Q6) wiring diagram

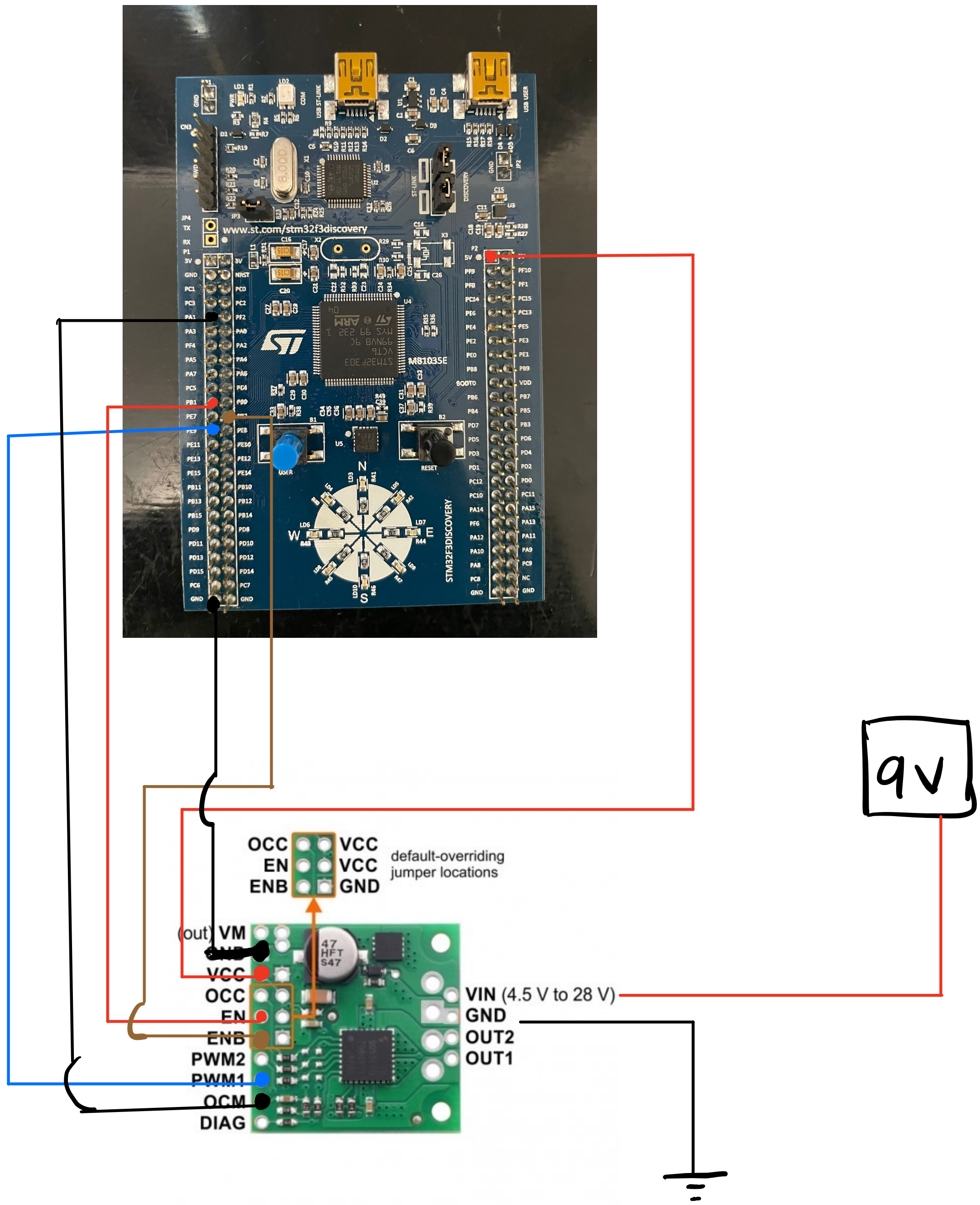
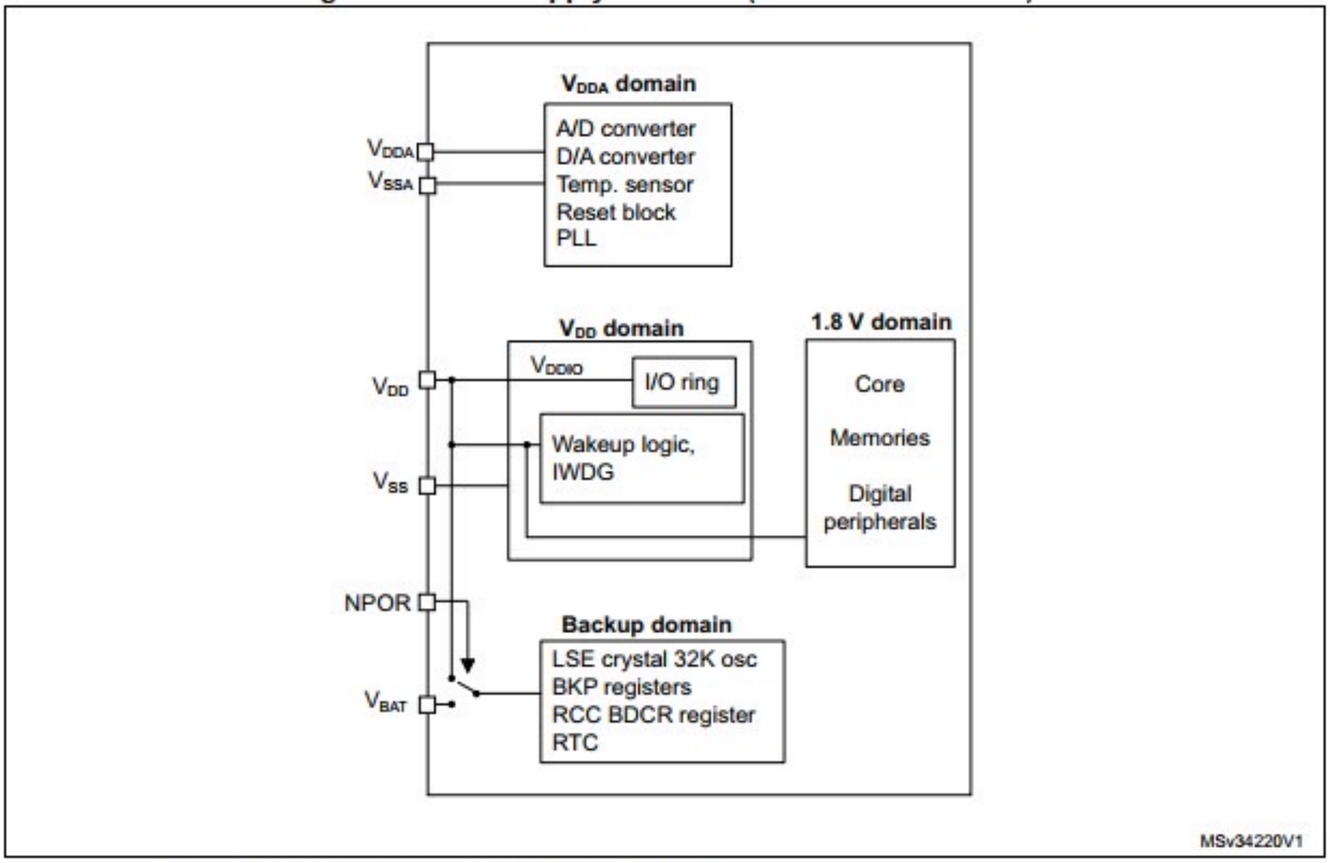


Figure 9. Power supply overview (STM32F3x8 devices)



The following supply voltages are available:

- V_{DD} and V_{SS}: external power supply for I/Os and core.
  These supply voltages are provided externally through V_{DD} and V_{SS} pins. V_{DD} = 2.0 to 3.6 V(STM32F303x6/8/B/C/D/E devices) or 1.8 V ± 8% (STM32F3x8 devices).
  When the 1.8 V mode external supply is selected, V_{DD} directly supplies the regulator output, which directly drives the VDD18 domain.
  V_{DD} must always be kept lower than or equal to V_{DDA}.
- VDD18 = 1.65 to 1.95 V (VDD18 domain): power supply for digital core, SRAM, and flash memory.
  VDD18 is either internally generated through an internal voltage regulator (STM32F303x6/8/B/C/D/E) or can be provided directly from the external V_{DD} pin when the regulator is bypassed (STM32F3x8).
- V_{DDA}, V_{SSA}= 2.0 to 3.6 V (STM32F303x6/8/B/C/D/E) or 1.65 to 3.6 V (STM32F3x8): external power supply for ADC, DAC, comparators, operational amplifiers, temperature sensor, PLL, HSI 8 MHz oscillator, LSI 40 kHz oscillator, and reset block.
  V_{DDA} must be in the 2.4 to 3.6 V range when the OPAMP and DAC are used.
  V_{DDA} must be in the 1.8 to 3.6 V range when the ADC is used.
  It is forbidden to have $V_{DDA} < V_{DD} - 0.4$ V. An external Schottky diode must be placed between V_{DD} and V_{DDA} to guarantee that this condition is met.
- V_{BAT}= 1.65 to 3.6 V: Backup power supply for RTC, LSE oscillator, PC13 to PC15 and backup registers when V_{DD} is not present. When a V_{DD} supply is present, the internal power switch switches the backup power to V_{DD}. If V_{BAT} is not used, it must be connected to V_{DD}.

⊗ According to documentation

STM32 operates with VDD supply range of 2.0V to 3.6V. operating at 3.3V is within acceptable range GPIO can use 3.3V logic for high signals. This confirms compatability with 3.3V logic signals for interfacing with other components, (motor driver / controller) in our case

PWM1	PWM2	EN	ENB	Mode	OUT1	OUT2	Description
High	Low	High	Low	Forward	High	Low	motor spins forward (clockwise)
Low	High	Low	High	Reverse	Low	High	Motor spins reverse (CCW)
Low	Low	High	Low	Short Brake	Low	Low	Motor brakes
High	High	Low	High	Short Brake	High	High	Motor brakes
Z	L/H	Low	Low	Coast	Z	Z	Motor Coasts, no active drive or brake
L/H	Z	High	High	Coast	Z	Z	Motor Coasts no active drive or brake

7.1. Motor Driver Output Circuit

The output circuit operates according to the following function (Table 7.1-1). In the Table 7.1-1 to 7.1-3, each letter means; X: Don't care, H: High, L: Low, and Z: High impedance.

Table 7.1-1 Motor function

	PWM1	PWM2	EN	ENB	**DIAG pin**	**OUT1**	**OUT2**
Forward	H	L	H	L	**H**	**H**	**L**
Short brake	L	L	H	L	**H**	**L**	**L**
Reverse	L	H	H	L	**H**	**L**	**H**
Short brake	H	H	H	L	**H**	**L**	**L**
EN Disable	X	X	L	X	**L**	**Z**	**Z**
ENB Disable	X	X	X	H	**L**	**Z**	**Z**
EN Disconnected	X	X	Z	X	**L**	**Z**	**Z**
ENB Disconnected	X	X	X	Z	**L**	**Z**	**Z**
PWM1 Disconnected	Z	L/H	H	L	**H**	**L**	**L/H**
PWM2 Disconnected	L/H	Z	H	L	**H**	**L/H**	**L**

Note 1: When the motor is set to the reverse from the forward, or to the forward from the reverse, be sure to perform after setting the brake between them. Otherwise the IC may be broken.

Note 2: In the current limitation control, the operation is different from the above table of the motor function. For details, refer to current limitation control (Section 7.3).

(*) TB9051FTG Motor driver datasheet

Explanation

⇒ we have different behaviors
- ↳ forward
- ↳ Reverse
- ↳ Coast
- ↳ Brake

based on the logic levels of EN, ENB and PWM1.

Coast mode

* Config 1

PWM1 = Z (disconnected) EN = Low

PWM2 = Low / High ENB = Low

* Config 2

PWM1 = Low / High EN = High

PWM2 = Z (disconnected) ENB = High

→ when both outpts (output 1 and output 2) are in high impedence (Z), the motors terminals are disconnected from the drive circuitry. This allows the motor to spin freely

forward mode

PWM1 = High EN = High

PWM2 = Low ENB = Low

Out 1 = High

OUT 2 = Low

=> When EN is high and ENB low -> motor controller is enabled. And high signal on PWM1 (with PWM2 low) Sets the output to drive the motor forward.

(output 1 high
output 2 low) creates +ve current flow)

Reverse Mode

PWM1 = low
PWM2 = High

EN = Low
ENB = High

Out 1 = low ; OUT2 = high

=> When setting EN low & ENB high motor controller is enabled but this time current flow is reversed.

Short Brake Mode

We have two config

① PWM1 = Low
PWM 2 = Low

EN = High
ENB = Low

② PWM1 = High
PWM 2 = High

EN = Low
ENB = High

Both configurations makes the out1 and OUT2 to either go low or high and this causes the motor to brake by shorting the terminals, stopping it quickly.