Robust and Low-Cost Proximity Sensor for Line Detection Robot using Goertzel Algorithm

Rayi Yanu Tara
EE & IT Department
Gadjah Mada University
Yogyakarta, Indonesia
tara_s210@mail.te.ugm.ac.id

Teguh Barata Adji
EE & IT Department
Gadjah Mada University
Yogyakarta, Indonesia
adji@mti.ugm.ac.id

Abstract—**The usage of proximity sensor for marker detections such as color line detection is the easiest technique in robotic competition. Most of the contestants use simple photo-reflective circuit as proximity sensor with constant light source and voltage-based threshold to deduce the presence of marker line. By using this technique, noises such as various changes in indoor lighting exposure and camera flash may cause negative effect against system performance. To overcome this situation, a proximity sensor model is presented utilizing photo-reflective circuit with modulated light source. Goertzel algorithm is then used to calculate the power spectrum value of the received signal from photo-reflective circuit. The presence of marker line is deduced by simple thresholding technique based on the previous calibrated power spectrum value. Using this model, marker line detection method based on proximity sensor becomes more robust against noises and disturbances.**

Keywords- detection; proximity sensor; robust; goertzel algorithm;

I. INTRODUCTION

University-level robotic competition is one of many astounding ways to spur college students to do research on robotic technology. The contributions of robot competition for research development are also shown by the growing of several scientific communities [1]. Nowadays, robotic competitions are already held periodically at many competition levels (e.g national and international levels). In the competitions such as ABU Robocon, white colored grid line is used for robot navigation in the competition arena [2]. The other robot competition i.e Trinity Fire-Fighting Robot Contest uses white line marker in every room entrance and utilizes white circle object to mark the surrounding candle location [3].

Eight pieces of infra red photo-reflective sensors were used in ABU Robocon 2007 competition as line detection sensors [4]. The data received by the sensors is fed into robot control system as references for the robot movement. Another model of simple photo-reflective sensors was utilized to detect navigation line in the competition arena. Each sensor consisted of one LED as the light source, one LDR (Light Dependant Resistor) for receiving the reflected light, and one voltage comparator circuit using an operational amplifier for determining the navigation line. Whilst, our proposed system use modulation technique for detection and this sort of technique will not be affected by various lighting exposure, thus more robust than the system reported in [4]. Another solution using an embedded camera to detect the navigation line was used in [6]. The solution utilized embedded vision system for line detection and therefore was expensive and required much more computing resources.

In this paper, a proximity sensor consisting of photo-reflective circuit and a low-cost microprocessor is proposed. For the photo-reflective circuit, a pair of photodiode and LED is used. A modulated signal with fixed frequency is used to drive the LED to generate a light source. The photodiode received the reflected light from the LED and transformed the light into voltage signal. The signal is then sampled and processed by the low-cost microprocessor using goertzel algorithm to calculate the signal power spectrum value. Afterward, the calculated spectrum value is compared with a calibrated value to define the presence of the marker line. Goertzel algorithm is used since it is more efficient than the Fast Fourier Transform (FFT) for computing an *N*-point DFT and suitable for constrained performance computing platforms such as microprocessor and DSP [7]. Goertzel algorithm is often used for tone decoding application, therefore is applicable for this research. Using this technique, it can be shown that the proposed signal processing technique could be implemented in a low-cost computing platform and could be employed to overcome external noises which occur during its use in line detection robot. The proposed technique is implemented as the proposed proximity sensor that will be explained in Section 2.

II. PROPOSED PROXIMITY SENSOR SYSTEM

The scheme diagram of the proposed system is illustrated in Fig. 1. The system consist of two main components i.e a photo-reflective circuit and a low-cost microprocessor.

An 8-bit AVR ATmega8 microcontroller is used as the low-cost microprocessor. The microcontroller is an 8-bit RISC architecture with hardware multiplier and complex built-in peripherals such as ADC and timers [8]. According to its characteristic and performance, the microcontroller is suitable for calculating goertzel algorithm and thresholding method. The microcontroller I/O and UART are used to send output signals, which represent detection results.

The photo-reflective circuit uses photodiode to receive reflected light from ground surface and converts it into a

978-1-4577-1460-3/11/$26.00

voltage signal. The signal is then sampled by the microcontroller using internal 8-bit ADC. The conversion result is then stored into a data buffer.

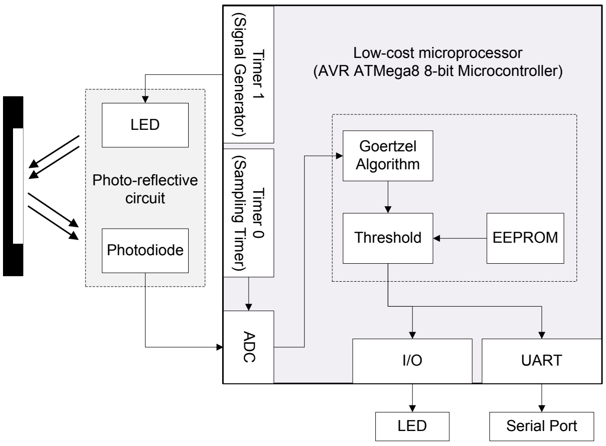

Figure 1. Scheme diagram of the proposed system.

Sampling process is executed periodically by utilizing Timer 0 interrupt. Timer 1 is used to generate fixed-frequency signal, which drives the LED state. Thus, the LED will be emitting a modulated light. Fig. 2 shows the process of signal sampling.

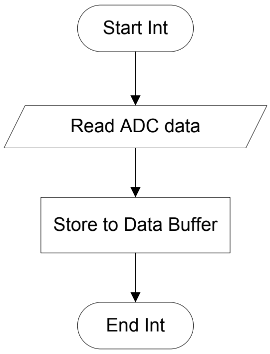

Figure 2. Signal sampling using timer interrupt.

An optimized version of goertzel algorithm is employed to reduce computation time. Optimized goertzel calculate the relative magnitude squared rather than the real and imaginary frequency components [9]. To do the goertzel calculation, some preliminary calculations are made. First, 3 kHz sampling rate is used. According to the nyquist rule, the proposed system will be able to sample a signal with frequency up to 1.5 kHz. The sampling rate is sufficient since the light source frequency will be set below 1 kHz. Second, the goertzel block size (symbolized as N) that determines the frequency resolution or bin width. The number of N doesn't have to be a power of two. The higher the N the higher the computation time is. Thus, by setting N to 50, each bin will have 60 Hz bin width.

Signal generator frequency is derived by (1):

$$f = \frac{sampling\_rate}{N} \times (bin\_number) \tag{1}$$

where bin_number is constant between 1 and N/2 and sampling_rate is input signal sampling rate. To achieve the frequency of close to 1 kHz, bin_number is set to 15. This will result in signal generator frequency of 960 Hz.

To apply the goertzel calculation then a constant (symbolized as coeff) is calculated using (2):

$$coeff = 2 \times \cos\left(\frac{2\Pi}{N} \times \left(0.5 + \frac{N \times f}{sampling\_rate}\right)\right) \tag{2}$$

In this calculation, -0.85156 is obtained as the value of the coeff.

Here is the optimized goertzel algorithm flowchart.

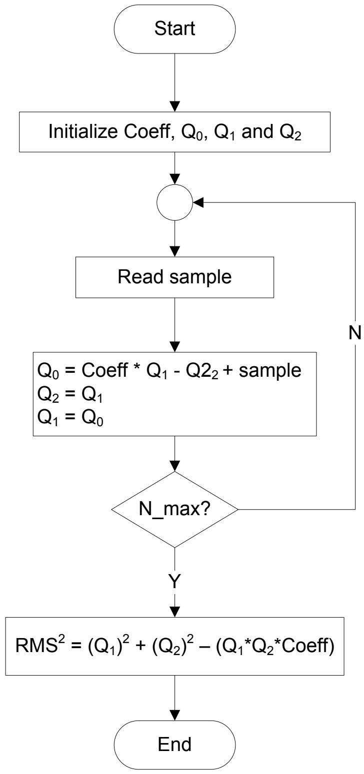

Figure 3. Optimized goertzel algorithm.

The optimized goertzel is running on per-sample processing [9]. Three temporary variables are used to run the processing mechanism; there are Q_0, Q_1 and Q_2. Variables Q_1 and Q_2 must be initialized to zero at the beginning of each sample block and (3) ~ (5) must be run for every sample.

$$Q_0 = coeff \times Q_1 - Q_2 + sample \tag{3}$$

$$Q_2 = Q_1 \tag{4}$$

$$Q_1 = Q_0 \tag{5}$$

After N iterations, (6) is used for the calculation of relative magnitude squared (rms) value, which represents power spectrum value.

$$rms^2 = Q_1 + Q_2 - (Q_1 \times Q_2 \times coeff) \qquad (6)$$

The main process implementation is shown in Fig. 4. During startup, the main process is halted until the data buffer gets filled with more than N. Before the goertzel calculation start, N data is being read from the data buffer. The flowchart of goertzel calculation process is drawn in Fig. 3. Thus, after the goertzel calculation has been completed, the result is compared (using thresholding technique) with a calibrated value, which is stored in microcontroller EEPROM. If the comparison result is positive, the microcontroller will send an output signal indicating a line has been detected, and vice versa.

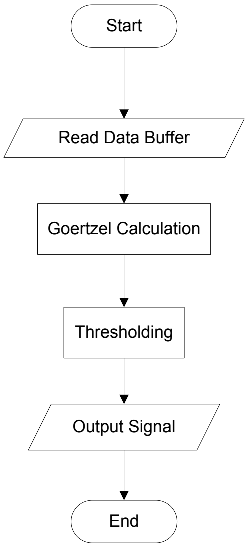

Figure 4. Main process flowchart.

III. Experiment and Result

Experiment is divided into three parts; frequency response test, computation time test and overall system performance test.

In frequency response test, a signal generator is used to generate a sinusoidal signal. Fig. 5 shows the frequency response test that is illustrated in a bar chart. The signal frequency sweeps from 10 Hz to 1500 Hz with 10 Hz increment. Then, the signal is fed into the microcontroller ADC input and the microcontroller sends the goertzel calculation result into PC via serial port. The system is able to measure the magnitude value of input signal having frequency range between 940 Hz and 980 Hz in which 960 Hz is the optimum measured frequency value. Based on this result, it can be concluded that the goertzel algorithm is well implemented in ATmega8 microcontroller.

In computation time test, AVR Studio 4 simulator software is used. This test is performed using 16 MHz crystal oscillators as microntroller clocking source. Times consumption of three procedures (i.e startup time, per-sample processing time, and magnitude calculation time) are measured. Computation test result is shown in Table 1. Startup time takes the longest computation time because it is a condition where the system waits for the data buffer to be filled with N samples and this only happens one time. Per-sample processing time is measured by time consumed execution from the first iteration until N^{th} iteration. Magnitude calculation time is measured by time consumed execution of relative magnitude squared value calculation. Total calculation time is a summation of per-sample processing time, and magnitude calculation time. It's about 1022.19 us, it means that this proposed system has 978.29 Hz refresh rate.

TABLE I. COMPUTATION TIME

Procedures	Time (us)
Startup	16582,38
Per-sample processing	986,31
Magnitude calculation	35,88
Total	1022,19

To measure overall system performance, two consecutive

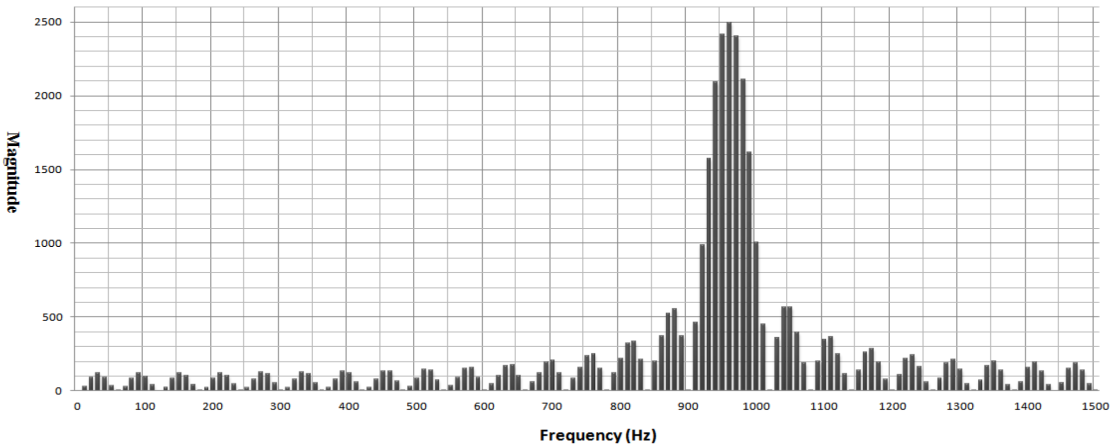

Figure 5. Frequency response test result.

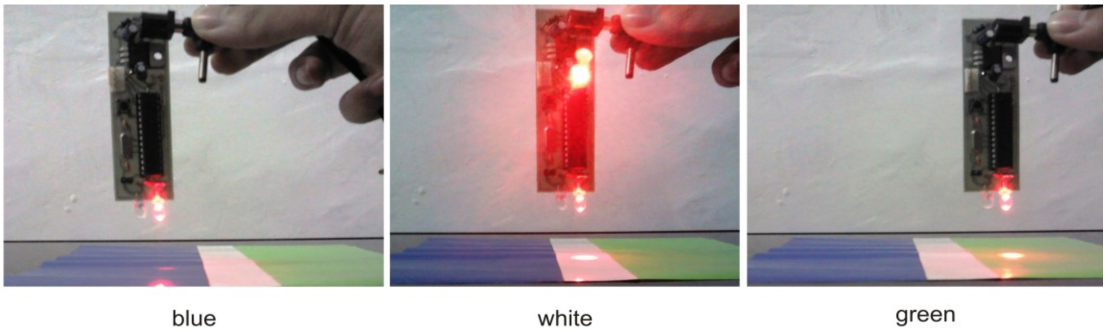

Figure 6. Overall system performance test. The LED turns on when the photo-reflective circuit placed over white marker. Otherwise, the LED turns off when the photo- reflective circuit placed over blue and green marker.

tests are conducted. The first test is the measurement of the system successfulness to deduce white line marker (on black colored background) in various lighting condition. This test is conducted under three lighting conditions; indoor fluorescent lamp, indoor bulb lamp, and outdoor lighting i.e direct sunlight. The photo-reflective circuit is not enclosed during this test, so the influence of various lighting exposure levels will obviously affect the system performance. For every test with different lighting conditions, this system has to be calibrated first. Each test is performed 10 times. The result of this test is shown in Table 2. Based on this result, it can be concluded that the proposed system performance is robust against various lighting exposure levels.

TABLE II. SYSTEM PERFORMANCE TEST

Lighting Condition	Successful Experiment
Indoor fluorescent lamp	10
Indoor bulb lamp	10
Direct sunlight	10

The second test is the assessment of the system capability to distinguish white line marker on other color backgrounds i.e green and blue. This test is conducted under indoor fluorescent lamp. This system has to be calibrated before the test is conducted. Fig.6 shows overall system performance test results.

IV. CONCLUSION AND FUTURE WORK

This work presents a proximity sensor system using goertzel algorithm, which is applied to marker line detection. The presented system uses photo-reflective sensor with modulated light source and uses a low-cost microprocessor to do the goertzel calculation. The results demonstrate that the goertzel algorithm is successfully implemented in a low-cost microcontroller and able to detect line marker. Experimental result shows that this system has refresh rate of 978.29 Hz when implemented in AVR ATmega8 microcontroller with 16 MHz oscillator. Moreover, the measured frequency ranges between 940 Hz and 980 Hz. The presented system can detect the line marker without any enclosure mounted on it. Thus, this system is more robust against various lighting exposure, which may affect the performance of another proximity sensor model.

The main advantage of the presented system is that it only needs initial calibration. It is used to measure the respective power spectrum value of the line marker and calculate the threshold value for comparison. Moreover, it has shown that a simple and low-cost system could adopt a signal processing technique to build a robust proximity sensor.

In the future, IR photo-reflective circuit and higher microcontroller performance (e.g 32-bit ARM Cortex) should be used to increase overall system performance and robustness, especially for the system refresh rate.

REFERENCES

[1] Fonseca, P. Lima, F. Ribeiro, V. Santos, "Mobile Robot Competitions: Fostering Advances in Research, Development and Education in Robotics", *CONTROLO' 2000 4th Portuguese Conference On Automatic Control*, 2000.

[2] ABU RoboCon (2011,May 20). Rulebook_2011_ final_20May11_English [pdf]. Available: http:// aburobocon2011.com/downloads/AmendedRules_Eng20May2011.pdf

[3] Trinity College Fire Fighting Robot Competition (2010,Dec 8). 2011 Trinity College Fire Fighting Home Robot contest Rules and Regulations [pdf]. http://www.trincoll.edu/events/robot/Rules/Rules%202011/Rules_2011-Final-3-A4.pdf

[4] Yunzhou Zhang, Chengdong Wu, Dingyu Xue, Hao Chu, "Design of Control System for Mobile Robot Based on DSP Processor," cccm, vol. 2, pp.528-532, *2008 ISECS International Colloquium on Computing, Communication, Control, and Management*, 2008.

[5] M. Z. Baharuddin, I. Z. Abidin, S. S. Kaja Mohideen, K. S. Yap, and J. T. C. Tan, "Analysis of Line Sensor Configuration for the Advanced Line Follower Robot", in Proc. of *Student Conference on Research and Development (SCOReD)*, Bangi, Selangor, Malaysia, 2005.

[6] S. Skaff, G.A. Kantor, D. Maiwand, and A.A. Rizzi. Inertial navigation and visual line following for a dynamical hexapod robot. In Proc. of *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vol 2, pp808–1813, October 2003.

[7] A.V. Oppenheim and R. W. Schafer, "*Discrete- Time Signal Processing*", Prentice-Hall, 1990

[8] Atmel Corporation, "ATmega8, 8-bit AVR with In-System Programmable Flash", ATmega8 datasheet, Mar. 2002 [Revised Oct. 2004].

[9] K. Banks. "The Goertzel Algorithm," *Embedded Systems Programming*, September 2002, pp. 32-42.