LIFE Test Bench Design

Eric Danese
Collegio di Ingegneria Elettrica ed Energetica
Politecnico di Torino
Torino, Italia
danese.eric@gmail.com

Supervisors:
Aldo Canova
Alessandro Fasolo
Pierre Lefranc
Xinchang Liu

***Abstract*—The LIFE time test bench project aim, is to advance ETEL S.A. equipment for testing the lifespan of insulating materials in torque motors. This project involved the design of an advanced test bench that can perform various motor tests, like AC (alternating current) dielectric, surge, and high-voltage PWM (pulse-width modulation) tests, while also controlling various aspects such as temperature and humidity. The test bench is required to automate the testing process and manage each motor coil independently to address the destructive nature of the tests.**

The design takes into account both electrical and mechanical components, and uses a modular approach with Eurocard subracks to house the electronic components, which will facilitate possible future expansions. The system's architecture needs to be able to handle both high-current and high-voltage operations, therefore presenting challenges in circuit design, component selection, and safety measures. Even more, the project includes the development of firmware for microcontroller integration and software for a PC graphical interface which will control the entire testing process.

***Index Terms*—Test Bench, Electric Motor, Windings, Dielectric, Heating, Relays, PCB, Subrack**

I. Introduction

LIFE time test bench is a project whose intent is to increment the existing equipment for testing the lifetime of insulating materials inside the torque motors produced by ETEL S.A.

The new test bench required several key implementations. Firstly, it needed the capability to test motor dielectrics at their operating temperatures. Additionally, it was essential to control the air humidity within the motor testing enclosure. Automation was another critical requirement of the new system, which necessitated individual control of each motor coil due to the destructive nature of the tests. These aspects are illustrated in Figure 1, which provides an overview of the system.

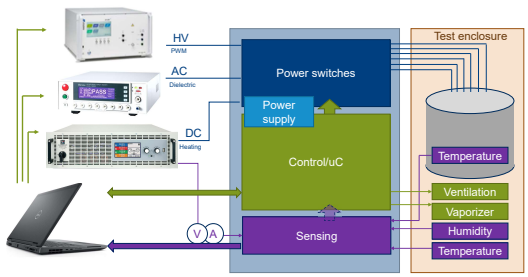

Fig. 1. Test bench overview

The fundamental requirement is to be able to design and construct a single subrack and establish its control system, which includes aspect like circuit and PCB (printed circuit board) design, mechanical aspects, and firmware programming. Even more, it involves establishing the initial steps of the software development.. This will serve as a prototype to demonstrate the feasibility and functionality of the overall design before scaling up to additional subracks.

As for the electrical components, the objective is to build an electronic enclosure (rack) which contains some electronics boards (modules) for coil control.

II. Hardware

First of all, it was fundamental to define all the requirements for the component choice. In particular, the voltage and current levels to operate all the different tests. Which was quickly followed by the design of the circuit to control each coil separately, thanks to different relays connected as shown in Figure 2.

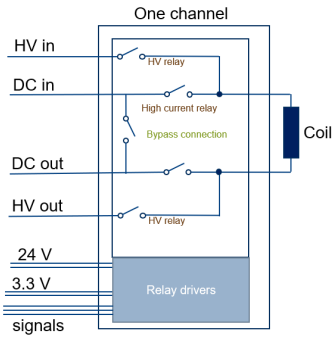

Fig. 2. Channel configuration

Once the components chosen and the relay driver circuit designed (Fig. 3), it was possible to build the first prototype, meant to validate its functionality. Even more, the prototype showed insightful aspects which needed to be addressed. As an example, the safety measures were better ensured by adding a delay circuit on the micro-controller signals to the relays.

The following step into electrical hardware design is to realise the PCBs to install in the electronic enclosure. In order to achieve it, many mechanical constraints needs to be taken care of, starting from the cards dimensions, ensured

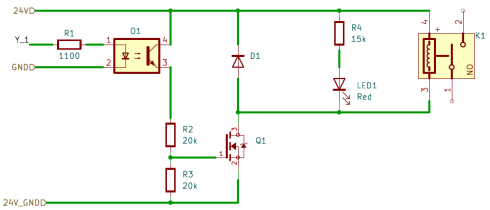

Fig. 3. Relay driver circuit

by the IEEE Std 1101.1, up to components placement and front panels design. This aspects also defined the number of channels per child board and the number of child boards per subrack, as well as the choice of dedicating one board to the management of the power inputs and a place where to install the μcontroller.

The PCB layout can now be carried out, which meant introducing all electrical components and nets into the EDA (Electronic Design Automation) KiCad electrical schematic and then finally placing the components in a way to fit all of them into the card defined space. Moreover, all of the components need to be connected to their correct net which implied addressing the electrical constraints on the copper layer, which were the width in case of the high current tracks (heat dissipation issues) and the distance between traces for the high voltage ones (dielectric strength). This process had to be repeated for each of the boards required by the setup, therefore for the child board (Fig. 4), the inputs board and the motherboard.

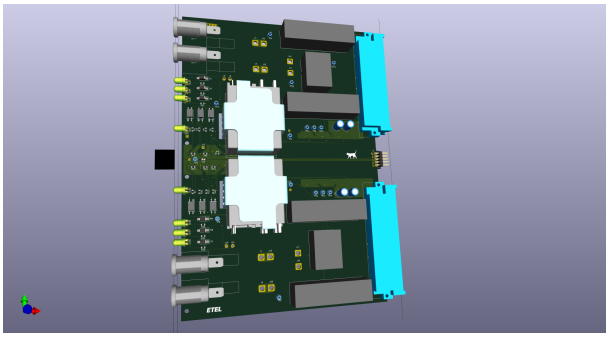

Fig. 4. Child board 3d view

III. Software

Finally, the software side needs to be dealt with, two distinct software parts are required: firmware for the NUCLEO boards (μC) and software for the PC (Personal Computer).

The firmware is responsible for controlling the relays by altering the state of the board's output pins, in accordance to the message received from its master connection. The code was developed using C language within the STM32CubeIDE environment, whereas for the communication it meant establishing a commands protocol. The main challenge was represented by developing an object-oriented program in C,

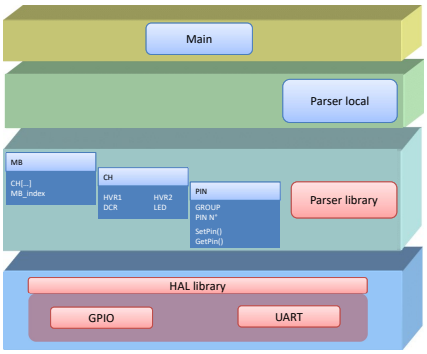

Fig. 5. Firmware layers

a language which is not inherently designed for this approach. The objects, like the channel and the relays, are represented inside Figure 5, which shows all of the code parts and their hierarchical level. As for the communication, the solution was found by basing it on SCPI commands.

Moving on to the PC software, it was developed in MATLAB. The software is meant to provide a graphical user interface (GUI) whose intents are multiple, for example, it needs to show each channel state, with each of its relays.

The graphical side needs to be combined with the object-oriented program, whose structure replicates the one applied to the firmware.

IV. Conclusion

The project's achievement is depicted in Figure 6, which presents a subrack in its testing phase, fitted with its input board and a child board. The white front panel hides a bypass for the high current circuit.

In conclusion, the project has laid a solid foundation for the expansion of ETEL S.A. facilities for testing the lifetime of the insulating materials. The electrical and mechanical designs were successfully developed, the firmware is functional, and the software is capable of performing manual operations on the channels, therefore achieving the fundamental requirement of the internship.

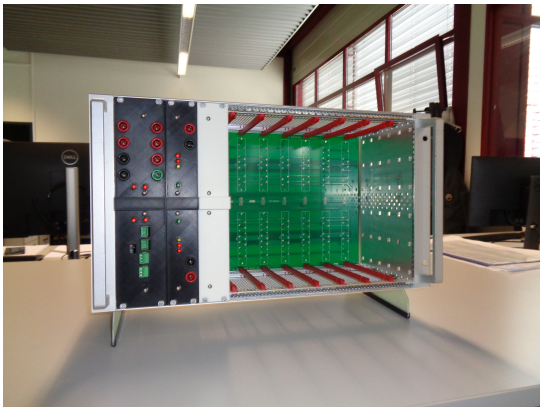

Fig. 6. Final product