

Integrating project

by Jason Hopkins

FILE	11839_JASON_HOPKINS_INTEGRATING_PROJECT_173939_1819588035.DOCX (2.83M)		
TIME SUBMITTED	04-MAY-2019 02:54 PM (UTC+0100)	WORD COUNT	1884
SUBMISSION ID	106313828	CHARACTER COUNT	8933

University of Wales Trinity Saint David

School of Applied Computing

Lego Mindstorms Project

Integrating Project M4X01506

Jason Byron Hopkins

1805507

04/05/2019

Table of Contents

1	Introduction.....	1
1.1	Aims.....	1
1.2	Objectives	1
2	Project Plan.....	2
2.1	Group Structure.....	2
2.2	Time Allocation	2
3	Design Process.....	3
3.1	Robot Kit.....	3
3.2	Robot Design	3
3.3	First Built Design.....	5
3.4	Final Design	6
3.5	Obstacle.....	7
4	Programming.....	8
4.1	Line following.....	8
4.2	Obstacle.....	9
4.3	Maze.....	9
4.4	Final program.....	10
5	Conclusions.....	11
6	References.....	12

1 Introduction

1.1 Aims

As a group, build and program a robot that will be able to navigate a set course from a starting point to an end point. The robot will need to follow a line on the ground, deal with an obstacle on the lined path and then pass a coloured line on the floor. Once passed move through a 3D built maze and reach a red line or obstacle the robot will then indicate that it has reached the end.

1.2 Objectives

To achieve this the following will be done:

- Set key roles for each member of the group.
- Research and evaluate what methods may be best suited for the different tasks in the course.
- Use Lego Mindstorms [1] hardware and software to build and program the robot.
- Trial different designs and methods of the robot to find the most convenient of solutions.

2 Project Plan

2.1 Group Structure

The group consists of three people; Jason Hopkins, David Isaac and Hafiz Uddin. After discussions it was decided that the design of the robot would be built as a group, but the programming of the robot will be split into individual sections of the obstacle course. These sections would be led by individual leaders who would be supported by the others as follows;

- Line leader – Jason Hopkins
- Obstacle leader- David Isaac
- Maze Leader- Hafiz Uddin

Jason Hopkins would oversee the program of the robot so it would be able to follow the line. David Isaac would oversee the programming of the robot as it meets the obstacle. Hafiz Uddin would program the robot so that it would be able to navigate through the maze portion of the course.

2.2 Time Allocation

Once the team roles were allocated the group worked out a 5-week plan on what to achieve each week to optimise resources and not waste time on non-critical matters.

For the first week the group would research the main requirements of the robot, what sensors would it need and how would it negotiate the obstacle. Then find a suitable design for the robot that would be able use the chosen sensors.

The second and third week would be allocated to programming and testing the robot. Two weeks were selected in case of any changes to the design that were required based on findings from the testing.

The fourth week would be allocated to the final designing of the robot and testing of said design.

For the fifth and final week the robot will be given its final testing then would be placed onto the course.

3 Design Process

3.1 Robot Kit

The Lego Mindstorms robot comes with a kit that provides building blocks for the robot, an ev3 brick that can be programmed, three motors (two large, one medium) and a selection of sensors.

From conducting research, the LEGO Mindstorms website [2] indicated that to allow the robot to follow a line, a colour sensor would be needed. The colour sensor would allow the robot to measure the colours that passed in front of it or could pick up the light intensity of a source. With this information, the robot could be programmed to move in certain directions when certain conditions are met.

Hafiz Uddin indicated for the maze portion of the course, an ultrasonic sensor would be required for the maze to detect the distance of the robot from an obstacle. By using this sensor, the robot could react before it collided with a maze wall.

For the obstacle on the line David Isaac suggested that a medium motor could be used to move the obstacle out of the way by building an arm or claw. The ultrasonic sensor could also be used for this part of the course as well. For movement, the robot would use two large motors which could be used for tracks or wheels.

3.2 Robot Design

Once the main parts were decided on the next part was to build the robot itself so that the sensors could be placed on it. Three main designs were considered as a basis for the robot.



Figure 1- Riley Rover Design

The first considered design was the Riley Rover robot [3]. This design had been used by the group before for several tasks to learn about Lego Mindstorms. While this design allowed for plenty of room for sensors to be added on, the group felt that the design was not unique and was a basic design.

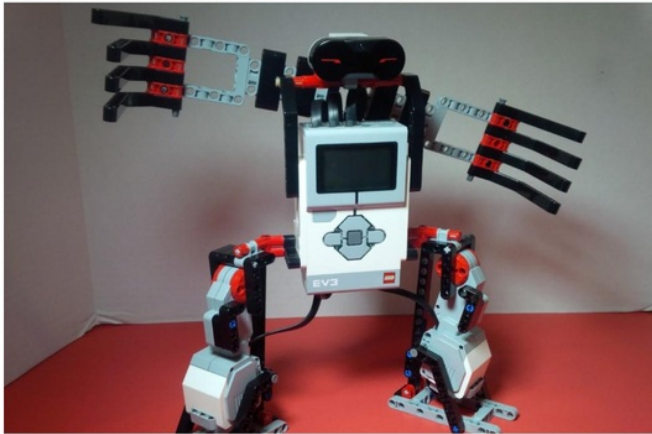


Figure 2 Ev3 Basket Ball Robot [4]

The next design considered was figure 2. This design could allow the robot to move the obstacle away with its arms and provide a unique design by walking through the maze. However, by using this robot there would be little room for adding on extra parts like the colour sensor and after measuring the maze walls it was deemed too tall for the ultrasonic sensors to detect the walls.



Figure 3 BOBB3E Robot [5]

The third design figure 3 [5] was the base design that the group had decided upon. The BOBB3E design featured a forklift which could lift the obstacle and had rubber tracks which would provide more grip on the surfaces of the course. Several modifications were needed to adapt it to the group's requirements.

3.3 First Built Design

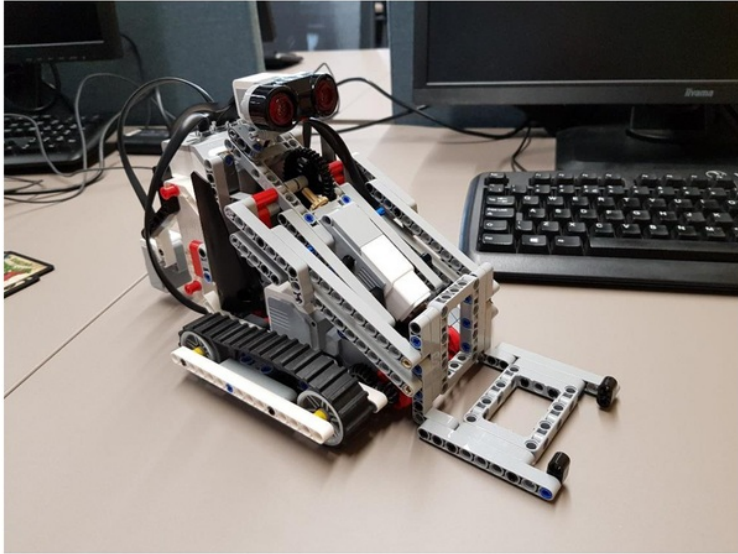


Figure 4 First Design Front

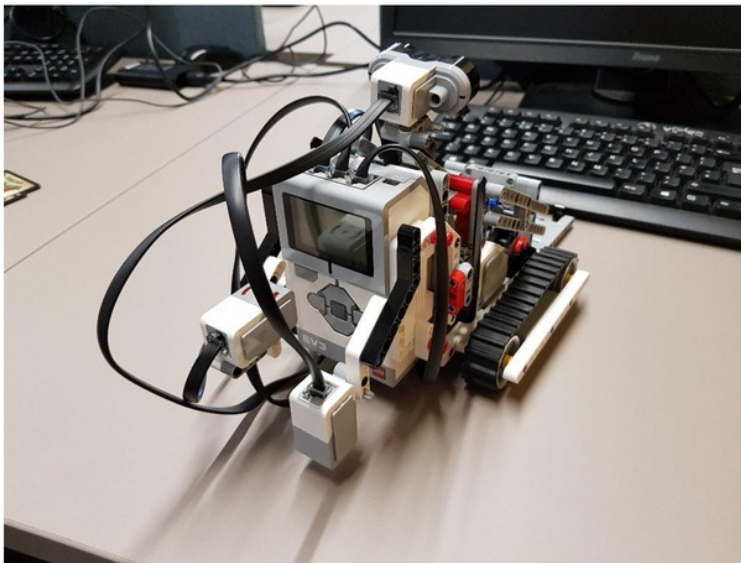


Figure 5 First Design Back

As shown in figure 4 the robot design was mostly kept the same in the front. The colour and gyro sensor were placed at the back of the robot as seen in figure 5. The gyro sensor would measure the amount of degrees that the robot has turned.

After some minor testing, some issues were discovered. The first issue was that the colour sensor was behind the robot and so was making it harder for it to direct the robot because the robot was moving ahead of the sensor. The second issue was that the ultrasonic was still too high for the maze walls. The design of the front of the robot however did not allow much room for modifications. To fix this issue the group decided to flip the robot around.

3.4 Final Design

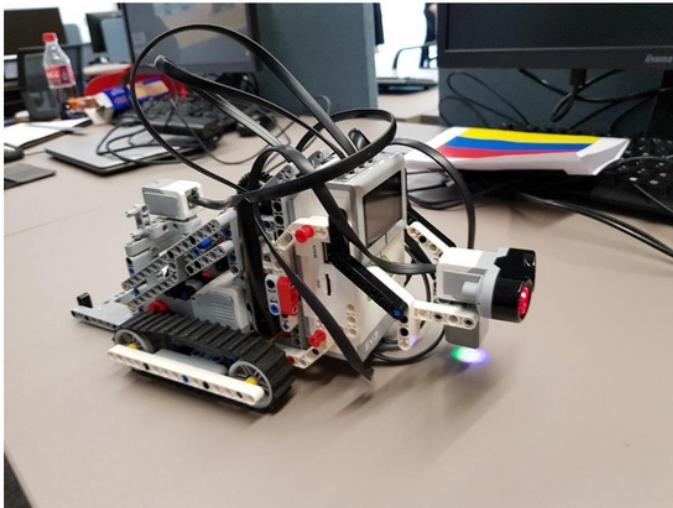


Figure.6 Final Design Front

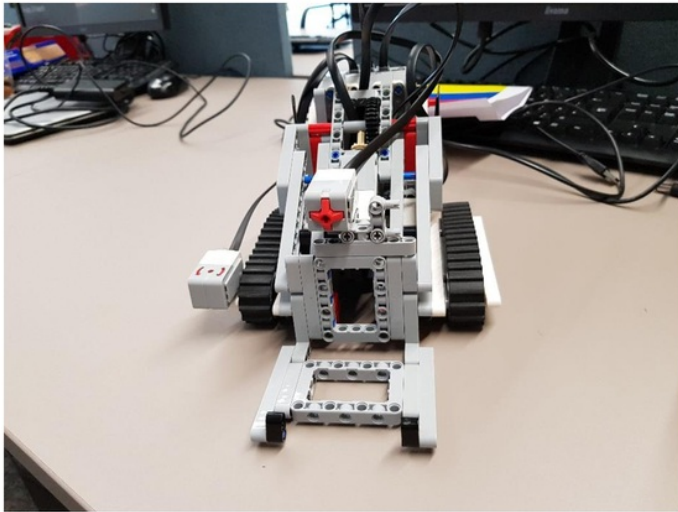


Figure 7 Final Design Back

For the final design shown in figure 6 and 7. The ultrasonic sensor and colour sensor were now the front of the robot and had been placed together. The gyro sensor was placed to the side of the robot and a touch sensor was added to the forklift so that the robot could detect when it had hit the object.

3.5 Obstacle



Figure 8 Course Obstacle

For the obstacle of the course David Isaac built an object from spare Lego parts. The base of the object had an opening so that the forklift could fit under and then lift it out of the way.

4 Programming

4.1 Line following

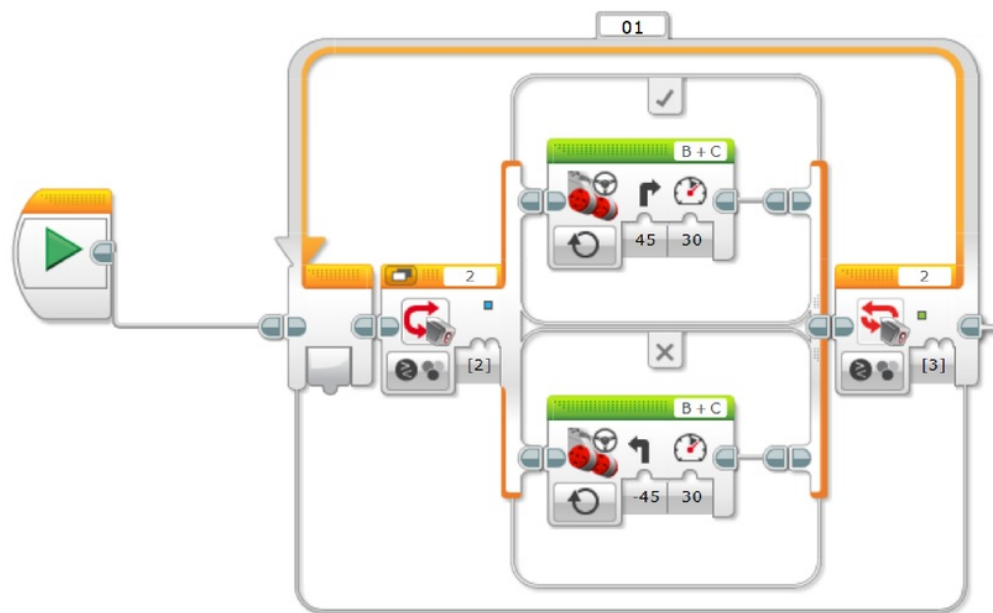


Figure 9 Line Following Program

Using Lego Mindstorms software each person programmed their own section of the robot. Jason Hopkins programmed the line following part, shown in figure 9. This program would make the robot move with certain conditions. The line that would be used for the course would be two lines of two colours, blue and red.

If the robot detected the colour blue with the colour sensor, the robot would move forward and right using both motors at a speed of 30. If it did not detect the colour blue, then it would turn forward and left at a speed of 30. The robot will loop these conditions until it detected the colour green which would be the colour at the start of the maze.

Several speeds were attempted for the robot. At slower speeds the robot would still be able to follow the lines accurately, but it would not be time efficient. Higher speeds would make the robot follow the path quicker however when it reached a corner the robot would more likely overshoot the line and go off course.

4.2 Obstacle

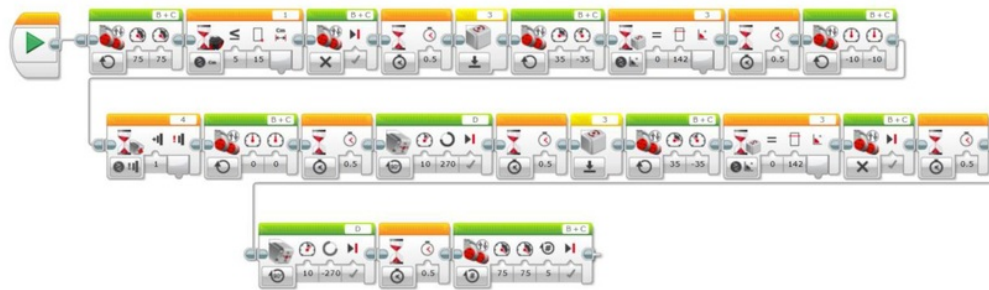


Figure 10 Obstacle Program

David Isaac's program figure 10, allowed the robot to deal with the obstacle in the path. Once the robot detects an object less than 15cm away from the ultrasonic sensor the robot will turn 180 degrees using the gyro sensor and then reverse until the touch sensor gets pressed. Once pressed the medium motor will move the forklift which will cause the obstacle to lift off the ground. The robot then turns 180 degrees again and then drops the obstacle to the ground. After which the robot carries on moving.

4.3 Maze

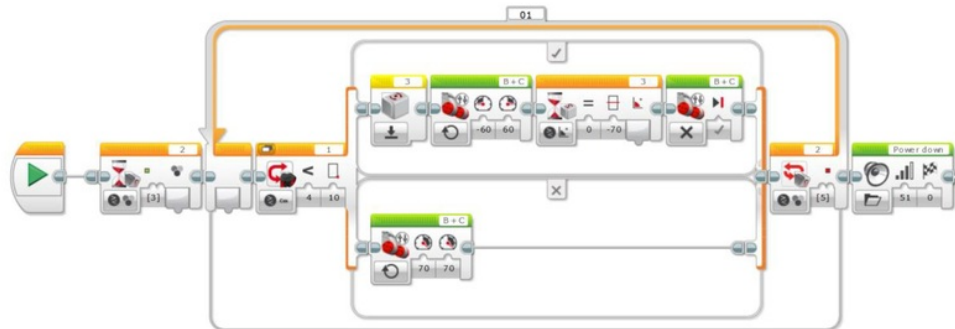


Figure 11 Maze Program

The maze program shown in figure 11 was created by Hafiz Uddin. The robot would wait until it detects the colour green. Once detected the robot will move forward until it detects an object with its ultrasonic sensor less than 10cm away. If it detects an object the robot will then using the gyro sensor turn left 90 degrees. The robot will keep repeating this program until it senses a red colour, signalling the end of the maze. Once it has stopped it will play a tune.

4.4 Final program



Figure 12 Final Program

Figure 12 shows the final programming of the robot. The final build combines each group members program together. The Robot now follows the line until it detects the obstacle, or it detects a green colour. If it detects an obstacle it will carry out the obstacle program and then loop back into the line following program. Once it detects the colour green it will then follow the maze program.

5 Conclusions

With the design and programming the robot was successful in completing the course with no issues. If the group were to attempt this project again there would be a few changes to the design of the robot to improve the results further. For the line following the robot would follow a single line smoother rather than turning repeatedly and slowing it down. For the maze the group could add different sensors to the robot to help adapt it to sudden changes to the layout of the maze rather than have it turn in only one direction.

6 References

- [1] LEGO, “Lego Downloads,” *LEGO Mindstorms EV3*, [Online]. Available: <https://www.lego.com/en-gb/mindstorms/downloads/download-software>. [Accessed May 1, 2019].
- [2] LEGO, “How to make a LEGO MINDSTORMS robot follow a line – Discover the EV3MEG fan robot,” *LEGO Mindstorms EV3*, [Online]. Available: <https://www.lego.com/en-us/mindstorms/videos/fan-robot-ev3meg1-87c0bd0d1dc843f1a907e20903d15596>. [Accessed May 1, 2019]
- [3] D. Kee, “RILEYROVER - EV3 CLASSROOM ROBOT DESIGN,” Damien *Kee*, Aug. 2, 2013 [Online]. Available: <http://www.damienkee.com/rileyrover-ev3-classroom-robot-design/>. [Accessed May 1, 2019].
- [4] Teach Kids Engineering, “Our Favorite Lego Mindstorms EV3 Projects,” *Teach Kids Engineering*, Nov. 6, 2015. [Online]. Available: <https://teachkidsengineering.com/our-favorite-lego-mindstorms-ev3-projects/>. [Accessed May 1, 2019]
- [5] K. R. Madsen, “BOBB3E- Build a Robot,” *LEGO*, [Online] Available: <https://www.lego.com/en-us/mindstorms/build-a-robot/bobb3e/>. [Accessed May 1, 2019].

Integrating project

GRADEMARK REPORT

FINAL GRADE

81 / 100

GENERAL COMMENTS

Instructor

Report (60%) overall 77%

Introduction - 4/5

clear aim and SMART objectives. introduction could have been improved with more discussion of the required tasks.

Background research - 10/15

good reference list and in text citation. there is evidence of the engagement with the literature

Project plan - 8/10

clear project planing process.

Design Process - 35/40

very detailed design process. highlighting of the iteration in designing the hardware and the software.

Evaluation and Results - 20/30

the results analysis is mostly combined in other section. the conclusion is very brief. but the over all quality of the results is of a good standard.

Demo (40%) - 86% overall

Technical knowledge/skill - 17/20

Analytical and problem-solving skills - 17/20

Line followed smoothly, object removal negotiated well, Design of root is robust and very suitable. Maze is negotiated well, but the gyro is not always accurate on different surfaces. A side sensor to mitigate collisions (eg push or ultrasonic) would help for different environment conditions. We enjoyed the Star Wars finale!

Overall mark for module: 81%

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14
