the resident timers, it is suggested that the user become thoroughly familiar with ARM Timer operation. Full pin-outs for timers can be referenced in fig 2.1, and operations of the timers can be found in the reference manual

ADC:

This is an internal very high performance peripheral that can be accessed via most Port(A) and Port (B) pins.

DAC, Sound, Touch Etc. Interfaces:

These peripheral controllers are rarely used except for special cases.

Further Processor Devices:

Please see the ST Micro-controller Reference Manual (RM0394), Data Sheet (DS11451) and Programming Manual (PM0214) for the STM32L432 device. Available from ST here

Electrical characteristics

Power input: 5V at 100 mA

All IO are 5V tolerant, and trigger at 3.3V levels.

Further electrical information can be found in the STM32L432 datasheet.

Chip peripherals

FIG 2.0

peripheral	pinout	peripheral	pinout
ADC1	ADC1_IN5: A00	LPUART1	LPUART1_TX: A02
	ADC1_IN6: A01		LPUART1_RX: A03
	ADC1_IN7: A02		LPUART1_CTS: A06
	ADC1_IN8: A03		LPUART1_RTS_DE: B01
	ADC1_IN9: A04	OPAMP1	OPAMP1_VINP: A00
	ADC1_IN10: A05		OPAMP1_VINM: A01
	ADC1_IN11: A06		OPAMP1_VOUT: A03
	ADC1_IN12: A07	QUADSPI	QUADSPI_BK1_NCS: A02
	ADC1_IN15: B00		QUADSPI_CLK: A03
	ADC1_IN16: B01		QUADSPI_BK1_IO3: A06
CAN1	CAN1_RX: A11		QUADSPI_BK1_IO2: A07
	CAN1_TX: A12		QUADSPI_BK1_IO1: B00
COMP1	COMP1_OUT: A00, A06, B00, A11		QUADSPI_BK1_IO0: B01
	COMP1_INM: A00, A04, A05, B01	SAI1	SAI1_EXTCLK: A00, B00
	COMP1_INP: A01		SAI1_MCLK_A: A03
COMP2	COMP2_OUT: A02, A07, B05		SAI1_FS_B: A04, A14, B06
	COMP2_INM: A02, A04, A05, B03, B07		SAI1_SCK_A: A08
	COMP2_INP: A03, B04, B06		SAI1_FS_A: A09
DAC1	DAC1_OUT1: A04		SAI1_SD_A: A10
	DAC1_OUT2: A05		SAI1_SD_B: A13, B05
I2C1	I2C1_SMBA: A01, A14, B05		SAI1_SCK_B: B03
	I2C1_SCL: A09, B06		SAI1_MCLK_B: B04
	I2C1_SDA: A10, B07	SPI1	SPI1_SCK: A01, A05, B03
I2C3	I2C3_SCL: A07		SPI1_NSS: A04, B00, A15
	I2C3_SDA: B04		SPI1_MISO: A06, A11, B04
LPTIM1	LPTIM1_OUT: A14		SPI1_MOSI: A07, A12, B05
	LPTIM1_IN1: B05	SPI3	SPI3_NSS: A04, A15
	LPTIM1_ETR: B06		SPI3_SCK: B03
	LPTIM1_IN2: B07		SPI3_MISO: B04
LPTIM2	LPTIM2_OUT: A04, A08		SPI3_MOSI: B05
	LPTIM2_ETR: A05		
	LPTIM2_IN1: B01		

Chip peripherals continued

FIG 2.1

peripheral	pinout	peripheral	pinout
TIM1	TIM1_BKIN: A06	USART1	USART1_CK: A08, B05
	TIM1_BKIN_COMP2: A06		USART1_TX: A09, B06
	TIM1_CH1N: A07		USART1_RX: A10, B07
	TIM1_CH2N: B00		USART1_CTS: A11, B04
	TIM1_CH3N: B01		USART1_RTS_DE: A12, B03
	TIM1_CH1: A08	USART2	USART2_CTS: A00
	TIM1_CH2: A09		USART2_RTS_DE: A01
	TIM1_CH3: A10		USART2_TX: A02
	TIM1_CH4: A11		USART2_RX: A03, A15
	TIM1_BKIN2: A11		USART2_CK: A04
	TIM1_BKIN2_COMP1: A11	USART3	USART3_CTS: A06
	TIM1_ETR: A12		USART3_CK: B00
TIM2	TIM2_CH1: A00, A05, A15		USART3_RTS_DE: B01, A15
	TIM2_ETR: A00, A05, A15	USB	USB_CRS_SYNC: A10
	TIM2_CH2: A01, B03		USB_DM: A11
	TIM2_CH3: A02		USB_DP: A12
	TIM2_CH4: A03		USB_NOE: A13
TIM15	TIM15_CH1N: A01	WKUP	WKUP4: A02
	TIM15_CH1: A02		WKUP1: A00
	TIM15_CH2: A03	Standalone	CK_IN: A00
	TIM15_BKIN: A09		JTCK-SWCLK: A14
TIM16	TIM16_CH1: A06		JTDI: A15
	TIM16_BKIN: B05		JTDO-TRACESWO B03
	TIM16_CH1N: B06		JTMS-SWDIO: A13
TSC	TSC_G3_IO1: A15		IR_OUT: A13
	TSC_G2_IO1: B04		LSCO: A02
	TSC_G2_IO2: B05		MCO: A08
	TSC_G2_IO3: B06		NJTRST: B04
	TSC_G2_IO4: B07		PVD_IN: B07
SWPMI1	SWPMI1_IO: A08		RTC_TAMP2: A00
	SWPMI1_TX: A13		
	SWPMI1_RX: A14		
	SWPMI1_SUSPEND: A15		

Back to TOC

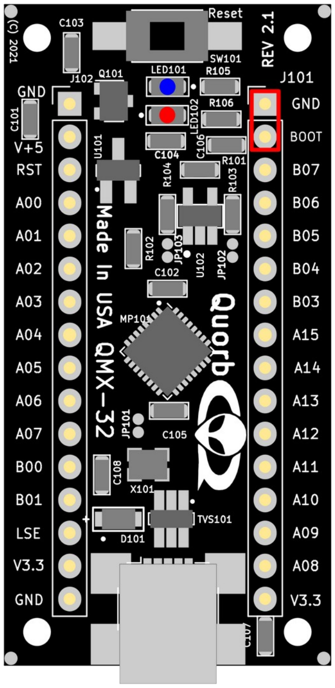
Activating and loading a program.

- As soon as you open your QMX-32 plug it into a USB port on your computer and hit the reset button. You should see the red LED on the board turn on and off in a series of short and long intervals. This shows that your board is in good working order and that you are ready to start your projects. The below instructions assume that you have compiled your program into either a .hex or .elf file.

Recommended programming steps

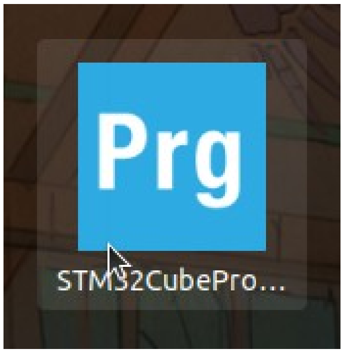
- Notes:
  - No default selections or configurations need to be changed in the programmer application.
  - The programming method we will be using in these instructions is STM32 Cube programmer through the built in USB plug. Please consult your manual if you use a different programming method.

1. Remove the SEL jumper that shorts together the Boot and Ground pin.

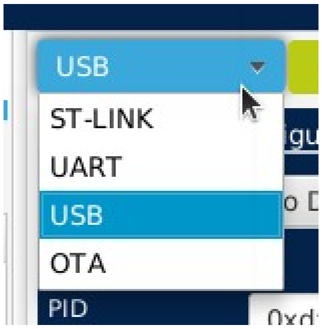


Back to TOC

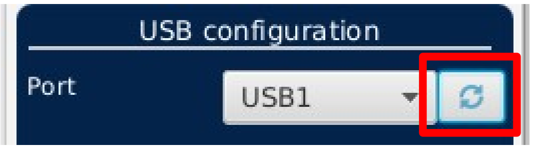
2. Press the reset button. If the red and blue LED are both on, then the board is now in programming mode. Open the STM32 Cube programmer.



3. On the right side of the programmer window, you will see a blue dropdown menu. Click the drop-down and select USB.



4. Click the refresh button next to the port drop-down, and USB1 should appear in the drop-down menu.



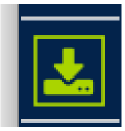
USB configuration

Port USB1

   1. Note: If the port selector says ‘No DFU detected,’ then hit the reset button on the module then refresh, and it should appear.

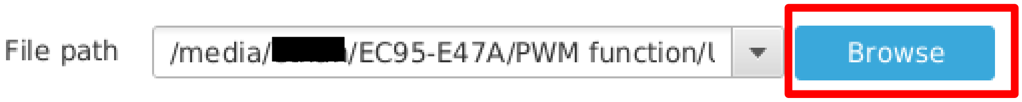
5. Click the connect button and now you are ready to load your program.

6. Select erasing and programming on the left side of the programmer window.



Back to TOC

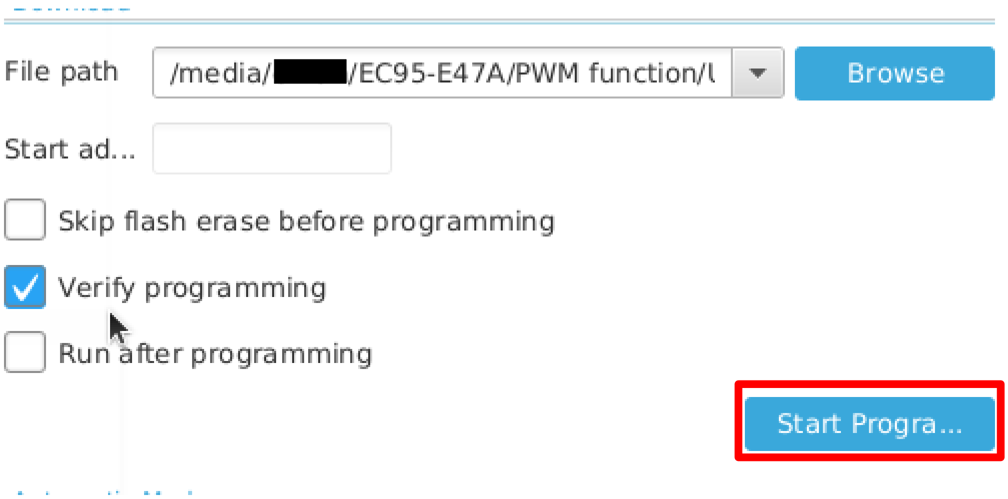
7. Next to the file path drop-down menu, select browse, and your file manager should open up.



File path /media/[redacted]/EC95-E47A/PWM function/l

8. Navigate to the location where you saved and compiled your project, and select the appropriate .hex or .elf file.

9. Click start programming and your program should load straight into the chip.



File path /media/[redacted]/EC95-E47A/PWM function/l

Start ad...

□ Skip flash erase before programming

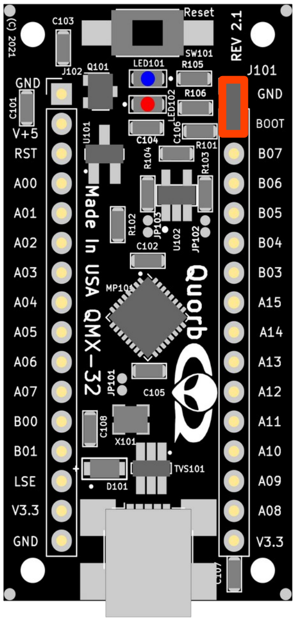
☑ Verify programming

□ Run after programming

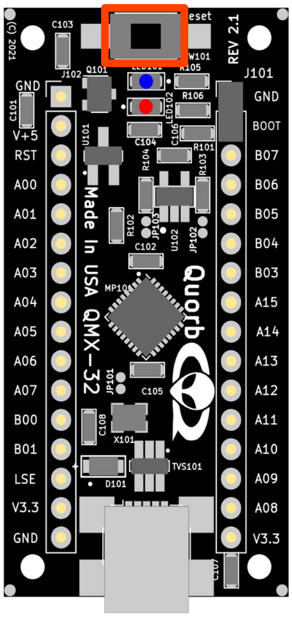
10. If the last line in the log says that your program loaded successfully. At this point, the chip is now programmed, and you are ready to run your program.

Running your program.

1. Replace the SEL jumper back onto boot and ground.



2. Press the reset button.



3. Congratulations, your new program should be running.

Other reference material

- STM32L432 datasheet
- STM32L432 reference manual
- STM32L432 programming manual
- STM32 Cube programmer user manual

Back to TOC