

Development of an Automated Multifunctional Motor Test Bench

Candidate: Farshad Hassanalzadeh

Supervisors: Prof. Luca Ferraris; Ing. Emir Poskovic – Laboratory Technician: Ing. Fausto Franchini

General framework and objective

This thesis is an experimental and implementation-oriented work on the realization of an automated multifunctional test bench for a Permanent Magnet Synchronous Motor (PMSM). The aim was to transform a generated STM32 motor-control project into a repeatable laboratory platform able to start and stop the motor, acknowledge faults, command validated speed ramps, tune speed and current regulators, monitor physical quantities, and obtain final performance results from measured data. The activity connects PMSM theory, power electronics, STM32 MCSDK, Field Oriented Control (FOC), embedded firmware, serial communication, LabVIEW monitoring, PID tuning, fault recovery and efficiency-map generation. The main value is practical: the motor was physically connected to the inverter and brake/load system, commanded from PuTTY, monitored in LabVIEW and evaluated from measured voltage, current, speed, torque, input power, output power and efficiency.

Experimental platform and methodology

The test bench uses an STM32F302 platform with the NUCLEO-F302R8 control board and the X-NUCLEO-IHM08M1 3Sh three-phase inverter. The laboratory system includes the PMSM under test, DC power supply, inverter, three-shunt current sensing, bus-voltage and temperature feedback, a mechanical brake/load unit, speed and torque measurement chain, ST Motor Profiler, ST Motor Control Workbench, PuTTY and LabVIEW. The motor was characterized with Motor Profiler and imported into Workbench as a surface-mounted PMSM with four pole pairs, nominal voltage close to 14.1 V, 6 A peak current limit and 30 kHz PWM frequency. This hardware-first approach was necessary because the measured behavior depends on the complete chain: motor, inverter, brake, observer, current loops, ramp duration, firmware state machine and measurement software.

After the introduction, the thesis first explains motor types, drives, PMSM modelling and control methods. FOC was selected because it regulates the flux current i_d and torque current i_q independently in the rotating dq reference frame, while Space Vector PWM generates the inverter switching signals. The generated MCSDK project was then customized to become a real test-bench application rather than only a default firmware example.

Firmware implementation, synchronization and tuning

A custom USART2/UART interface was implemented and tested through PuTTY at 115200 baud. The firmware receives characters by interrupt, reconstructs the command string, validates the input and executes motor-control actions in the main loop. The implemented features include Start, Stop and Ack commands; numerical speed-reference commands; storage of a speed command before motor start; safe speed limitation from 500 rpm to 3000 rpm; motor-state reporting; fault-code reporting; and commissioning messages. The Ack command is essential to clear fault states and restart the laboratory sequence safely. The firmware also reports the speed-feedback fault code 0x0020, which was observed during unstable operating conditions.

A key experimental problem was the mismatch between the rpm value entered in PuTTY and the real mechanical speed. The MCSDK function used by the custom firmware required an internal speed unit, not direct mechanical rpm. Experimental tests showed that the real speed was approximately six times the internal command, so the implemented correction was $MCSDK\ command = requested\ rpm / 6$. After this correction, the speed command was validated at 500, 1000, 1500, 2000, 2500 and 3000 rpm, with speed errors approximately between 1.00% and 0.13%.

The speed-loop PI regulator, the I_q torque-current loop and the I_d flux-current loop were tuned experimentally with the motor connected to the brake/load system. The investigated final configuration included $K_{p, speed} = 290$, $K_{i, speed} = 20$, $K_{p, I_q} = 200$, $K_{i, I_q} = 300$, $K_{p, I_d} = 200$ and $K_{i, I_d} = 300$. The measured responses showed that aggressive gains increase overshoot, oscillation and fault probability, while a 10000 ms ramp improves stability.

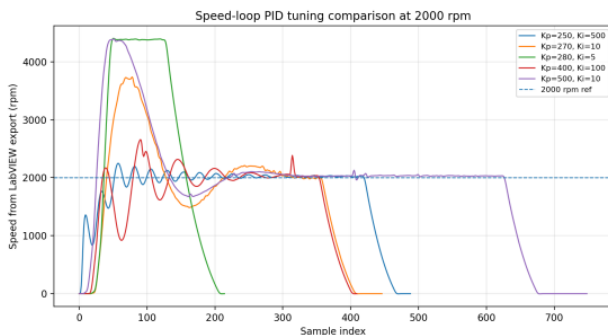


Figure 1: Speed-loop PI/PID tuning comparison at 2000 rpm.

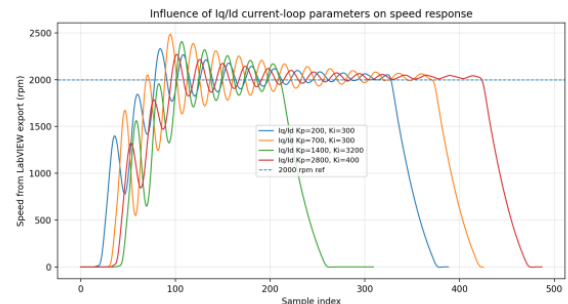


Figure 2: Influence of I_q/I_d current-loop parameters on speed response.

LabVIEW monitoring and final measured results

LabVIEW was used to monitor speed, torque, voltage, current, input power, output power, efficiency and dynamic curves. PuTTY and LabVIEW had complementary roles: PuTTY commanded and debugged the firmware, while LabVIEW observed and recorded the electromechanical response. A repeatable procedure was defined: acknowledge faults, enter or store the speed command, start the motor, apply the ramp, observe the response, acquire data, stop the motor and repeat the test for another speed or braking condition. This procedure converts the setup into a reusable PMSM-drive test bench.

The final performance activity was the experimental efficiency map. The map was created from 45 measured operating points at 500, 1000, 1500, 2000, 2500 and 3000 rpm while progressively changing the braking load. Efficiency was calculated as $\eta = P_{out}/P_{in}$; therefore $\eta = 0.60$ corresponds to 60%. The best measured efficiency was approximately 60.3% near 2000 rpm and 27.6 mNm, while the maximum measured torque of the campaign was about 28.05 mNm at 2500 rpm. The low-speed/high-torque region contains fewer stable points because the brake is unstable in that operating range. The brake is an asynchronous motor with DC-current excitation; its torque-speed characteristic is locally very steep at low speed, so when high braking torque is applied the control has difficulty stabilizing the speed. For this reason, only stable and repeatable low-speed points were used in the final map.

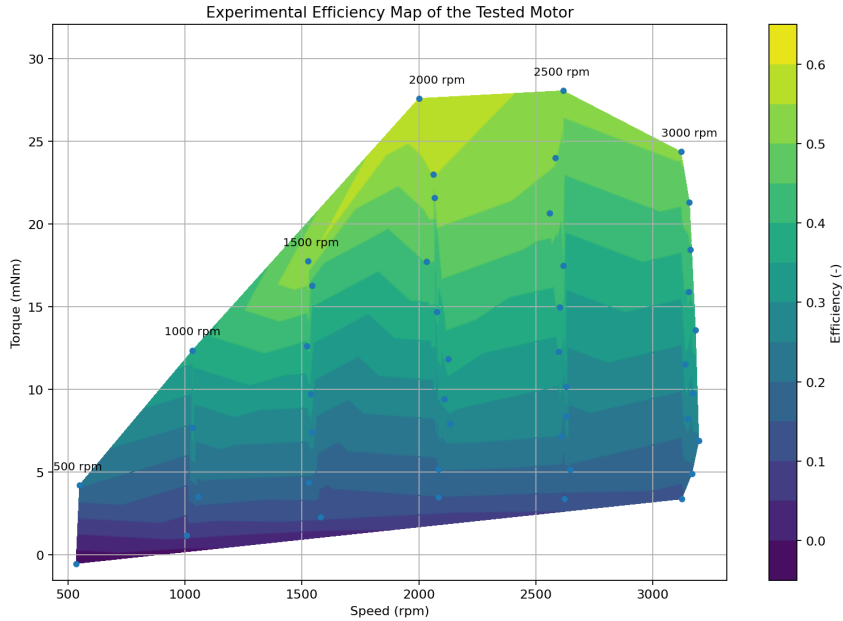


Figure 3: Experimental efficiency map of the tested PMSM drive.

Table 1: Summary of the measured efficiency campaign.

Speed (rpm)	Points	Peak eff. (%)	Max T. (mNm)
500	2	20.4	4.20
1000	4	37.3	12.36
1500	7	54.5	17.74
2000	10	60.3	27.59
2500	11	52.4	28.05
3000	11	51.0	24.36

Academic value, limitations and future development

The thesis delivers a functional and experimentally validated PMSM test bench. The most relevant outputs are: a configured MCSDK project, a customized UART firmware layer, the validated rpm/6 speed conversion, a repeatable laboratory procedure, PID-tuning evidence for speed and current loops, documented fault recovery, LabVIEW measurement screens, exported datasets and a final efficiency map. These results demonstrate the complete engineering path from motor identification to embedded implementation, experimental debugging, controller tuning, measurement and final performance assessment.

The limitations observed during the work were treated as engineering results rather than hidden problems. Sensorless start-up, brake inertia, friction, measurement noise, fault thresholds, ramp duration and current-loop tuning all influenced the final response. In particular, the low-speed/high-torque area could not be fully populated because the available brake made the operating point difficult to stabilize. Therefore, the map represents the stable and repeatable points that could be measured with the available test bench.

The platform can be extended to automatic LabVIEW test sequences, longer calibrated data logging, automatic generation of speed–torque–efficiency maps, comparison of different motors, repeated tests with improved torque calibration, and further optimization of PMSM-control strategies. A secondary axial-flux motor was prepared and profiled only as a future-work starting point; no complete validation campaign was performed on that motor. Therefore, the final contribution of the thesis is a complete experimental foundation for future automatic motor testing at laboratory level.