Chapter 4 Tests, Results and Discussions

4.1 MATLAB simulation results

4.1.1 SVPWM generation module

Given the rotor electric angle from 0 to 360 degrees, set the reference V_d, V_q to be 0 and 1 respectively, the SVPWM generation module was tested. The result of dwelling time of each phase with respect to rotor electric angle was shown as,

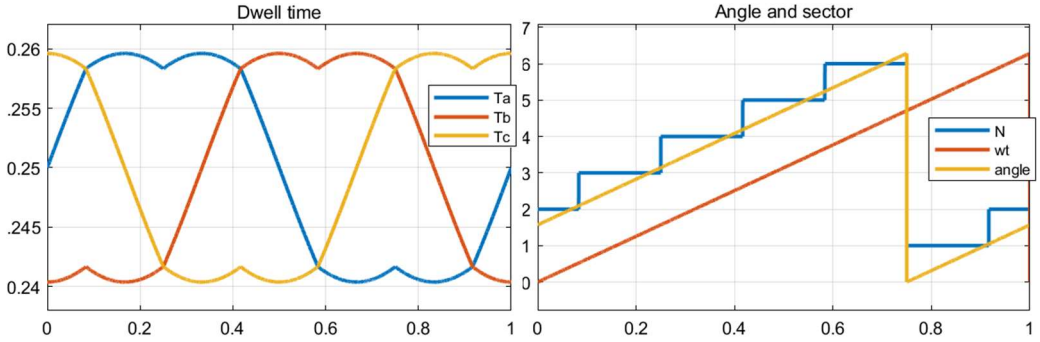

Figure 4-1 Test results of the SVPWM module

The modulation pattern obtained by the SVPWM algorithm is saddle shape, which is beneficial to improve the utilization rate of DC and restrain harmonics effectively. The variable N is the sector in which the resultant space voltage vector resides, the variable named angle is the angle of the resultant space voltage vector, and the wt in the result is the input rotor electric angle. Theoretically, when V_d, V_q were given to be 0 and 10, the space voltage vector would always lead the rotor electric angle by 90 degrees ($\frac{\pi}{2}$). As shown in the figure, the experimental results are completely consistent with the expectation. The corresponding output PWM pattern is also consented with the prediction when the space voltage vector is in the third sector (surrounded by U_{60} and U_{120}), the length of the high-level time for phase B is the longest, then phase A and phase C, shown in Figure 4-2.

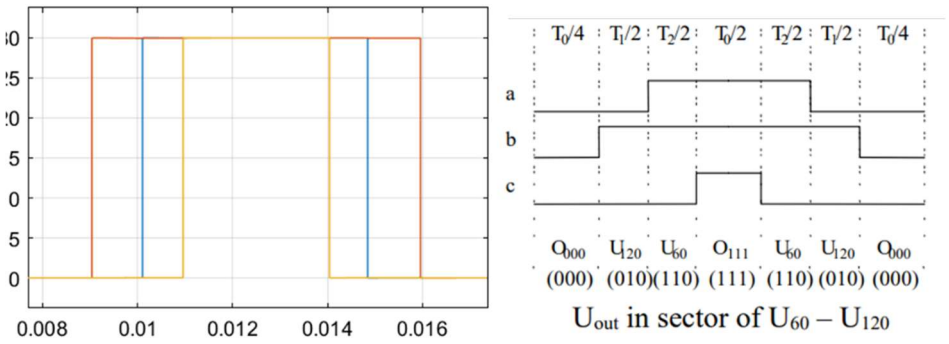

Figure 4-2 PWM pattern comparison

4.1.2 FOC open-loop

The open-loop test is carried out by directly setting the V_d, V_q were given to be 0 and 10, observing the motion of the rotor, theoretically, the rotor should constantly accelerate as the generated space voltage vector always leads the rotor electric angle, which means the generated field is always perpendicular to the rotor field and giving a maximum torque. Shows the simulation model for FOC open-loop test, and the result was given by,

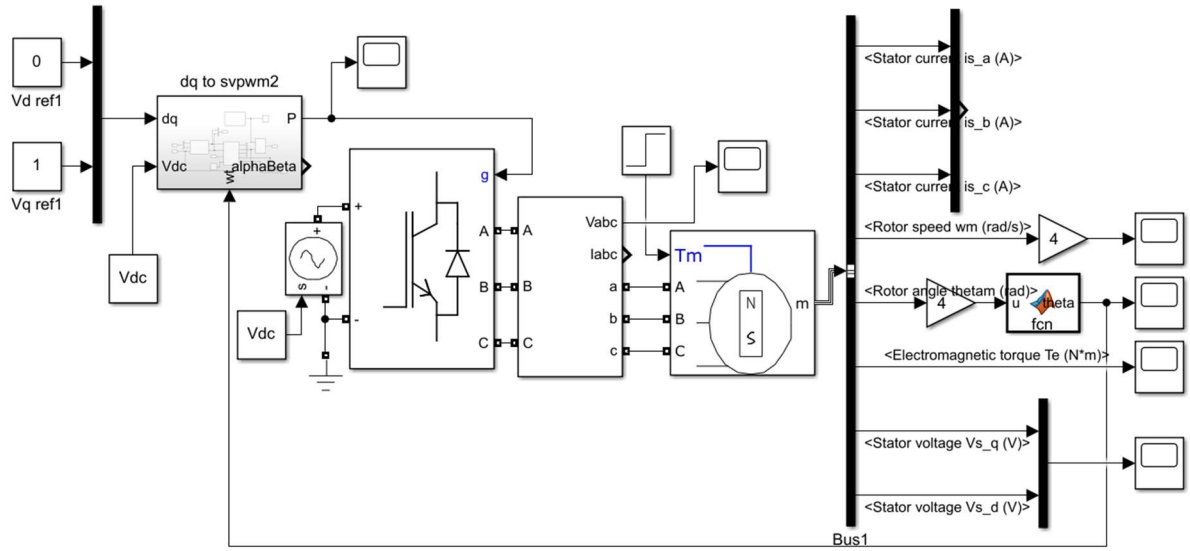

Figure 4-3 FOC open-loop model

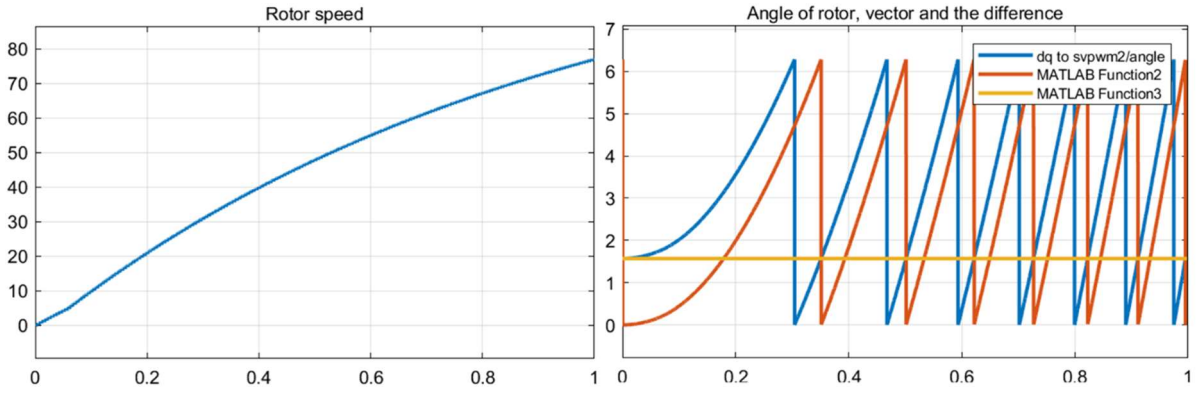

Figure 4-4 Open-loop simulation result

4.1.3 FOC current loop

The current loop is built based on the open-loop model, when setting the reference I_d, I_q to different values, the rotor speed, and actual values of I_d, I_q are measured, as the current loop is the core loop of the FOC algorithm, the system response should be fast. To test the system response, the quadrature current was set to 0.5 A initially and was changed to 2 A and 1A at

time 0.25 and 0.75 respectively. Overall, the system can respond to step changes in a very short time and has good tracking. As the output torque is proportional to the quadrature current, higher torque should result in a higher acceleration rate, as shown in the speed result, the speed of the rotor has an increasing trend and the acceleration rate increases first and then decreases.

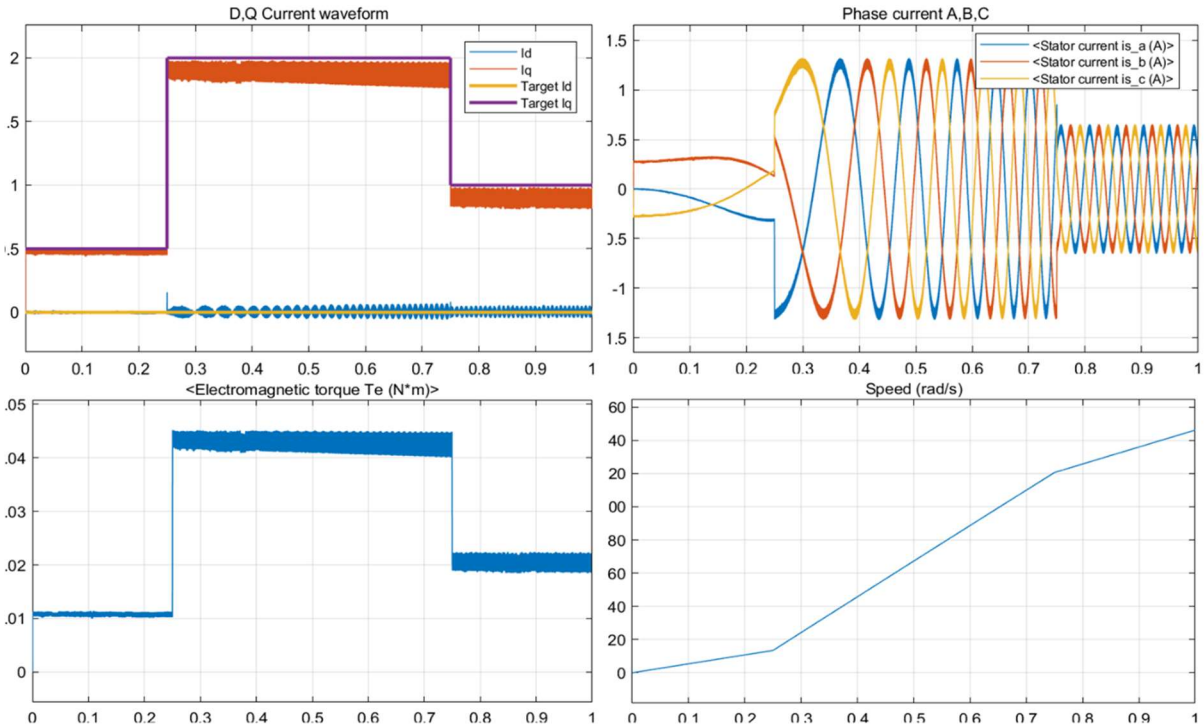

Figure 4-5 Current loop results

4.1.4 FOC speed loop

The reference speed was initially set to 200 rad/s and was changed to 100 rad/s, 300 rad/s at time instant 0.25s and 0.75s respectively. In addition, to test the anti-interference of the system, initially, external load torque was set to 0.2 N. m and at $t = 0.5$ s, it was reduced to $T = 0.1$ N. m. The system response was shown as,

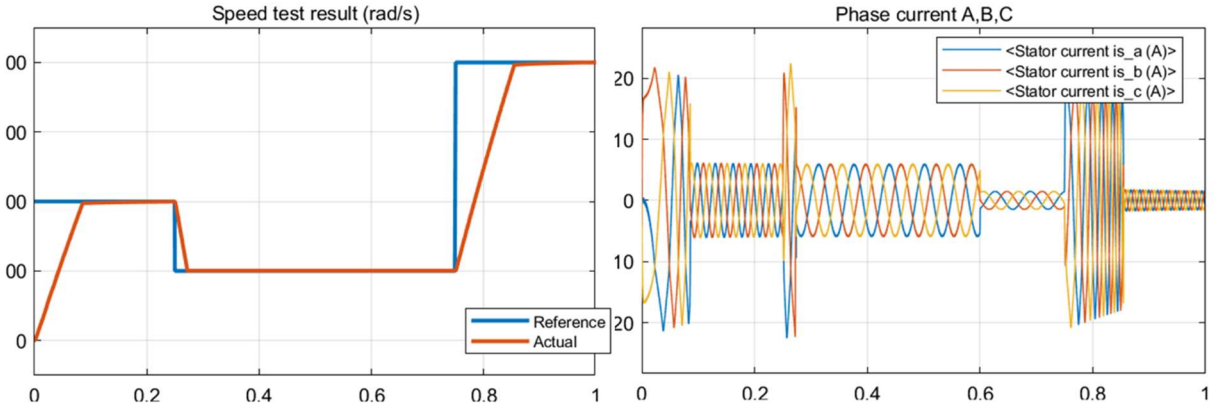

Figure 4-6 Speed loop simulation results

Overall, the system response was at an acceptable level, constrained by the motor parameters and supply voltage, when the speed step change was large, the response time would be longer.

4.1.5 FOC position loop

For the position-current double loop, the initial target position was set to 1 rad and was changed to 5 rad at 0.25 s, 3 rad at $t = 0.75$ s. Though the motor speed has some overshoot at the beginning, it has a fast dynamic response. The overall performance was satisfactory.

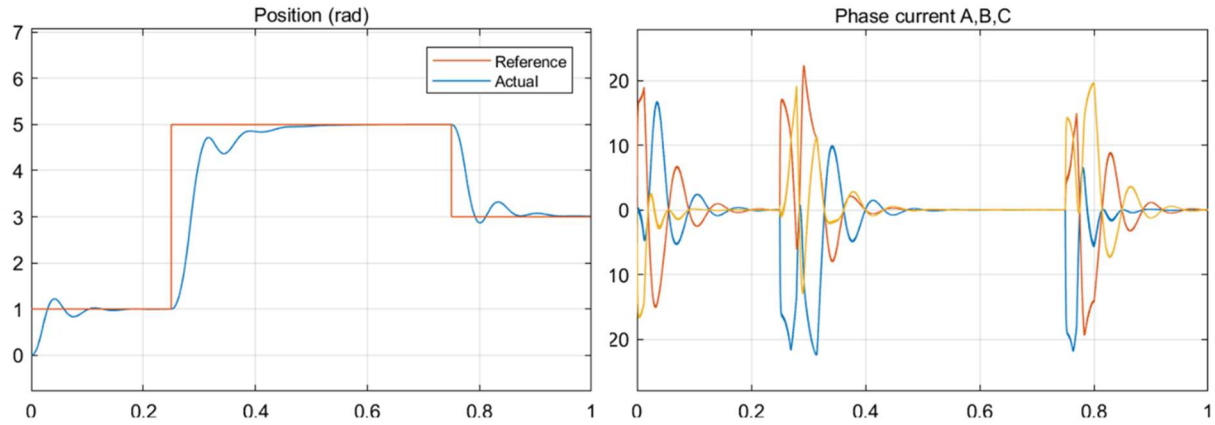

Figure 4-7 Position-current loop simulation results

The system performance for the position-speed-current triple loop was tested under the same condition, the response rate was good but compared with the position-current double loop as shown in, less overshoot was created. However, the overall response mainly depends on the parameters of the PI controllers, it could be argued that by setting appropriate parameters, the same performance could be developed.

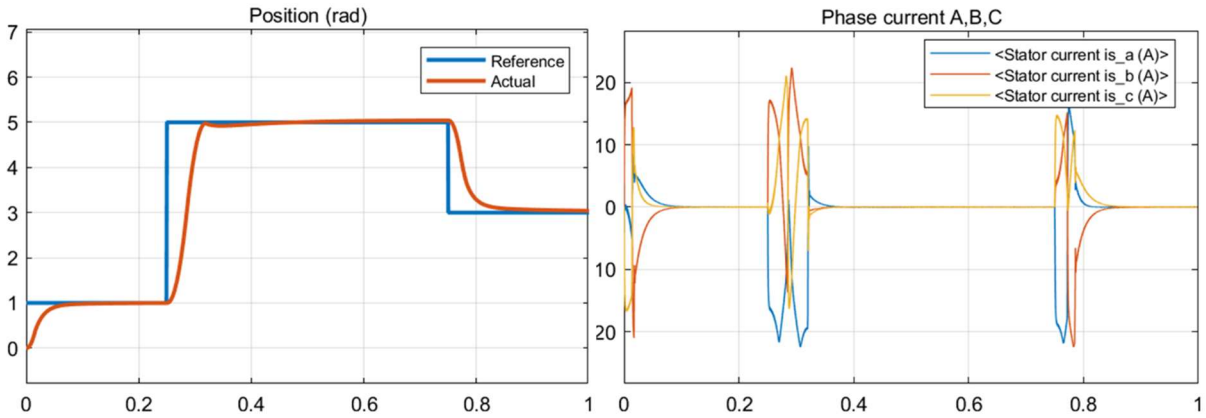

Figure 4-8 Position-speed-current loop simulation results

4.2 STM32

4.2.1 Encoder Angle Acquisition

Both I2C mode and PWM mode for the encoder were tested, and as Figure 4-9 shows, the performance for PWM mode was lower because the accuracy of PWM generation and duty cycle calculation caused the existence of fluctuation. Moreover, the fluctuation of the original mechanical angle would further affect the reliability of the calculated electric angle, thus, the I2C mode was used.

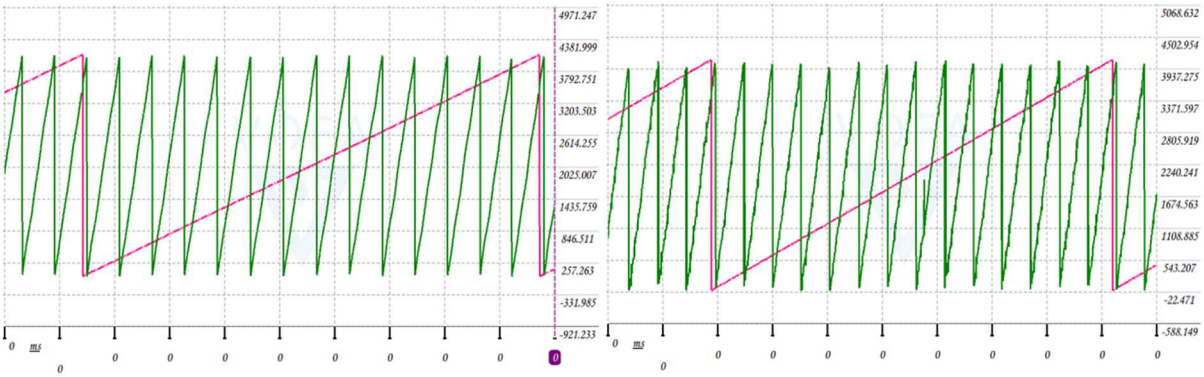

Figure 4-9 Encoder feedback for I2C mode (left), PWM mode (right)

4.2.2 SVPWM Generation

SVPWM generation function was tested with the same input variables as the 4.1.1 SVPWM generation module, which were 0 to 360 degrees of electric angle and reference $V_d = 0\ V, V_q = 10\ V$, the output is printed through the serial port (Figure 4-10).

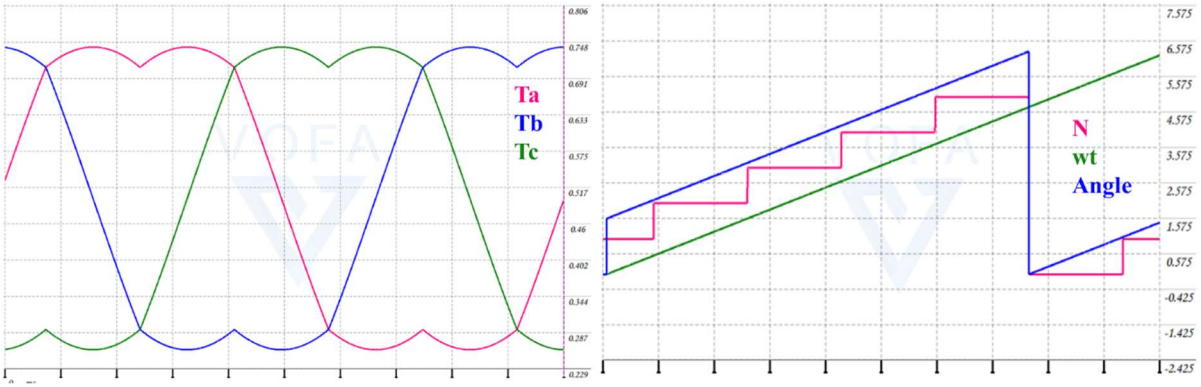

Figure 4-10 STM32 SVPWM generation result

It should be noted that the N in Figure 4-10 was the sector minus one, whereas in Figure 4-11 was the number of the sector. The two results were consistent.

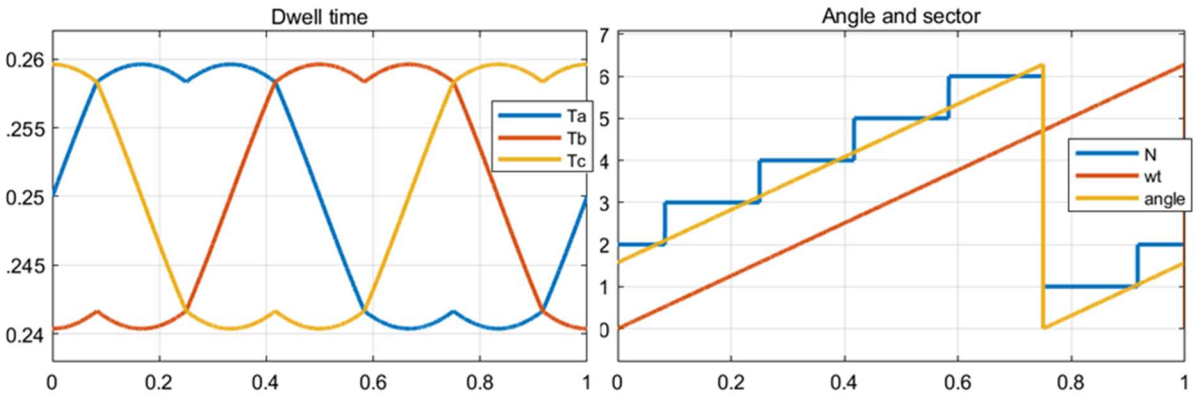

Figure 4-11 MATLAB SVPWM generation result

4.2.3 Open-Loop Test

The open-loop test is carried out using the same method as the MATLAB simulation test, that is, by directly setting the V_d, V_q were given to be 0 and 1, observing the motion of the rotor. The rotor position and speed with respect to time was shown in Figure 4-12. It seems that the rotor was rotating at a constant speed, which was inconsistent with the expectations of sustained acceleration. This might be because friction was not considered in the simulation part, as the speed increased, the force developed by the rotor torque was soon balanced by the friction resulting from the shaft.

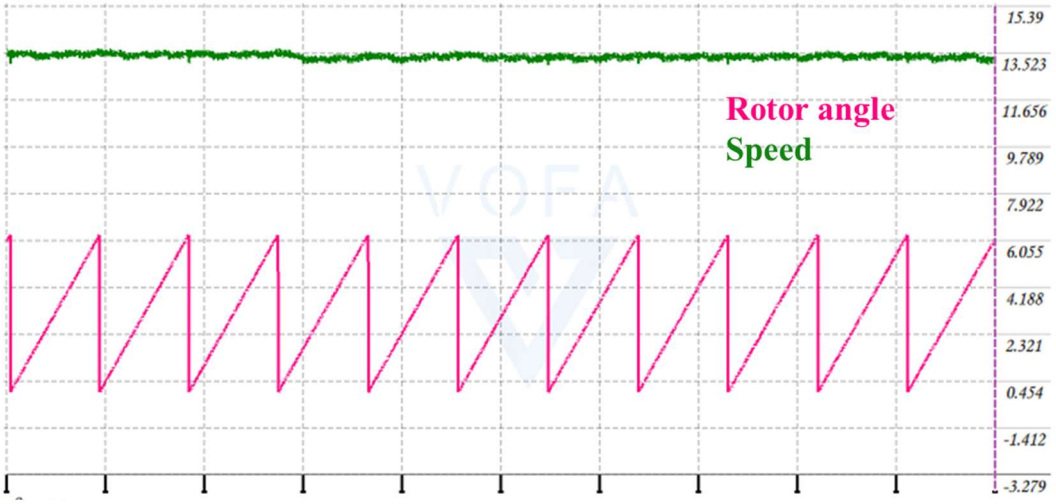

Figure 4-12 Open-loop test result

4.2.4 Current Sample Test

The sampling frequency was set to 50 kHz for better performance of the FOC algorithm, two channels of ADC were used to sample the current of phase A and phase B, the phase C current was reconstructed as introduced, the gain of 40V/V was selected and both raw data and filtered data was printed through UART communication port, shown as Figure 4-13.

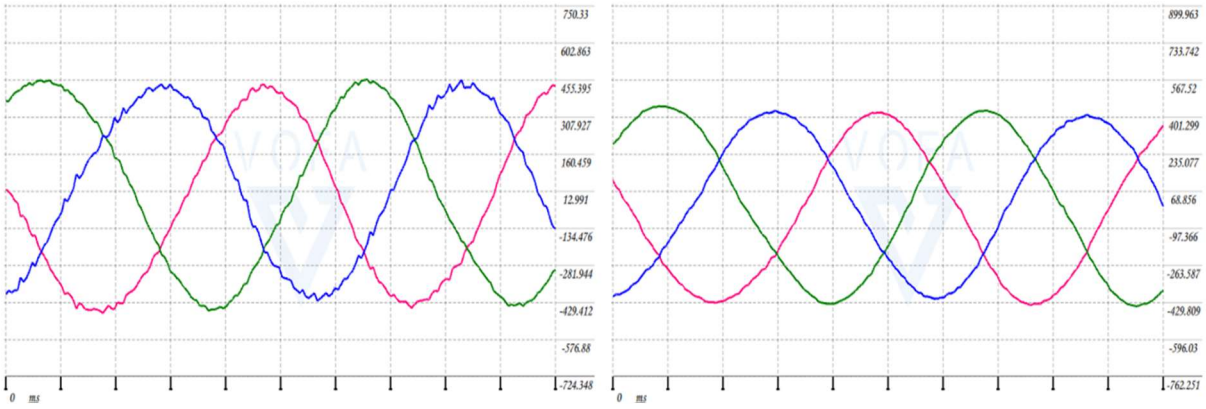

Figure 4-13 Current sampling result without filter (left), with filter (right)

As stated, a first-order low pass filter was used and the filter coefficient was set to 0.07. as shown in the result, fluctuation (mainly high-frequency noise) was filtered out by the filter and the data became smoother. Another thing that should be noted was that the average phase current is not zero, which means the output of the current shunt amplifier in DRV8302 is not exactly the reference voltage (1.65 V). Therefore, a calibration should be performed to obtain a more accurate result. This could be realized using the DC calibration feature of DRV8302 together with ADC, to calibrate the reference output voltage level when the system starts.

4.2.5 Angle Calibration

One important thing to be noticed for the practical realization of the FOC algorithm was the calibration of the rotor mechanical angle and electric angle. In other words, because of the relative mounting position of the encoder, the mechanical zero-point might not be exactly aligned with phase A (electric zero point). The calibration idea was thus measuring the angle offset. This can be achieved by setting the opening time of the three-phase MOSFETs to generate a space voltage vector towards phase A, and measuring the rotor mechanical Angle when the rotor is stabilized, which is the offset of electrical angle zero point and mechanical angle zero point. The actual electric angle can be determined by,

$$Angle_E = (Angle_M - Offset) * nPolePairs \qquad \text{eqn. 4.1}$$

4.2.6 Current Loop Test

The PI controller parameters can be determined by the phase resistance and inductance of the motor and the bandwidth of the controller, from the motor specification,

$$\begin{cases} R_s = 5.32\ \Omega \\ L = 2.66\ mH \end{cases} \qquad eqn.\ 4.2$$

By convention, the bandwidth is given as

$$BW = \frac{f_{sampling}}{20} = 2.5\ kHz \qquad eqn.\ 4.3$$

The theoretical PI parameters were determined by,

$$K_p = \frac{L \times BW \times 2\pi}{Gain} = 0.0407 \qquad eqn.\ 4.4$$

$$K_i = \frac{R \times BW \times 2\pi}{integration\ time} = 0.00169 \qquad eqn.\ 4.5$$

The system response of I_d was tested and the practically PI parameters could be different because of the error of the practical values of the motor. The final result was shown as, with $K_p = 0.02, K_i = 0.001$.

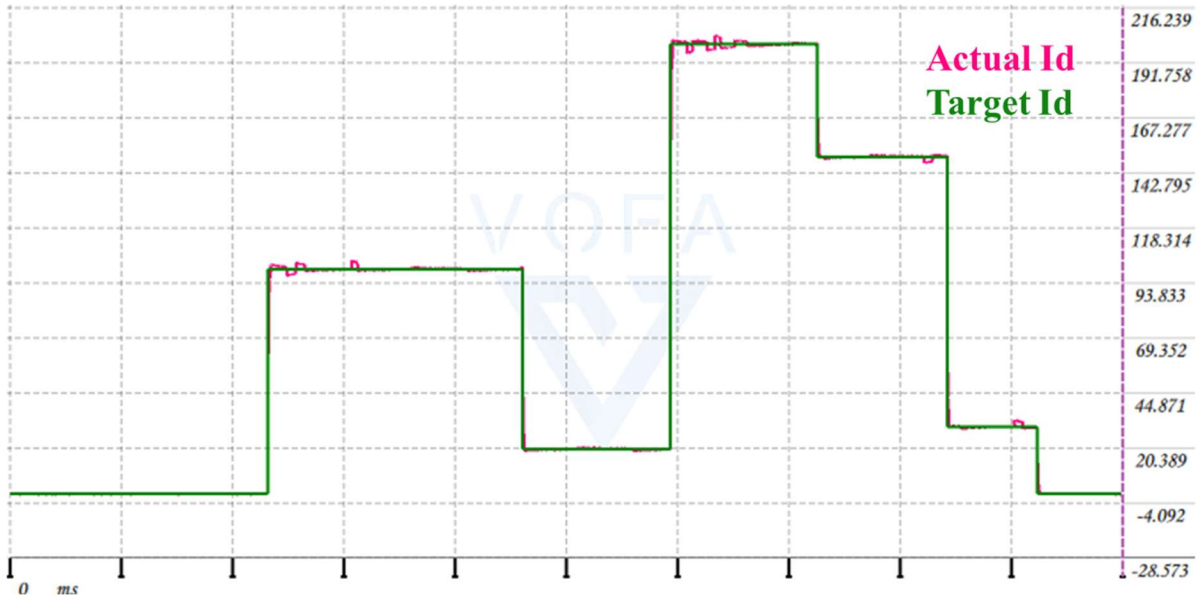

Figure 4-14 Id response

Overall, the system response was fast, though some overshoot existed.

4.2.7 Speed Loop Test

The overall response with positive and negative speed targets was shown as,

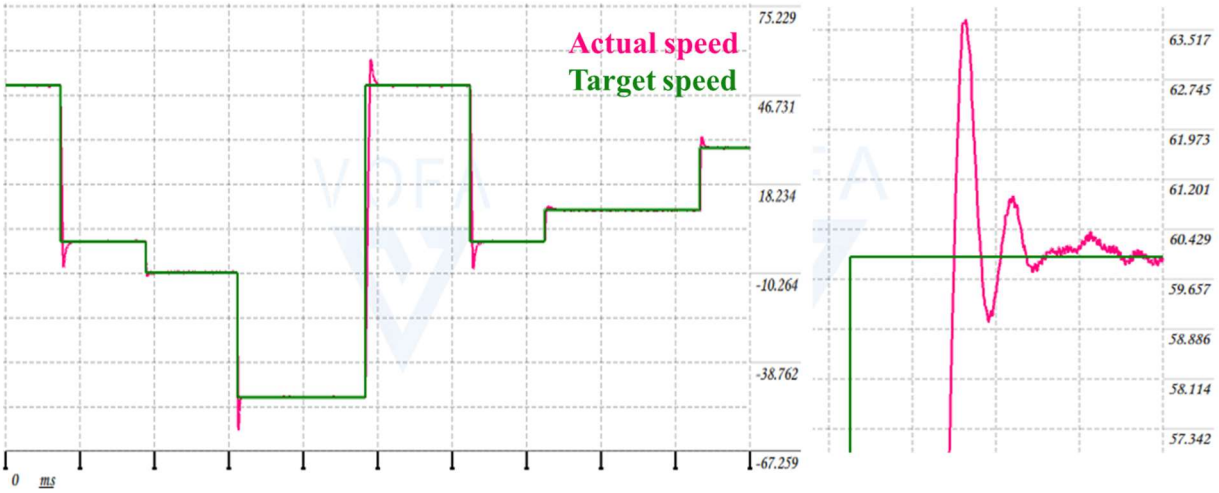

Figure 4-15 Speed response

To get a fast system response, the whole system was adjusted to have little overshoot. The PID parameters for the speed loop were set to be,

$$\begin{cases} K_p = 40 \\ K_i = 0.1 \\ K_d = 10 \end{cases} \qquad eqn.4.6$$

4.2.8 Position Loop Test

For maximizing the system stability, the position-speed-current triple loop topology was selected to be the final position loop, only a PD controller was used, and the corresponding parameters were,

$$\begin{cases} K_p = 0.5 \\ K_d = 0.1 \end{cases} \qquad eqn.4.7$$

The final result was shown in Figure 4-16, target position was set to be 2048 initially, then the external torque was exerted on the rotor to test the anti-interference of the system, then the target position was set to be 3000, the same process was carried out. It is shown that the overall system response was agile and after the external torque, the rotor soon returned to the target position, which illustrated the interference tolerance of the system was satisfactory.

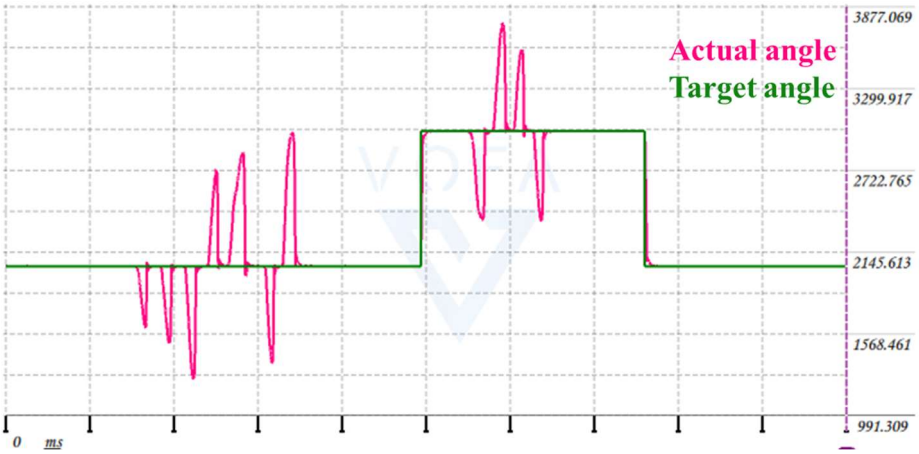

Figure 4-16 Position loop result

4.3 Dual Motor Control board

4.3.1 FOC Core Functions

With all parameters determined, the structure of the whole program library was adjusted to achieve double motor control. multithreads technique was used to ensure the system performance, this was achieved with the help of the freeRTOS operating system middleware of the STM32. The overall logic was demonstrated as,

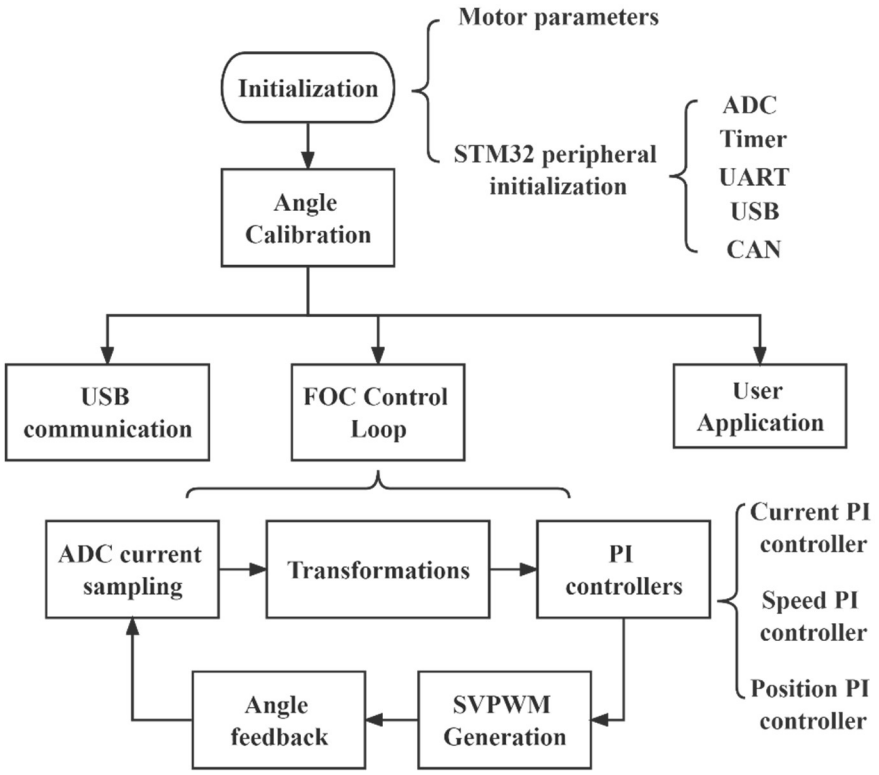

Figure 4-17 Program structure

The control frequency of the FOC algorithm was set to be 50 kHz and was measured using the oscilloscope, the recorded waveform was obtained by toggling an output pin at every control cycle, thus the measured frequency represents half of the control frequency.

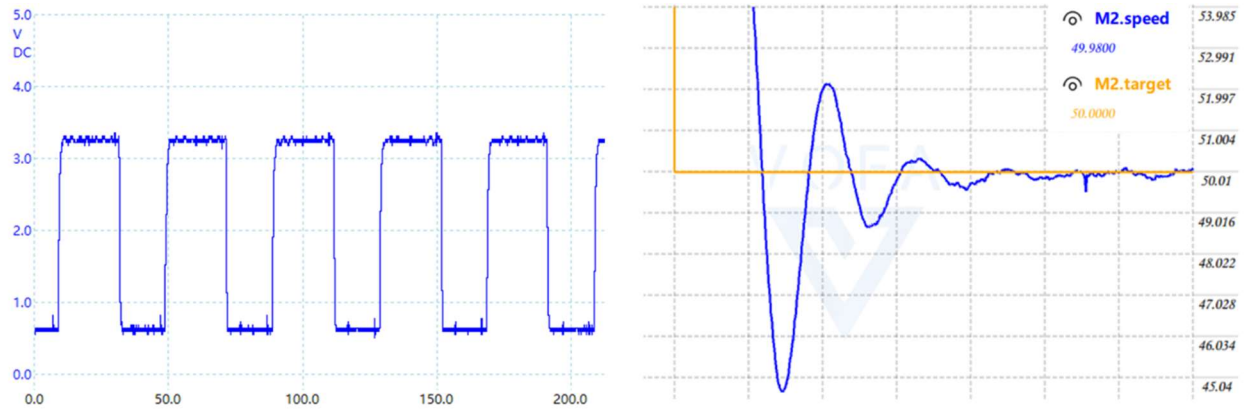

Figure 4-18 (a) Control frequency (b) Overshoot for speed mode

The speed and position mode for the double motor was tested, and the system response was shown in Figure 4-19 (a).

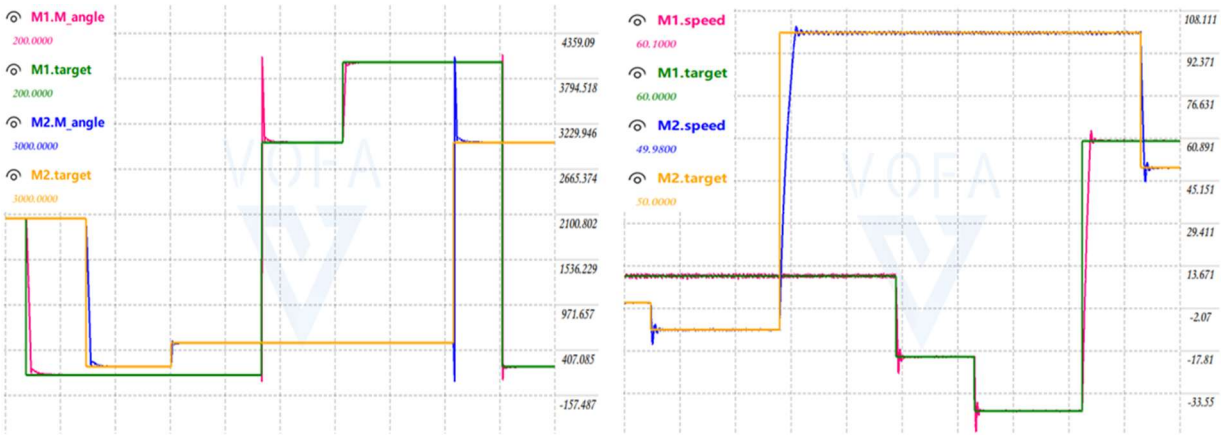

Figure 4-19 (a). Position mode response (b). Speed mode response

Still, the overall response was swift but due to a large Kp, high overshoot existed, especially when a large difference angle was given as the targets. Figure 4-19 (b) shows the system response at the speed mode.

Similar to the position loop, overshoot exists but short response time. Take the second motor as an example, the overshoot has reached nearly 5 rpm (Figure 4-18 (b)). This might be a result of large Kp, large Ki and small Kd as expected. Overall, both motors can respond quickly according to the controlled quantities.

4.3.2 Force Feedback

The Force feedback function was built based on motor position tracking, and the level of feedback force that was being produced can be represented by the magnitude of quadrature current of the controller motor as stated in $eqn.$ 4.8,

$$T_{e-non-salien} = \frac{3}{2} p_n i_q \varphi_f \qquad eqn. 4.8$$

The test result was demonstrated in Figure 4-20, the magnitude of currents in the figure was the ADC sampled value which equals the current in amperes multiplied by the amplifier gain (approx. 662), and angles were mapped to 500 to 909.6 for the convince of illustration. External torque was applied on motor M2 at time instants of t1 and t3 and was released at t2 and t4.

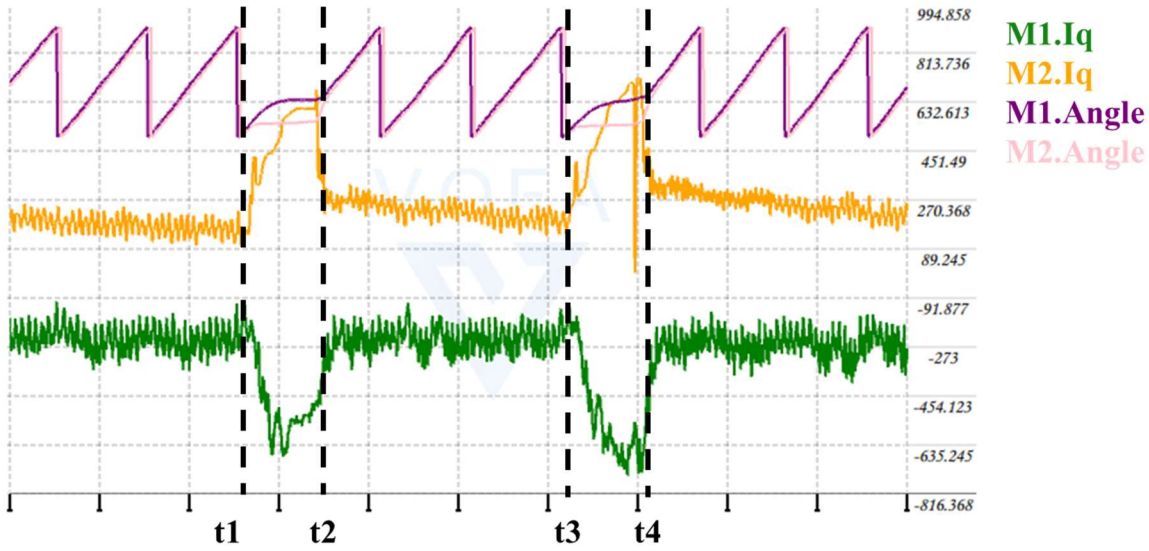

Figure 4-20 Quadrature Currents for M1, M2 with reference to rotor angle

It was shown that after applied external torque, the motor M2 was stopped and the magnitude of i_q for both motors were increased, which means larger inverse torque was created by the controller motor M1. In other words, the user needs a higher force to rotate the motor. The motion state of the two motors is restored after the torque was withdrawn after t2 and t4.