2022 IEEE International Conference on Consumer Electronics (ICCE)

Robot-on-Chip: Computing on a Single Chip for an Autonomous Robot

Young Woo Jeong
Department of Electronic Engineering
Seoul National University of Science and Technology
Seoul, Replublic of Korea
jeongyoungwoo@seoultech.ac.kr

Kwang Hyun Go
Department of Electronic Engineering
Seoul National University of Science and Technology
Seoul, Replublic of Korea
gokwanghyun@seoultech.ac.kr

Seung Eun Lee*
Department of Electronic Engineering
Seoul National University of Science and Technology
Seoul, Replublic of Korea
*seung.lee@seoultech.ac.kr

***Abstract*— The interest in autonomous robots is growing due to diverse usability. Autonomous robots are equipped with various sensors for stable operation. As the sensor data increases, the system for sensor signal processing and actuators controlling is complicated. In this paper, we propose the robot-on-chip (RoC) which processes all functions for an autonomous robot on a single chip mounted on a robot. In order to realize the RoC, we designed an autonomous robot with a lightweight algorithm and a hardware-friendly architecture. We demonstrated the feasibility of the RoC that the robot moves successfully without bumping into people in a building by recognizing the environment.**

Keywords*— *Robot-on-Chip, autonomous robot, sensor fusion

I. Introduction

Recently, studies on autonomous robots and self-driving vehicles which have an advantage that increases human convenience have been conducted [1]. The most important assignment in implementing the autonomous driving is generating steering signals accurately when the vehicles or robots encounter people or obstacles. Existing autonomous robots and vehicles are equipped with sensors such as several cameras or multichannel LiDAR sensors with large size of data for generating the steering signals accurately [2]. As the size of the sensor data to process increases, the complexity of the sensor signal processing algorithm grows. Accordingly, an architecture that adopts servers to run the sensor signal processing algorithm has emerged. Since the stability of the robot executing the algorithm with servers increases when network latency decreases, a high performance network such as 5G was adopted to the robot [3].

On the one hand, researches on applying edge computing, a paradigm that computes data with an edge device, to the autonomous driving field have been conducted [4]. Edge computing provides an advantage to achieve complex tasks in an efficient way [6]. Tesla, the leader in this field, proposed a full self-driving chip that includes neural network accelerators, CPUs, and GPUs. to process camera data in real-time [5]. This chip has a processing speed up to 2300 frame per second with low cost operation compared to their previous hardware. They developed the stability of autonomous driving with this chip which realized a system-on-chip.

In this paper, we propose a robot-on-chip (RoC) for an autonomous robot. The RoC is an architecture that a single chip mounted on a robot computes functions about robot control such as sensor signal processing and actuator control. In order to realize the RoC, we designed a robot with a lightweight sensor signal processing algorithm and the hardware-friendly architecture. To the best of our knowledge, it is the first work that proposes the architecture of RoC.

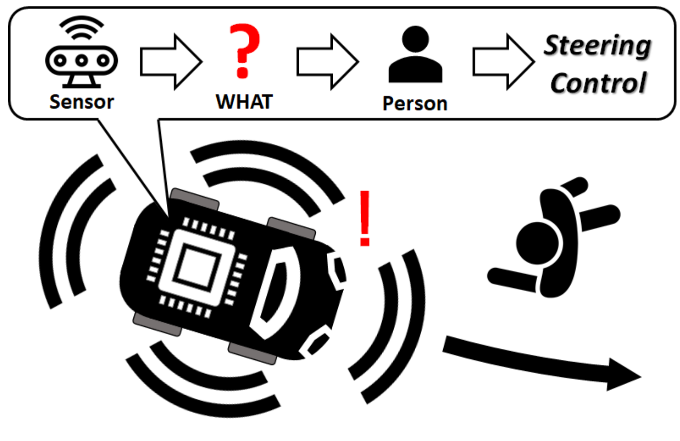

Fig. 1. Overview of the RoC

II. System Architecture

Fig. 2 shows the architecture of the proposed RoC. In order to implement the RoC, we designed an autonomous robot that consists of a single-channel LiDAR sensor which is a main sensor, three ultrasonic sensors which are assistance sensors, and two motor drivers for actuator control. The LiDAR sensor and ultrasonic sensors are employed for local path planning, and two motor drivers are utilized for four-wheel drive. We employ a processor called steering MCU, which is responsible for the role of the steering controller to control the LiDAR sensor and to execute a sensor signal processing algorithm for local path planning. Moreover, another processor called motor MCU to take the role of the motor controller is utilized for controlling the motor drivers and ultrasonic sensors. The conventional procedure that computes the sensor values simultaneously has to modify the entire algorithm whenever additional sensors are attached. There-

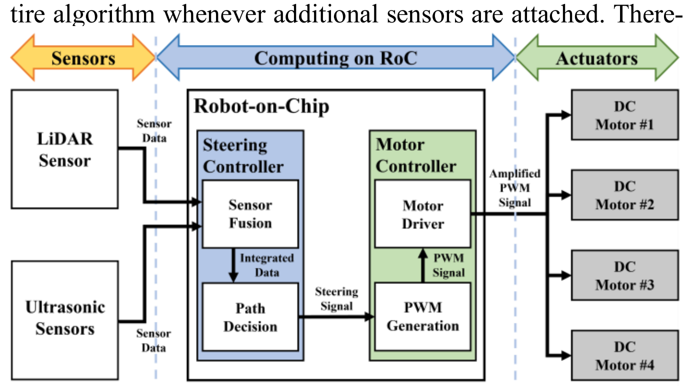

Fig. 2. The architecture of the RoC

2022 IEEE International Conference on Consumer Electronics (ICCE) | 978-1-6654-4154-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCE53296.2022.9730399

fore, by computing the values of the sensors separately, we not only increase the flexibility of sensor attachment, but also reduce the complexity of the algorithm. An operation of the robot is in the following order:

Step 1: The robot receives the values of the sensors and executes the sensor signal processing algorithm at the steering MCU for the LiDAR sensor, the front ultrasonic sensors, and the rear ultrasonic sensor respectively.

Step 2: Three steering signals are generated for each sensor when the sensor signal processing algorithm is executed. Accordingly, the robot generates the final steering signal by integrating these steering signals.

Step 3: The final steering signal is transmitted from the steering MCU to the motor MCU. The robot updates local path according to the final steering signal.

III. Sensor signal processing algorithm

The sensor signal processing algorithm is implemented by modeling the driving procedure of person. A sensor signal processing first operates for the LiDAR sensor. The LiDAR sensor outputs 630 number of distance values for 360 degrees. Since using all of these values for the signal processing increases the complexity, a pre-processing for the sensor values is essential. We converted 630 number of the distance values into 360 number of distance values for 0° to 360° by rounding the angles. When more than one distance values exist for one angle, the minimum value is set as the representative value. In this process, the sensor recognizes the distance value for the empty space is zero. Therefore, zero is set as the lowest priority when setting the representative value because the distance value is extracted incorrectly when the minimum value is set.

Fig. 3 (a) shows the process of reducing the number of distance values to 36 by grouping the distance values with 10 degrees. However, this procedure has the challenge of detecting the obstacles in the driving route because the values of the front of the robot are divided into left and right. Therefore, the values of two degrees in the front are grouped separately as shown in fig. 3 (b). As a result, 630 number of the distance values are reduced to 37 number of distance values. The robot employs only 23 number of distance values out of the 37 number of the distance values and the corresponding areas are shown in fig. 3 (c). The white areas are not employed because the robot is not affected although obstacles exist in these areas. The robot generates one of the nine steering signals according to the 23 number of the distance values of the LiDAR sensor. The nine steering signals include steering signals for basic driving as well as for straight correction.

After the steering signal for the LiDAR sensor is generated, the robot performs a sensor signal processing for the front ultrasonic sensors. The processing for the ultrasonic sensor is similar to the processing for the LiDAR sensor. Since the value of the ultrasonic sensor is unstable due to the vibration of the robot while driving, the ultrasonic sensor signal processing is designed in a way that deals with the previous two values and the current value simultaneously in order to improve the stability of the robot. This process allows the robot to operate stably although the sensor outputs an incorrect value twice out of three times. The robot generates one of seven steering signals for the distance values of the front ultrasonic sensors excluding steering signals for the straight correction because the front ultrasonic sensor detect the front of the robot. Finally, the robot performs a sensor signal processing for the rear ultrasonic sensor. Since the rear ultrasonic sensor detects obstacles behind the robot, the robot only generates one of the two steering signals. After generating the steering signal is completed for each sensor, the robot generates a final steering signal by integrating three steering signals.

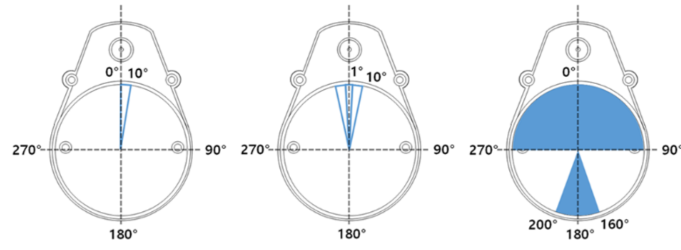

Fig. 3. Pre-processing procedure for the LiDAR sensor

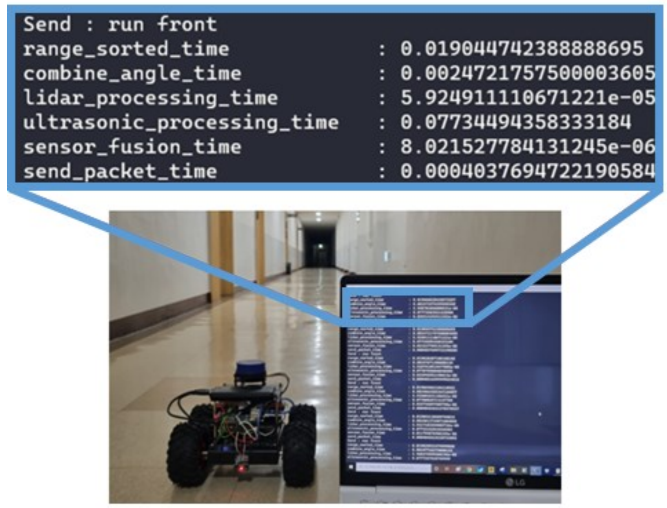

Fig. 4. Experimental environment

IV. Experiment

Fig. 4 shows the experimental environment. We operated the robot in a building with obstacles and verified that the robot moves stably without collisions through the local path planning. In addition, we measured the processing time of each step of the robot. The signal processing time for the LiDAR sensor and the ultrasonic sensors was 21.6 msec 77.3 msec respectively due to the slow speed of the serial interface of the sensors. The overall processing time was measured at 99.17 msec which is a suitable speed for real-time processing despite the slow speed of the interface.

V. Conclusions

In this paper, we proposed the RoC which computes all functions for an autonomous robot on a single chip mounted on a robot. We implemented the autonomous robot with the lightweight algorithm and the hardware-friendly architecture for realizing the RoC. Additionally, we demonstrated the feasibility of the RoC that the robot moves successfully without bumping into people in a building by recognizing the environment.

ACKNOWLEDGMENTS

This research was funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea) under Industrial Technology Innovation Program. No. 20016354, 'Developed proximity/healthcare convergence sensor SoC for TWS'.

REFERENCES

[1] Y. Kim, J. Jang and S. Yun, "End-to-end deep learning for autonomous navigation of mobile robot," IEEE International Conference on Consumer Electronics (ICCE), 2018, pp. 1-6.

[2] Y. Fan, B. Wu, C. Huang and Y. Bai, "Environment Detection of 3D LiDAR by Using Neural Networks," IEEE International Conference on Consumer Electronics (ICCE), 2019, pp. 1-2.

[3] K. Yaovaja, P. Bamrungthai and P. Ketsarapong, "Design of an Autonomous Tracked Mower Robot using Vision-Based Remote Control," IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE), 2019, pp. 324-327.

[4] Qingqing, Li, et al. "Edge computing for mobile robots: multi-robot feature-based lidar odometry with FPGAs." Twelfth International Conference on Mobile Computing and Ubiquitous Network (ICMU). IEEE Computer Society, 2019, pp. 1-2.

[5] E. Talpes et al., "Compute Solution for Tesla's Full Self-Driving Computer," in IEEE Micro, vol. 40, no. 2, pp. 25-35, 2020.

[6] Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu and W. Lv, "Edge Computing Security: State of the Art and Challenges," in Proceedings of the IEEE, vol. 107, no. 8, pp. 1608-1631, Aug. 2019.