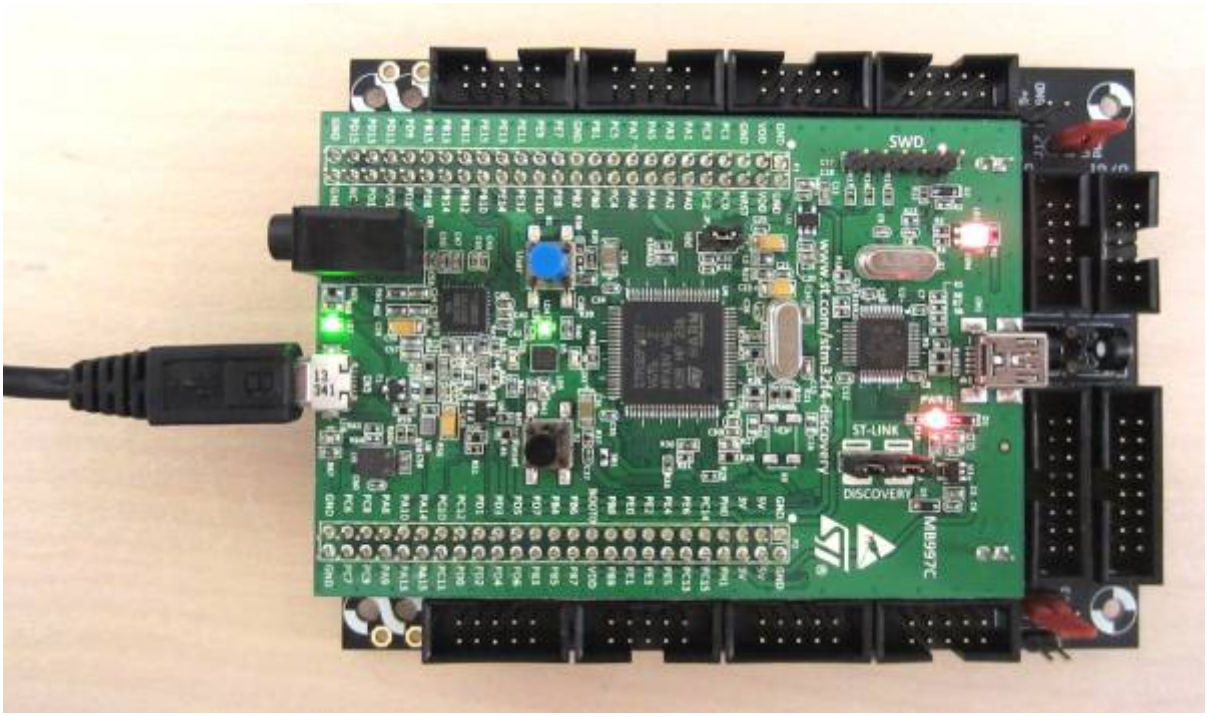
Core 32 STM32F4



Introduction

Features

- **1MB Flash and 192k RAM**
- **Easily available** for private persons in small quantities and no SMD soldering skills required
- ARM Cortex-M4 CPU is widely supported by the GNU toolchain, accordingly source code can be compiled under Windows/Linux/MacOS w/o spending money for an expensive ARM compiler
- **Increased performance** compared to STM32F1xx and LPC17xx because of the higher frequency (168 MHz)
- **Enhanced instruction set** with DSP operations makes Cortex-M4 very interesting for audio applications (e.g. Synths and Fx)
- MIOS32 applications running more than 2..3 times faster than on STM32F103RE @72 MHz, and ca. 50% faster than on a LPC1769 @120 MHz
- Very useful integrated peripherals for MIDI and Synthesizer applications, such as **UART, SPI, I2C, USB, CAN, Timers, ADC, DAC**
- The USB OTG (Micro-USB) socket supports **MIDI Host**, which means that it's possible to connect a USB MIDI Keyboard or Synth
- The on-board Audio DAC predestinates this module for synthesizer projects - it could even be used standalone without the MBHP_CORE_STM32F4 breakout board for this purpose.
- in distance to STM32F103RE, **CAN and USB can be used in parallel**
- It's easy to interface external components like SD Cards and MBHP_ETH
- STM32F4xx is supplied at 3V, but **5V tolerant inputs allow to access all existing MBHP modules without additional hardware**.
- **Performance boost by using the DMA** for background operations, such as scanning DIN/DOUT modules, analog inputs but also I2S transfers (Audio Output) w/o loading the CPU

- **Realtime Operating System: FreeRTOS**
- **no external flash programmer required**, ST-LINK/V2 is part of the STM32F4DISCOVERY board to program the MIOS32 bootloader into a “virgin” device
- Fast upload of application code via USB MIDI (ca. 50kb/s) - thanks to the MIOS32 Bootloader

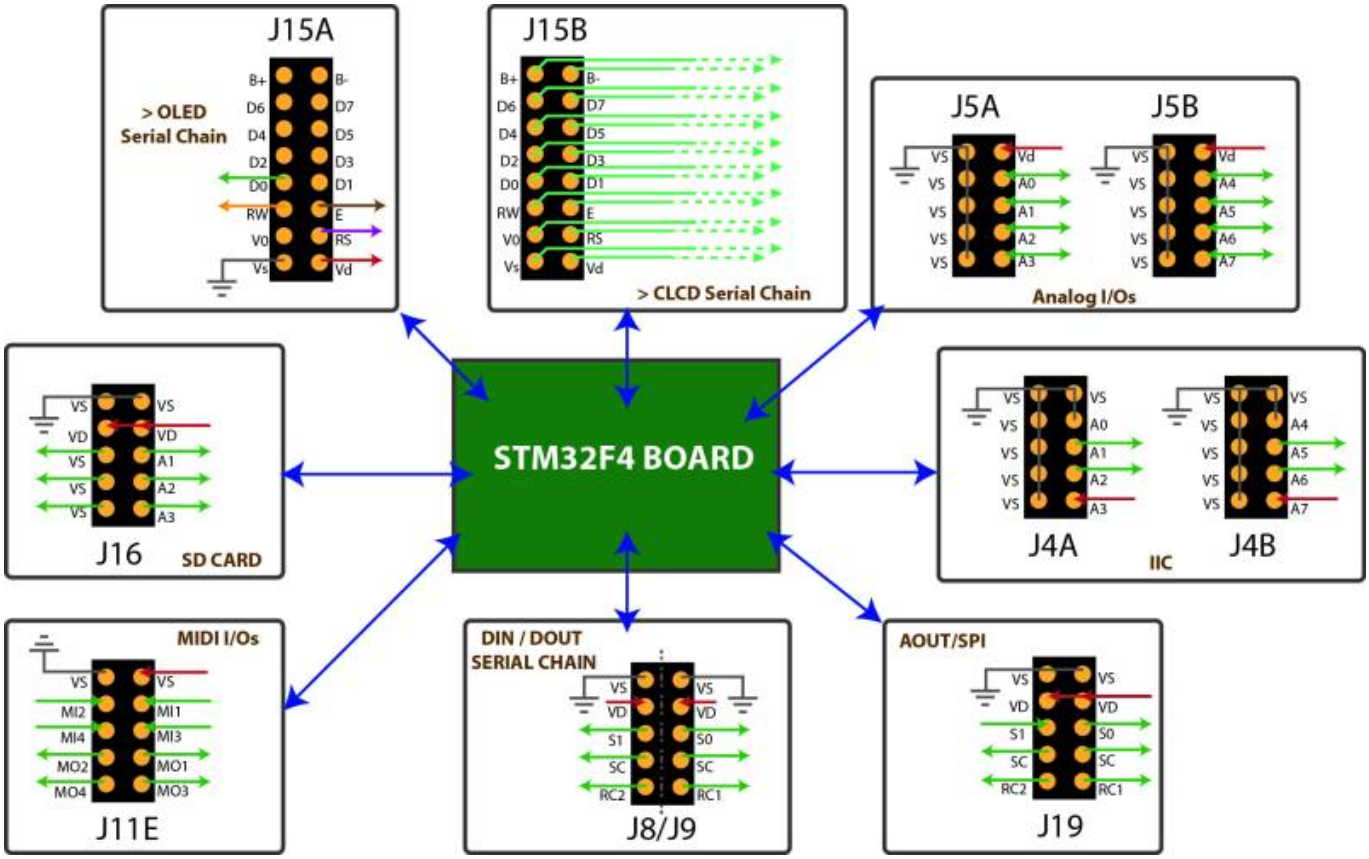
Known Issues

- ADCs cannot be supplied at 5V, 3.3V is the limit, and the STM32F4DISCOVERY board uses an even lower voltage (3V). This decreases the signal/noise ratio. The MBHP_AINSER64 and MBHP_AINSER8 module helps to overcome this by using an external 5V ADC.
- Pin mapping suboptimal in some cases. E.g., the outputs of the integrated DAC (not the on-board Audio-DAC) overlay SPI1 pins. However, one DAC channel has been freed up with the small drawback that SPI1 can't be used in slave mode.
- The IO pins of the SDIO peripheral (high speed SD Card access) conflicts with the on-board Audio DAC, so that it can't be used.
- Although the chip contains 6 UARTs, only 4 RX and TX pins could be found for MIDI IN/OUT. IN3/OUT3 even have to use two different UARTs.
- No on-board Ethernet PHY
- No PHY for the second USB peripheral - would be nice to have, e.g. to support USB MIDI device and host (e.g. for the connection of a USB MIDI Keyboard), USB Memory Sticks or even a USB harddisk in parallel.

Parts List

See Core32 Parts List

Inputs/Outputs Ports



Buy Places

From:
https://wiki.midibox.org/ - **MIDIbox**

Permanent link:
https://wiki.midibox.org/doku.php?id=stm32f4_module&rev=1470245774

Last update: **2016/08/03 17:36**