'Cruise-and-Collect' algorithm for an ARM-based autonomous robot

M.N. Zarina[1], M.S Sani Irwan, M.Y. Norhidayah, A. Nur Alisa and M.S. Muzalifah

[1]Fakulti Kej. Elektronik & Kej. Komputer (FKEKK), Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia
[2]Oita University, Oita 870-1121, Japan
(Tel: +6-06-5552058, Fax: +6-06-5552112)

[1]zarina.noh@utem.edu.my

Abstract: The 'cruise-and-collect' algorithm proposes in this research is intended to be applied for an autonomous robot system. The robot is designed to be used in a sport such as table-tennis, to help the players collect the off-side ball during a game. The controller for the robot system will utilize ARM-based microcontroller chip from Atmel, AT91SAM7S128, as it is the classic ARM7TDMI processor that had marked a great success of ARM processor market. The algorithm is developed via *Flowcode* software which exploits the flowchart as its design entry. The E-block board that has an integrated microcontroller chip is used for the purpose of its hardware prototype and software testing. It is hoped that the development and design process exposed in this research will be benefitted for researchers who are interested in the area of prototype development and robot design, especially via the ARM7TDMI processor platform**.**

Keywords: ARM-based processor, AT91SAM7S128, E-block, *Flowcode*.

1 INTRODUCTION

The concept of autonomous robot reflects that a robot can perform a desired task in an unstructured environment without continuous human guidance, or with minimal human control [1]. Depending on its application, the design approach might cover several modular areas; each carries a certain degree of autonomous in the robot's development.

In developing an autonomous robot for assembly line (typically used in warehouse or factory), the focus is given more to the assembly plan itself [2]. Successful assembly plan ensures that each of the pieces to be assembled in the warehouse is defined clearly in each of its single step. An algorithm related to each of these assembly steps are written such that the robot will perform those task consequently and independently. A feedback scheduling might also be provided in the algorithm to avoid repetition of its assembly step [2]. Compared to the design of an Autonomous Underwater Vehicle (AUV), the focus is more towards the navigational path of the autonomous robot because the robot is moving [3]. In this case, the algorithm design concentrates on collision avoidance and escape route by gathering the object geometries and its relative distance to the robot (reactive system). Furthermore, advancement to the reactive system was proposed, such that a configured database (known surrounding information and maneuver reaction) is added to the component. With this configured database, the reactive system will have additional information that helps the autonomous robot to plan for its navigational path effectively besides its real-time information gathering [3]. Later, the algorithm for autonomous robot navigation was improved, resulting in a real-time hierarchical POMDP (Partially Observable Markov Decision Processes) [4]. This improved algorithm act as a unified framework for autonomous robot navigation in dynamic environment, where it decides for the robot movement in every time step execution. The algorithm also caters for planning, localization and local obstacle avoidance; those are the traditional approach for an autonomous navigational system [2].

2 AUTONOMOUS ROBOT DESIGN ISSUES

Gathering from the literatures [2][3][4], there are several design issues need to be addressed in the development of an autonomous robot. For the purpose of this paper, only two of the design issues will be highlighted. The first issue is the navigational path of the robot; either it is predetermined or unknown. If it is unknown, several sensors need to be listed as part of the feedback circuit in configuring the robot's paths and movements. If it is predetermined, the navigational algorithm needs to be manipulated such that it reflects the predetermined path. The paths need to be clearly specified, so that the robot will be fully autonomous.

Next issue is on the definition of the application of the autonomous robot itself. As such, the algorithm and sensors related need to be configured to perform the required task. The application should also reflect the autonomous robot requirement in its design.

With these two design issues clearly described, the autonomous robot proposed in this research can be developed. Still, the core target in this project will not be forgotten that is, to explore the functionality of ARM-processor platform for robotics application.

3 DEVELOPMENT APPROACH

The prototype of the autonomous robot developed in this research is intended to be used in table-tennis (ping-pong) sport, as a ball-collector (applying the 'cruise-and-collect' algorithm in its system).

The prototype will take a route such that it will cover all the space along its way and collect the ping-pong ball. It is assumed that there is a wall at one side of the navigational path to ease the collecting work and the route is free from any obstacles as in **Fig. 1**. All those mentioned above are the 'cruise' information used in the algorithm of the system in this research.

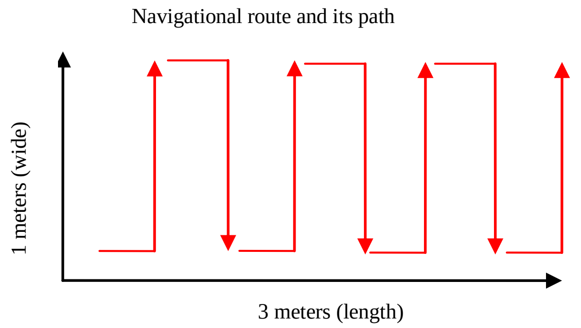

Fig. 1. Navigational path of the autonomous robot

As the application of the autonomous robot in this research is to navigate through its designated route while collecting the ping-pong ball along its way, several sensors needs to be used to realize this system. **Fig. 2** illustrates the block diagram of this autonomous robot prototype.

The Controller Module is the most important part of the autonomous robot as this module consists of the processor unit in the system. The Sensor 1 Module will cater the 'cruise' operation or its navigational route, while the Sensor 2 Module will cater the 'collector' operation. Sensor 3 is of no use at the moment because it is intended to be developed for further enhancement of the system. The 3 sensor modules are the input of the system, to be processed by the Controller Module, in order to produce the respective output to the Cruise Module and Collect Module.

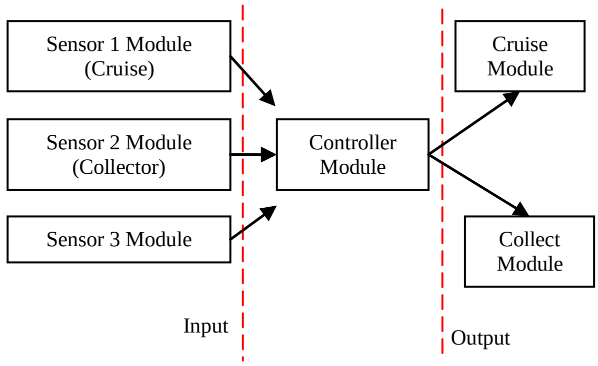

Fig. 2. Block diagram of the autonomous robot prototype

4 ALGORITHM DESIGN

The functional operation of the system is as summarized in the **Fig. 3**. **Fig. 4** detailed the flowchart of the Cruise Module, while **Fig. 5** shows the flowchart of the Collect Module. The Cruise Module inputs consist of 1 switch for prototype On/Off indicator, 1 switch for Start/Stop module and 3 IR sensors to guide the motor in its movement. The IR sensor act as the prototype eyes in order to avoid collision with obstacles in its surrounding. The Collect Module will make use of the same IR sensors in Cruise Module, such that it moves its motor to collect the ball in every turn the prototype made.

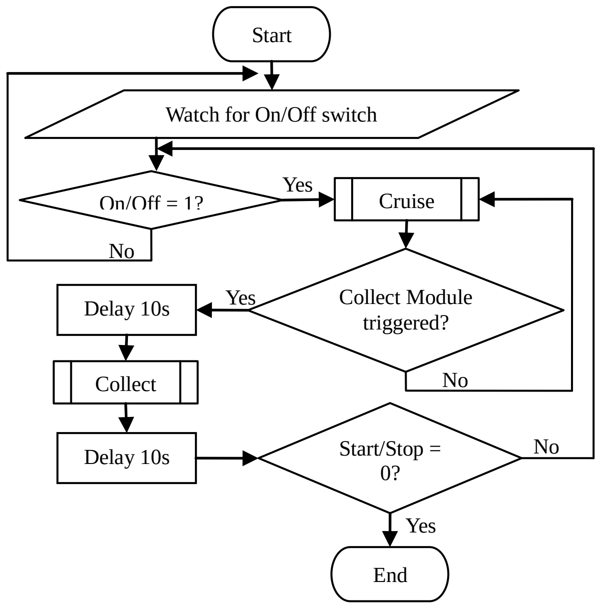

Fig. 3. Flowchart of the system operation

The designed algorithms as shown in the flowcharts are then realized via the *Flowcode* software, software that uses flowcharts as its design entry [5]. The *Flowcode* software has the capability to compile the flowcharts into the user's required form; either into Hex code, C-language code or compiled to the target board in the program. In this project, the designed flowcharts are compiled to its target board, EB031.

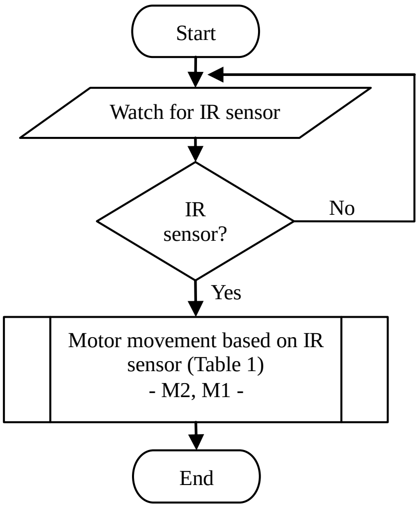

Fig. 4. Flowchart of Cruise Module

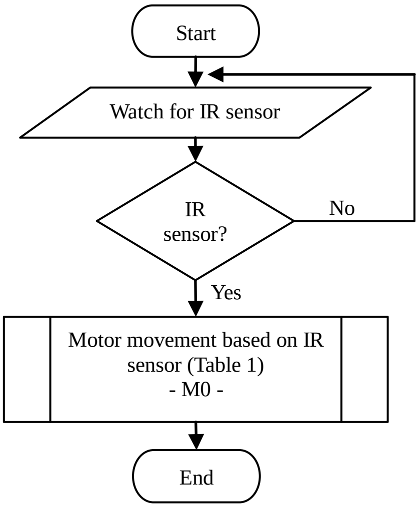

Fig. 5. Flowchart of Collect Module

Table 1. IR sensor and Motor movement

IR Sensor			Motor		
IR2	**IR1**	**IR0**	**M2 (Left)**	**M1 (Right)**	**M0**
0	0	0	Forward	Forward	OFF
0	0	1	Forward	Forward	OFF
0	1	0	Reverse	OFF	OFF
0	1	1	OFF	OFF	Forward
* after delay *			OFF	Forward	Reverse
1	0	0	Forward	Forward	OFF
1	0	1	Forward	Forward	OFF
1	1	0	OFF	OFF	Forward
* after delay *			Forward	OFF	Reverse
1	1	1	OFF	OFF	OFF

5 PROTOTYPE TESTING

The autonomous robot in this research is developed via the E-blocks boards, which can be obtained from *Matrix Multimedia Ltd.*, a company based in United Kingdom [5]. Due its low cost and free software download (trial version); these resources are chosen to be applied in this research.

The Controller Module will make use of EB031, an E-block ARM programmer board that is used to load the developed algorithm into the AT91SAM7 microcontroller chip. The microcontroller chip is housed on the removable EB034 board (another E-block), also called as ARM Daughter board. The removable ARM Daughter board allows users to detach the board and have the freedom to develop their own ARM application circuit. Both EB031 and EB034 board are as shown in **Fig. 6**.

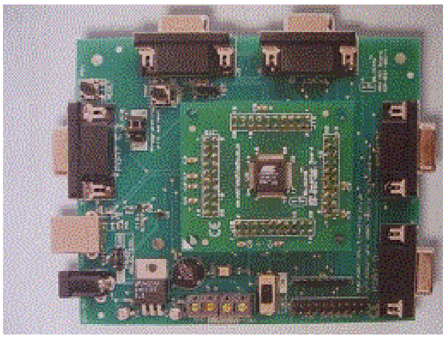

Fig. 6. EB031 and EB034 E-blocks

The supply voltage for EB031 board is an external regulated 6-9V DC supply or powered via the USB port (attached to computer) [6]. The programmer board is connected to any of its required input and output peripherals via its 5 serial ports (Port A, B, C, D and E). In this research, a motor driver board EB022, as shown in **Fig.7** is needed to drive the DC motor for wheel movement [7].

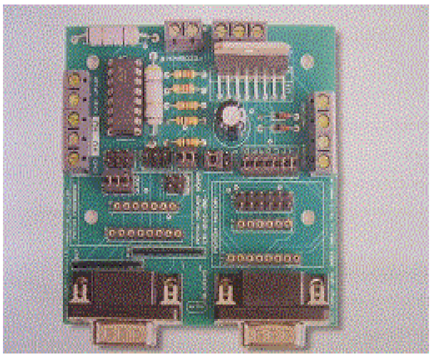

Fig. 7. EB022 E-block

The EB022 board allows users to drive two motors simultaneously and also allow independent PWM control for each motor. This board needs an external supply of minimum 5V and also for the motor power supply in the range of 3.3-24V external DC source.

For the purpose of this paper, the IR sensor status will be simulated via the LED board EB004 (**Fig. 8**) as the intended circuit is still in its development phase at the time of this paper writing. With that, the Collect Module will also be simulated via the LED pattern display.

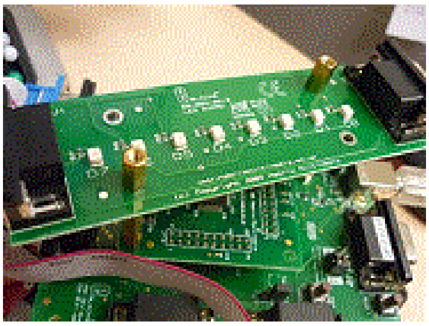

Fig. 8. EB004 E-block

The developed prototype is as shown in **Fig. 9.** With the EB031 and EB034 as the Controller Module, the EB004 will imitate the Collect Module while the EB022 will perform the Cruise Module. In this paper, the IR sensor condition is predetermined in the developed program as the IR sensor circuit is not fully developed at the time of this paper writing.

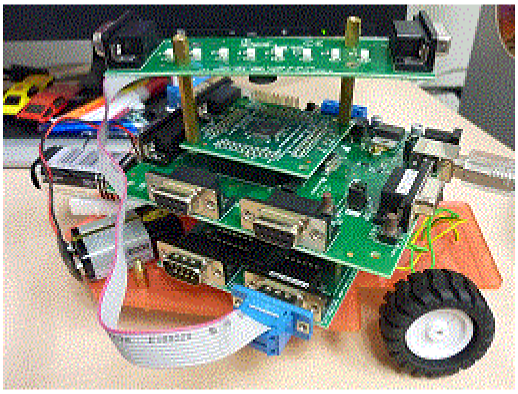

Fig. 9. Autonomous robot prototype

6 OBSERVATION AND CONCLUSION

From the developed prototype, it is observed that with a limited time constraint, researchers or even hobbyist may design their own application and develop the prototype with the use of the E-blocks and *Flowcode* software. The only constraint with the use of E-blocks is that each block must be connected via the serial port or E-block IDC cable if extension to the peripheral connection is needed. However, the removable ARM Daughter board allows customized circuit to be developed, besides restricting the development with only 5 serial ports given on the E-block ARM programmer board.

The 'cruise and collect' algorithm in this research is realized via the flowcharts without the hassles of writing the programming code. In this case, the steps involved in the development phase had been reduced, such that the system designer will only concern on its design requirement, but not the syntax or coding error. The use of flowcharts as the design entry form for the *Flowcode* software help ease the researchers work in testing its prototype. As such, the functional application can be fully tested and verified.

It is hoped that the development tools and design approach exposed in this paper will give the researchers or prototype developers other options in developing their system.

7 ACKNOWLEDGMENT

The authors would like to express the greatest appreciation to the Universiti Teknikal Malaysia Melaka (UTeM) for the continuos support, assistance and help given in completing this research.

REFERENCES

[1] Wikipedia - Autonomous robot (Accessed 2011). Citing Internet sources URL http://en. wikipedia.org/wiki/Autonomous_robot

[2] Daniel E. K. (2004), 'An Approach to Autonomous Robot Assembly', Robotica –Vol. 12, Feb 2004, pp137-155.

[3] Jonathan E., Pedro P., Ben S., David M. L. (2006), 'Design and Evaluation of a Reactive and Deliberative Collision Avoidance and Escape Architecture for Autonomous Robots', Journal of Autonomous Robots, 22 July 2006.

[4] Amalia R., Panos T. (2007), 'Real-time Hierarchical POMDPs for Autonomous Robot Navigation', Robotics and Autonomous Systems, Volume 55, Issue 7, 31 July 2007, pp. 561-571.

[5] Matrix Multimedia Ltd. (Accessed 2011). Citing Internet sources URL http://www.matrixmultimedia.com

[6] Matrix Multimedia Limited (2008), ARM progra mmer and daughter board (EB185-00-1) Technical Data sheet, Document Code: EB185-30-1.

[7] Matrix Multimedia Limited (2006), Motor driver board (EB022-00-1) Datasheet, Document Code: EB022 -30-1.