Development-board

BE-CREATIVE MINOR

Student Name: Job Heijlighen
Student number: 3111075
Study: Mechatronics

Student Name: Rick Stemkens
Student number: 3007065
Study: Mechatronics

Student Name: Ralph Heijden
Student number: 3095053
Study: Industrial design

Student Name: Rob van Kempen
Student number: 2727366
Study: Mechatronics

Student Name: Maarten Derks
Student number: 3082407
Study: Mechatronics

Student Name: Wojtek van de Heuvel
Student number: 2815079
Study: Electronics

1. Introduction

Most development boards often encounter the challenge that external drivers are needed to use various motors and sensors. This often makes it difficult to test components directly. An extensive development board with the necessary circuits will be an ideal solution.

With this project we wanted to go through the entire design process of the development board. In this way everyone in the project group could learn something about designing electrical schematics, researching components, determining the right components, designing the PCB (Printed Circuit Board) and actually producing and testing the final product.

The development board is an easy to use board with standard screw terminals and an onboard power supply. The hardware on the development board gives you different functions from controlling a low power DC, stepper or servo (2A) to onboard supply of 5V and 12V.

We also created some software templates that come with the development board. You can easily use some function like controlling the motor in different speeds and direction, IO driver, LCD (Liquid-Crystal Display) functions, UART (Universal asynchronous receiver-transmitter) and low power mode.

For us it was mainly about going through the design process. But in order to test all

functions an application has to be devised and developed to actually test and validate all functions. We chose to make a useless box. In this application we can apply all our functions and add functions as we go through the project.

2. Ease of Use

The PCB is designed in such a way that it is relatively easy to use. The focus is on standard functions that can be used from the software library, a robust hardware design and the necessary protections

There are indication LEDs that indicate whether an output is switched and buttons that allow an input to be switched manually. These aspects give the user the possibility to test parts of his software without having to connect peripheral equipment. It is also possible to control a DC motor, stepper motor or servo. It is possible to directly connect 2 DC motors or a stepper motor. Also it is possible to connect 4 Pulse With Modulation (PWM) controlled servos but only when the PWM signal is inverted.

The outputs of the IO expander are controlled by I2C (Inter-IC). However, the user can use the GPIO (General Purpose Input/Output) layer to switch these outputs, so no knowledge of I2C is required. Also, the user can easily send/receive a message using the

UART driver. Last but not least, there is a LED that indicates if the main loop is still running.

3. Research / tests

For creating a new development board, we have to collect and analyze data that is already available, looking at current boards like 'Arduino', that can be expanded to create a new development board.

More options to increase the range. This designed development board can be used for many different products, for our project we have chosen a Useless box. Other products can be, robots (for drawing our just picking products), alarm system etc. It could be everything. This has created a board that has been made user-friendly and extended to a higher level. More possibilities.

The development board is so extensive that there are many inputs and outputs present in combination with 5v and 12v input. Because of this versatility the board can be used in a wide range of applications. And with knowledge of STM32IDE written software programs can be applied to this development board in combination with the application.

The useless box is our application that we designed for the development board. Below you will find which elements are used and which can be controlled by the written software.

- Motor driver
    - Servo motors;
    - DC-motors;
- Sensors
    - Moisture Sensor;
    - Acceleration sensor;
    - Ultrasonic sensor;
- LCD- screen
- LED-lights
- Toggles for activation of the application or to stop an engine when it comes to its endpoint.

The points mentioned above have been applied in the application (Useless box). The Useless box is a nice product, and it is interactive so interesting for all categories.

4. Software

The main goal of the software part of this project is to create some templates that can be used for each project. With these templates everyone needs to understand how it works. The goal is to create the functions as easily as possible in an Arduino style.

There is a standard IDE which belongs to our STM32 microcontroller named STM32 IDE. In this IDE is an initialization code generator that integrates what's used to generate the basic software files for the microcontroller settings.

For the design of the software architecture, the software is divided into layers. The standard layers that comes from the STM32 code generator is the STM32 HAL (Hardware Abstraction Layer) In this layer contains all the functions directly related to the registers. At the top of this layers there is a development board layer. In this layer the functions in which the HAL functions are applied. The goal of this layer is to make the functions as easy as possible and to incorporate extra safety and feedback functionality. The top layer is the application layer. This layer can be used for the application.

The function that's been created in this minor are:

- GPIO
    - IoWrite(OutputNumber, State)
    - IoToggle(OutputNumber)
    - IoRead(InputNumber)
- UART
    - serialcom_SendString(Data)
    - serialcom_ReceiveChar()
- Motor driver
    - Motor(MotorType, Direction, Speed)
- LCD

        - lcdPrint(Data, Row, Column)
        - lcdScroll(State)
        - lcdCursor(State)
- LowPowerMode
        - EnterSleepMode()
        - EnterStopMode()
        - EnterStandbyMode()

All this functions are easy to use and can directly implemented in your application code. The pin numbers of this functions are represented on the development board.

5. Hardware

Besides the software there was also an aim for a piece of hardware. This will be realized with a PCB. All the different parts of the research will be added to this.

To realize this, we have to make a schematic design and a board design on the computer. For this there are a couple of software programs. We will use Eagle for this. The program is easy to use and you can easy add libraries to it.

First we wanted to make some test PCB's to test some parts of the total schematic. If everything works, a first concept of the development board can be made. Our planning is that we order the final PCB at the end of the minor at JLCPCB.
This is a company in China that makes PCB's at low costs with a big order amount. That means that JLC is not only accessible for big companies, but also for students like us. Furthermore there is an option to let them solder components for you on the board. This only applies SMD (Surface Mount Device) components. We have to take that into account for the design of the board because we are also going to use through hole components like pin headers and screw terminals.

At the end the PCB has to fit and work in our application. This will be a useless box. Inside that box will be a couple of motors and sensors to make it work.
That means that on the development board a motor driver must be fitted. Furthermore an I2C and UART control IC must be used to read the sensors.

6. Conclusion

The purpose of this project is to design and produce an easy to use development board. This is defined and subdivided into hardware and software based points:

- <u>Hardware</u>
        - Included screw terminals besides the pin headers.
        - Onboard power supply, switchable between 5V and 12V.
        - Controlling of low power DC, stepper or Servo motors (2A).
- <u>Software</u>
        - Software templates that come with the development board which support the use of functions for: LCD, UART, I2C, IO driver, Motor speed and Low power mode.
        -

As a result of the mentioned points above, a PCB is designed using Eagle and realized by JLCPCB. Together with this PCB there are software templates made in STM32CubeIDE to create functions as easily as possible in an Arduino style. The software design is divided into layers. The standard layers that comes from the STM32 code generator is the STM32 (HAL) and Register layer. The top layer is the application layer. This layer can be used for your own created application. All of the functions are easy to use and can directly implemented in your application code. The pin numbers of this functions are represented on the development board.

In order to prove that the development board supports all pre-defined requirements, a useless box has been created.