

# Design and Prototyping of An Autonomous Un-Guided Vehicle (AuGV) for Hazardous Areas Usage: Moon Rover Space Exploration Vehicle Prototyping

Sumaila Musa, Alexander Ajibola, Emmanuel Okafor, Ikenna Ijeh, and Jamie Gibson

Department of Control and Instrumentation, University of Derby, United Kingdom

**Abstract**— This paper presents the design of an AuGV (Autonomous un-Guided Vehicle) prototype capable of autonomously seeking brightest light source and deploying a pair of foldable solar-panel to recharge it depleted battery upon detecting a spot of brightest light intensity. The AuGV is embedded with the ability to detect & avoid obstacles in its path whilst it is navigating to the detected light-spot; also it simultaneously calculates the shortest or best route to achieving this navigation task.

While most AGV (Autonomous Guided Vehicle) journal papers focuses on simulations and abstract ideas, this paper focuses on the development of an actual working prototype - with emphasizes on hardware subsystems, controllers, and various communication protocols used for linking the different subsystems together. Decentralized embedded system solution was used to achieve the objectives of this project.

A fully functioning prototype was built, tested and its results presented herein as well as video demonstration. In the test carried-out, spontaneous obstacles of varying sizes and light-sources of varying intensities were employed in the test environment. The developed AuGV prototype has no prior knowledge of its environment nor were knowledge of the environment embedded as program codes in the AuGV controller.

**Keywords**— AuGV, locomotion, light-sensing & deployment of solar-panel, MBED Microcontroller, Obstacle-Detection & Avoidance, prototyping

Copyright© 2017. Published by UNSYSdigital. All rights reserved.  
DOI: [10.21535/just.v5i2.966](https://doi.org/10.21535/just.v5i2.966)

## I. INTRODUCTION

**A**UTONOMOUS Vehicle (AV) belongs to a class of cooperative driverless vehicles which are coordinated by either a centralized or distributed embedded control systems. Application of this class of vehicle include but not limited to academic research, space-exploration, underwater exploration, logistic transport delivery, UAVs (unmanned Aerial Vehicles) & precision formation flying, industrial manufacturing, etc. AGVs, a form of AV, are used in investigating areas, such as radioactive environment, that are too dangerous for human.

Corresponding author: Sumaila Musa  
(e-mail: [sumailaesq@yahoo.com](mailto:sumailaesq@yahoo.com))

This paper was submitted on Aug 14, 2017; revised on Aug 25, 2017; and accepted on August 30, 2017.

Currently AGVs are being employed in the decontamination of Japanese Fukushima nuclear power station [1].

Previous investigators have designed AGVs based on some guidance techniques, which are: laser-guided AGV, wire-guided AGV and tape-guided AGV; although a combination of these different guidance techniques is also possible [2]. These guidance techniques confines AGVs to indoor application only, such as manufacturing, and hence depriving the vehicle from fulfilling the area of application of AVs mentioned in the preceding paragraph.

Academic and industrial research communities including Google and other driverless car researchers are developing AGVs that can successfully navigate unguided (both indoor and outdoor environments) to deliver payload to some locations. State-of-art [3][4] differentiate modern AGVs from classical AGVs as the former are free-ranging in their movement and are software driven as against the guidance techniques used by the latter classical AGVs. Hence, we span the AGV acronym to include un-Guided (AuGV).

In this project, the authors worked as a team to develop an intelligent AuGV prototype (based on “Himoto Rock Crawler RCF-8” chassis) to address two tasks. Primary task is detection of a location with brightest light intensity, navigating to that spot with the sole purpose of deploying its foldable solar-panels used for recharging its battery on-the-go.

Secondarily, the AuGV has no knowledge of its environment and it's customary to assume that the AuGV will be operating in environments marred with different kind of obstacles. The developed prototype must successfully detect and avoid any impeding obstacles that may hamper it from achieving the aforementioned primary task. Hence, the secondary design task is the ability of the prototype to provide the vehicle central-controller with real-time qualitative informed decision on which best alternate route to navigate and avoid any detected obstacle(s).

Being electrical in nature, power management is a critical issue of importance for these vehicles, although this is usually omitted from most AVs research publications and journals. The ability for AVs to autonomously recharge their batteries on-the-

go is of desirable research interest for these vehicles especially when they are employed in mission critical environment, like space exploration, logistic delivery, etc. State-of-art [5] describes three types of charging schemes for AV: 1) Opportunity charging – using the vehicle idle time to charge it battery. 2) Automatic charging – the AV run until it batteries are depleted to some predefined threshold value, after which the vehicle automatically schedule itself for charging. 3) A combination of both. This project utilizes the automatic charging technique.

A power supply combination of batteries supplemented with solar-panel is an effective strategy which has proven it worth overtime. NASA space exploration vehicles, Spirit-Rover used for Mars exploration for instance, use this power management strategy. This strategy is herein applied for this prototype.

II. AUGV DESIGN OBJECTIVES AND SPECIFICATION

A. Prototype Design Objectives

The design objective of this prototype is as follows:

- Design a self-charging electric powered vehicle capable of recharging it battery autonomously on-the-go via a foldable solar-panel
- Design a prototype AuGV that can detect, avoid and navigate obstacles successfully in any unplanned or unknown environment.

B. Prototype Design Specification

The choice of using SPI (Serial peripheral interface), I2C (Inter-integrated circuit) and Async communication protocols for connecting the various subsystems stems out of the need to demonstrate the feasibility of incorporating various protocols together while still achieving the design objectives of this project.

Table 1 depicts the technical specifications of the various subsystems used for the AuGV design, while Figure 1 shows the various communication protocols used to interconnect these subsystems.

The choice of using SPI (Serial peripheral interface), I2C (Inter-integrated circuit) and Async communication protocols for connecting the various subsystems stems out of the need to demonstrate the feasibility of incorporating various protocols together while still achieving the design objectives of this project.

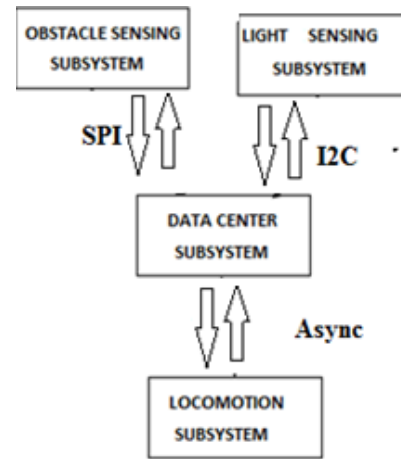


Figure 1 Our AuGV subsystems interconnections

Power supply for this project was generated via a 6volts rechargeable Lead-Alkaline battery. Power architecture used for this project is shown in Figure 2; as shown in the figure, independent regulated voltages of magnitude shown was drawn and distribution to the various subsystems of the vehicle.

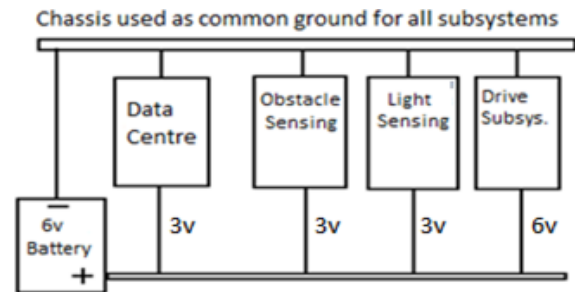


Figure 2 Power architecture

III. AGV PROTOTYPE DESIGN

Our AuGV consist of a body-module built on a “Himoto Rock Crawler RCF-8” chassis. This chassis, with specifications shown in Figure 3, comprises of four independently actuated wheels with two motors (one for each pair of wheels) and two servo-motors (one for steering each pair of wheels). The vehicle’s battery, deployable solar-panel, controller units and the remaining hardware of the AuGV prototype were all mounted on the body module and chassis frame as shown in Figure 11.



Approx. Dimensions

Vehicle Length:	645mm	Width:	380mm
Wheelbase:	465mm	Twin Motor:	RC 540
Wheel Width:	82mm	Wheel Diameter:	182mm
Battery:	6v 2000mAh	225mm between left & right wheels	

Figure 3 Himoto RCF-8” chassis & tire configuration

Table 1 AGV subsystems

Subsystems	Technical Specification
Locomotion	Receives data via data center from all peripheral systems and provides PWM drive signals to motor and steering sub-assembly.
Obstacle Detection	Identifies the presence of obstacles ahead of the AGV, and communicates this with the central controller.
Light-sensing & deployable solar panel	Identifies direction of light source, when optimum light condition have been found and vehicle is at rest, deploys solar panel. This however must be retracted at all times that the vehicle is in motion. Panel dimension is 25000mm <sup>2</sup> X 10000mm <sup>2</sup> when open.
Data Centre & Diagnostics	Manages and coordinates the activities of all subsystem. It receives and transmits all messages and display same on LCD

A decentralized-embedded system of mutually connected intelligent subsystems was employed as the solution strategy for this prototype design; similar strategy was successfully employed by state-of-art [6].

The decentralized-embedded solution employed for this prototype design is divided into:

- Hardware, subsystems & Programming
- Controller

#### A. Hardware, Subsystems, & Programming

This section discusses the hardware, subsystems and control strategy employed in the design of the AuGV prototype.

##### 1) MBED microcontroller

The intelligence of each subsystem is provided by a programmable Cortex-M3 LPC1768 mbed microcontroller; these controllers are connected via bi-directional communication protocols as shown in Figure 1. Each controller was programmed using MBED online proprietary C-language compiler (i.e. ARM mbed development environment and ARM API features).

The program code embedded in this hardware contains the control commands used for executing the desired control actions for the vehicle as well as coordinating the vehicle's overall activities.

This controller hardware is shown in Figure 4; more details about the controller can be found on the MBED website at [www.mbed.org](http://www.mbed.org) [7].

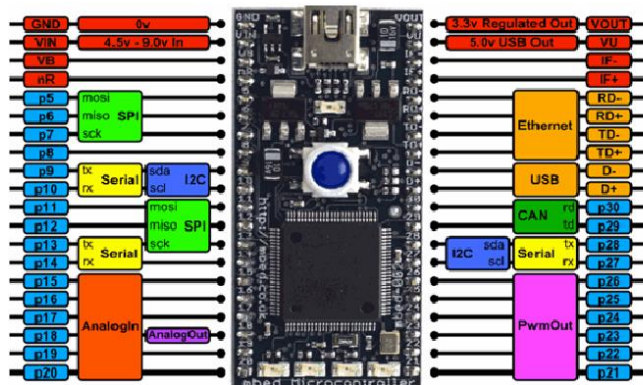
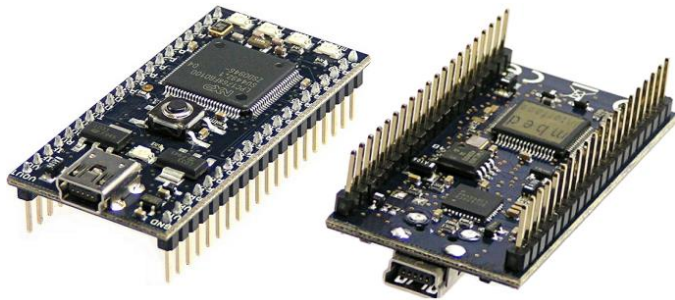


Figure 4 MBED microcontroller [7]

##### 2) Intelligent light sensing & deployable solar-panel subsystem

This subsystem is responsible for achieving the aforementioned primary design task of this project and it comprises of:

- A foldable, deployable solar-panel power supply, controlled by a dedicated servo-motor.
- An MBED microcontroller
- 4-LDR light sensing sensors situated on each side of a square box atop a 0.5m pole as shown in Figure 5 and Figure 11.

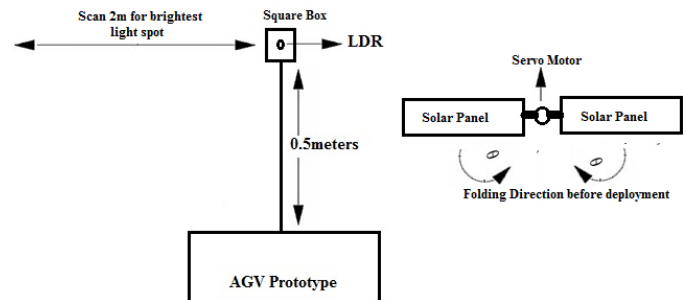


Figure 5 LDR and solar-panel block diagram representation

With this sensor arrangement and the program coded in this subsystem's controller, the vehicle can scan up to 2m radius of its environment for a spot with brightest light intensity when its battery needs to be recharged, then calculate the intensity of received bright light from all its sensors, determine the location with the brightest light source (if any detected) and transmit this location to the data-center subsystem (or data-center for short). This data is used by the data-center to successfully navigate the AuGV to the spot with brightest light source by guiding the locomotion-subsystem.

Data communication between the data-center and this subsystem is via asynchronous-protocol at a predetermined data or baud rate of 9600bps. An excerpt of the program code for this is as under:

```
async_port.baud(9600); //set baud rate to 9600
if (async_port.readable () = 1); //is there a character to read?
read_val = async_port.getc(); //if yes, then read it
read_val = read_val & 0x0F; //AND out unwanted bits
```

The switch word or control code for this data communication which represents the state of scanning and detection of brightest light spot by this subsystem is presented in Table 2.

When the battery needs recharging and there exist differential light sources, Table 2 represents the location of detected brightest light spot; this data is transferred to the data-center as hexadecimal switch-word. When the battery needs recharging and the vehicle is in a uniformly bright environment, the prototype remains still, deploy its solar-panel and start recharging its battery. But when the vehicle is in a pitch dark environment and its battery needs recharging, the prototype remains still.

**Table 2 Light condition switch word**

Light Condition	Switch word – hex	Switch word – binary
No light	0x00	0000 0000
Front	0x01	0000 0001
Front – Right	0x02	0000 0010
Right	0x03	0000 0011
Back – Right	0x04	0000 0100
Back	0x05	0000 0101
Back – Left	0x06	0000 0110
Left	0x07	0000 0111
Front – Left	0x08	0000 1000
Full Light	0x09	0000 1001

3) *Intelligent Drive (Locomotion) Subsystem*

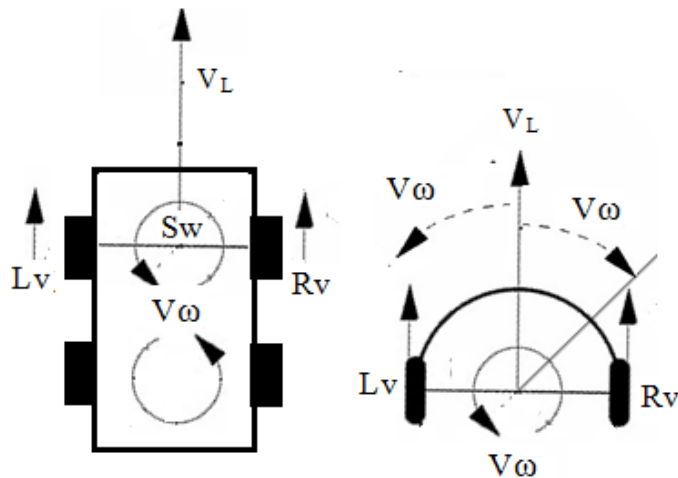
This subsystem is responsible for the vehicle locomotion and navigation and it comprises of:

- Two servos motors (one on each wheel) for steering and propelling the AuGV forward or reverse.
- Electric motor ESC (Electronic Speed Controller) for driving and controlling the speed of the vehicle
- An embedded MBED microcontroller for controlling the operation of this subsystem.

4) *Locomotion kinematic*

Two independent servo-motors connected to each front wheel of the AuGV-prototype drives the vehicle independently. This configuration results into two linear (one for each wheel) and a circular (or arc) trajectories for the vehicle; in other words, the vehicle changes speed and direction by the application of differential PWM signals to the servo-motors of it front wheels [8].

Our kinematic model follows suggestion by state-of-art [9][10] and is shown in Figure 6.



**Figure 6 Locomotion kinematics**

(note: 15° angular turn was used in this project demonstration)

To estimate the resultant linear ( $V_L$ ) and angular velocity ( $V_w$ ) of steering the vehicle, we define  $R_v$  &  $L_v$  as the linear velocity of the right and left wheel respectively,  $S_w$  as wheel distance = 225mm (see Figure 3).

The estimated resultant linear velocity is hence defined as  $V_L = (R_v + L_v)/2$  while  $V_w = 2*(R_v - L_v)/S_w$ . With this kinematic approach, the vehicle does not require too large an area for turning, maneuvering and other types of movements. To propel the vehicle forward, equal pwm-signal were fed to each front wheel motor, resulting to a linear speed of:

$$V_L = 0.5R_v \quad \text{or} \quad V_L = 0.5L_v \quad (1)$$

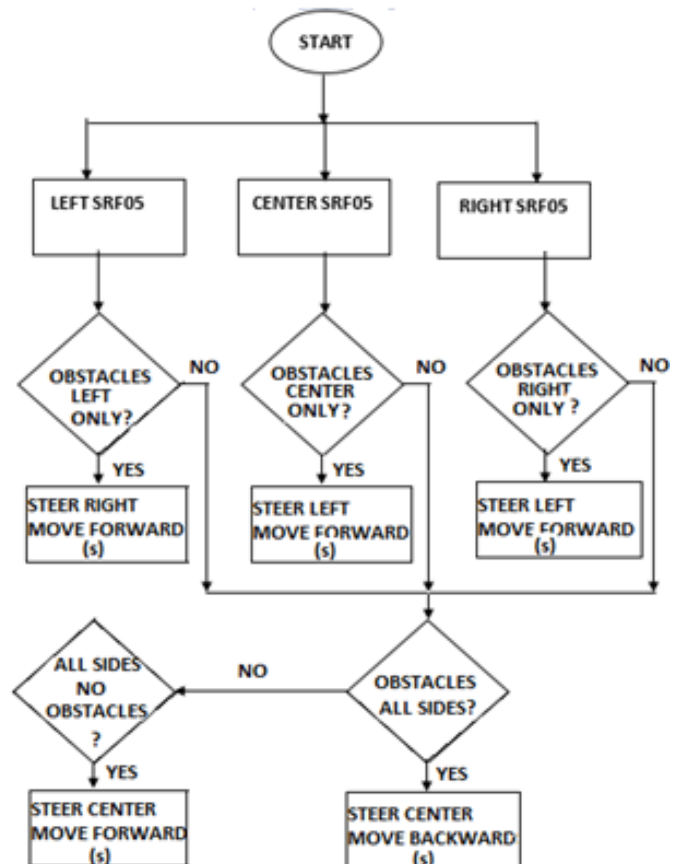
To turn the vehicle right or left, we programed the controller to cut the pwm-signal supplied to the motor in the required direction; the result is a turning moment about the center of mass of the vehicle, leading to an angular steering motion defined by:

$$2(R_v - L_v)/S_w \quad (2)$$

Data communication between the data-center and this subsystem is via I2C protocol, an excerpt of the program code is as shown under:

```

I2Cslave slave (p9, p10); //Configure I2C slave
PwmOut steering (p21); //Define PWM Output to wheels
Servo motor (p22); //Define PWM Output to the ESC
i2c_port.start(); //force a start condition
i2c_port.write(addr); //send the address
i2c_port.write(switch_word); //send one byte of data
i2c_port.stop(); //force a stop condition
wait (0.00000002);
    
```



**Figure 7 Navigational obstacle avoidance flow-chart**

Figure 7 depicts our self-explanatory navigational flow-chart which was employed for coordinating the movement of the vehicle using obstacle data from the obstacle-subsystem; while Figure 8 shows the interfacing of the locomotive and obstacle subsystems via the central-controller.

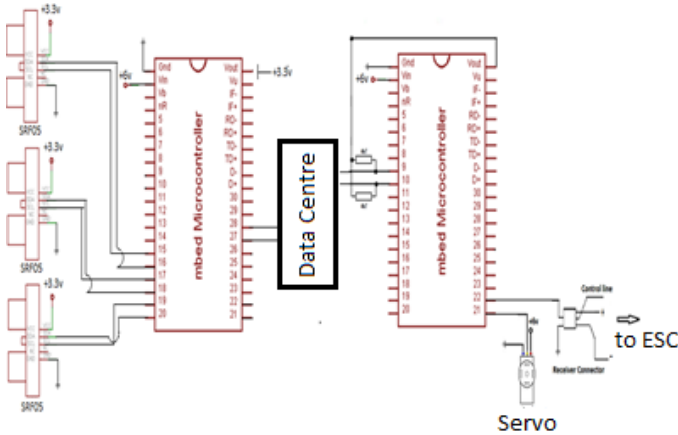


Figure 8 Interfacing of locomotion and obstacle subsystems

### 5) Obstacle detection and avoidance (ODA) subsystem

Different techniques have been proposed and employed in AV obstacle detection & avoidance. State-of-art [11] recommended a deadlock detection and prevention algorithm which assumes that AGV reserves grid-blocks in advanced to prevent collisions; while State-of-art [12] designed an AGV collision detection & avoidance system using fuzzy logic, rotational ultra-sonic sensor and PLC. State-of-art [13] stipulated that the design of AV must be simply smart such that the vehicle can successfully navigate in any unknown environment (boulders, rock, hazardous situations, etc.) without any knowledge of that environment.

Our ODA approach in this AuGV design follows stipulation by state-of-art [6] and comprises of three low cost SRF05 rangefinder sensors arranged in the configuration shown in Figure 9. These sensors are connected to the MBED controller of this subsystem and controlled by the controller via instruction codes programmed in this controller. Sensors, MBED microcontroller and program-codes were the hardware-software platform used for achieving obstacle detection and avoidance in this prototype.

Communication between the data-center and this subsystem is via SPI protocol, an excerpt of the program code is as shown under:

```
SPI ser_port(p11, p12, p13); // mosi, miso, sclk, ssel
ser_port.format(8,0);
ser_port.frequency(4000000);
if (ser_port.receive()) //test if data transfer has occurred
recd_val1 = ser_port.read (); //Read byte from master
recd_val1 = recd_val1& 0x0F;
```

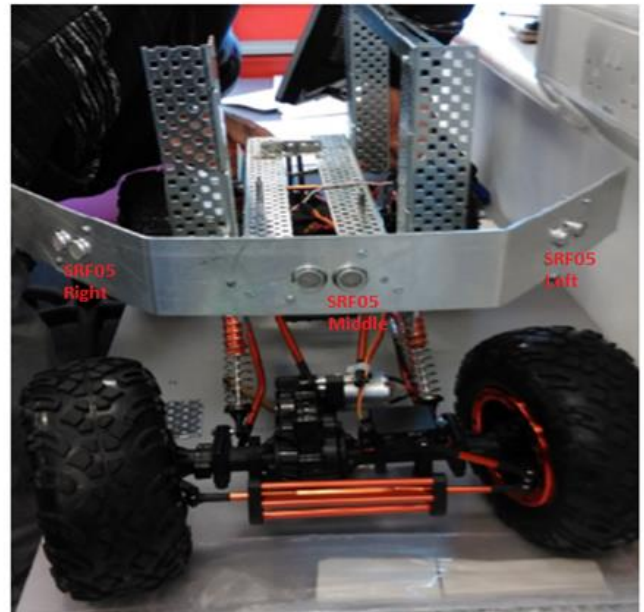
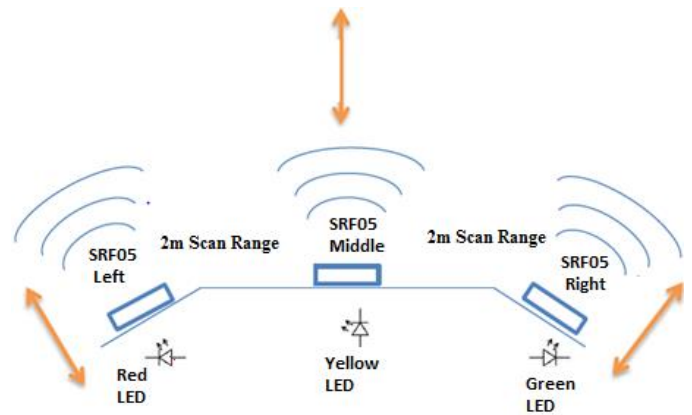


Figure 9 SRF05 configuration on AuGV prototype

Table 3 Obstacle position switch word

Obstacle Position	Switch word – hex	Switch word – binary
None	0x00	0000 0000
Ahead	0x01	0000 0001
Right	0x02	0000 0010
Left	0x03	0000 0011
All Position	0x04	0000 0100
Front – Right	0x05	0000 0101
Front – Left	0x06	0000 0110

The switch word or control code for data communication which represents the position of the detected obstacle(s) by this subsystem is shown in Table 3. When no obstacle exists in the path of travel of the vehicle, 0x00 hexadecimal word is relayed to the data-center. When obstacle exists directly in the path of the vehicle, 0x01 is relayed to the data-center; similarly for other code-words shown in the table.

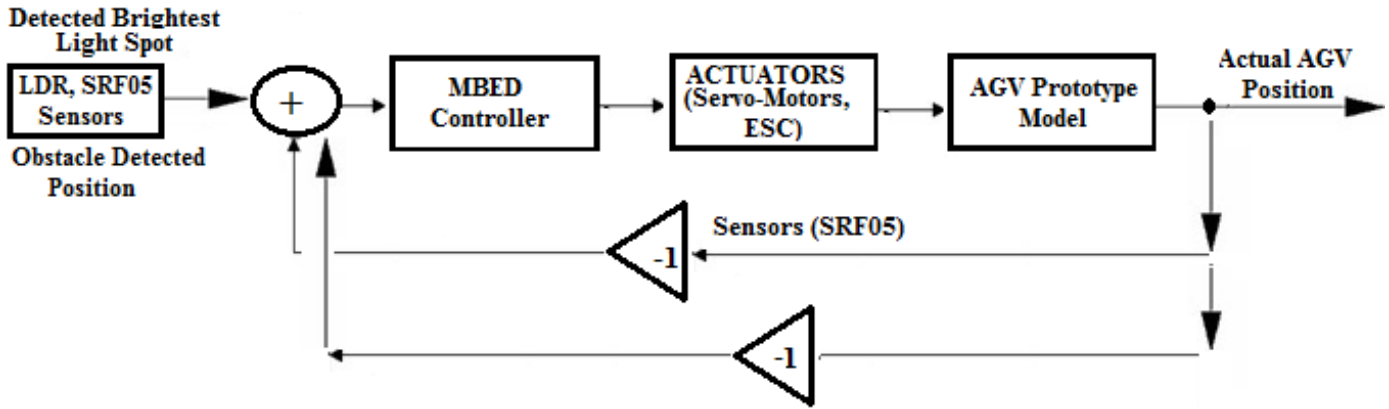


Figure 10 Block diagram of AGV controller

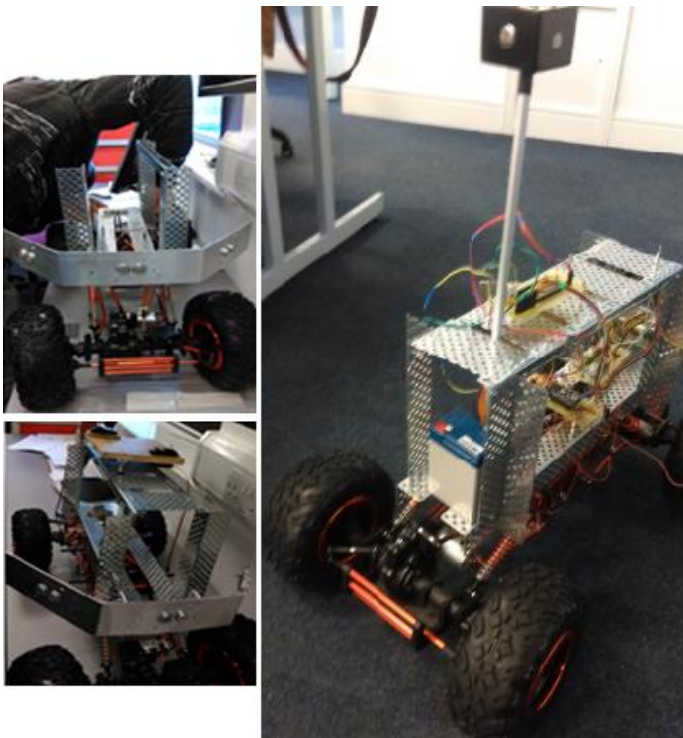


Figure 11 AuGV assembled prototype

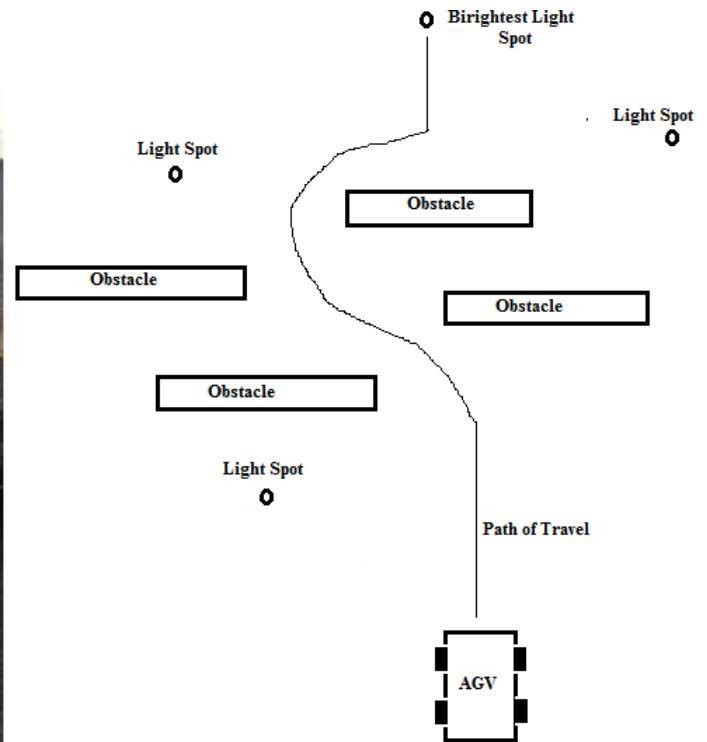


Figure 12 Design task demonstrations. All obstacles positions were unknown to the vehicle

6) Smart data-centre subsystem (central controller)

Our prototype data-center or central controller comprises of a 16×2 LCD Display monitor, an MBED-microcontroller and associated program codes for centrally controlling all the various activities of the vehicle.

At the heart of this AuGV prototype is the data-center subsystem which receives, processes and make informed decisions based on the data received from all other subsystems. In other word, this subsystem act as the interconnecting link between the different subsystems by transmitting processed communication data it receives from other subsystems to the locomotion subsystem. As an instance, it receives data from the light-sensing subsystem about detected spot of brightest light source, it processed the hexadecimal switch word for this data in conjunction with similar data received from the obstacle subsystem about obstacle(s) it detects and then communicates

the necessary directional action required to locomotion subsystem for navigation.

It communication with all other subsystems were programmed as event-triggered activities since the time taken for either tasks to occur is not predictable.

B. Controller Solution

The task of our AuGV controller is to regulate the vehicle motion on detecting bright source of light when the vehicle battery needs recharging; as well as avoidance of obstacles while navigating to this detected light source. Our control solution is shown in the control-diagram in Figure 10.

The reference trajectory inputs to the mbed-controller are data from LDRs and SRF05s sensors about the location of detected brightest light source and detected obstacle(s) that may hinder free flow of the vehicle to this spot respectively. SRF05

sensor data and position data were negatively feedback to the controller, this feedback data were used to continuously guide the vehicle in reaching this spot successfully. Actuators were provided by servo-motors and ESCs of the locomotion subsystem by ways of propulsion and steering the vehicle.

### C. Picture of Complete Prototype

Picture of the assembly and completed prototype of the developed AuGV is presented in Figure 11.

## IV. TESTING AND COMMISSIONING

Testing of the developed AuGV prototype was conducted in a controlled environment of approx. 30m×30m with comparatively even ground surface; the test environment was equipped with light sources of varying intensities and these were spontaneously situated. Obstacles of different sizes (but none higher than 600mm) were spontaneously distributed but a minimum of 1m space gap was observed between any two obstacles - the essence of this was to allow the vehicle navigates successfully in the ensuing space due to its size.

Testing of our developed AuGV prototype began by simulating the test environment with uniform light sources, spontaneous distribution of different obstacles and the vehicle battery needs recharging. The vehicle scanned its environments, calculated the received light intensities from light-sensors and upon deciding that it is in a uniformly light environment by its central controller, it remained still as programmed and deployed its foldable solar-panel to begin charging its battery (see Figure 13a). The next testing stage was simulating a dark test environment by turning off all light sources while the vehicle battery needs to be recharged; obstacles were spontaneously distributed. The vehicle scans its environment for light sources, continues its movement in its original direction while it continues to scan for light source; it detects and avoids obstacles in the process. When no light sources were detected, the vehicle remained still as programmed to conserve its remaining battery (this was simulated on the basis that the vehicle battery has been depleted to a critical level and urgently needs recharging or else the vehicle will shut down).

Simulating the test environment with varying light sources of different intensities was carried out while the vehicle battery needs to be recharged. Spontaneous distribution of obstacles in the vehicle path of travel was also included in this stage of the test. The vehicle scans its environment, calculates the intensities of detected light source and determines the location with the brightest light source. It navigates past the various impeding obstacles to the spot where the light intensity was brightest, and then deploys its solar panel to begin charging its battery (see Figure 13). Also see Figure 12 which shows one such demonstration of the AuGV testing.

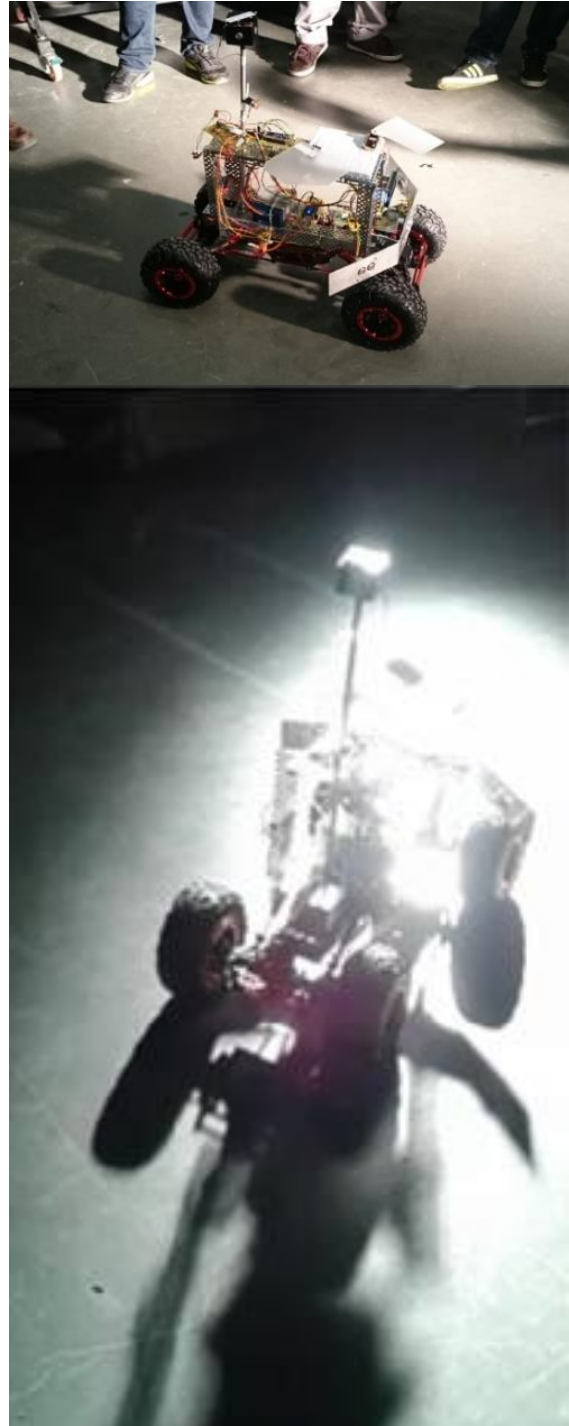


Figure 13

(a) Prototype testing under uniform light;

(b) Testing of AGV prototype

(See [supplementary file](#) for testing demonstration video)

(See [supplementary file](#) for testing demonstration video)

## V. CONCLUSIONS

In this paper, the authors developed and presented a working AuGV prototype that simulated AV power management in terms of scanning its environment for brightest light source & deploying a foldable solar-panel to charge its battery when the battery charge level falls below a predefined minimum threshold.

Decentralized embedded system technologies incorporating different communication protocols & programming codes were used to achieve the intelligent design of the developed AuGV prototype; also hardware in the form of sensors technologies, microcontroller, Himoto RCF-8 chassis, body modules, etc. were incorporated to achieve the design task of this project.

The prototype developed was tested on its ability to detect brightest light sources in uniformly light, pitch-dark and differentially light test environment. Also, the vehicle was tested on its ability to successfully detect & navigate obstacles while navigating to the spot where brightest light-intensity was detected. Without sophisticated algorithms such as deep-learning, SLAM, etc. the developed prototype was able to navigate spontaneous objects in an unfamiliar environment successfully and still recharge its batteries at the spot of brightest light detected by deploying its foldable solar-panel. Hence, the developed prototype successfully achieved its primary and secondary design tasks.

Testing repeatability of developed prototype was almost 100% except for two occasions where the vehicle angle of turn were not accurate enough to avoid an obstacle, leading the vehicle to collide with that obstacle.

Additional notes:

- The complete programming code for this project has been uploaded onto MBED developer website (at [www.mbed.org](http://www.mbed.org)) and can be downloaded for free without any copyright issue.
- The chassis for this vehicle was not fabricated but purchased solely for this prototype development.
- Actual charging up of the battery was not demonstrated due to financial constraint. The authors of this project assume a simulated scenario in which direct charging up circuitry was in place.

## VI. FUTURE WORK

Although this prototype AuGV achieved its designed tasks and objectives, future work & recommendation are:

- Use of CAN Protocol: Considering the environment in which AV are conventionally employed to operate in, CAN protocol will be an excellent fit. Communication protocols used for this project are susceptible to corruption and doesn't offer state-of-the-art features like error-checking, etc. as CAN
- Incorporate of SLAM Algorithm: While the developed prototype navigated successfully without prior knowledge of its environment, it is paramount that those future similar

vehicles collect and build data simultaneously about their environment as well as being virtually aware of their environments.

- Incorporate Camera or (and) LIDAR: Environment perception is key to AV as it provides the vehicles with crucial information on their driving environment like free drivable areas, velocities, surrounding obstacles' locations, etc. Based on the sensors implemented in this project, environment perception task for future work will be better handled by the incorporation of devices like LIDAR, camera or a combination of both devices.

## ACKNOWLEDGEMENT

The designs, researches and results herein presented in this paper is the combined project work of the above listed authors, undertaken at the University of Derby, United Kingdom under the supervision of Tim Wilmschurst.

## REFERENCES

- [1] The IET (2017). "Swimming and Wriggling robots unveiled for Fukushima clean-up". [VIEW](#) [last accessed 29th July, 2017].
- [2] Yaghoubi S., Khalili S., Nezhad R. M., Kazemi M. R. and Sakhaifar M (2012). Designing and methodology of AGV vehicle Robots/ Self-Guided Vehicle systems, future trends. IJRRAS.
- [3] Junemann R. and Schmidt T. (2000). Material flow system. Springer.
- [4] Tompkins J.A., White J. A., Bozer Y. A. and Tanchoco J.M. (2003). Facilities Planning, 3rd Edition. John Wiley & Sons, Inc.
- [5] McHaney R. (1995). "Modelling battery constraints in discrete event automated guided vehicles simulations". International Journal of Production Research. pp. 3023 – 3040. [CrossRef](#)
- [6] Kumar S. and Singh R. (2011). "Advanced speed control of an automated guided vehicle". Proceeding of the International Multi-conference of Engineers and computer scientists. Vol II.
- [7] MBED Website (2015). Available at:
- [8] < <http://www.MBED.org> > [Accessed: 20th July, 2017].
- [9] Ehrlin E. and Tornqvist M. (2012). Networked control of Autonomous Guided Vehicles.
- [10] Antonelli G., Chiaverini S. and Fusco G. (2005). "A calibration method for odometry of mobile robots based on the least-squares techniques: theory and experimental validation". IEEE Transactions on vol. 21, pp. 994 – 1004. [CrossRef](#)
- [11] Bethencourt J., Qiang L. and Fernandez A. "Controller design and implementation for a differential drive wheeled mobile robot", pp 4038 – 4043.
- [12] Kim K.H., Jeon S.M. and Ryu R.K. (2006). "Deadlock prevention for automated guided vehicle in automated container terminals". Spectrum, vol. 28,, pp. 659 – 679. [CrossRef](#)
- [13] Yahyaei M., Jam J. and Hosnavi R. (2009). Controlling the navigation of automatic guided vehicle using integrated fuzzy logic controller with programmable logic controller. Springer-Verlag, London. pp. 795 – 807.
- [14] Darren Krasny, (2003). "The Autonomous Maintenance Robot for confined space Maintenance Application", Battelle Memorial Institute.