Principle and Realization of Micromouse Fast Dashing Turning Based on STM32

Haoming Zhang[1,*], Hong Li[2] and Yinghai Wang[3]

[1] *School of computer science & technology, Soochow University, Suzhou, 215006, China*

[2] *Department of Electrical and Information Engineering, Tongling University, Tongling, 214000, China*

[3] *Department of Electrical and Information Engineering, SIPIVT, 215321, Suzhou, China*

Abstract: Micromouse is an intelligent maze robot. Affected by the ability of MCU, micromouse based on MCS-51 can not guarantee the performance of quick dashing of the micromouse. High speed micromouse servo controller based on STM32 was designed. In order to meet the fast dashing of micromouse in a complicated maze environment, different speed-time curves were used. To realize the fast turning of micromouse in dashing, the turning trajectory was divided into five paths, different control parameters and the navigation sensors were used to track the turning. Experimental results show that five-path tracking method can not only realize micromouse dashing turning in high speed, which helps improve the dynamic performance and stability of micromouse, but also reduce its dashing time.

1. INTRODUCTION

Micromouse is an intelligent walking robot, with embedded microcontrollers and sensors. As a maze robot, micromouse technology covers many subjects. Micromouse has been used to solve the maze for nearly 30 years. Its principle can be transformed into many kinds of practical industrial robots. In recent years, the technology has been introduced in China, and it has gradually become a new competition. Micromouse can automatically explore different mazes and memorize its path, and use the corresponding algorithm quickly to reach the set destination. In order to promote the development of this technology, now every year, micromouse international competitions are held in Japan, USA, Singapore, UK, and the state of China, such as Hong Kong, etc. [1-6].

The basic function of a micromouse is to travel from the start square to the destination square. This is called a run. The time it takes is called the run time. Traveling from the destination square back to the start square is not considered a run. The total time from the first activation of the micromouse until the start of each run is also measured. This is called the maze time. If a mouse requires manual assistance at any time during the contest, it is considered touched. By using these three parameters the scoring of the contest is designed to reward speed, efficiency of maze solving, and self-reliance of the micromouse [7-11].

The technology of the micromouse in China is relatively backward. The structure of designed micromouse is shown in Fig. (**1**).

Many problems are found after long dashing experiments: stepper motors are used as the execution unit of a micromouse, which often generate excessive heat by their high internal resistance. PWM drive signals of motors are often missed without compensation, which leads to positional errors, causing the micromouse to be unable to find the dashing destination occasionally. Relatively large through hole technology components are used in the system, which makes the micromouse large and heavy, and unable to meet the requirements of high speed dashing. Due to the relatively low-level algorithms that are used to solve the maze, the dashing path becomes very long. In general, it takes a long time to dash from the start to the end. Due to unstable interference of the surrounding environment, especially light, which often makes the controller abnormal, and causes the micromouse to be out of control.

Therefore, the existing micromouse controller based on MCS-51 needs to be redesigned.

2. NEW TYPE MICROMOUSE BASED ON STM32

STM32F103 is manufactured by STM. It is a 32-bits flash microcontroller, and it has an ARM Cortex-M3 in its core. Its cost is low enough to challenge traditional 8-bit (such as C8051F series) microcontrollers and 16-bit microcontrollers (such as ADSP series). The STM32F microcontrollers combine high performance with first-class peripherals low-power, and low-voltage operation. The STM32 family is specifically designed for embedded applications, especially suitable for micromouse uses [12-14].

In order to overcome the problem of traditional micromouse based on MCS-51, domestic micromouse operating mode is abandoned. Referring to modern and advance design

*Address correspondence to this author at the School of computer science & technology, Soochow University, Suzhou, 215006, China; shizi road, Suzhou, 215006, China; Tel: 13013801792, E-mail: zhm20060616@163.com
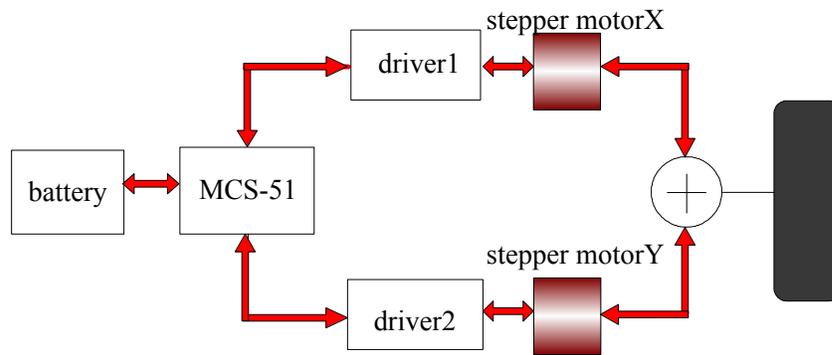
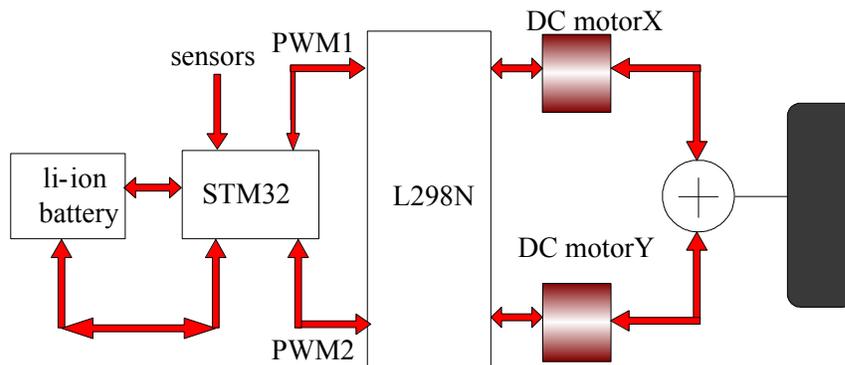**Fig. (1).** Principle of domestic micromouse.

Fig. (2). Principle of micromouse based on STM32.

Fig. (3). Two-dimensional structure of micromouse.

ideas, a new type of micromouse based on STM32 was developed, as shown in Fig. (**2**).

To meet the need of micromouse high speed dashing, DC motors are used in the system instead of the traditional stepper motors. The new designed micromouse has STM32F103 as the core processor, and has six infrared sensor, S1~S6, to judge the surrounding maze wall information. The infrared sensors are arranged as shown in Fig. (**3**).

As shown in Fig. (**3**), sensors S1 and S6 judge the front wall information of the maze; S2 and S3 judge the left wall information of the maze; S4 and S5 judge the right wall in-

formation of the maze. Then the synchronous servo control signals of the two DC motors are generated in real time, depending on the maze wall information, enabling the micromouse position adjustment in fast dashing.

3. RAPID DASHING TURNING PRINCIPLE

3.1. Dashing mode judgment

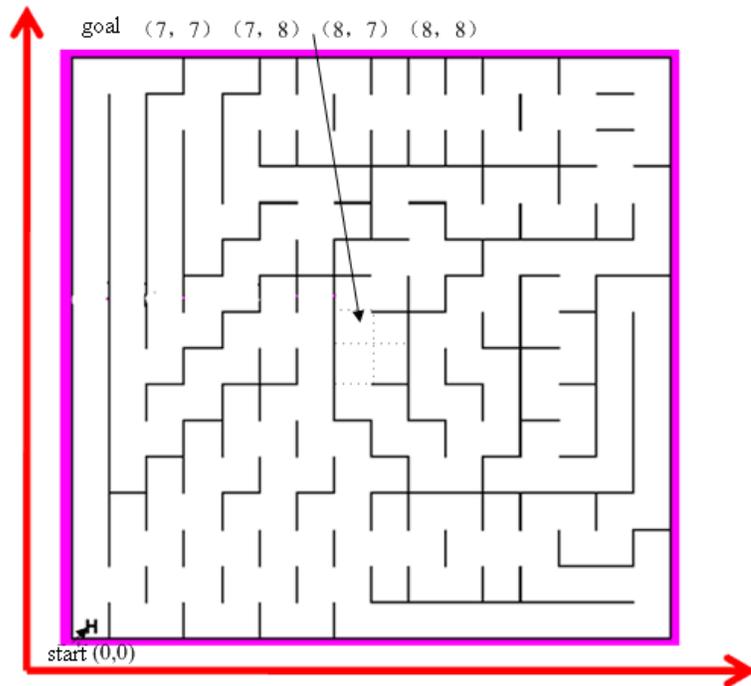When micromouse begins to dash, normally, it is placed on the starting cell, generally assigned the coordinate (0, 0)
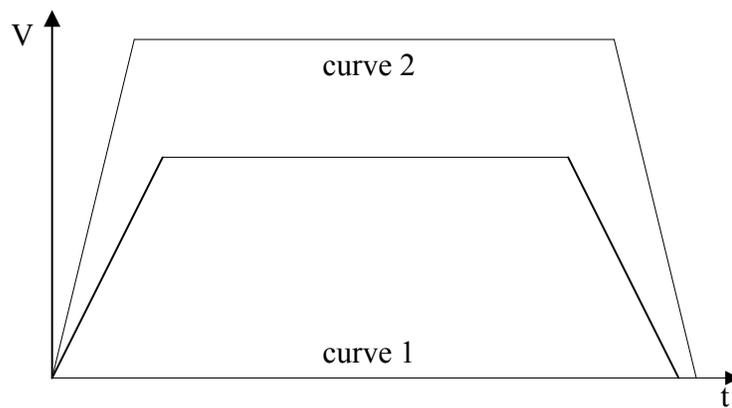
Fig. (4). Map of micromouse maze.

Fig. (5). Micromouse motion curve.

in order to facilitate memorizing the path location, as shown in Fig. (**4**).

When the power is turned on, the SOC of the battery will be judged firstly, and if the battery has low power, all the commands will be ignored. The controller issues the warning to replace the battery. If the battery is in good working condition, the system initialization is finished and the controller waits for the key information. In order to prevent wrong dashing, after receiving the dashing command, the front sensors S1 and S6 judge the front maze again, and determine that there are no retaining walls in the range of the dashing motion command. If present, interruption request is generated, and the PWM driving signals of motor X and Y are disabled and the micromouse can not dash. After receiving

the key information, the controller triggers different dashing modes: if the START key is pressed, micromouse will dash in the first high speed; if RESET and START keys are pressed, it demonstrates that the optimal maze has been confirmed. According to the number of times the START key is pressed, the dashing speed is determined, then the micromouse dashes from the starting position to the destination with a faster speed.

In order to meet the rapid dashing of micromouse in complicated mazes, this paper abandons the traditional single acceleration dashing mode. According to the distances of dashing, different speed and acceleration curves are used, as shown in Fig. (**6**). The area under the curves represents the distances of micromouse movement.
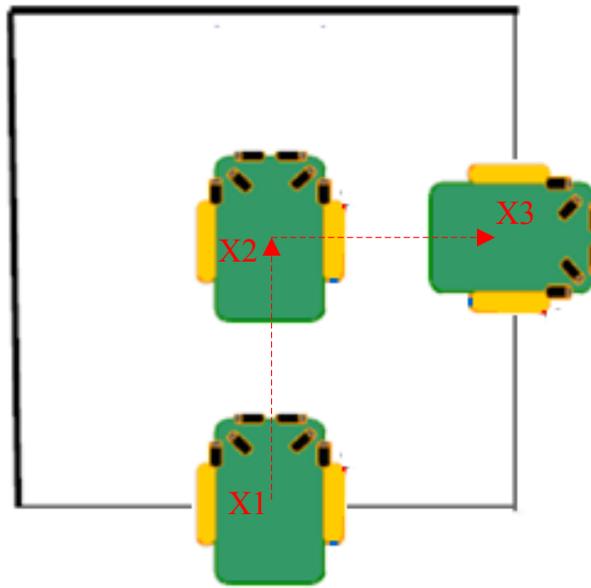
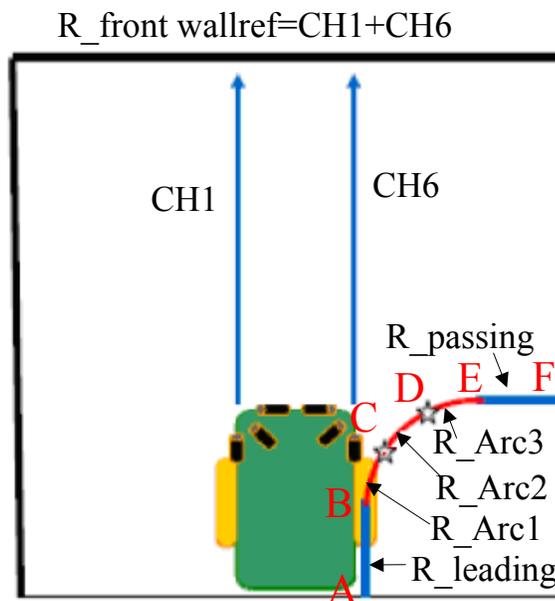**Fig. (6).** Principle of the traditional turning.

Fig. (7). Principle of five-path motion trajectory.

3.2. Principle of Five-path Turning

In the process of the micromouse dashing forward along the Y axis, if there is a wall in the front cell and a wall in the left cell at the same time, the only way for the micromouse can go is in the right hand direction, relative to the micromouse position.

The traditional method of turning right is shown in Fig. (**6**): If the micromouse needs to turn right, it begins to do deceleration from position X1 to position X2, and accurately parks in position X2, then the controller makes only the Y wheel move forward and the X wheel move backward, causing the micromouse to rotate 90 degrees at position X2, and then accelerates to position X3. Examining the principle of this movement, it can be seen that this method of turning is inefficient and takes a long time.

In order to reduce the dashing time and improve the stability of the micromouse, the actual turning action is considered. This paper proposes a method fast turning as shown in Fig. (**7**). The whole turning trajectory is divided into five parts as follows: the straight navigation before turning R_leading, slow turning R_ARC1, fast turning R_ARC2, slow adjustment turning R_ARC3, and straight navigation after turning R_passing.

Before turning right, with the help of sensor S1 and S6, the micromouse will first adjust its position along the straight path, and then does position error compensation until reaching A.

Once the micromouse reaches the predetermined position A, according to the different requirements of dashing command, STM32 first converts R_leading into different velocity and acceleration parameter instructions, and then combines with the feedback currents and photoelectric encoder values of motor X and motor Y. The servo control software generates PWM waves of the X and Y motors, and then drives the X and Y wheels forward with the same acceleration and velocity until reaching B.

When the micromouse reaches the predetermined position B, sensors S1 and S6 begin to work again, with reference value R_frontwallref giving error compensation. After the error compensation, the controller converts R90_Arc1 into different velocity and acceleration parameter instructions, and then combines with the feedback currents and photoelectric encoder values of motor X and motor Y. The servo control software generates PWM waves of the X and Y motors. The speed of motor X and motor Y is adjusted to R90_Vel-X1 and R90_VelY1, respectively, to drive the micromouse to enable the curve turning of R_arc1. In the whole turning of R_arc1, the ratio of two motor speeds C1= R90_VelX1/R90_VelY1 is constant.

When micromouse reaches the predetermined position C, the controller converts R90_Arc2 into different velocity and acceleration parameter instructions, and then combines with the feedback currents and photoelectric encoder values of motor X and motor Y. The servo control software generates the PWM waves of the X and Y motors, speed of motor X and motor Y will be adjusted to R90_VelX2 and R-90_VelY2, respectively, to drive the micromouse to enable the curve turning of R_arc2. In the whole turning of R_arc1, the ratio of two motor speeds C2= R90_VelX2/R90_VelY2 is constant.

When micromouse reaches the predetermined position D, the controller converts R90_Arc3 into different velocity and acceleration parameter instructions, and then combines with the feedback currents and photoelectric encoder values of motor X and motor Y. The servo control software generates the PWM waves of the X and Y motors, speed of motor X and motor Y will be adjusted to R90_VelX3 and R90_Ve-lY3, respectively, to drive the micromouse to enable the curve turning of R_arc3. In the whole turning of R_arc3, the ratio of two motor speeds C3= R90_VelX3/R90_VelY3 is constant.

When micromouse reaches the predetermined position E, controller converts R_passing into different velocity and acceleration parameter instructions, and then combines with the feedback currents and photoelectric encoder values of motor X and motor Y. The servo control software generates the PWM waves of the X and Y motors, speed of motor X and motor Y will be adjusted to the same value, respectively, to drive the X and Y wheels forward with the same acceleration and velocity until reaching predetermined position, F.

Through this method, turning right, relative to the position of the micromouse, is done. The micromouse updates its coordinates to (X+1, Y), and then compares if they coincide with any of the following coordinates (7, 7), (7, 8), (8, 7), or (8, 8). If so, then the micromouse has reached it final destination.

Principle of turning left is same as the right turn. The whole left turning trajectory is also divided into five parts. Due to the possible different performances of X and Y motors, parameters of turning left may not be the same as turning right.

4. SOFTWARE DESIGN

The micromouse control system designed in this paper is divided into two parts: the host system and motion control system. The host system realizes the function of maze reading, optimization of the shortest path, coordinate positioning, and online output; and the motion control system realizes the function of micromouse system servo control, data storage, I/O control, etc.

In the process of micromouse running, with the help of its six sensors, the micromouse searches the unknown maze, solves the maze information, memorizes the search paths, and with the intelligent algorithm calculates the optimal path, and then dashes in high speed to realize the shortest possible running time. The principle diagram of dashing is shown in Fig. (**8**).

5. EXPERIMENT

The designed micromouse based on STM32 is shown in Fig. (**9**).

PWM waveform of micromouse when it enters a maze cell is shown in Fig. (**10**). It can be seen from Fig. (**10**) that PWM and its duty ratio of two channels are the same. With the help of sensors, the micromouse moves in the straight path, and the speed of motor X equals the speed of motor Y.

PWM waveform of micromouse when it leaves a maze cell is shown in Fig. (**11**). From the figure it can be seen that PWM and its duty ratio of two channels are the same. With the help of sensors, the micromouse moves in a straight path, the speed of motor X equals the speed of motor Y. As a result, ratio of motor X speed to motor Y speed is constant.

PWM waveform of micromouse through arc 2 is shown in Fig. (**12**). As a result of entering and leaving a cell with different speeds, so the duty ratios of micromouse around the arc is different.

Picture of micromouse before a turning is shown in Fig. (**13**). As can be seen from the graph, with the help of the sensors S1 and S6, micromouse can be navigated accurately to the setting position before a turning, which can ensure the accuracy of the oncoming turn.

Picture of micromouse after a turning is shown in Fig. (**14**). As can be seen from the graph, with the help of sensors S4 and S5, micromouse can be navigated accurately to the
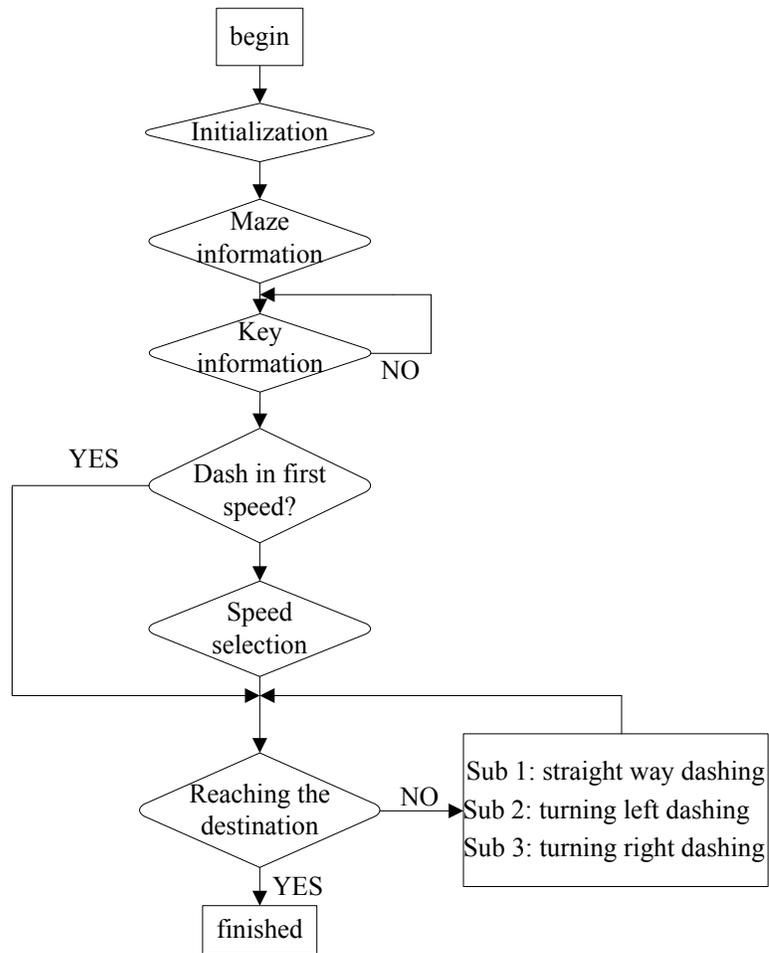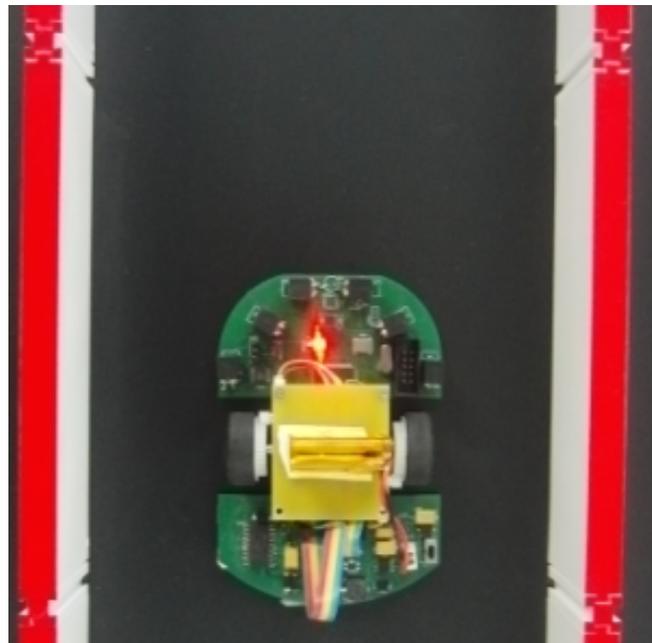
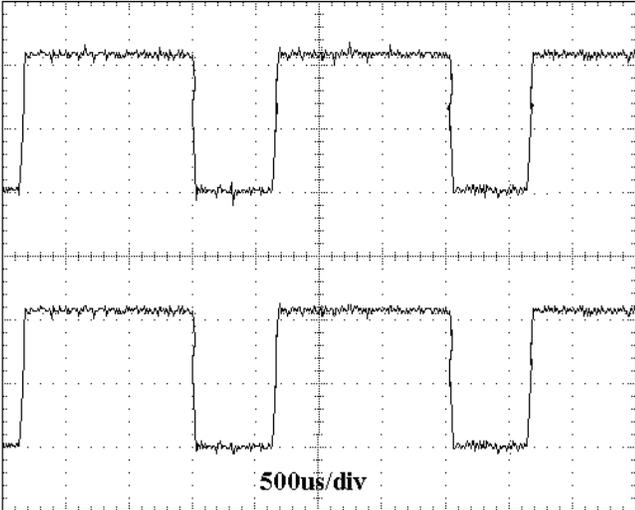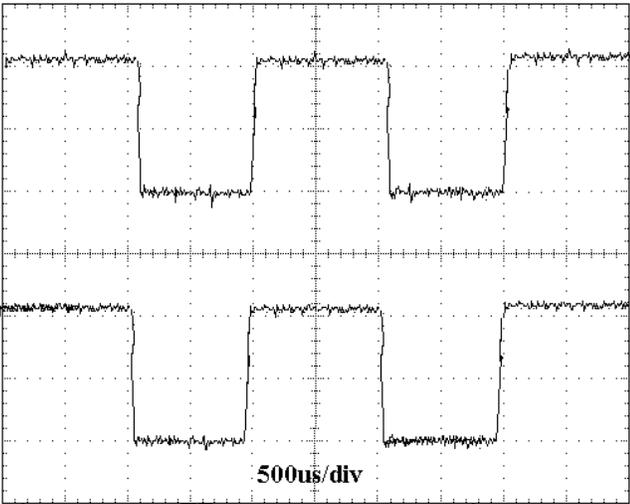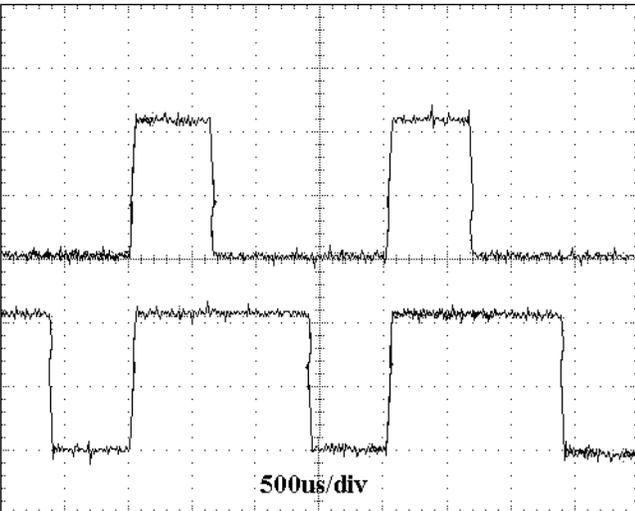**Fig. (8).** Principle diagram of micromouse dashing.

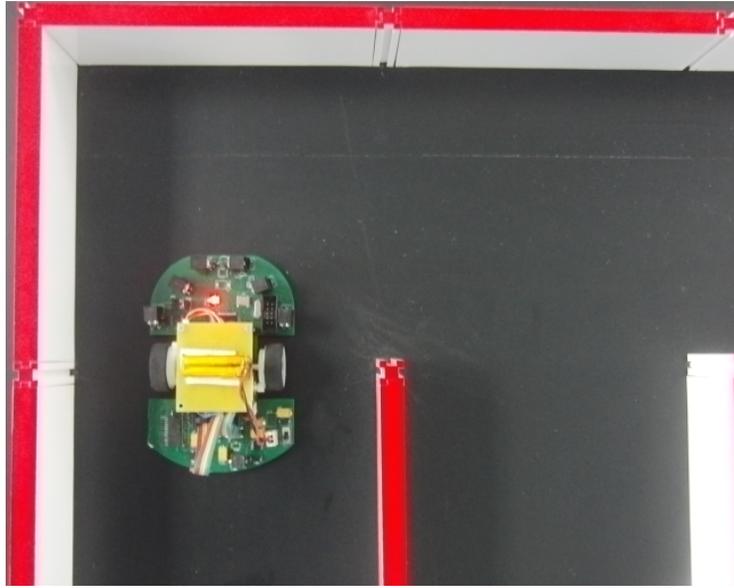
Fig. (9). Prototype of designed micromouse.

Fig. (10). PWM waveform of micromouse enters a maze.
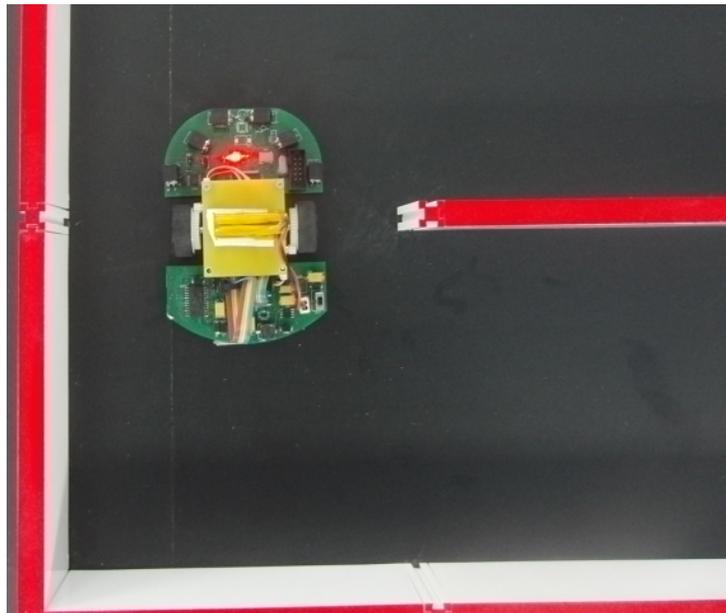

Fig. (11). PWM waveform of micromouse leaves a maze.

Fig. (12). PWM waveforms of micromouse through ARC2.

Fig. (13). Picture of micromouse before a turning.

Fig. (14). Picture of micromouse end a turning.

setting position after a turning with compensation, which greatly ensures the accuracy of the turning at high speed.

CONCLUSION

1. In the process of micromouse dashing, the effect of battery to the system is fully considered. Controller based on STM32 monitors its SOC, which can avoid high current discharging, and reduce the damage to battery in high speed dashing.

2. Surface mounted technology components are used in the system. They not only reduce the volume and weight of the micromouse but also lower its centre of gravity, which is conducive to high speed dashing.

3. The controller uses STM32 to process maze reading and algorithm of dashing. STM32F103 outputs dashing PWM signals, which greatly improves the efficiency and stability of micromouse dashing.

4. In order to reduce the dashing time, micromouse turning left or right trajectory, relative to the position of the micromouse, is divided into five parts. During the whole turning, turning error in every path has been compensated depending on the navigation sensors, which increases the stability of high speed turning.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

REFERENCES

[1]   J. H. Su, C. S. Lee, H. H Huang, and J. Y Huang, "A micromouse kit for teaching autonomous mobile robots", *Int. J. Elect. Eng. Edu.*, vol. 48 , no. 2, pp. 188-201, 2011.

[2]   M. Swati, and B. Pankaj, "Maze solving algorithms for micro mouse", In: *Proceedings of the 2008 IEEE International Conference on Signal Image Technology and Internet Based Systems*, vol. 11, 2008.

[3]   S. G. Kibler, A. E. Hauer, D. S. Giessel, C. S. Malveaux, D. Raskovic, "IEEE Micromouse for mechatronics research and education", In: *Mechatron. (1CM), IEEE Int. Conf.*, pp. 887-892, 2011.

[4]   P. J. Cheng, C. H. Cheng, and C. C. Tung, "Design of DC motor's torque control using DSP", *J. Inform. Optimizat. Sci.,* vol. 30, no. 6, pp. 1197-1207, 2009.

[5]   X. Jiang, H. Ren, and Z. Ma, "Research on how arm-based micromouse gets out of maze", *Mod. Elect. Tech.*, vol. 34, no. 8, pp. 14-16, 2011.

[6]   A. D. Tondr, and H. Drew, *The Inception of chedda: a Detailed Design and Analysis of Micromouse*, University of Nevada: Las Vegas, 2004.

[7]   Y. Zhou, and J. Fu, "speed control of micro-mouse based on fuzzy control", *Microcomput. Inform.*, vol. 26, no. 13, pp. 25-29, 2010

[8]   X. Fu, and H. Zhang, "Design of the MicroMouse hardware based on the ARM-M3", *J. Jilin Instit. Chem. Technol.*, vol. 29, no. 01, pp. 47-49, 2012.

[9]   R, Wen, Y. Xu, and Z. Wang, 'Study and application on maze path planning based on improved ant colony algorithm", *J. Jiangxi Univ. Sci. Technol.*, vol. 31, no. 02, pp 26-28, 2010.

[10] S. He, and K. Sun, "Design and optimization of micro-mouse solving the maze algorithm based on central method", *Comput. Syst. Appl.,* vol. 9, pp. 79-82, 2012.

[11] Y. Chen, and J. Yang, "Improvement and search algorithm of micromouse circuit", *Mod. Elect. Tech.,* vol. 34, no. 10, pp. 68-70, 2011.

[12] E. Bo, and Y. Yu, "Design and implementation of micromouse based on STM32", *Value Eng.,* vol. 28, no. 20, pp. 136-137, 2011.

[13] Z. Chou, A. Li, and W. Zhang, "Design of micromouse system based on ARM and FPGA", *Instrum. Anal. Monitor.,* vol. 55, no. 2, pp. 88-91, 2011.

[14] B. R. Chaudhuri, "On the determination of the emission wavelength of an infrared LED with common laboratory instruments", *Europ. J. Phy.,* vol. 32, no. 4, pp. 935-945, 2011.