

## STM32-P405 development board USER'S MANUAL

Revision A, December 2015  
Designed by OLIMEX Ltd, 2015



All boards produced by Olimex LTD are ROHS compliant

## DISCLAIMER

---

© 2015 Olimex Ltd. Olimex®, logo and combinations thereof, are registered trademarks of Olimex Ltd. Other product names may be trademarks of others and the rights belong to their respective owners.

**The information in this document is provided in connection with Olimex products. No license, express or implied or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Olimex products.**

The hardware design of STM32-P405 development board is considered propriety. The board's schematics are available in pdf format. The original schematic and design files are not available to the public.

The part of the software written by Olimex is released under GPL. However, note that the examples that we distribute are based on the libraries and projects released by IAR Systems. Make sure that you are acquainted with the license agreement of IAR Embedded Workbench before using, publishing, or distributing any code based on the software released by Olimex.

It is possible that the pictures in this manual differ from the actual appearance of the latest revision of the board.

The product described in this document is subject to continuous development and improvements. All particulars of the product and its use contained in this document are given by OLIMEX in good faith. However all warranties implied or expressed including but not limited to implied warranties of merchantability or fitness for purpose are excluded. This document is intended only to assist the reader in the use of the product. OLIMEX Ltd. shall not be liable for any loss or damage arising from the use of any information in this document or any error or omission in such information or any incorrect use of the product.

This evaluation board/kit is intended for use for engineering development, demonstration, or evaluation purposes only and is not considered by OLIMEX to be a finished end-product fit for general consumer use. Persons handling the product must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards.

Olimex currently deals with a variety of customers for products, and therefore our arrangement with the user is not exclusive. Olimex assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.

**THERE IS NO WARRANTY FOR THE DESIGN MATERIALS AND THE COMPONENTS USED TO CREATE STM32-P405. THEY ARE CONSIDERED SUITABLE ONLY FOR STM32-P405.**

## Table of Contents

<a href="#">DISCLAIMER.....</a>	<a href="#">2</a>
<a href="#">CHAPTER 1 OVERVIEW.....</a>	<a href="#">5</a>
<a href="#">1. Introduction to the chapter.....</a>	<a href="#">5</a>
<a href="#">1.1 Features.....</a>	<a href="#">5</a>
<a href="#">1.2 Target market and purpose of the board.....</a>	<a href="#">6</a>
<a href="#">1.3 Similar boards.....</a>	<a href="#">6</a>
<a href="#">1.4 Document Organization.....</a>	<a href="#">6</a>
<a href="#">CHAPTER 2: BOARD DESCRIPTION.....</a>	<a href="#">7</a>
<a href="#">2.1 Layout (top view).....</a>	<a href="#">7</a>
<a href="#">2.2 Prototype area pinout (top view).....</a>	<a href="#">8</a>
<a href="#">CHAPTER 3 SETTING UP THE STM32-P405 BOARD.....</a>	<a href="#">9</a>
<a href="#">3. Introduction to the chapter.....</a>	<a href="#">9</a>
<a href="#">3.1 Electrostatic warning.....</a>	<a href="#">9</a>
<a href="#">3.2 Requirements.....</a>	<a href="#">9</a>
<a href="#">3.2.1 Hardware requirements.....</a>	<a href="#">9</a>
<a href="#">3.2.2 Software requirements.....</a>	<a href="#">10</a>
<a href="#">3.3 Powering the board.....</a>	<a href="#">11</a>
<a href="#">3.4 Programming the board.....</a>	<a href="#">11</a>
<a href="#">3.4.1 Using JTAG debugger.....</a>	<a href="#">11</a>
<a href="#">3.4.2 Using USB cable and bootloader software.....</a>	<a href="#">11</a>
<a href="#">3.4.3 Using RS232 cable and bootloader software.....</a>	<a href="#">12</a>
<a href="#">3.5 Prebuilt software.....</a>	<a href="#">12</a>
<a href="#">CHAPTER 4 THE STM32F405 MICROCONTROLLER.....</a>	<a href="#">13</a>
<a href="#">4. Introduction to the chapter.....</a>	<a href="#">13</a>
<a href="#">4.1 The STM32F405RG features.....</a>	<a href="#">13</a>
<a href="#">CHAPTER 5 CONTROL CIRCUITY AND HARDWARE MODULES.....</a>	<a href="#">15</a>
<a href="#">5. Introduction to the chapter.....</a>	<a href="#">15</a>
<a href="#">5.1 Reset.....</a>	<a href="#">15</a>
<a href="#">5.2 Clocks.....</a>	<a href="#">15</a>
<a href="#">5.3 Power supply circuit.....</a>	<a href="#">15</a>
<a href="#">CHAPTER 6 CONNECTORS AND PINOUT.....</a>	<a href="#">17</a>
<a href="#">6. Introduction to the chapter.....</a>	<a href="#">17</a>
<a href="#">6.1 USB connector.....</a>	<a href="#">17</a>
<a href="#">6.2 JTAG/SWD debug connector.....</a>	<a href="#">17</a>
<a href="#">6.3 CAN connector.....</a>	<a href="#">18</a>
<a href="#">6.4 UEXT connector.....</a>	<a href="#">18</a>
<a href="#">6.5 PWR JACK connector.....</a>	<a href="#">19</a>
<a href="#">6.6 BAT 3V connector.....</a>	<a href="#">19</a>
<a href="#">6.7 RS232 2 connector.....</a>	<a href="#">20</a>
<a href="#">6.8 SD/MMC card connector.....</a>	<a href="#">20</a>
<a href="#">6.9 Extension pins and prototype area.....</a>	<a href="#">21</a>
<a href="#">6.4 Jumper description.....</a>	<a href="#">21</a>
<a href="#">6.4.1 BOOT memory select jumpers.....</a>	<a href="#">21</a>
<a href="#">6.4.2 RTS E.....</a>	<a href="#">21</a>
<a href="#">6.4.3 CTS E.....</a>	<a href="#">21</a>
<a href="#">6.4.4 R-T jumper.....</a>	<a href="#">22</a>
<a href="#">6.4.5 CAN0 T.....</a>	<a href="#">22</a>

---

6.4.6 LED E.....	22
6.4.7 USBP E.....	22
6.4.8 WP E jumper.....	22
6.4.9 3.3V MCU E jumper.....	23
6.5 Additional hardware components.....	23
CHAPTER 7 BLOCK DIAGRAM AND MEMORY.....	24
7. Introduction to the chapter.....	24
7.1 Processor family block diagram.....	24
7.2 Physical memory map.....	25
CHAPTER 8 SCHEMATICS.....	26
8. Introduction to the chapter.....	26
8.1 Eagle schematic.....	26
8.2 Physical dimensions.....	28
CHAPTER 9 REVISION HISTORY AND SUPPORT.....	29
9. Introduction to the chapter.....	29
9.1 Document revision.....	29
9.2 Board revision.....	29
9.3 Useful web links and purchase codes.....	30
9.4 Product support.....	31

---

# CHAPTER 1 OVERVIEW

---

## 1. Introduction to the chapter

---

Thank you for choosing this general-purpose development board designed and assembled by Olimex! This document provides a user's guide for the Olimex STM32-P405 board. As an overview, this chapter gives the scope of this document and lists the board's features. A comparison between boards similar to STM32-P405 is presented. The document's organization is then detailed.

The STM32-P405 development board enables code development of applications running on the microcontroller STM32F405RG, manufactured by STMicroelectronics.

STM32-P405 board is a medium-sized development board with a small prototype area. Using STM32-P405 you can explore the features of STM32F4 family on budget. The board has everything necessary for a large number of different applications: USB port for communication and powering; separate power jack for powering; reset and oscillator circuits; JTAG port for programming and debugging; serial driver and DE9 connector; UEXT expansion connector; prototype area; programmable LED; programmable button and much more.

There are plenty of GPIOs available at the GPIO area which can be used to connect your additional circuits.

### 1.1 Features

---

The board has the following list of features:

- MCU: STM32F405RG ARM Cortex M4 in LQFP64 package that operates up to 168 Mhz; with 1024KB flash memory; (192+4)kB SRAM; 2×USB; 2×CAN; 2 x I2C; 3×12-bit ADC×16 channels; 2×12-bit DAC; 2×UART; 3×SPI; 17 timers; with up to 51 GPIOs
- Standard JTAG (SWD) connector with ARM 2×10 pin layout for programming/debugging with ARM-JTAG
- USB type B connector
- User and RESET buttons
- User and power supply LEDs
- CAN driver and connector
- RS232 driver and connector
- UEXT connector which allows easier hardware connection to different extension modules (like MOD-IO2, MOD-MAG, etc.)
- SD-MMC connector
- On-board voltage regulator 3.3V with up to 800mA current
- Single power supply: takes power from either the USB port or the power supply jack
- 8 Mhz crystal oscillator
- 32768 Hz crystal and RTC backup battery connector
- Prototype area with 0.1" step for all uC ports
- PCB: FR-4, 1.5 mm (0.062"), red soldermask, white silkscreen component print
- Dimensions: (100 × 90)mm ~ (3.9 × 3.5)"

---

## 1.2 Target market and purpose of the board

---

STM32-P405 is a development board featuring a ARM Cortex-M4 microcontroller. The board can be powered by a number of different sources. It can be programmed with a debugger via JTAG or SWD, or alternatively with a cable via USB. The board has a lot of GPIO pins exposed.

The board is suitable for numerous tasks and situations. The power of ARM and the creativity of OLIMEX come at the best price and the well-known quality.

Every ARM enthusiast would see STM32-P405 as quite capable board for its low price.

---

## 1.3 Similar boards

---

The STM32H405RG microcontroller is also present in two other Olimex-made boards – STM32-H405 and STM32-405STK.

- STM32-H405 is a smaller and cheaper header board – it is a good choice if you want to embed STM32H405 in your own product or design. It is not recommended for initial evaluation of STM32H405RG since it lacks in peripherals and interfaces available.
- STM32-405STK has a different set of peripherals compared to STM32-P405 – micro SD card interface with connector, mini USB, 84x48 BW display; microphone input with connector; headphone output with connector; UEXT connector; two male extension headers. The board can be powered by a 1.5V AA battery – there is a battery holder provided.

Another board, quite similar to STM32-P405, is STM32-P103; the board designs are almost identical – the main difference is that STM32-P103 uses STM32F103RB.

If you are looking for a board with similar processor and Ethernet functionality take a look at STM32-E407 board. It uses STM32H407ZG.

---

## 1.4 Document Organization

---

Each section in this document covers a separate topic, organized as follow:

- Chapter 1 is an overview of the board usage and features
- Chapter 2 contains the board's layout
- Chapter 3 provides a guide for quickly setting up the board
- Chapter 4 describes the component that is the heart of the board: the STM32F405 microcontroller
- Chapter 5 is an explanation of the control circuitry associated with the microcontroller to reset. Also shows the clocks on the board
- Chapter 6 covers the connector pinout, peripherals and jumper description
- Chapter 7 shows the block diagram and the memory map
- Chapter 8 provides the schematics
- Chapter 9 contains the revision history, useful links and support information

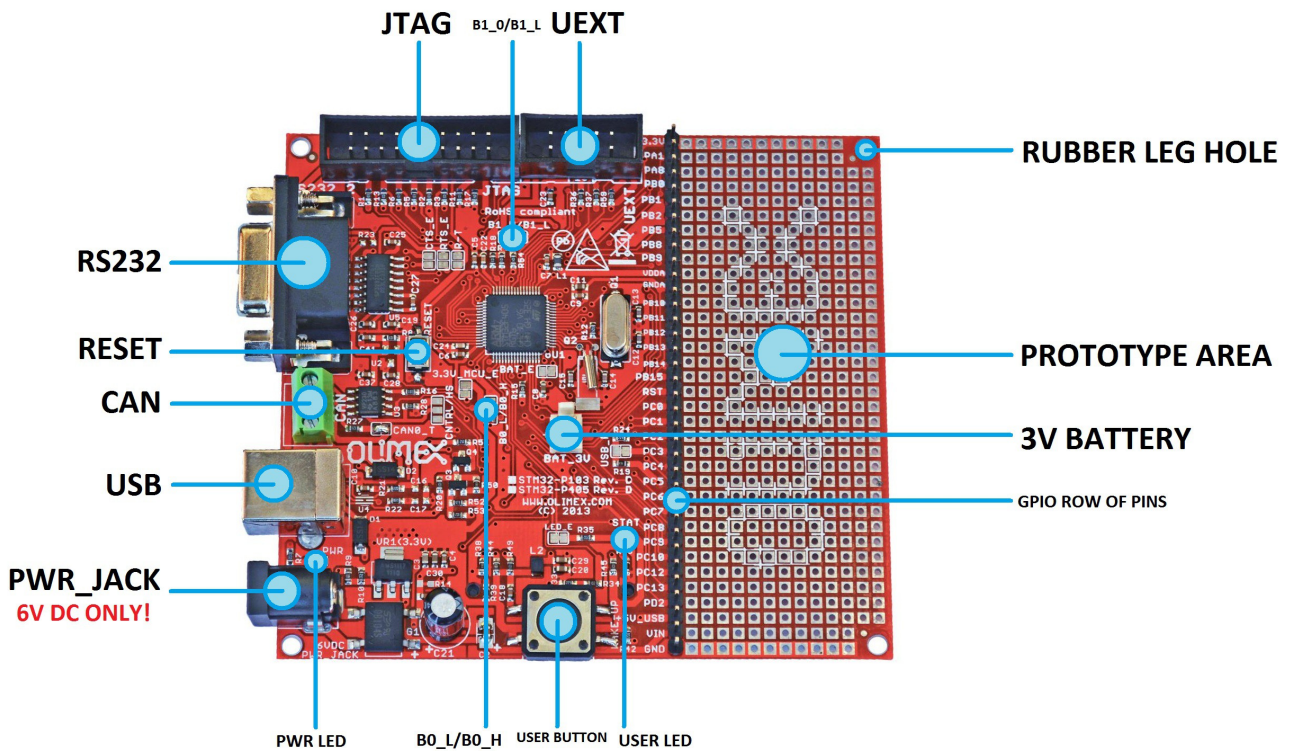
## CHAPTER 2: BOARD DESCRIPTION

In this chapter you will get acquainted with the main parts of the board. Note the names used on the board differ from the names used to describe them. For the actual names check the STM32-P405 board itself.

### 2.1 Layout (top view)

The picture below shows the top side of the board and highlights the most important parts.

Notice that all on-board jumpers are SMD type – in order to change the position of the jumpers you would either need to cut between the pads of the jumper or solder the pads together.



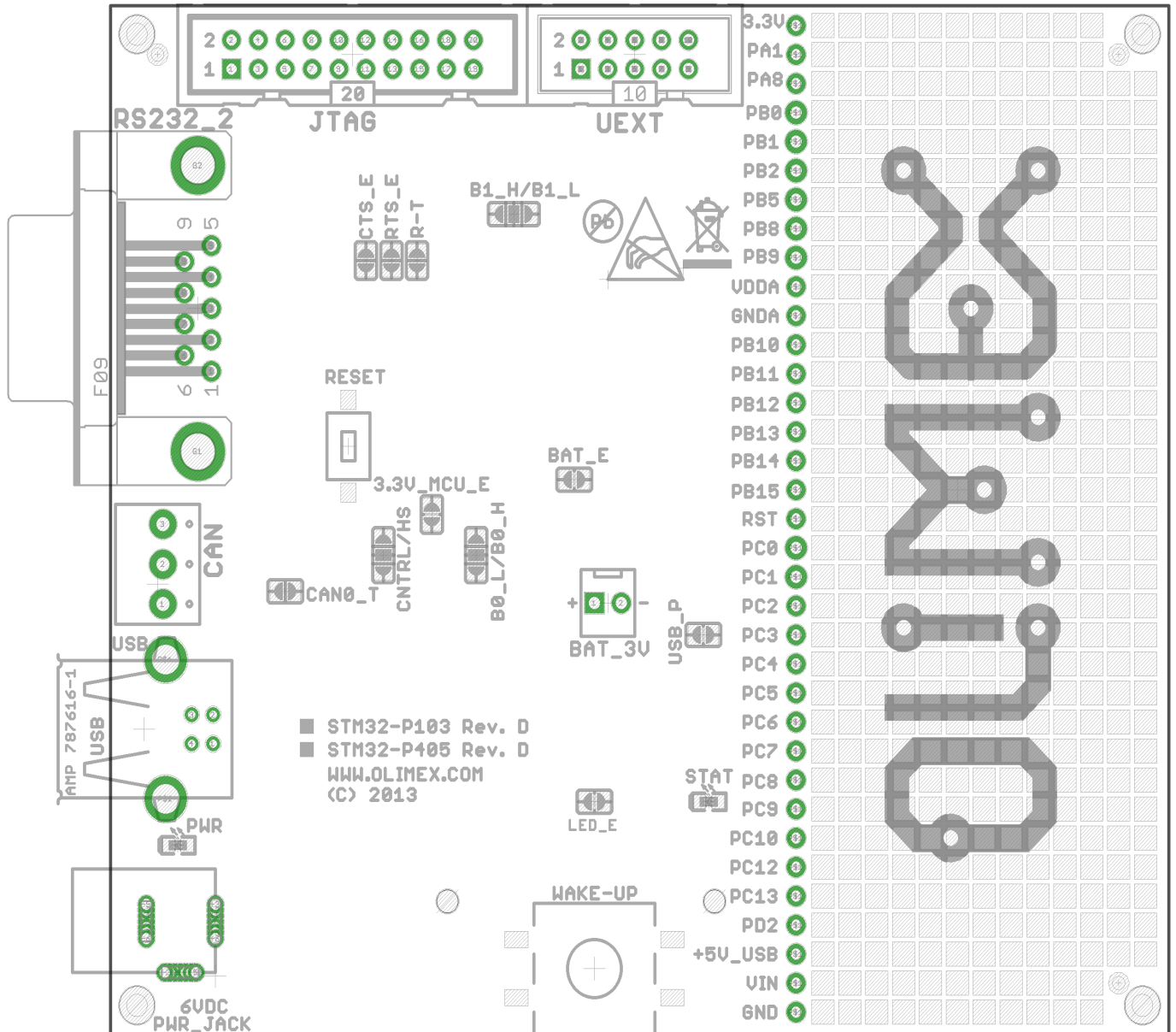
The bottom view of the board is not shown since there is only one important component of the board at the bottom – the SD/MMC connector. Note that it is a full-sized MMC connector.



## 2.2 Prototype area pinout (top view)

Below you would find the signals available at the male headers and the prototype area. Each pinhole of the same row of the prototype area has the same signal available.

The picture also includes the positions of all connectors available at the top, all jumpers, the LEDs, the buttons. Refer to chapter 6 for signal pinout of the rest of the connectors.





---

## CHAPTER 3 SETTING UP THE STM32-P405 BOARD

---

### 3. Introduction to the chapter

---

This section helps you set up the STM32-P405 development board for the first time.

Please consider first the electrostatic warning to avoid damaging the board, then discover the hardware and software required to operate the board.

The procedure to power up the board is given, and a description of the default board behavior is detailed.

#### 3.1 Electrostatic warning

---

STM32-P405 is shipped in a protective anti-static package. The board must not be exposed to high electrostatic potentials. A grounding strap or similar protective device should be worn when handling the board to avoid electrostatic discharge.

In the package there is a bag with 4 rubber pads which can be placed in the holes near the corners of the board. That way the board will be safe from short-circuiting on the bottom if placed on low-resistance surface. The placement stability of the board will also increase.

#### 3.2 Requirements

---

In order to set up the STM32-P405 optimally you would need a hardware debugger tool and compatible software tools. More detailed information may be found in the sub-chapters below.

##### 3.2.1 Hardware requirements

---

- Standard ARM JTAG/SWD debugger – it can power the board and gives you the ability to debug your code in real-time – to select the correct tool make sure that it is compatible with the software tools which you are going to use during development. Also makes sure that the tool is compatible with the STM32F405 processor.

**IMPORTANT:** The board can be programmed using only a USB cable. However, it is a slow process – this way of programming is not suitable for evaluation, research and development. Using the USB DFU bootloader might be good idea only if you need to program the board once or twice.

It is not possible to use the RS232 connector and cable for programming (via the chip's serial bootloader). It is not implemented by hardware design.

Additional components can be acquired in order to increase the functionality of the board:

- USB cable with type A and type B connectors – in case you want to power the board from the USB port of your personal computer or, alternatively, if you want need to utilize the board's USB interface
- serial cable to interface the board's female DE-9 connector
- jumper wires – allow solderless access to the exposed pins; if you want to directly connect to the extension headers use wires with a female connectors.
- power adapter with suitable connector capable of providing exactly 6V of DC voltage
- UEXT extension boards – such can be used to expand the features of the board

Some of the suggested items can be purchased by Olimex, for instance:

**ARM-USB-TINY-H** – high-speed OpenOCD ARM JTAG debugger

**ARM-USB-OCD-H** – high-speed OpenOCD ARM JTAG debugger with buffer protection

**USB-A-B-CABLE** – cable with USB type A and USB type B connectors at each end

**RS232-CABLE** – cable with male DE-9 connector at one side and female DE-9 connector at the other

**JW-200x10-FM** – ten jumper wires with female-male connectors

### 3.2.2 Software requirements

In order to change the code on the board you would need an ARM C compiler and debugger software, the possible software options are:

- free open-source software: OpenOCD (all Olimex OpenOCD JTAG debuggers are supported in OpenOCD)

- free software: CooCox IDE (all Olimex JTAGs are supported in CooCox)

- commercial software: EW-ARM from IAR Systems AB; CrossWorks from Rowley (all Olimex OpenOCD JTAG debuggers are supported in IAR and Rowley Crossworks; note that our debugging tools doesn't work in Keil MDK-ARM).

Refer to the table below for some guidance when choosing development tools.

Comparison table						
	Is it free or commercial	Initial struggle and set-up time	STM32-P405 examples provided by Olimex	Is it open-source	Does it work with Olimex OpenOCD debuggers	Quality of customer support
OpenOCD	Free	Highest	No	Yes	Yes	Bad
CooCox IDE	Free	High	No	No	Yes	Mediocre
Rowley Crossworks for ARM	Commercial (free time-limited evaluation version)	Low	No	No	Yes	Good
Keil MDK-ARM	Commercial (free size-limited evaluation version)	Low	No	No	No	Good
IAR EW for ARM	Commercial (free size-limited version and free time-limited evaluation version)	Low	Yes	No	Yes	Good

It is a good idea to study the example that is provided by Olimex for your board. If you are not willing to purchase the software just use the evaluation version initially.

Specifically, for STM32-P405 Olimex provides only IAR EW for ARM examples. It is a good idea to download the latest time-limited evaluation version.

### 3.3 Powering the board

---

The board can be powered in one of the following ways:

- 1) Using the JTAG or SWD debugger – the board gets powered via the 20-pin 0.1” step JTAG connector; note that some tool might hold the board in RESET state – make sure to release the microcontroller from RESET to be able to see the execution of your code
- 2) Using USB cable – the board gets powered via the USB connector; the power is provided, typically, by your computer's USB port
- 3) Using external power supply – in this case, the board gets powered via the VIN line – provide +6V DC to pin (pad) named “VIN”; then connect the power supply's ground line to a pin (pad) named “GND”.

The board has BAT\_3V connector which can be used to keep alive the RTC, external clock 32 kHz oscillator and backup registers (through power switch) when VDD is not present. BAT\_3V requires voltage in the 1.65V – 3.6V range! The BAT\_3V connector can not be used to fully power the board.

**IMPORTANT:** By default you can have either battery or external power supply connected (via the 6V DC barrel jack). If you connect both you might damage the board or the battery. If you need to keep the physical connection consider changing the default (closed) position of the BAT\_E jumper!

### 3.4 Programming the board

---

The programming of the board is also fairly straight forward (unlike the software development for the board).

#### 3.4.1 Using JTAG debugger

---

This is the recommended method of changing the software of the board. This is fairly simple from hardware point of view – simply plug the male 20-pin JTAG connector of your hardware tool to the 20-pin female JTAG connector of STM32-P405.

Refer to the documentation of your debugger and also to the documentation of the software of choice on instructions how to set the whole development environment.

#### 3.4.2 Using USB cable and bootloader software

---

The board can be programmed without a debugger tool. In this case you only need a personal computer that is able to run Windows software and an USB cable.

STM32-P405 has a built-in USB DFU bootloader that allows such programming.

Programming STM32-P405 via the bootloader is a slow process that requires the re-configuration of SMD jumper. It is not recommend for initial evaluation, research and development (use any ARM JTAG debugger instead). Using the bootloader might be good idea only if you need to program the board once or twice.

How to do it:

1. Change the position of one of the boot SMD jumpers – B0\_H/B0\_L (the bottom one). By default it is set to position B0\_L – you need to change it to B0\_H. This requires soldering.

2. After the adjustment is done you can connect to the DfuSe software released by STM. It available for download from STM's official web-site. Search for “STSW-STM32080”.
3. Install the software and run it.
4. Connect the board via the USB cable to the personal computer.
5. The board should now be recognized in the software. The bootloader application provided by STM32 is a bit strange. It works only with dfu files. To upload other binary files (hex, bin, s19) you would need to use additional converter tool (“DFU File Manager”; it converts hex, bin, s19 to dfu and vice versa). I highly recommend you to read carefully the document about the usage of the Dfuse called “Getting started with DfuSe USB device firmware upgrade”. Search for “UM0412”.
6. Note that once you've uploaded the software you would need to change the jumpers back to the original position – else the bootloader would start every time and the user program (typically stored in the flash memory) would not execute.

### 3.4.3 Using RS232 cable and bootloader software

---

The access to the serial bootloader is not implemented in STM32-P405. You can't access the bootloader of STM32F405 via the serial interface.

## 3.5 Prebuilt software

---

Upon powering the board the PWR LEDs should turn on and the STAT LED should blink fast 10 times.

The board comes with the software which was downloaded and used during the testing of the board. This software and its source code could not serve any purpose to the customer, who purchased the board, since it requires additional custom hardware. That is why there is a more meaningful example available for download at the web-page of the board. Please download and extract the following archive which contains the recommended initial demo:

[https://www.olimex.com/Products/ARM/ST/STM32-P405/resources/STM32-P405\\_example.zip](https://www.olimex.com/Products/ARM/ST/STM32-P405/resources/STM32-P405_example.zip)

Note that the demo can opened and compiled out-of-the-box ONLY with IAR Embedded Workbench for ARM. You can get a free code-size or time-limited evaluation version of IAR EW for ARM from their official web-site (requires registration). Refer to the “readme.txt” file inside the archive for more information.

There is also a separate CAN-specific demo available for download from this link:

<https://www.olimex.com/Products/ARM/ST/STM32-P405/resources/STM32-P405-CAN-DEMO.zip>

Again refer to the “README.txt” inside of the archive for more information about the demo.

---

## CHAPTER 4 THE STM32F405 MICROCONTROLLER

---

### 4. Introduction to the chapter

---

Some of the details about the main processor in the design of STM32-P405 (STM32F405RG) are mentioned in this chapter. The information is a modified version of the datasheet provided by its manufacturers from ST.

It is highly recommended to refer to the original datasheet which might be downloaded from the ST's official web-side.

#### 4.1 The STM32F405RG features

---

STM32-P405 uses STM32F405RG – a microcontroller designed and manufactured by STMicroelectronics. It has the following features that distinguish it from the rest of the family:

- Core: ARM 32-bit Cortex™-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 168 MHz, memory protection unit, 210 DMIPS/1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
- Memories
  - 1 Mbyte of Flash memory
  - 192+4 Kbytes of SRAM including 64-Kbyte of CCM (core coupled memory) data RAM
  - Flexible static memory controller supporting Compact Flash, SRAM, PSRAM, NOR and NAND memories
- LCD parallel interface, 8080/6800 modes
- Clock, reset and supply management
  - 1.8 V to 3.6 V application supply and I/Os
  - POR, PDR, PVD and BOR
  - 4-to-26 MHz crystal oscillator
  - Internal 16 MHz factory-trimmed RC (1% accuracy)
  - 32 kHz oscillator for RTC with calibration
  - Internal 32 kHz RC with calibration
  - Sleep, Stop and Standby modes
  - VBAT supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
- 3×12-bit, 2.4 MSPS A/D converters: 24 channels and 7.2 MSPS in triple interleaved mode
- 2×12-bit D/A converters
- General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
- Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 168 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input
- Debug mode
  - Serial wire debug (SWD) & JTAG interfaces
  - Cortex-M4 Embedded Trace Macrocell™
- 114 I/O ports with interrupt capability
- Up to 15 communication interfaces
  - 3 × I2C interfaces (SMBus/PMBus)
  - 4 USARTs/2 UARTs (10.5 Mbit/s, ISO 7816 interface, LIN, IrDA, modem control)
  - 3 SPIs (37.5 Mbits/s), 2 with muxed full-duplex I2S to achieve audio class accuracy via internal audio PLL or external clock
  - 2 × CAN interfaces (2.0B Active)

- SDIO interface
- Advanced connectivity
  - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
  - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
  - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
- 8- to 14-bit parallel camera interface up to 54 Mbytes/s
- True random number generator
- CRC calculation unit
- 96-bit unique ID
- RTC: subsecond accuracy, hardware calendar

For comprehensive information on the microcontroller visit the ST's web page for a datasheet.

At the moment of writing the microcontroller datasheet can be found at the following link:

<http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00037051.pdf>

---

## CHAPTER 5 CONTROL CIRCUITRY AND HARDWARE MODULES

---

### 5. Introduction to the chapter

---

Here you can find information about reset circuit and quartz crystals locations, the power supply circuit is discussed.

#### 5.1 Reset

---

The reset circuit is connected to pin #7 (NRST) of the STM32F405RG microcontroller. STM32-P405's reset circuit includes R8 (10 k $\Omega$ ), C28 (100nF) and a RESET button.

The RST signal can be found at an extension pin and has own row of pin holes at the prototype area – check for the pin named “RST”.

The reset is also available at pin #15 of the JTAG connector. Furthermore, you can connect RST and TRST of the JTAG together if you close the SMD jumper “R-T”.

The reset signal can be connected to pin #25 of the EXT1 header.

#### 5.2 Clocks

---

There are two quartz crystals available on the board:

8 MHz quartz crystal Q1 is connected to pins 5 and 6 of the ARM microcontroller.

Quartz crystal Q2 is a 32768Hz RTC and it is connected to microcontroller's pins 3 and 4.

#### 5.3 Power supply circuit

---

The power supply circuit of STM32-P405 allows the board to be fully powered from four locations.

- 1) The recommended location to power the board is the “PWR” barrel jack. You should provide between 6V DC and 12V DC. There is a diode rectifier at this power input so you can also power the board with 5V AC to 9V AC. The barrel jack is center positive.
- 2) The most used interface for powering is the USB type B connector. It works at +5V DC.
- 3) The board can also be powered via the on-board JTAG connector. In that case the board receives +3.3V DC from the debugger tool.
- 4) The board can be fully powered from the extension pins (or the pin holes at the prototype area). The best location to power the board would be the two pins at the bottom of the column of male pins – provide between +6V DC and +12V DC to the VIN pin. Double check the pin locations before providing the power. Important: VIN signal line is connected after the diode rectifier – do not provide negative voltage at VIN! Do not provide AC voltage to VIN!

If you want to use a back-up battery consider using the BAT\_3V connector. It works with batteries providing voltage in the (1.65 – 3.6)V range. This VBAT pin allows to power the device VBAT domain



from an external battery, an external supercapacitor, or from VDD when no external battery nor an external supercapacitor are present. VBAT operation is activated when VDD is not present. The VBAT pin supplies the RTC, the backup registers and the backup SRAM.

The board consumes around 0.25W (e.g. 0.04A at 6V) with both LEDs on. You can enable one of the low-power modes to reduce the consumption.

By default, if you have successfully powered the board its PWR LED would turn on. The STAT LED would blink 10 times. In case you you have modified the software (or if you have changed the positions of the jumpers responsible for the boot location) the STAT LED might not turn on at all since it is initialized by the code loaded on the flash memory of the board.

The red LED should always turn on upon powering. Note that it is possible to have the PWR LED on even if there isn't enough power for proper operation of the board and all the peripherals currently connected.

## CHAPTER 6 CONNECTORS AND PINOUT

### 6. Introduction to the chapter

In this chapter are presented the connectors that can be found on the board all together with their pinout and notes about them. Jumpers functions are described. Notes and info on specific peripherals are presented. Notes regarding the interfaces are given.

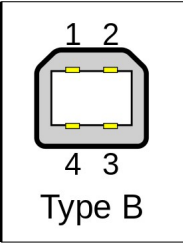
#### 6.1 USB connector

The USB type B connector allows you to access the USB functionality of STM32-P405. This connector is can be used to power the board. The connector is sturdy and typically lasts much longer during extensive use compared to the USB mini or USB micro connectors.

Note the DFU bootloader uses the USB port.

The signals follow the familiar and standard USB pattern:

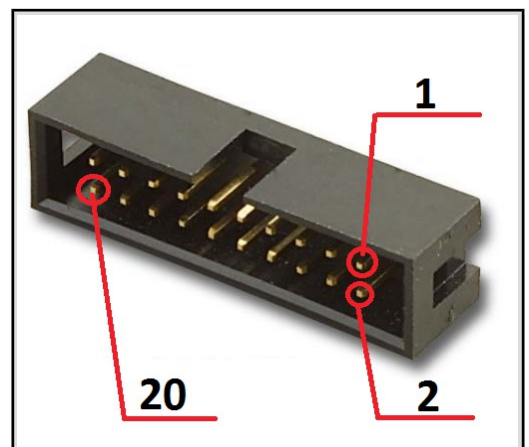
USB Type B connector	
Pin #	Signal name
1	+5V_USB
2	USBDM
3	USBDP
4	GND



#### 6.2 JTAG/SWD debug connector

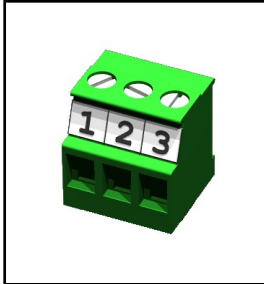
The board can be debugged from the 20-pin JTAG connector either by a debugger with compatible connector. The on-board JTAG connector has 0.1" step of its pins. Below is the table with the signals of the JTAG. This interface can be used with the Olimex's OpenOCD debuggers. It is also compatible with most of the ARM debuggers available on the market.

JTAG/SWD debug connector			
Pin #	Signal name	Pin #	Signal name
1	3.3V	2	3.3V
3	TRST	4	GND
5	TDI	6	GND
7	TMS	8	GND
9	TCK	10	GND
11	GND	12	GND
13	TDO	14	GND
15	RST	16	GND
17	GND	18	GND
19	GND	20	GND



## 6.3 CAN connector

The available CAN interface is compliant with the 2.0A and B (active) specifications with a bitrate up to 1 Mbit/s. The on-board CAN interface uses CAN bus transceiver SN65HVD230. The transceiver itself is connected to the main microcontroller's pins 61 (port PB8) and 62 (port PB9).



CAN connector	
Pin #	Signal name
1	GND
2	CANL
3	CANH

The CAN interface behavior can be influenced by the two jumpers CAN0\_T and CNTRL/HS.

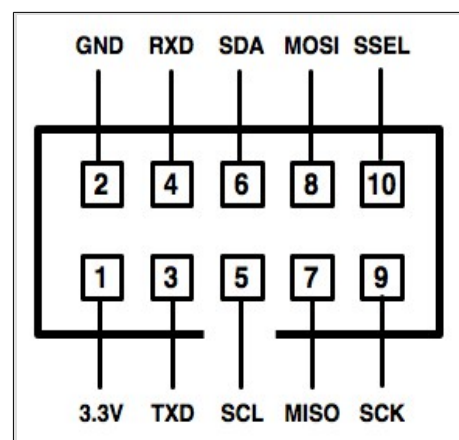
## 6.4 UEXT connector

STM32-P405 board has UEXT connector and can interface Olimex's UEXT modules. UEXT is a board to board connector which supports three serial communication interfaces – I2C, SPI and RS232. There is also a +3.3V output line and GND.

For more information on UEXT please visit:

<https://www.olimex.com/Products/Modules/UEXT/>

UEXT connector		
Pin #	Signal name	MCU pin #
1	3.3V	-
2	GND	-
3	USART1_TX	42
4	USART1_RX	43
5	I2C1_SCL	58
6	I2C1_SDA	59
7	SPI1_MISO	22
8	SPI1_MOSI	23
9	SPI1_SCK	21
10	SPI1_NSS	20

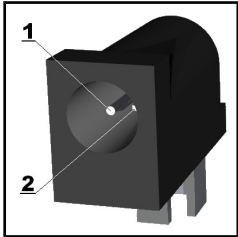


Note that the I2C lines have 10k pullup resistors. SPI1\_NSS also has a 10k pullup.

## 6.5 PWR\_JACK connector

The power jack used is the typical one used by Olimex in most of our products – the DC barrel jack has 2.0mm inner pin and 6.3mm hole. More information about the exact component might be found here: <https://www.olimex.com/wiki/PWRJACK>.

You should provide either (6 – 12)V direct current or (5 – 9)V alternating current. The required current may vary depending on the peripherals connected to the board. Do not provide more than 12VDC! Do not provide more than 9VAC!



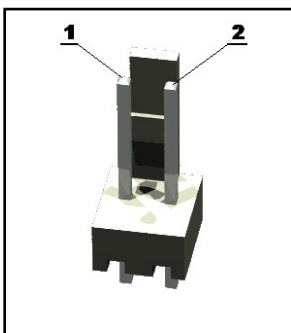
PWR_JACK connector	
Pin #	Signal name
1	Power input
2	GND

More info about the power supply can be found in chapter 4 of this manual.

## 6.6 BAT\_3V connector

When using the battery connector keep in mind that it is an energy solution that wouldn't be able to power the board and all the peripherals!

The board has BAT\_3V connector which can be used to keep alive the RTC, external clock 32 kHz oscillator and backup registers (through power switch) when VDD is not present. BAT\_3V requires voltage in the 1.65V – 3.6V range!



BAT_3V connector	
Pin #	Signal name
1	VBAT
2	GND

The pin polarity is also printed near the connector.

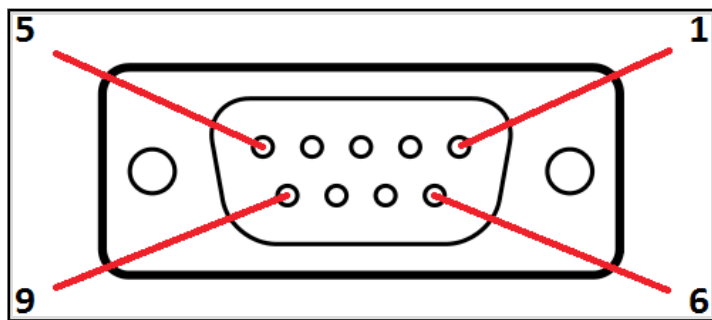
## 6.7 RS232\_2 connector

The STM32F405xx embeds four universal synchronous/asynchronous receiver transmitters (USART1, USART2, USART3 and USART6) and two universal asynchronous receiver transmitters (UART4 and UART5). The RS232\_2 connector gives access to USART2 with throughput of 5.25 Mbit/s.

By default RTS and CTS signals are disconnected. You can enable them by closing the jumpers RTS\_E and CTS\_E.

The circuit for the serial connection includes the ST3232 IC – it adjusts the external levels of USART to suitable levels.

RS232_2 connector	
Pin #	Signal name
1	Not connected
2	USART2_TX
3	USART2_RX
4	Not connected
5	GND
6	Not connected
7	UART2_RTS
8	WAKE-UP (CTS)
9	NC



## 6.8 SD/MMC card connector

SD/MMC slot is an SD card slot connector, located on the bottom of STM32-P405. It works with the “large” SD and MMC cards.

SD/MMC card connector			
Pin #	Connector signal name	Wire name	Microcontroller pin #
1	CD/DAT3/CS	SPI2_NSS	33
2	CMD/DI	SPI2_MOSI	36
3	VSS1	VSS (GND)	Any GND
4	VDD	SD_VCC1	Any 3.3V pin
5	CLK/SCLK	SPI2_SCK	34
6	VSS2	VSS (GND)	Any GND
7	DAT0/DO	SPI2_MISO	35
8	DAT1/RES	Pulled-up (held at logical 1)	Not connected
9	DAT2/RES	Pulled-up (held at logical 1)	Not connected

Additionally, there are the WP (write protect) and CP (card presence) switches that are responsible, respectively, for sensing whether the card is locked for reading, and whether there is a card inserted.

You can manipulate the availability of WP and CP by manipulating WP\_E and CP\_E jumpers. Disconnecting these jumpers would disable the WP and CP wires from the connector.

---

## 6.9 Extension pins and prototype area

---

At the right side of the board are is a column of pins. Each of them is connected to a signal – the name of the signal available at a pin is printed next to it. Each of these extension pins carry the same signal as the row of pinholes next to it. This prototype area provides very nice GPIO option – you can interface the signals exposed with your breadboarding wires; you can use them as headers; you can take signal readings; etc.

The distance between two adjacent pinholes is 0.1".

Refer to sub-chapter “2.3 Prototype area pinout (top view)” for visual representation of each pin near the connectors.

---

## 6.4 Jumper description

---

Please note that all of the jumpers on the board are SMD type. If you feel insecure in your soldering/cutting technique it is better not to adjust the SMD jumpers.

---

### 6.4.1 BOOT memory select jumpers

---

B1\_H/B1\_L and B0\_H/B0\_L are SMD jumpers which define the initial boot default priority of the microcontroller (aka which memory to boot from). By default they are set so that the board would boot from the user Flash memory.

Notice that these two jumpers must be moved together – they are responsible for the boot mode if bootloader is present. The board can search for bootloader on three places – either the user Flash memory, the System memory, or the Embedded SRAM.

By default, B0\_H/B0\_L is set to position B0\_L; the other jumper B1\_H/B1\_L is set to B1\_L – in this configuration the board will try to boot from user Flash memory.

To boot from the system memory you need to change the position of B0\_H/B0\_L jumper to position B0\_H. The boot loader is located in system memory (if you want to program the board via the USB and DFUse software – use this mode).

Changing both high positions (B0\_H/B0\_L to B0\_H; and B1\_H/B1\_L to B1\_H) would attempt to boot from the Embedded SRAM.

Refer to the “STM32 microcontroller system memory boot mode” document (“AN2606”) for more information about the bootloader available on STM32F4xx microcontrollers.

---

### 6.4.2 RTS\_E

---

Closing this jumper enables RTS line on the RS232\_2 connector.

The default position of jumper RTS\_E: open.

---

### 6.4.3 CTS\_E

---

Closing this jumper enables the CTS line on the RS232\_2 connector via the WAKE-UP button.

The default position of jumper CTS\_E: open.

---

---

#### 6.4.4 R-T jumper

---

If you close this jumper RST and TRST at the JTAG will be connected together.

The default position of jumper R-T: open.

6.4.4 CNTRL/HS  
The RS pin (pin 8) on the SN65HVD230 provides three different modes of operation: high speed mode, slope control mode, and low-power mode. CNTRL/HS jumper allows you to select a mode. The jumper has three valid states – either both sides open; either the pads of CNTRL position connected; or the pads of HS position connected. It is illegal to have both CNTRL pads soldered together and HS pads soldered together. Each of the valid cases is described below:

- Both sides of CNTRL/HS jumper open – 10kΩ resistor is connected to the control pin of SN65HVD230 CAN driver – with a resistor value of 10kΩ the device will have a slew rate of ~15 V/μs
- HS pads soldered together – high speed of output CAN drivers (>20v/uS) – no slope control
- CNTRL/HS is connected to CNTRL side – enable PC10 (pin 51) of STM32F405 to control CAN driver modes. Logical 1 of PC10 disables the CAN driver. Logical 0 of PC10 enables CAN driver with high speed mode.

Refer to the documentation of SN65HVD230 for more information about different modes.

The default state of jumper CNTRL/HS: both sides open (slope control mode).

---

#### 6.4.5 CAN0\_T

---

Control whether there is a terminating resistor between the CAN buses. Closing the jumper connects 120Ω terminating resistor between CAN\_L and CAN\_H.

The default position of jumper CAN0\_T: closed.

---

#### 6.4.6 LED\_E

---

Opening this jumper will disable the STAT LED.

The default position is closed.

---

#### 6.4.7 USBP\_E

---

Opening this jumper would disable the option to power the board from the USB connector.

The default position is closed.

---

#### 6.4.8 WP\_E jumper

---

Controls the write protect pin of the MMC card connector. If you disconnect this jumper the WP would be disconnected from the SD/MMC card connector.

The default position of the jumper is closed – write protect pin (microcontroller pin #37 – PC6) is available at the card connector.



### 6.4.9 3.3V\_MCU\_E jumper

---

Connects 3.3V regulated voltage to STM32F405 power pins. The 3.3V\_MCU\_E jumper is usually used to measure the current consumption of the microcontroller. It is not recommended to disconnect this jumper.

The default state of the jumper is closed.

## 6.5 Additional hardware components

---

The hardware components below are mounted on STM32-P405 but are not discussed above. These are listed here for completeness:

**RESET button** – used for hardware reset of the board; typically cannot be re-programmed; connected to STM32F405's pin #7 (NRST);

**WAKE-UP button** – customizable button; can be re-programmed by the user – connected to STM32F405's pin #14 (PA0); also available at the first pad of jumper CTS\_E;

**PWR LED** – indicates when the board is powered; cannot be re-programmed; it is not connected to the main microcontroller;

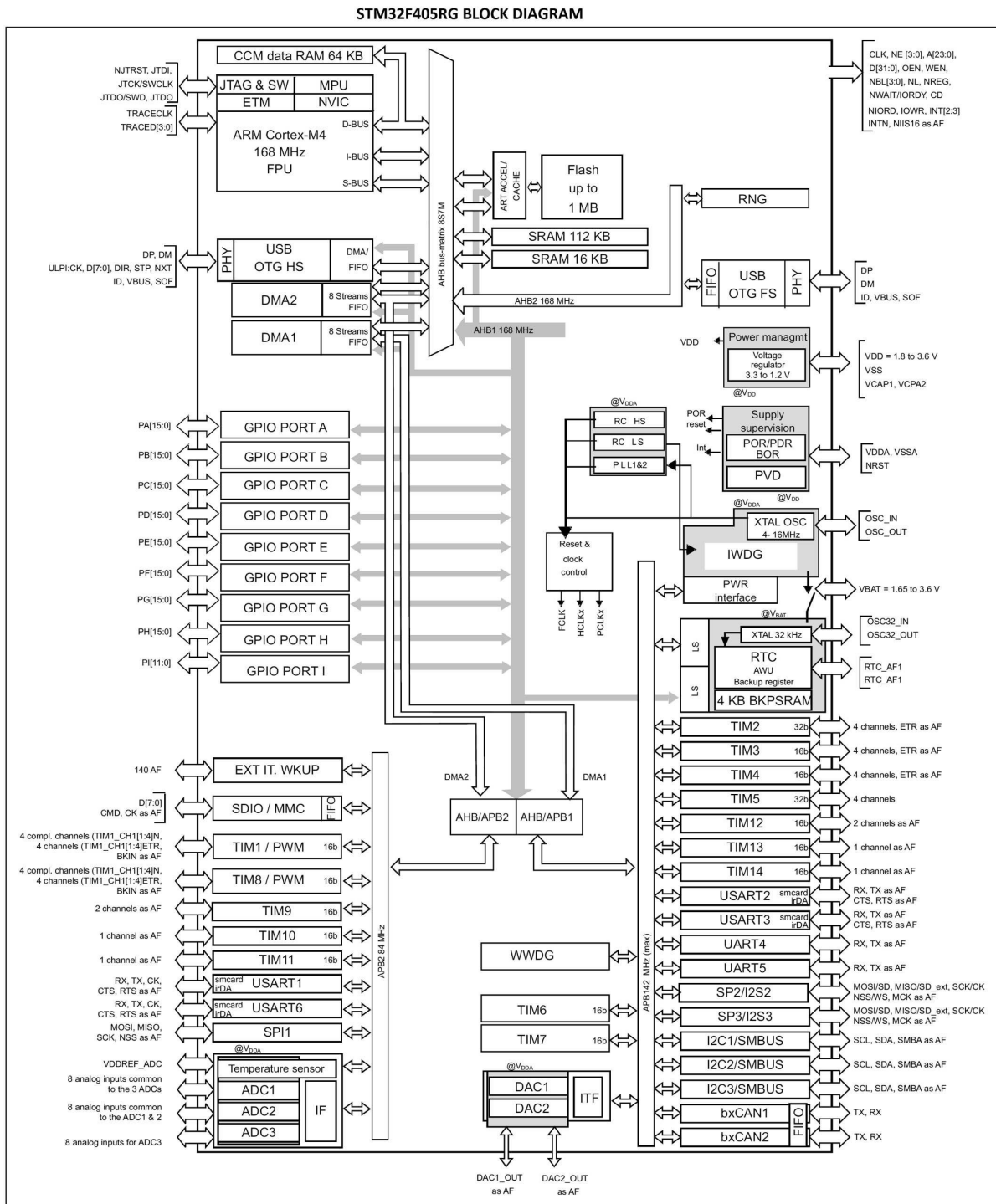
**STAT LED** – customizable LED; can be re-programmed by the user – connected to STM32F405's pin #53 (PC12); also available at male header of the board and the prototype area; this LED can also be disabled if you disconnect jumper LED\_E; the jumper pads can also be used for measurements.

# CHAPTER 7 BLOCK DIAGRAM AND MEMORY

## 7. Introduction to the chapter

On the next page you can find a memory map for this family of processors. It is strongly recommended to refer to the original datasheet released by STMicroelectronics for one of higher quality.

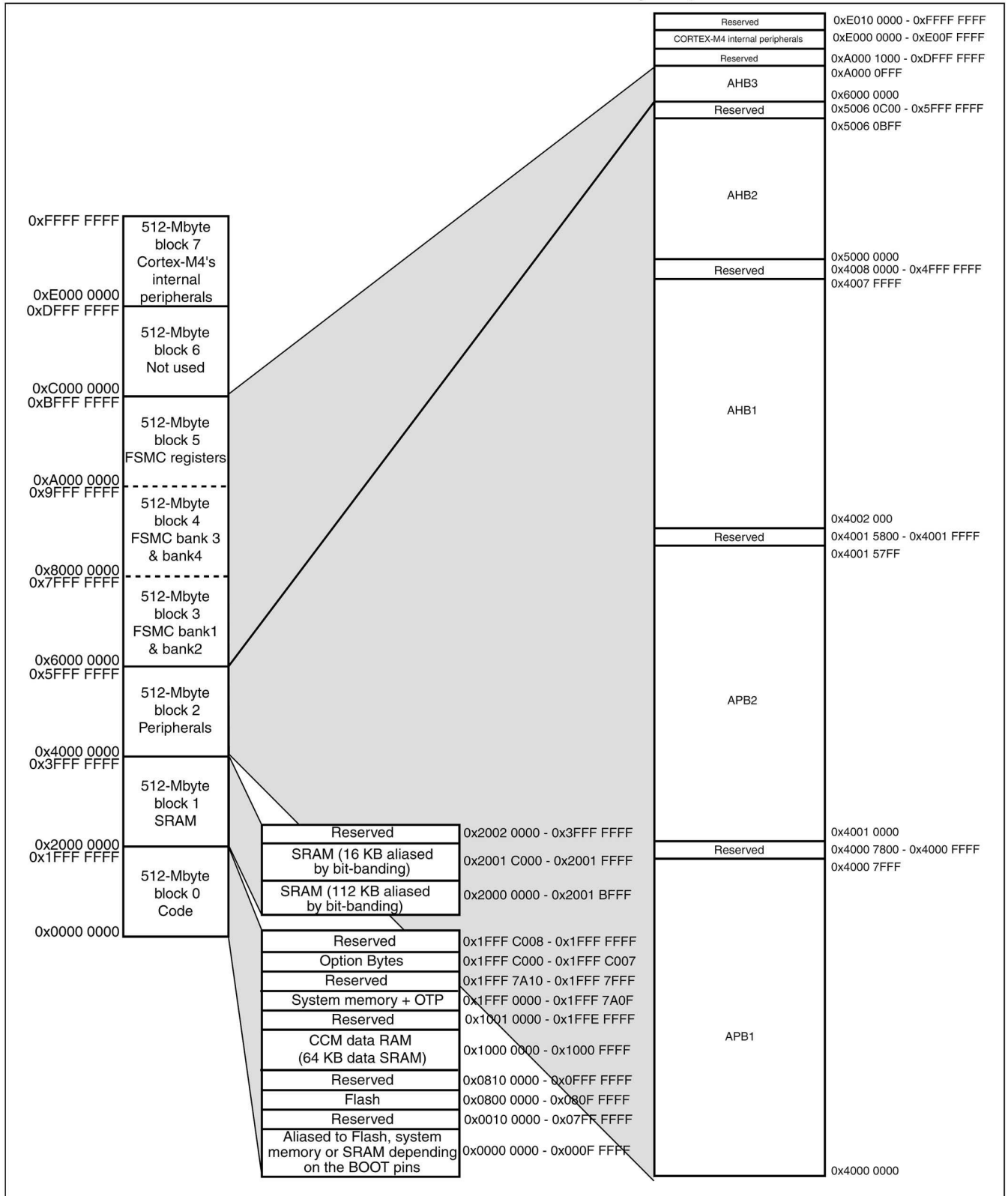
### 7.1 Processor family block diagram



## 7.2 Physical memory map

STM32F405RG has 1024Kbytes of flash memory and (192 + 4)Kbytes of SRAM. Inspecting the map below don't forget that all STM32F405YY microcontrollers have neither Ethernet controller nor camera interface. STM32F405RG also lacks FSMC memory controller.

STM32F40xxx memory map



## CHAPTER 8 SCHEMATICS

---

### 8. Introduction to the chapter

---

In this chapter are located the schematics describing logically and physically STM32-P405.

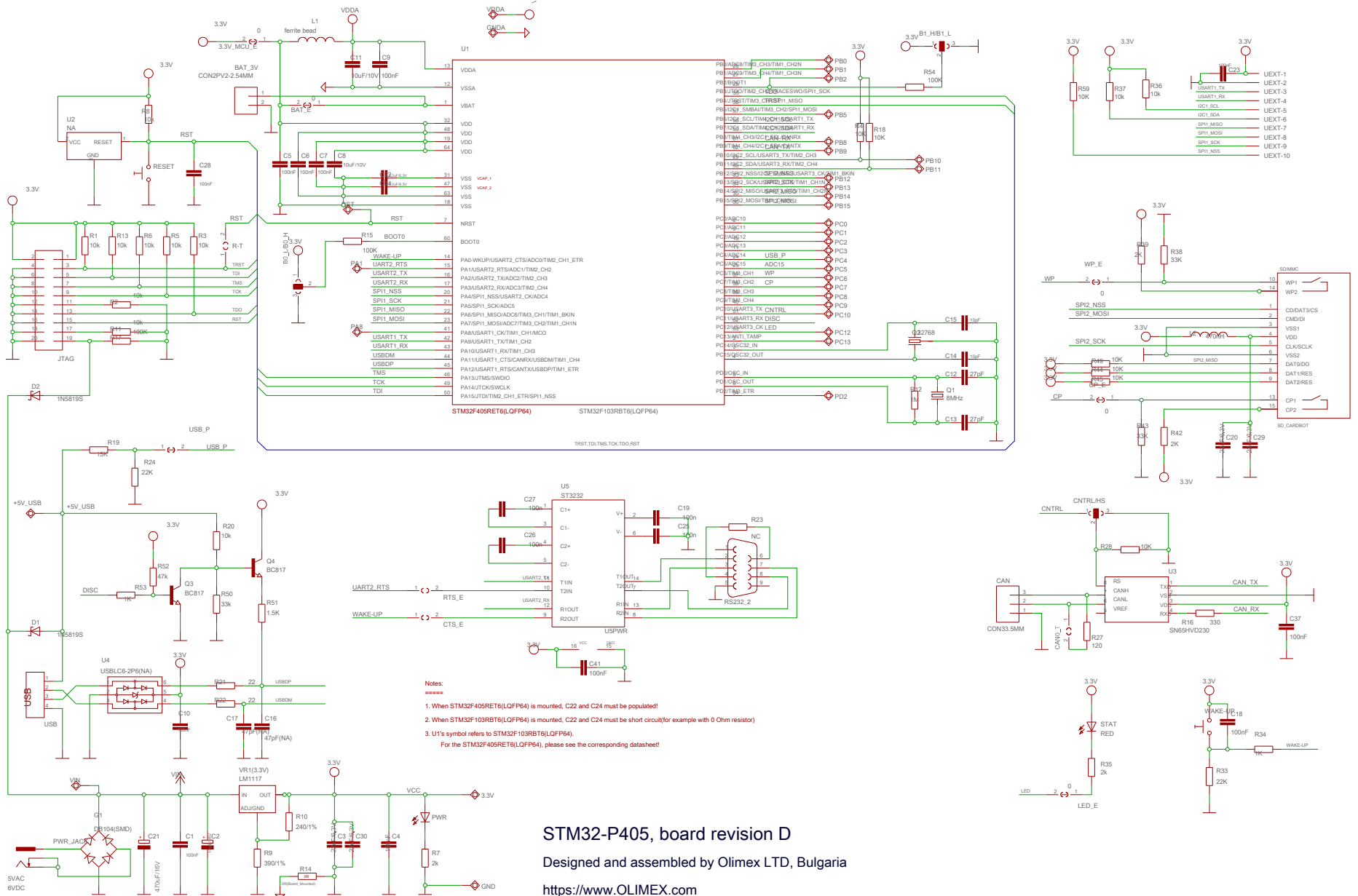
#### 8.1 Eagle schematic

---

STM32-P405 schematic is visible for reference here. You can also find them on the web page for STM32-P405 at our site: <https://www.olimex.com/Products/ARM/ST/STM32-P405/>

They are located in HARDWARE section.

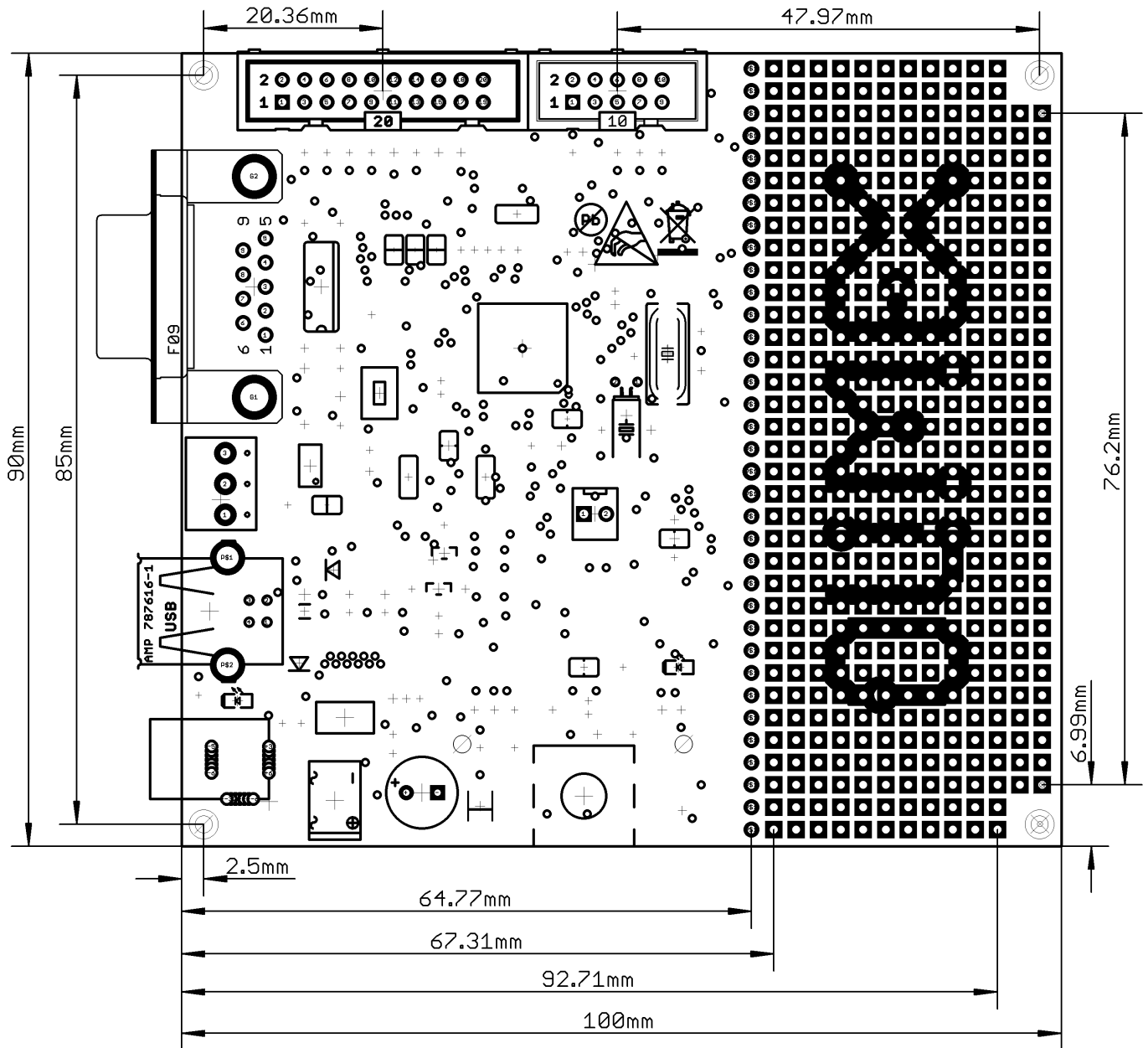
The EAGLE schematic is situated on the next page for quicker reference.



STM32-P405, board revision D  
 Designed and assembled by Olimex LTD, Bulgaria  
<https://www.OLIMEX.com>

## 8.2 Physical dimensions

Note that all dimensions are in millimeters.



---

## CHAPTER 9 REVISION HISTORY AND SUPPORT

---

### 9. Introduction to the chapter

---

In this chapter you will find the current and the previous version of the document you are reading. Also the web-page for your device is listed. Be sure to check it after a purchase for the latest available updates and examples.

#### 9.1 Document revision

---

Revision, date	Changes	Modified pages
A, 04.12.15	Initial release	All

#### 9.2 Board revision

---

Revision, date	Revision notes
D, 21.02.14	Initial release



### 9.3 Useful web links and purchase codes

---

The web page you can visit for more info on your device is:

<https://www.olimex.com/Products/ARM/ST/STM32-P405/>

The official Olimex forums are located here: <https://www.olimex.com/forum>

#### **ORDER CODES:**

**STM32-P405** – the version of the board discussed in this document

**STM32-H405** – a similar board with less peripherals

**STM32-405STK** – a similar board with even more peripherals

**ARM-USB-TINY-H** – OpenOCD compatible high-speed debugger/programmer with JTAG interface

**ARM-USB-OCD-H** – OpenOCD compatible debugger/programmer with JTAG interface, protection buffers and better power supply circuit

How to order?

You can order directly from our web-shop or via any of our distributors. List of Olimex distributors may be found here: <https://www.olimex.com/Distributors/>

**Please check <https://www.olimex.com/> for more information.**

## 9.4 Product support

For product support, hardware information and error reports mail to: [support@olimex.com](mailto:support@olimex.com). All document or hardware feedback is welcome. Note that we are primarily a hardware company and our software support is limited. Please consider reading the paragraph below about the warranty of Olimex products.

All goods are checked before they are sent out. In the unlikely event that goods are faulty, they must be returned, to OLIMEX at the address listed on your order invoice.

OLIMEX will not accept goods that have clearly been used more than the amount needed to evaluate their functionality.

If the goods are found to be in working condition, and the lack of functionality is a result of lack of knowledge on the customers part, no refund will be made, but the goods will be returned to the user at their expense.

All returns must be authorized by an RMA Number. Email [support@olimex.com](mailto:support@olimex.com) for authorization number before shipping back any merchandise. Please include your name, phone number and order number in your email request.

Returns for any unaffected development board, programmer, tools, and cables permitted within 7 days from the date of receipt of merchandise. After such time, all sales are considered final.

Returns of incorrect ordered items are allowed subject to a 10% restocking fee. What is unaffected? If you hooked it to power, you affected it. To be clear, this includes items that have been soldered to, or have had their firmware changed. Because of the nature of the products we deal with (prototyping electronic tools) we cannot allow returns of items that have been programmed, powered up, or otherwise changed post shipment from our warehouse.

All returned merchandise must be in its original mint and clean condition. Returns on damaged, scratched, programmed, burnt, or otherwise 'played with' merchandise will not be accepted.

All returns must include all the factory accessories which come with the item. This includes any In-Circuit-Serial-Programming cables, anti-static packing, boxes, etc.

With your return, enclose your PO#. Also include a brief letter of explanation of why the merchandise is being returned and state your request for either a refund or an exchange. Include the authorization number on this letter, and on the outside of the shipping box.

Please note: It is your responsibility to ensure that returned goods reach us. Please use a reliable form of shipping. If we do not receive your package we will not be held liable.

Shipping and handling charges are not refundable. We are not responsible for any shipping charges of merchandise being returned to us or returning working items to you.

The full text might be found at <https://www.olimex.com/wiki/GTC#Warranty> for future reference.