STM32F407VGT6

"do it yourself Mr. OE"

TOP

(RIGHT)

PWM / E11 o o E12 / PWM
PWM / E13 o o E14 / PWM
E15 o o B10 I2C2 SCL / PWM / SPI2 SCK / USART3 TX / (MIC*)
I2C2 SDA / PWM / USART3 RX / B11 o o GND
USART3 CK / B12 o o 3.3V
SPI2 MISO / PWM / B14 o o B13 / PWM / SPI2 SCK
USART3 TX / D8 o o B15 / PWM / SPI2 MOSI
USART3 CK / D10 o o D9 / USART3 RX
(LED2*) / PWM / D12 o o D11
(LED3*) / PWM / D14 o o D13 / PWM / (LED1*)
USART6 TX / PWM / C6 o o D15 / PWM / (LED4*) / BTN1
USART6 CK / PWM / C8 o o C7 / PWM / USART6 RX / (AUDIO*)

(LEFT)

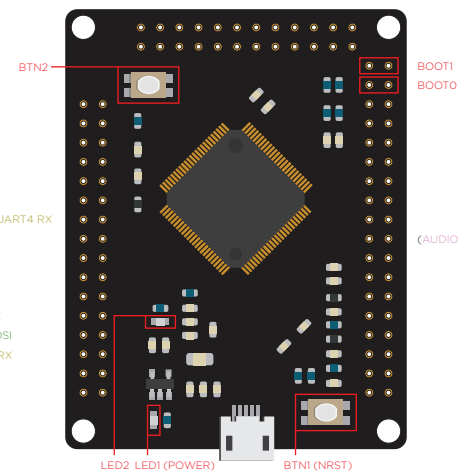
QUICK DEBUG

- onboard BTN2 is connected to D15 via a 10k pull-up resistor
- onboard LED2 is connected to E0 & is sinked by uC when ON
  (in other words, the logic is reversed as demonstrates the following example)
  turn LED on: digitalWrite(E0, 0);
  turn LED off: digitalWrite(E0, 1);

```
/* Mr.OE board test */
setWatch(function() {
  digitalWrite(E0, 0);
  setTimeout(function () {
    digitalWrite(E0, 1);
  }, 20);
}, D15, true);
```

LEFT

GND o o VCC
I2C3 SCL / PWM / USART1 CK / A8 o o C9 / I2C3 SDA / PWM
USB / PWM / USART1 RX / A10 o o A9 / USB / PWM / USART1 TX
USB / A12 o o A11 / USB / PWM
JTAG / A14 o o A13 / JTAG
(AUDIO*) / SPI3 SCK / UART4 TX / USART3 TX / C10 o o A15 / JTAG
(AUDIO*) / SPI3 MOSI / UART5 TX / USART3 CK / C12 o o C11 / SPI3 MISO / USART3 RX / UART4 RX
D1 o o D0
D3 o o D2 / UART5 RX
USART2 TX / D5 o o D4 / (AUDIO*)
USART2 CK / D7 o o D6 / USART2 RX
PWM / SPI1 MISO / SPI3 MISO / B4 o o B3 / PWM / SPI1 SCK / SPI3 SCK
(AUDIO*) / I2C1 SCL / PWM / USART1 TX / B6 o o B5 / PWM / SPI1 MOSI / SPI3 MOSI
E0 o o B7 / I2C1 SDA / PWM / USART1 RX
GND o o 3.3V
GND o o VCC



/!\ PCB SILKSCREEN ON ACTUAL BOARD IS WRONG:
'K2' IS NRST (RESET) & UNLABELLED 'K1' (USER BTN) IS WIRED TO D15

RIGHT

BOOT1 o o GND
BOOT0 o o GND
PWM / E10 o o B2 [BOOT1]
PWM / E8 o o E9 / PWM
(MEMS*) / E1 o o E7
I2C1 SCL / PWM / B8 o o B9 / I2C1 SDA / PWM / (AUDIO*)
ADC / PWM / B0 o o B1 / ADC / PWM
ADC / C4 o o C5 / ADC
(MEMS*) / ADC / PWM / SPI1 MISO / A6 o o A7 / ADC / PWM / SPI1 MOSI / (AUDIO*)
(AUDIO*) / ADC / DAC / USART2 CK / 3.3V / A4 o o A5 / 3.3V / ADC / DAC / SPI1 SCK / (MEMS*)
(MIC*) / ADC / SPI2 MOSI / C3 o o A3 / ADC / PWM / USART2 RX
ADC / PWM / UART4 RX / A1 o o A2 / ADC / PWM / USART2 TX
ADC / SPI2 MISO / C2 o o A0 / ADC / PWM / UART4 TX / (BTN1*)
ADC / USB / C0 o o C1 / ADC
3.3V o o GND
PWM / E6 o o C13
E4 o o E5 / PWM
E2 o o E3 / (MEMS*)

BOOT MODES

- BOOT0 & BOOT1 low (both jumpers set) -> normal boot (boot from flash to user firmware / 2nd bootloader)
- BOOT1 low (jumper set) & BOOT0 high (no jumper) -> STM32 bootloader & allow firmare upload through USART1
- BOOT1 high (no jumper) & BOOT0 high (no jumper) -> boot from RAM (if implemented)
- BOOT1 high (no jumper) & BOOT0 low (jumper set) -> N/A => defaults to normal boot from flash (case 1)

Note: keeping BOOT1 low & using a button on BOOT0 allows to:
- hold it & press reset button to run from flash
- press only reset & run STM32 bootloader

FLASHING & CONNECTING

FLASHING:
- for the STM32F4DISCOVERY board, pins A9 & A10 won't work cuz have caps for USB
  (seems to connect of for Mr.OE board, although there are 10uF & 0.1uF caps on Vcc
  USART1 Tx: A9, Rx: A10
  USART2 Tx: A2, Rx: A3 (differs from "Tx: B10, Rx: B11" as says stm32f4-discovery.net)
  USART3 Tx: C10, Rx: C11
- if connected via USB, the console is automatically moved to USB & needs
  'SerialX.setConsole()' to be typed via USB (X is the UART number)
- default UART is Serial2, as Serial1 will not work with USB plugged in

CONNECTING:
- use an HC-05 (3.3V without breakout !) on USART2 & set use its baud (def 9600)
- do the same using an USB to UART converter (USB-TTL, FTDI232 & Cie) (def 115200)