# Stm32f407vet6 development board schematic pdf

Sale! Rs 3,640.00 Rs 3,390.00 STM32F407'ET6 развития борту M4 STM32F4 основной совет по разработке руки коры-M4 1 на складе STM32F407VET6 развития борту M4 STM32F4 основной борту руку борту борту коры-M4. This board has more IOs, more interfaces and definitely better than the STM32F4Discovery board. Processor: STM32F407-ET6 , Cortex-M4 32-bit RISC Features: single-call DSP Instructions Operating frequency: 168 MHz 210DMIPS / 1025 DMIPS / MHW Processor Operating Voltage: 1,1,. 8-3.6V CPU Package: L'F100 Resources Storage Resources 512kB Flash 192 - 4kb SRAM - on board Flash In Port Programming System (UART) WIRELESS Connector Interface for NRF24L01 Modules 3x Custom 1x Button 3x 2x LEDs 1x Rab USB 2.0 Interface 2x 48-pin GPIO Headers 1x FMSC TFT Display Interface Port 1x On-Board W25'16 SPI Data Flash 1x Micro-SD Interface RTC Battery Backup JTAG/SWD The interface for programming and debugging on-board devices and interfaces Documents and downloads Package includes 1x STM32F4 ARM Cortex M4 Processor Core Board - STM32F407VET6 (KK-OK) 1x USB cable 10x women to female prototyping cables Keyword Key Words : Better than the STM32F4DISCOVERY Development Board, STM32F4 ARM Cortex M4 Core Board Development Processor - STM32F407VET6 Pakistan ×Sort interrupt errors and questions about boards that may work MicroPython but do not have a special forum. Target audience: All interested in launching MicroPython on other hardware. Mcauser Post: 482 Posted by: 15, 2015 8:03am P1 pm to feb 27, 2017 2:59 pm I created a stm32 board of definitions for these two MCUDev boards from China, tagged STM32F407X. For lack of a better name, I just called them BLACK STM32F407xx. Originally located at: ... -stm32f407 Now, moved to their own REPO: New narrower version: VCC GND boards: They are very similar boards with major differences in CPU size (more io pins), led by the color and rearrange components. STM32F407VET6 ... 22721.html $11.50 USD Brand: MCU Dev Markings: STM32F4XX STM32_F4VE V2.0 1509 Specs: STM32F407VET6 ARM Cortex M4 168MHz, 210 DMIPS / 1.25 DMIPS / MHz 1.8V - 3.6V operating voltage 8 MHz crystal system 32,768 KHz RTK Crystal 2.54 mm step pins JTAG / SWD headline 512KByte Flash, 192 and 4Byte SRAM 3x SPI, SPI, 3x USART, 2x UART, UART 2x I2S, 3x I2C 1x FSMC, 1x SDIO, 2x CAN 1x USB 2.0 FS/HS controller (with DMA dedicated) 1x USB HS ULPI (for external HS PHY) Micro S D Winbond W25'16 16Mbit SPI Flash RTC батарея CR1220 1x 10/100 Ethernet MAC 1x 8 до 12-разрядный интерфейс Параллельной камеры 3x ADC (12-разрядный (12-разрядный 16-channel) 2x DAC (12-bit) 12x common timers, 2x advanced timers AMS1117-3.3V: 3.3V LDO voltage control, max current 800mA Micro USB power and comms Red Power LED D1 Red User LED D2 (PA6) active low red user LED D3 (PA7) active low 2x jumper to download a selection of reset button, Wake button, 2x custom buttons K0 (PE4) and K1 (PE3) 2x24 side pins 2x16 bottom pins and 1x4 ISP pins 2x16 FMSC LCD interface NRF24L01 Nest M3 mounting holes Dimensions: 85.1 mm x 72.45 mm STM32F407-16 ... 62341.html $14 USD Brand: MCU Dev Markings: STM32F4XX STM32_F4XX V3.0 1606 Specs: STM32F407-ET6 ARM Cortex M4 168 MHz, 210 DMIPS / 1.25 DMIPS / MHz 1.8V - 3.6V operating voltage 8 MHz system crystal 32,768 KHz RTK Crystal 2.54 mm step pins JTAG / SWD headline 512KByte Flash, 192 y 4Byte SRAM 3x SPI, SPI, 3x USART, 2x UART, 2x I2S, 3x I2C 1x FSMC, 1x SDIO, 2x CAN 1x USB 2.0 FS/HS controller (with dedicated DMA) 1x USB HS ULPI (for external USB HS PHY) Micro SD Wind W25'16 16Mbit SPI Flash RTC battery CR1220 1MB SRAM Trail, uninhabited (IS622WV51216-1M) 1x 10/100 Ethernet MAC 1x 8 to 12-bit Interface Parallel Camera 3x ADC (12-bit/16-channel) 2x DAC (12-bit) 12x common timers , 2x advanced timers AMS1117-3.3V : 3.3V LDO voltage control, max current 800mA Micro USB power and comms yellow user LED D1 (PF9) active low yellow user LED D2 (PF10) active low yellow power LED D3 2x jumper to download choice of restart button, Wake button, 2x custom buttons K0 (PE4) and K1 (PE3) 2x30 side pins - 2x16 bottom pins - 1x4 ISP pins 2x16 FMSC LCD Interface NRF24L01 socket M3 mounting holes Dimensions: 95.. 1mm x 74.6mm They also look like the VCC-GND STM32F407VET6 board I previously added: Installed MicroPython v1.8.7-333 on both using ST-Link V2 (clone). ... 77845.html $2 USD Installation Steps: ST-Link V2 - JTAG Connector: Code: Select allSWDIO --- Pin 7, PA13/TMS/JTMS_SWDIO GND ----- Pin 4, GND SWCLK --- Pin9, PA14/TCK/JTMS_SWCLK 3.3V ---- Pin 1, 3V3 JTAG pinout connector: Code: Choose ----- all 1 to 2 inches Pin 1 and 3v3 3 to 4 inches Pin-4 - GND 5 to 6 inches q 7 to 8 inch Stift 7 - SWDIO 9 to 10 Degrees Pin-9 - SWCLK No. 11 12 13 14 Connect ST-Link to usb port-----, making sure there are no other power sources to the board. Terminal 1 Terminal 2 (replace F407VE with F407'E for a larger board) Code: Select allcd stm32 do BOARD-BLACK_F407VE hand-not-eabi-gdb-build-BLACK_F407VE/firmware.elf (gdb) target extended localhost:4242 (gdb) load (gdb) exit Terminal 1 Look for Flash written and verified! Hilarious good! Control to exit st-util ST-Link, connecting Mini-USB to your computer and connecting to a board. Code: Select allscreen/dev/tty.usbmodem1422 MicroPython v1.8.7-333-gfb1e7e2-dirty for 2017-02-27; BLACK STM32F407VE STM32F407VE STM32F407VE Type Help () for more information. It works! Progress: Toggling LEDs (x) Flashing LEDs with timers (x) Toggling IO pins (x) Reading values ADC (only one job - K0) - I2C is a timeline for scanning on all buses () All 5 or 6 UARTs, seems to be running the installation of the RTC random number generator (x) - DAC - SPI - W25'16 SPI Flash - Servo - NRF24L01 - Add SRAM to the back of the board OF THE6 Last edited mcauser on Wed July 17, 2019 7:28 am, edit 4 times in total. SpotlightKid Messages: 420 Joined: St. Apr 08, 2015 5:19am Post SpotlightKid Mon Feb 27, 2017 3:07pm PST I booked the VET6 board too three weeks ago, but the tracking information says the package hasn't even left China yet mcauser Messages: 482 Joined: Mon June 15, 2015 8:03am Post Mcauser' Mon February 27, 2017 3.52pm Mine took 33 days to get delivered to Australia. Chinese New Year celebrations have caused some of my orders to get delayed. lmamakos Posts: 3 Joined: Sun January 22, 2017 6:37am Post lmamakos Thu Mar 02, 2017 4:25am I got a pair of VET6 black boards and hacked configuration for them too. I was able to get a nRF24L001 SPI connector to work between two boards, having them talk to each other running the driver test settings in the tree source. Also, I was able to talk to the I2C SSD1306 display works as well. Good/bad news - I also get SD cards functional, but this board apparently doesn't have a card currently available. So I had to add #ifdef to several places in stmhal/sdcard.c to avoid the initialization of the GPIO pin if the map of this signal was not available, and always go back true to the sdcard_is_present () function, and relying on I/O to fail. It seems to work just fine! Also, I was able to successfully load the micropython into the board over THE USB using a dfu-util which was a bit of a time-saver. Kid06 Posts: 14:30 Pm 03, 2017 4:21 pm To 3:21 PM Post from 03 P 03, 2017 4:30pm I confirm STM32F407VET6 is running normally. Composite MicroPython's Docker (ized) Ubuntu on OrangePI ONE. Downloaded firmware.dfu via dfu-util. Tested servo on PA0 (-----gt;X1, pyb. servo (1)) and PA2 (-----gt;X3, pyb. servo (3)). I see some reaction, but totally wrong. This is my day 0 at MicroPython. Looks very promising.... Kid06 Posts: 14:37 Pm: Fri Mar 03, 2017 4:21 pm By kid06 Sat Mar 04, 2017 10:37am RTC exibits reset the default time on a power cycle. Can someone confirm? Kid06 Posts: 14 Posted by Fri Mar 03, 2017 4:21pm Post time: 16 Sat Mar 04, 2017 4:21pm 5:40 p.m. I did some more investigations. pyb.delay (nnn) seems 3x slower, for example, as 3xnnn I use lmamakos source patch. Maybe the problem is time is the problem with the servo. Kid06 Posts: 14:03 Pm 4:21 PM To 4:21 PM Post: 6:44 pm Et on Aug 14, 2017 6:44 pm Ok, as I assumed, wrong timing. @lmamakos, there is a wrong definition in your your Line 104 file, the external OSC frequency should be 8 MHz not 25 (at least on my board). After changing the servo and delay is normal. RTC has not yet been tested. Thanks kid06 Messages: 14 Joined: Fri Mar 03, 2017 4:21pm Post by kid06 Sat Mar 04, 2017 9:41pm Well, RTC seems to have serious problems: Code: Choose all of rtc and machine. RTC () Traceback (последний звонок последний): Файл <stdin>, линия 1, в <module>AttributeError: 'модуль' объект не имеет атрибута 'RTC' >>> machine.freq() (168000000, 16800000, 42000000, 840000000) >>> machine.unique_id() b'G'x001'x00'x16'606478' ... There is no RTC in machine fashion... Using low-cost STM32F407VET6 boards with mbed. It provides an affordable (about $9 on eBay) and a flexible way for users to try out new ideas and build prototypes. The board is equipped with a STM32F407VET6 microcontroller compatible with the Seeed Arch Max platform. The microcontroller has the STM32F407VET6 in the ARM package®32-bit ARM®32-bit Cortex®-M4 and FPU 168 MHz maximum VDD processor frequency from 1.8 V to 3.6 V 512 KB Flash 1 92'1 4 Kbytes SRAM including 64-Kbyte CCM (basic connected memory) data RAM GPIO (82) with external interruption capabilities 12-bit ADC (3) with 16 channels 12-bit DAC (1) with 2 channels RTC Timers (3) with 16 channels 1 2-bit DAC (1) with 2 channels RTC Timers (3)14) I2C (3) Interfaces (SMBus/PMBus) I2S (2) USART (4) SPI (3) USB 2.0 full-cost USB 2.0 OTG CAN (2) Tip has JTAG/SWD headline Micro SD Winbond W25 6 16Mbit SPI Flash RTC Battery CR1220 10/100 Ethernet MAC 3.3V LDO Voltage Control Mini USB Connector 1x Power LED 2x user LEDs D2 (PA6) D3 (PA7) 2x jumper to download select reset button , The Awakening button 2x custom buttons K0 (PE4) and K1 (PE3) 2x24 side pins - 2x16 bottom pins - 1x4 ISP contacts 2x16 FMSC LCD Interface NRF24L01 pinout connector to enhance in zoom in More details on contact definitions see Table 9 in the data table and reference guide. Scheme View scheme. Using the online compiler Mbed Create the program as if it were for the Seed Arch Max board (choose Seeed Arch Max as a target platform for an online compiler). Programming STM32F407VET6 board NUCLEO ST-LINK/V2-1 and Drag and Drop You can use the NUCLEO virtual drive for the STM32F407VET6 board program (drag and drop programming). This requires an additional NUCLEO board (any type equipped with ST-LINK/V2-1). Remove two jumpers from the CN2 connector, As shown in Figure 8: Connect the NUCLEO connector to the STM32F407VET6 connector using flying wires as follows: NUCLEO board CN4 connectors TM32F407VET6 SWD SWD'LT attention that VDD_TARGET is not connected. This works with an ST-Link programmer, but it can potentially damage the target microcontroller if it works at a lower voltage (e.g. 2V5) than a programmer (e.g. 3V3). That's why it's also recommended to connect the VDD_TARGET line when an external programmer, such as Segger J-Link, is connected. to program the board. Provide power for the STM32F407VET6 board via 3.3V pin, 5V pin or USB cable. (VDD_TARGET on the NUCLEO CON4 board does not work as a power source). Connect the NUCLEO board to your computer via a USB cable. To program the STM32F407VET6

board, click on the Compilation button and save the binary to the NUCLEO virtual disk. For more information, take a look at the User's Guide, Chapter 6.2.4 Using ST-LINK/V2-1 for the program and debugging the external STM32 application. ST-Link V2 USB key and STM32 ST-LINK utility If you want to use the ST-Link V2 USB key (aka ST-Link V2 Programming Group) to program the board to apply the same wiring as above. If not yet done, install the ST-Link/V2 driver on your computer. Connect the ST-Link V2 key to your computer. Then click on the Compilation button and save the binary file on the local drive. Install and launch the STM32 ST-LINK. Once the program runs open binary is built with an online compiler and click on the program verification button. The STM32 USART and Flasher-STM32 memory loader look at the built-in STM32 loader. Download: FLASHER-STM32 For more information read: Note AN2606 app. Using a serial port (not just for debugging) Connect a FTDI or similar USB to a serial TTL converter to your computer and to a serial port on board (e.g. PC_6, PC_7 - NOTE: Connect to these contact devices also when printing to THE USART by default with a simple printf command). Make sure you plug on board the TX pin for the RX converter pin and board the RX pin for the TX converter pin. In the code, create a serial object (using TX and RX pins of a connected sequential port). Use the printf function to send serial messages to a connected computer. In case you would like to rid the external USB serial converter for other purposes, then there is an alternative solution offered by X M (bitman). You can also use the ST-Link virtual comport to debug programs running on the STM32F103C8T6 board. However, this will require solder iron (and probably some solder skills). According to the User Guide, Chapter 6.8 USART Communications, solder bridges (on the back of the NUCLEO board) SB62 and SB63 should be ON, SB13 and SB14 should be off. In this case, you can connect another USART to the CN3 NUCLEO (ST-Link) connector using flying wires. For example, on the STM32F103C8T6, you can use USART2, available on PA_2 (TX) and PA_3 (RX). Two flying wires must be connected as follows: STM32F407VET6 board, contact PC_6 (Serial2 TX) qlt"gt;NUCLEO board CN3 connector, contact RX STM32F407VET6 board, contact PC_7 (Serial2 RX) Nothing Special makes even solder Meaning is to redirect UART to the NUCLEO board using software (without changing the solder bridges on the back of the NUCLEO board) NUCLEO) turn it into a debugger. On the NUCLEO board that you're going to use as a programmer/jack, select any serial port other than Serial2 (except the default port used for standard UART) to be initiated as a standard UART. In the program below (using NUCLEO-F103RB as a programmer/fucker) was selected Serial1 (PA_9, PA_10) was selected. #include mbed.h / Declarations needed to change the external parameters of the UART stdio serial_t stdio_uart; extern int stdio_uart_inited; int main () - serial_init (No stdio_uart, PA_9, PA_10); Except Serial2 stdio_uart_inited No 1; printf (Ready to debug); After compiling (don't forget to select the NUCLEO board used for programming/debugging as a target for the online compiler), download the Debugger program to the NUCLEO board. Please make sure you have two jumpers in place on the CN2 connector while programming the NUCLEO board. Once the binary Debugger has been loaded onto the NUCLEO board, remove the two jumpers again. The Ethernet interface there are affordable affordable (about $2.50 on eBay) LAN8720 small footprint RMII 10/100 Ethernet modules suitable for the STM32F407VET6 board. For more information, see the LAN8720 data table. We can connect the LAN8720 module to the black board of the STM32F407VET6 as follows: LAN8720 moduleSTM32F407VET6 board TX1'lt; PB_13 TX_EN'lt; PB_11 the PB_12 of the RX0'lt'PC_4 RX1'lt'PC_5 nINT/nINT/nINT/RETCLK'lt; PA_1 CRS'lt'PA_7 MDIO'lt; PA_2 MDC'PC_1 lt Note: that because the line RX_ER not used the LAN8720 module should not be changed. No modifications to MBED libraries are required. However, please note that since the pin-PA_7, which is the driving force on board the LED2 (D3), is also used by the Ethernet interface (like the CRS) the LED will emit light. Demonstrations for the Ethernet TCPSocket WebSwitch CAN peripheral interface Unfortunately, no CAN interface configuration is available for the Seeed Arch Max board in mbed. However, we can easily add such by changing the mbed-dev library as described below. Imports CAN_Hello in an online compiler. Choose Seeed Arch Max as a target platform for an online compiler. Open and change the main.cpp as follows: ... #if (TARGET_STM32F103C8T6) #include stm32f103c6.h #define LED_PIN PC_13 const int off No 1; const int ON No 0; #else //#define LED_PIN LED1 //const int OFF No 0; const int ON No 1; #define LED_PIN PA_6 const int OFF No 1; const int ON No 0; #endif ... Connect the STM32F407VET to the CAN bus, as shown in the Schematic CAN_Hello. Remove the mbed library from the project. Import library mbed-dev into the project. Open the file и вставьте следующий код в конце файла незадолго до директивы C preprocessor #ifdef __cplusplus: typedef enum - CAN_1 (int)CAN1_BASE, CAN_2 ( int)CAN2_BASE int)CAN2_BASE Откройте файл mbed-dev/targets/TARGET_STM/TARGET_STM32F4/TARGET_STM32F407xG/TARGET_ARCH_MAX/PeripheralPins.c и прикрепите следующий код: const PinMap PinMap_CAN_RD PA_11 CAN_1, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN1) , PB_8, CAN_1, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN1) , PD_0, CAN_1, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN1) , PB_5, CAN_2, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN2) , PB_12, CAN_2, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN2), NC, NC, 0; const PinMap PinMap_CAN_TD, PA_12, CAN_1, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN1), PB_9, CAN_1, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN1) (PD_1, CAN_1, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN1) , PB_6, CAN_2, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN2) , PB_13, CAN_2, STM_PIN_DATA (STM_MODE_AF_PP, GPIO_NOPULL, GPIO_AF9_CAN2) , NC, NC, 0 Добавьте файл mbed_app.json со следующим содержанием в проект: «target_overrides»: «ARCH_MAX»: «target.device_has»: «CAN» » » » » » Компиляция проекта и загрузка двоичного файла на доску STM32F407VET6. Затем продолжить, как объясняется в CAN_Hello демо. Слот для карт Micro SD Доска оснащена слотом для микро-SD-карт , который подключен к интерфейсу SDIO микроконтроллера. К сожалению, MBED не поддерживает такой тип периферийных устройств. Тем не менее, это еще можно использовать на борту микро SD-карты слот. Взгляните на программу STM32F407VET6_SDCard, чтобы увидеть, как это сделать. На борту установлена на борту 16Mbit SPI Flash память Winbond W25-16 16Mbit SPI Flash. Смотрите пример STM32F407VET6_SPIFlash, как его использовать. NRF24L01 разъем nRF24L01 Мастер (Transmitter) nRF24L01 Раб (приемник) Дополнительные примеры программ Mbed OS-5 Благодаря Иоганнес Стратманн теперь мы можем создавать также Mbed OS-5 проектов для Seeed Arch Max борту с Mbed CLI, Mbed Studio и Mbed Online Компилятор. Для проектов Mbed OS-5 можно использовать также модуль DP83848 в качестве интерфейса Ethernet, а не модуль LAN8720. Последний не работает по неизвестной причине. EDIT: Посмотрите на как сделать LAN8720 работать с Mbed OS-5. Проводка DP83848 модульSTM32F407VET6 борту VCC<->'3.3V GND<'>GND MDIO<>PA_2 MDC<'>PC_1 OSCIN<'>PA_1 CRS< B >PA_7 RX0<>PC_4 RX1<< >PC_5 TX_EN<->PB_11 TX0<>PB_12 TX1<>PB_13 Обратите внимание, что из-за того, что линия RX_ER не используется модуль DP83848, не нуждается в модифицировании. Изменения. stm32f407vet6 development board schematic pdf