

Path Control iRobot Create2 Based Sensors

Narong Aphiratsakun^{1,*}, Santi Liwsakphaiboon²

¹Department of Mechatronic Engineering; ²Department of Computer Engineering, Assumption University, Samut Prakan, Thailand

Received 13 January 2020; received in revised form 06 May 2020; accepted 10 August 2020

DOI: <https://doi.org/10.46604/peti.2020.4130>

Abstract

Robots become a part of humans' daily lives. They work faster, more accurate, and tireless compared to humans. As technology develops, robots are more capable of serving people in more tasks than before. The objective of this study is to explain the methodology of controlling iRobot Create2 by using Arduino Uno to command iRobot Create2 to use operation codes. Instead of using a brute-force way to control the robot motion by commands, which may cause the whole path to go wrong if only one command is deflected. However, with the sensors, the robot can adjust the direction path despite the deviation. The Arduino Uno communicates with iRobot through the serial communication using SoftwareSerial. Arduino requests a sensors data from iRobot and calculates deviations in order to adjust the path. The study includes the general knowledge of Arduino control iRobot Create2 based on the robot's sensors and the detailed process of completing the first round of iRobot Competition.

Keywords: arduino, coding, control, iRobot create2, sensors

1. Introduction

iRobot Create2 [1] is an affordable STEM resource for educators, students, and developers to learn robotic control. It is a vacuum cleaning robot that is programmable and is used to grasp the fundamentals of robotics, computer science, and engineering [2-3].

Some previous projects about iRobot controlling are Camera Bot [4] pairing your Create2 with a microcomputer, a camera then driving around and seeing what your robot sees. DJ Create [5] is a roaming of the robotic DJ for your party by integrating iRobot with a Bluetooth module. However, most of the programs, like DJ Create, cannot control the path of iRobot effectively.

iRobot company set the robot controlling competition [6], the Assumption team took part in and passed the first round, also qualified for the final round of this competition. This study shares their experience and details in the robot controlling and the method of how to achieve.

2. System

2.1. iRobot create2

iRobot Create 2 in Fig. 1 is explicitly designed for a development of the robotics. The iRobot Create 2 replaces its Roomba predecessor's vacuum cleaner hardware with a cargo bay that also houses a DB-25 port to provide a serial

* Corresponding author. E-mail address: oofpokemon@gmail.com

Tel.: +6687-349-3482, +4176-223-0109

communication, digital input & output, analog input & output, and an electric power supply. The iRobot Create2 also has a 7-pin Mini-DIN serial port connector which sensor datas can be read and motor commands can be issued using the iRobot Roomba Open Interface (ROI) protocol.



Fig. 1. iRobot Create2

2.2. Mini-DIN connector

To use the OI, a processor which is capable of generating serial commands such as a PC or a microcontroller must be connected to the external Mini-DIN in Fig 2 connector on Roomba. This connector provided two-way, serial communication at TTL (0-5V) levels. The connector also provided an unregulated direct connection to Roomba’s battery, which can be used to power the OI applications in Table 1.

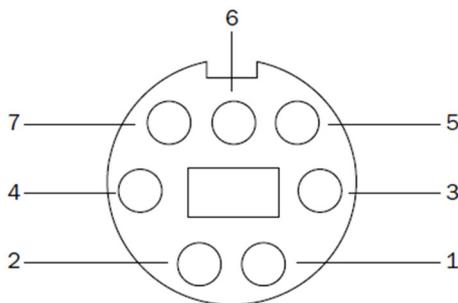


Fig. 2 Mini-DIN connector

Table 1 Mini-DIN pin table

Pin	Name	Description
1	Vpwr	Roomba battery + (unregulated)
2	Vpwr	Roomba battery + (unregulated)
3	RXD	0 – 5V Serial input to Roomba
4	TXD	0 – 5V Serial output from Roomba
5	BRC	Baud Rate Change
6	GND	Roomba battery ground
7	GND	Roomba battery ground

2.3. Controller: arduino uno



Fig. 3 Arduino Uno R3

Arduino Uno microcontroller board in Fig. 3 is based on the Microchip Technology ATmega328 8-bit Microcontroller (MCU). Arduino Uno features 14 digital input/output pins (six of which can be used as PWM outputs), six analog inputs, and a 16MHz quartz crystal. The ATmega328 on the Uno comes preprogrammed with a bootloader that allows the user to upload a new code to the MCU without the use of an external hardware programmer.

3. Task

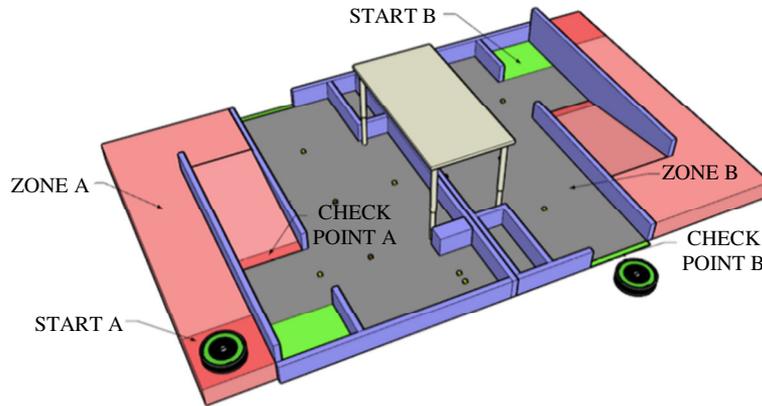


Fig. 4 Stage 3-D model

The competition’s rules and qualifications are as followed:

1. The robot used in the competition has to be iRobot Create2 which is given by the organizer at the beginning, then the robot must not be modified and/or added any sensor.
2. Only Arduino Uno Rev 3 is allowed to be used as a controller board.
3. The size and the position of objects (including micro bricks) in the stage are known before the competition day. (Figs. 4-5)

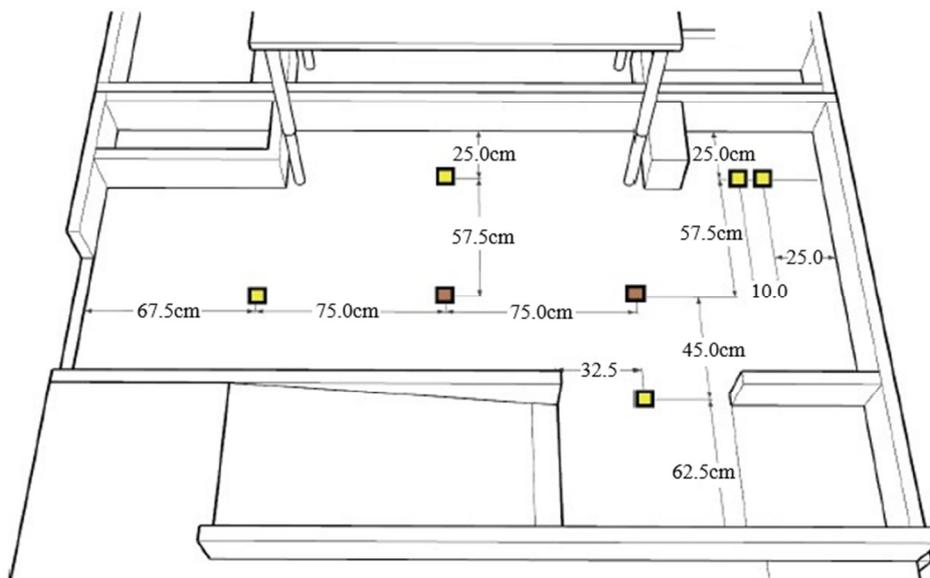


Fig. 5 Position of micro bricks

The task is divided into two stages: the first stage is for testing simple robot control skills and the second stage requires more control skills to precisely pick up yellow micro bricks and dodge brown micro bricks. Each stage’s rules are listed below.

The First Stage Rules:

- (1) When the robot reaches the first checkpoint, it gains 150 scores.
- (2) One chance for retrying if the robot falls from the stage cliff.

- (3) The robot hits the wall of the stage, it loses 25 scores per hit. Maximum loss 100 scores.
- (4) Two minutes is the limit for the first stage. If exceeds, 0 score for first stage and can proceed to the second stage.
- (5) The robot can either continue moving right after reaches the first checkpoint or stop.

The Second Stage Rules:

- (1) The robot reaches the destination area and the whole parts of the robot exits from the stage, it gains 100 scores.
- (2) The robot hits the wall of the stage, it loses 25 scores per hit. Maximum loss 100 scores.
- (3) Three minutes is the limit for the second stage. If exceeds, 0 score for the second stage.
- (4) Gain 50 scores per yellow micro brick (Fig. 6(a)) is collected (Total 5 yellow micro bricks in stage).
- (5) Lose 50 scores per brown micro brick (Fig. 6(b)) is collected (Total 5 brown micro bricks in stage).
- (6) If it does not use retry chance in the first stage, it can be used in the second stage instead.

The full score is 500 scores, the ranking is determined from the highest to lowest and fastest to slowest if the scores are equivalent.

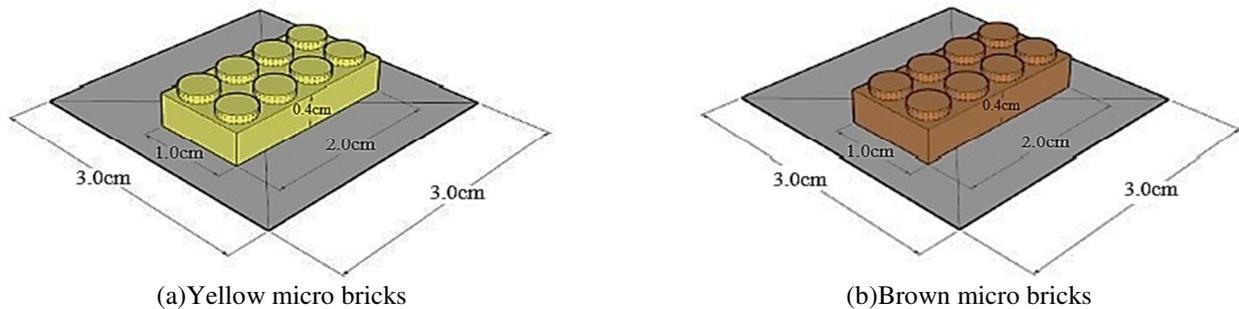


Fig. 6 Micro brick

4. Control Method

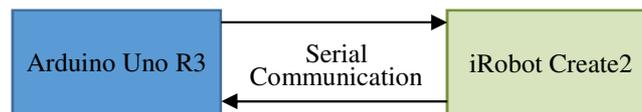


Fig. 7 Serial Communication

Serial Communication is the process of sending a data one bit at a time, sequentially, over a communication channel or computer bus. It is available in Arduino Uno (known as UART) to be used for the communication between the Arduino board and other devices.

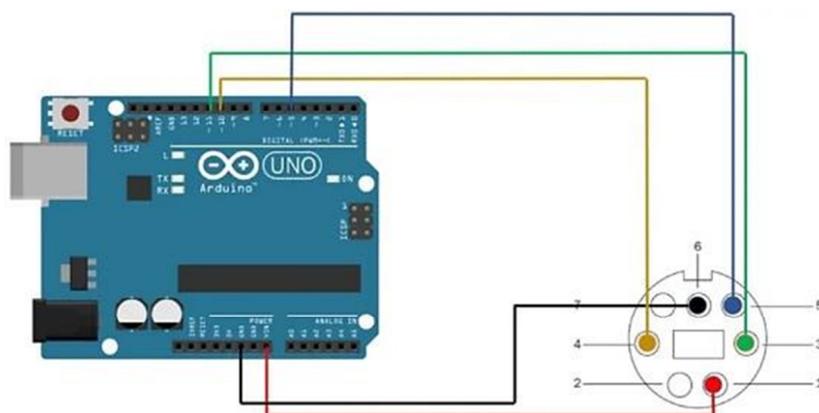


Fig. 8 Arduino-Create2 connection

The methodology of design a path control for iRobot used sensors to check the position, Fig. 7 used sensors of iRobot to receive data then send to Arduino, and Arduino made the decision which direction to go by sending opcode to iRobot through wires (Fig. 8).

5. Control Program

The program in Arduino was divided into 2 part for 2 stages and used the button on iRobot to run the program (Fig. 9).

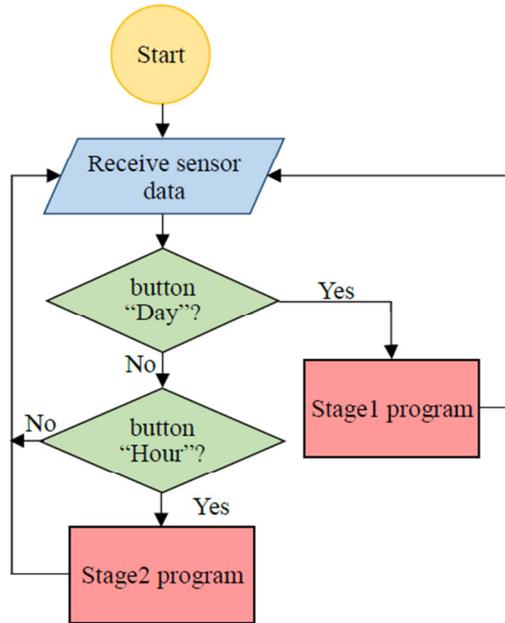


Fig. 9 iRobot control flowchart

Below is the flowchart and explanation for each program.

5.1. The first stage flowchart (Fig. 10)

- (1) After the program starts, first it resets the flag, sets distance and radius for iRobot.
- (2) Next, Arduino checks the value from the wheels encoder if iRobot reaches the distance, goes to check flag state.
- (3) If iRobot doesn't reach the distance set, Arduino requests a wall sensor and a cliff sensor from iRobot.
- (4) If detects the sensor of a wall or cliff, Arduino sends command to iRobot to adjust the direction angle to left or right.
- (5) If doesn't detect the sensor of a wall or cliff, iRobot keeps driving straight.
- (6) If reaches the distance, Arduino will check the flag status.
- (7) If a flag is not set, turns iRobot in radius and set the flag.
- (8) If a flag is set, stops iRobot and exits the program.

5.2. The second stage flowchart (Fig. 11)

- (1) Once program starts, iRobot moves out from the start point, turns the counter-clockwise, then initializes all variables.
- (2) Request the wheels encoders to compute the traveled distance and check if it equals to the distance set or not.
- (3) If doesn't reach the distance, checks for wall, here iRobot will keep moving along the wall. Otherwise, adjust the direction.
- (4) When reaches the distance set, iRobot moves in a sequence of the actions to collect the last micro brick as it position is not near the wall.

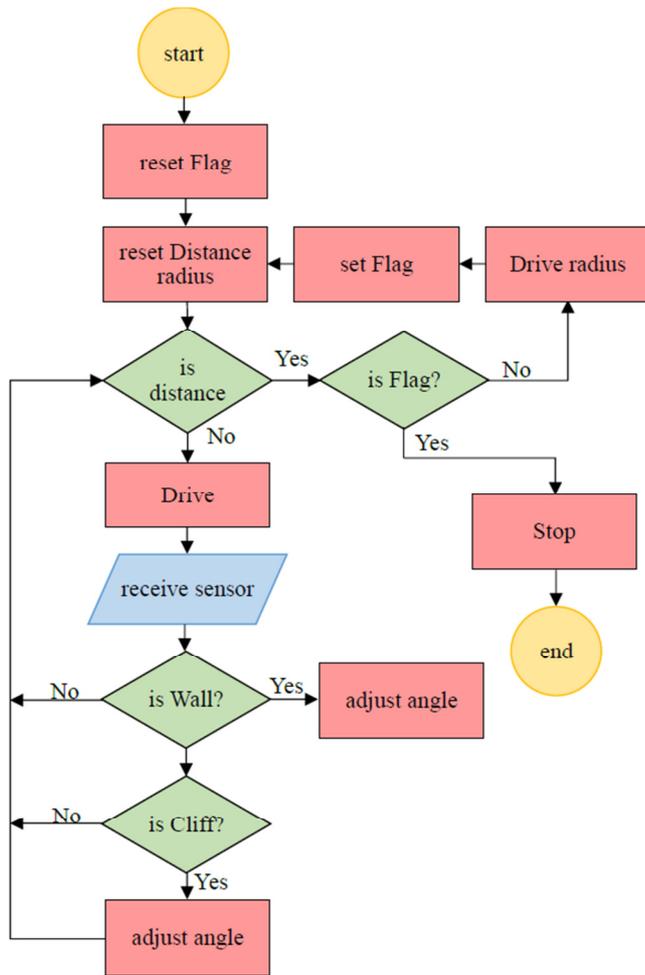


Fig. 10 Stage 1 program flowchart

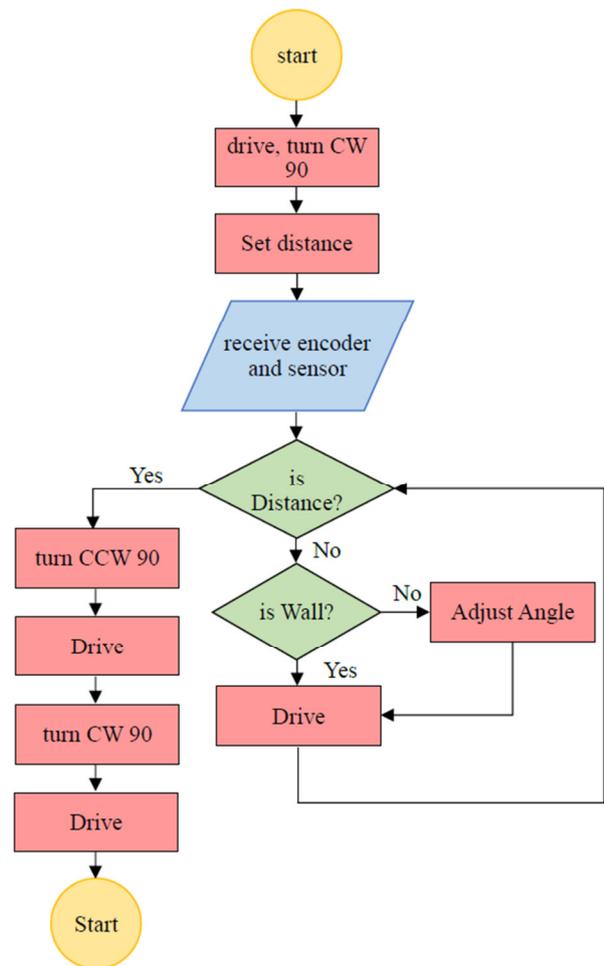


Fig. 11 Stage 2 program flowchart

6. Experiment and Result

The programs were tested 30 times per program. For the first stage (Fig. 12), the accuracy of iRobot moving path is 67% (20 success out of 30 times). The success here means iRobot can move from the start area to the checkpoint area without hitting the wall or falling from the cliff. Most of the errors occur is that the robot does not move in the expected direction. Sometimes it does not adjust the angle and hit the wall.

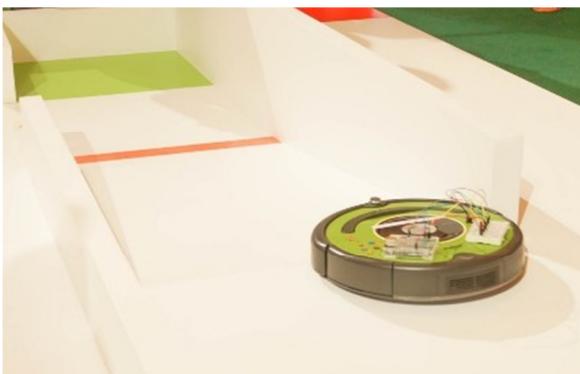


Fig. 12 First Stage testing



Fig. 13 Second stage testing

The root cause of this error happened when a duty cycle of the motor speed is more than 58%, iRobot receiving the data speed is slow down. The problem can be solved in 2 ways. The first method is by using Arduino Leonardo which has 2 internal modules UART (rs-232). The first UART connects to USB and the second one connects to Pin0 and Pin1 of the board. This way, Arduino can communicate with iRobot without using the SoftwareSerial. The second method is simply to adjust a duty cycle of the motors below 58%.

For the second stage (Fig. 13), the test result is that the robot can collect all yellow micro bricks 18 times out of 30 times without collect any brown micro brick. Collect 4 out of 5 yellow micro bricks (0 time) and collect 1 out of 5 yellow bricks (3 times).

7. Conclusion

This study explained the method of controlling iRobot Create 2 using Arduino Uno command and opcode. The control flowchart was used to qualify for the final round of iRobot competition.

In the competition, iRobot Create 2 passed the first stage with a full score (plus 150 marks) and managed to collect 4 yellow micro bricks (plus 200 marks) as robot hit the wall 4 times in the second stage, 100 scores are deducted for the penalty, resulting in having 350 scores. Total time using for stage1 and stage2 is 125.30 seconds.

Using the sensors and wheels encoders is efficient but will encounter a serial communication problem that it is different between the Arduino Uno's speed of receiving the sensor data speed and iRobot's sending speed. For further improvement, the control program is needed to be improved in order to increase the efficiency of the robot.

Acknowledgment

We wish to thank Smith Simargool and Mustafa Dadarananda for their assistance with the testing statistic used in this paper. We would like to acknowledge the support provided by Assumption University during the preparation before competition.

Conflicts of Interest

The authors declare no conflict of interest.

References

- [1] iRobot, "iRobot® Create® 2 Open Interface (OI) Specification based on the iRobot® Roomba® 600," https://www.irobotweb.com/-/media/MainSite/Files/About/STEM/Create/2018-07-19_iRobot_Roomba_600_Open_Interface_Spec.pdf.
- [2] iRobot, "Create 2 Programmable Robot," <https://edu.irobot.com/what-we-offer/create-robot>.
- [3] "iRobot Create," http://en.wikipedia.org/wiki/IRobot_Create.
- [4] iRobot, "Camera Bot," <https://www.irobotweb.com/-/media/MainSite/PDFs/About/STEM/Create/camerabot.pdf>.
- [5] iRobot, "Dirt Cheap DJ Create® 2 Bin," <https://www.irobotweb.com/-/media/MainSite/PDFs/About/STEM/Create/djcreate.pdf>.
- [6] N. Aphiratsakun, "iRobot Create – The Next Chapter of smart living," pp. 19, 22-23, 2018.
- [7] THrobotics, iRobot Rules of the competition of round 1, 2018.



Copyright© by the authors. Licensee TAETI, Taiwan. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY-NC) license (<https://creativecommons.org/licenses/by-nc/4.0/>).