



Mona: an Affordable Open-Source Mobile Robot for Education and Research

Farshad Arvin¹ · Jose Espinosa¹ · Benjamin Bird¹ · Andrew West¹ · Simon Watson¹ · Barry Lennox¹

Received: 19 December 2017 / Accepted: 2 May 2018 / Published online: 23 May 2018
© The Author(s) 2018

Abstract

Mobile robots are playing a significant role in Higher Education science and engineering teaching, as they offer a flexible platform to explore and teach a wide-range of topics such as mechanics, electronics and software. Unfortunately the widespread adoption is limited by their high cost and the complexity of user interfaces and programming tools. To overcome these issues, a new affordable, adaptable and easy-to-use robotic platform is proposed. *Mona* is a low-cost, open-source and open-hardware mobile robot, which has been developed to be compatible with a number of standard programming environments. The robot has been successfully used for both education and research at The University of Manchester, UK.

Keywords Mobile robot · Robotics for education · Open-hardware · Open-source

1 Introduction

Robotic systems are an excellent demonstration technology for STEM (Science, Technology, Engineering and Mathematics) teaching due to their inherent multidisciplinary nature [1]. The basic concept of a robot is something that students of all ages and abilities can understand. The learning activities that can be undertaken span the full range of abilities, from entry-level primary school students [2, 3] through to university undergraduates (UG) and post-graduates (PG) [4, 5]. Robots are also being used for non-technical learning, especially in support of those with learning disabilities [6].

The field of robotic systems brings together electrical engineering, mechanical engineering (or combined mechatronic engineering) and computer science [7] and therefore covers a large number of underpinning topics, as shown in Fig. 1. Robots can be used to directly teach robotics [8], or indirectly as a platform to teach other topics, both technical and non-technical (soft skills) [9]. One of the biggest strengths of using robots for teaching is the practical nature of the work; it is very easy to physically demonstrate or experiment on hardware subsystems or on full robots.

There have been several studies over the last few decades about how engineering students learn [10]. Whilst the studies showed that students have a range of learning styles, there is a notable preference towards active, sensing and visual learning [11]. Traditional university teaching methods, large-scale lectures with supporting laboratories, are designed for cost and time efficiency [12], but there is a growing demand from the latest generation of students to explore different learning approaches [13–16]. Mobile robots provide an excellent opportunity to explore these new methods of learning.

Unfortunately robots can be expensive and their use in teaching can be limited to a only a few laboratory sessions a year. Assuming a 50 week working year, with a 35 hour working week, mobile robots utilisation for teaching could be as low as 1%. This could be increased if the robots are used for project work, but in general teaching robots do not offer good value for money in terms of their use.

It is important therefore to try and use robots in teaching which can also be used for research. Not only does this improve the utilisation and value for money, but it also ensures that students are being taught using state-of-the-art technology.

This paper presents the design of the *Mona* robot, a low-cost, open-source platform which has been developed for both teaching and research. Details of the robot's application to research have been presented in [17]. The contribution of this paper is to its application to teaching.

✉ Farshad Arvin
farshad.arvin@manchester.ac.uk

¹ School of Electrical and Electronic Engineering,
The University of Manchester, M13 9PL, Manchester, UK

Fig. 1 Underpinning technologies for robotic systems



1.1 Robotics as a Teaching Platform Case Study

The School of Electrical and Electronic Engineering (EEE) at The University of Manchester, UK, uses robots in two ways; indirectly as an introduction to practical group project work for all UG students and directly in dedicated units as part of the B.Eng. and M.Eng. Mechatronic Engineering syllabi.

All second year UG students in the School take a compulsory unit entitled, Embedded Systems Project (ESP), which is a 20 credit unit run across both semesters of the academic year. Students are placed into teams of 5 or 6 and are given the task of building an autonomous white-line following robot buggy [18]. At the end of the year, a race day is held where the buggies compete against each other on a specially designed track.

The ESP is one of the flagship units in the School of EEE and provides students with an introduction to soft skills such as team working, project management and report writing as well as technical topics such as motor characterisation, embedded system programming, sensors, modelling and engineering drawing. The average Year 2 intake is approximately 280 students, split into 50 groups and the costs of running the unit are significant; the consumables budget is approximately £20k per year, in addition to 1200 hours of academic time commitment and 600 hours of support staff time. Whilst the ESP is highly successful and popular with both students and the School's Industrial Advisory Board, the operating model is not sustainable for more units in the syllabus.

Building on the ESP, third year Mechatronic Engineering students take a compulsory unit on Mobile Robots and Autonomous Systems (MRAS). This unit covers topics such as mobile robot kinematics (legged and wheeled), sensing & perception, planning & navigation and estimation & filtering. The average number of students on the course is between 40 and 50.

A key emphasis in the design of the MRAS unit was the use of real-world robotic systems. To achieve this, a new teaching style was utilised, which aimed to bridge the gap between the low-cost traditional lecture-based teaching

with supporting labs, which provides limited opportunity for practical learning, and the high-cost problem-based learning (PBL) approach of ESP.

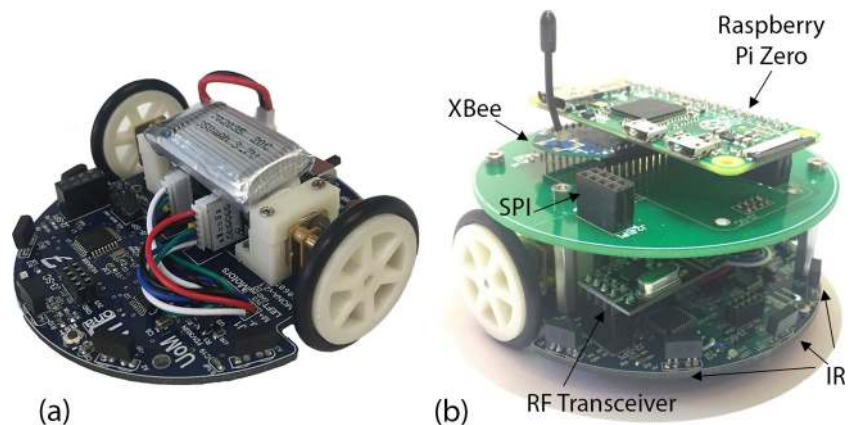
The teaching method employed is best described as practical lectures. All 12 lectures are given in a laboratory setting, where students have access to PCs (Fig. 2). This allows them to do simulation or experimental-based tutorial questions during the lecture, as well as being able to connect to robots to run simple experiments or exercises to observe the outcomes. In addition, there are two traditional laboratory exercises which build on the work done during the lectures.

Previous versions of the MRAS course have utilised practical lectures, but only using simulations (MATLAB-based tutorial questions). This has been popular with the students, however it was felt that significant improvements could be made if the students had access to robots as well. The challenge lay in identifying a suitable small-scale, low-cost platform which could be easily used by the students. To meet these needs, a new platform has been developed: the Mona robot (Fig. 3), which is an open-hardware and open-source, low-cost platform. At The University of Manchester, it is important that the teaching is related to the world-leading research for two reasons. Firstly, it ensures



Fig. 2 Practical lecture facility at the University of Manchester

Fig. 3 **a** Mona robot and **b** an expansion board for the off-the-shelf modules: Raspberry Pi Zero, RF transceiver, and XBee



that the students are being taught with the latest technology and they can see how the material links into real-world challenges. Secondly, it provides an excellent pathway for those students who are interested in pursuing a research career. It was therefore a requirement that the Mona robot could be used for both teaching and research.

This paper presents the design of i) Mona's hardware and modules and ii) the software and programming of the robot. The rest of the paper is arranged as follows: Section 2 provides a review of existing educational robotic platforms. Section 3 presents the Mona hardware and Section 4 presents software and robot programming platforms. Section 5 provides examples of experiments conducted with the Mona robots. Section 6 presents the results from the experiments and Section 7 discusses the outcomes and presents conclusions.

2 Robotics in Education

There are a number of important considerations to make when selecting a robotic platform for teaching including cost, size, functionality and interface. Educational robot platforms can broadly be split into three categories; manipulators (used for industrial robotics) [19], legged mobile robots [20] and wheeled mobile robots [21]. For the MRAS unit discussed in Section 1.1, only mobile robots will be discussed here.

In terms of practical demonstrations or experiments, legged mobile robots are more expensive, larger and more complex. Platforms such as the Nao robot [22] cost approximately £6.5k per robot, which make them infeasible for class sizes of more than 10 or 20. Other legged robot platforms; humanoids, quadrupeds and hexapods, are available, however low-cost alternatives do not provide the functionality required [20].

The difference between wheeled and legged robots is the means of locomotion. Considering Fig. 1, actuation (and the associated kinematics and dynamics) are only

one part of a robot system. All of the other subsystems can be made independent (for teaching purposes) of the locomotion method. Wheeled robots therefore offer a low-cost alternative for teaching.

Wheeled robots have been used as teaching aids for decades, but it has only been the last 10 years that small-scale, low-cost platforms have become widely available [23–25]. Platforms such as LEGO Mindstorms allow quick and easy access to robotics and their underpinning subjects across the education spectrum [26, 27]. For more advanced users, the e-puck became one of the first commercial platforms to have an associated robotics syllabus with it [23].

The e-puck also became widely used for research into swarms and collective control as its small scale allowed for large numbers of robots to be used in experiments [28]. A major drawback of the e-puck is its cost, retailing at around £700 per vehicle. This is significantly cheaper than legged robots, however for a class of 50 students, it is unaffordable to provide one robot per student. Since its release, there have been a number of other robots developed as a low-cost alternative for both education and research. These range in complexity, cost and functionality, and a selection of them are shown in Table 1.

2.1 Commercial Availability

Table 1 shows that whilst there have been a number of robots developed as low-cost platforms for education and research, very few of them have been successfully commercialised and are therefore not widely available. The majority of the robots are open-source in terms of hardware and software.

Instead of impacting the commercial feasibility of the robots, anecdotal evidence is that end users are still likely to buy a complete unit for convenience. They are then able to modify the hardware or software (including adding extra functionality) to tailor the system to their needs. This approach provides the full range of black-box uses for education or basic research through to full customisation and the understanding of the fundamental systems.

Table 1 Comparison of educational robots

Robot	Release date	Cost	Research or education	Open source	Commercialised
Aeris [29]	2017	Unknown	Education	Yes	No
Andruino-A1 [30]	2016	£30	Education	Yes	No
Colias [31]	2014	£25	Research	Yes	No
E-puck [23]	2009	£700	Both	Yes	Yes
Jasmine [32]	2006	Unknown	Research	Yes	No
Khepera IV [33]	2016	£3.5k	Both	No	Yes
MarXbot [34]	2009	Unknown	Research	Yes	No
microMVP [35]	2017	£70	Both	Yes	Yes
Rice r-one [36]	2013	£200	Both	Yes	No
Thymio II [24]	2010	\$105	Education	Yes	Yes

The Mona robot has been developed, in collaboration with a commercial partner, as a low-cost platform for robotic education and swarm/collaborative research. It currently retails at £100 per robot and is fully open-source (hardware and software). It has been successfully used for teaching on the undergraduate MRAS unit (Section 1.1) and MSc projects as well as being used for swarm research [17].

2.2 Teaching Resources

Most of the robots in Table 1 have aspirational plans for their use in a curriculum. Only the E-puck and Andruino-A1 robots have specific details of how they can be used in laboratories or as general teaching aids. The E-puck, being the first to be developed for teaching, is used in topics such as signal processing, automatic control, behaviour-based control, localisation and path planning and distributed intelligent systems. The Andruino-A1 is designed to be an introduction to mechatronic design and assembly and for embedded systems programming.

The Mona robot currently allows students to undertake practical experiments on system characterisation and motion planning. From a characterisation perspective (Section 5.1), students can do experiments on the sensors and actuation systems. For motion planning (Section 5.2), they can learn about open- and closed-loop control, obstacle detection and avoidance and more complex swarm algorithms.

A simulation environment is also being developed along with an add-on module to allow ROS control. This could enable low-cost practical ROS examples to be run. There are also plans to use the robots to for basic path planning (Wavefront, A*).

2.3 Research Capabilities

The primary research undertaken using the robots in Table 1 is focused on multi-agent systems or swarm robotics. The MarXbot has a specific focus on long-term autonomy [34]

and the Colias, E-puck and Jasmine robots have a number of publications in the general field of swarm robotics.

The perpetual swarm interface designed for Mona in [17] allows for large-scale, long-term autonomy and swarm scenarios to be investigated. The Monas are also currently being used to explore fault tolerant control of multi-agent systems, pheromone communication based swarm behaviour and human-robot-interactions using mixed-reality interfaces

3 Mona Robot Hardware Specification

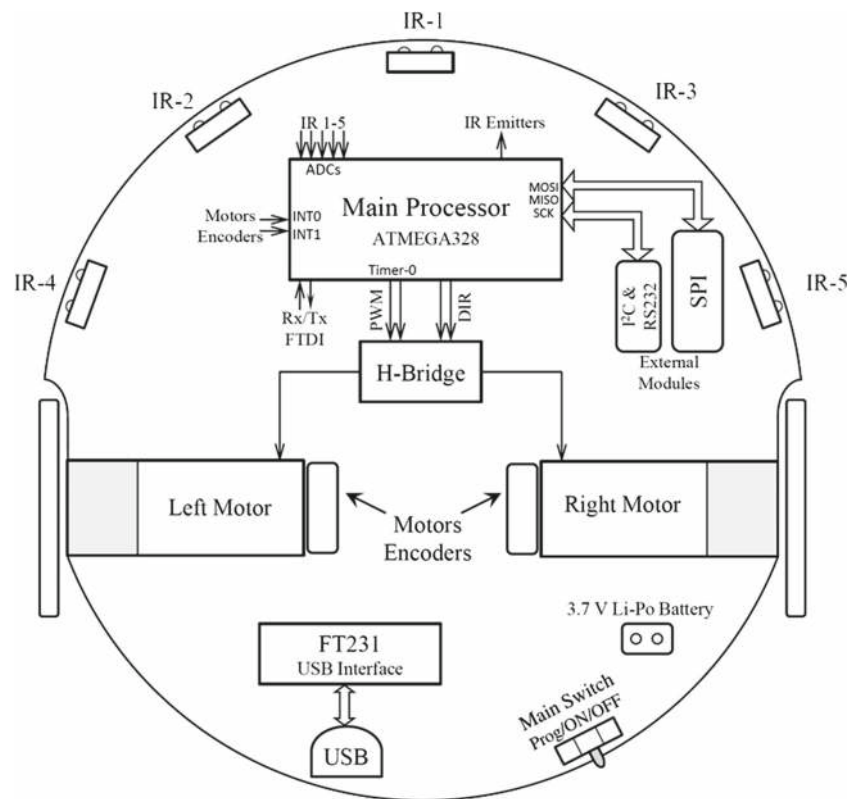
This section presents the robot's hardware systems. Mona was initially based on Colias [31], a low-cost and open-source robot, which has been used for swarm robotic research [37]. To allow flexibility for both teaching and research, a similar open-source and open-hardware approach was taken. Figure 4 shows the hardware architecture of the Mona robot.

3.1 Main Controller

Mona uses a low-cost ATmega 328 microcontroller (μ C) as the main processor. The primary reason for utilising this μ C was to develop the robot based on the Arduino Mini/Pro architecture and to be compatible with Arduino's open-source programming interface. Figure 5 shows the architecture of the main controller, the μ C's internal modules, and external modules which are connected to the main controller.

The utilised μ C has a low-power 8-bit AVR RISC¹ architecture with 131 instructions which are mostly executed with single clock cycle. It has several internal modules providing easy and reliable minimum system to develop the Mona robot. The μ C has an internal 32 KB flash memory, 1 KB EEPROM, and 2 KB SRAM, which

¹Reduced Instruction Set Computer

Fig. 4 Hardware architecture of a Mona robot

provide enough memory for programming and parameter allocation. The main clock source is an external 16 MHz crystal oscillator. An internal timer module is used to generate pulses for the motors' speed control and other function which requires an internal timer. Eight analogue to digital converter (ADC) channels are used to connect to the IR (infra red) sensors for obstacle range estimation and to monitor the battery level. The μC supports several serial communication methods such as RS232, I²C, and SPI, which are used for programming the flash memory or communication with external modules.

The μC directly controls the motors' driver and communicates with the PC using its USB driver. General purpose I/Os (input/output ports) are connected to LEDs and IR emitters. Moreover, there are two interrupt channels directly connected to the μC which are triggered by an external event. Mona uses these external interrupts for motors' encoders which will be discussed in Section 3.3.

3.2 Actuation

Two DC motors (Fig. 6) with direct reduction gears are used as the main actuation method. The architecture shown in Fig. 4 shows that a symmetrical differential driven configuration was used to control the robot's motion. The robot's wheelbase is 80 mm and the diameter of the wheels is $d_w = 28$ mm so the forward velocity of the robot is approximately 88 mm per revolution.

The rotational speed for each motor is controlled individually using pulse-width modulation (PWM). Each motor is controlled separately by an H-bridge DC motor driver, which requires approximately 74 mW to operate. As the motors are directly powered by the on-board battery, any voltage drop impacts the speed of the robot. Therefore, the battery's voltage must be considered in the robot's kinematic model.

The output voltage of the PWM (v_m) is a fraction of the maximum voltage of the source (E_{bat}) and the duty cycle of the PWM signal (p); $v_m = p \cdot E_{bat}$. The maximum velocity of the shaft, N_{max} , with a macroscopic model of the utilised motors [38], is shown in Eq. 1:

$$N_{max} = \alpha_m E_{bat} + \beta_m, \quad (1)$$

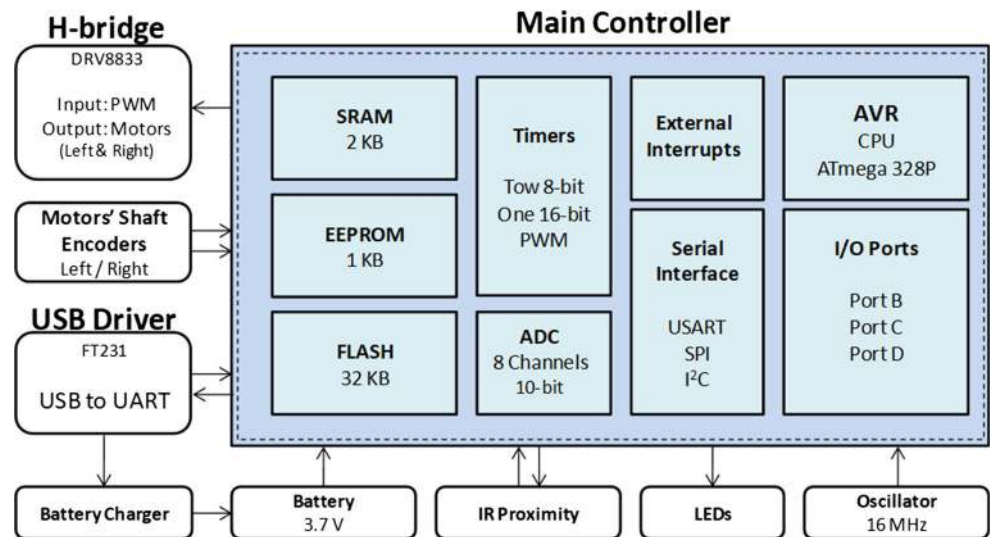
where α_m and β_m are two coefficients dependent on the motor's characteristics and the robot's design. Suitable values of α_m and β_m were estimated through empirical experiments.

The kinematic model of the robot follows general differential-driven kinematics:

$$\xi = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = f(\varphi_l, \varphi_r, \delta t), \quad (2)$$

showing the position estimation of the robot ($\xi = [x \ y \ \theta]^T$) depends on the speed of the left and right wheels ($\varphi_{l,r} =$

Fig. 5 Architecture of the main controller. The filled boxes indicates μC internal modules and white boxes indicate external modules for Mona



$p_{l,r} N_{max} \pi d_w$) in a δt time span. p is the duty cycle of PWM for each wheel, which is a variable between 0 and 1.

The required torque at motor's shaft can be calculated by: $\tau_m = \tau_{load} + \tau_{movement}$, and $\tau_{movement} = \tau_{wheel} / n_{gear}$. Since the motors' gearbox ratio, n_{gear} , is relatively high (250:1) and the robot is lightweight (45 g), the robot requires little torque to move ($\tau_{movement} < 2 \times 10^{-7}$ N.m). As a result, the acceleration of the motors is similar to the no-load condition which means that the velocity rise-time is a few milliseconds. This means that acceleration can be considered instantaneous so the dynamics of the robot do not need to be modelled.

3.3 Motion Control

Precursor robots such as Colias [39] and AMiR [40] used encoderless position estimation [38], however this resulted in low precision and required additional feedback from the motors' current to operate.

Each of Mona's motors has a small-package magnetic encoder attached to the motors' shaft. Figure 6 shows the configuration of the utilised motors. Each encoder includes a 6-pole magnetic disc and two hall effect sensors which generates 12 pulses per shaft revolution (before the gearbox). Due to the limited number of IO pins on the μC , only the output of one of the sensors was used which reduces the resolution to 6 pulses per pre-gearbox shaft revolution.

The post-gearbox resolution is therefore 1500 pulses per wheel revolution ($\approx 0.24^\circ/rev$). The encoders' outputs are connected to the μC 's external interrupt pins. Each pulse calls an interrupt routine in the main controller, which was used to increment an independent counter variable for each wheel.

The output of the encoders can be used as an input to a controller for closed-loop motion control.

3.4 Sensor Systems

Mona has been developed so that a range of sensors can be added through expansion boards, however there is a set of fixed short-range IR proximity sensors which are used for motion planning. Five sensors are mounted on the front half of the robot, with a 35° spacing between them (see Fig. 4). These are connected to ADCs on the μC which allow the estimation of the distance to an obstacle to be made.

Obstacle detection and distance estimation use fundamental principles of electromagnetic radiation and its reflection. The reflected IR signal intensity measured by a sensor is mathematically modelled using (3) [41]:

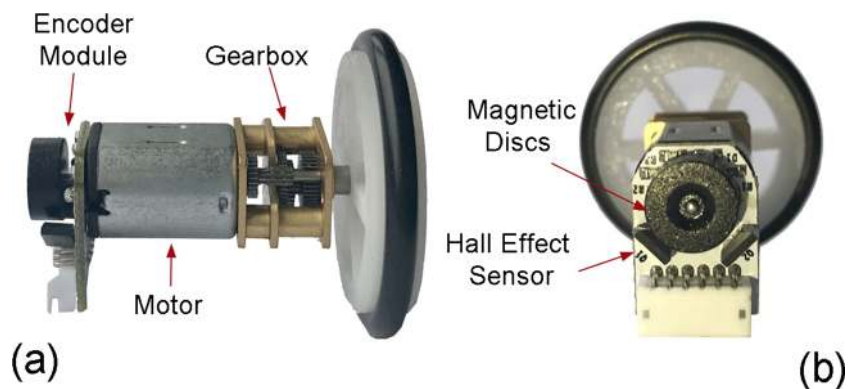
$$s(x, \theta) = \frac{\gamma \cos \theta}{x^2} + \kappa, \quad (3)$$

where $s(x, \theta)$ is the output value of sensor, x is the distance of obstacle, and θ is the angle of incidence with the surface. The model variable γ includes several parameters such as the reflectivity coefficient, output power of the emitted IR light and the sensitivity of sensor. κ is the offset value of the amplifier and ambient light effect. White body and black body surfaces reflect and absorb IR radiation with different ratios, which is a significant issue in selecting between an obstacle and a wall in robotic environments. The model parameters (γ and κ) were estimated empirically.

Mona translates the received IR signal intensity to estimate the distance and bearing of the obstacle and neighbouring robots in the case of a swarm scenario. The distance is calculated from the amplitude of the received IR shown in (3). As Mona's IR sensors are placed symmetrically at a known angle, the relative angular position of the obstacle can be estimated using (4):

$$\phi = \text{atan} \left(\frac{\sum_{i=1}^5 \hat{s}_i \sin(\psi_i)}{\sum_{i=1}^5 \hat{s}_i \cos(\psi_i)} \right), \quad (4)$$

Fig. 6 Configuration of the utilised motor and its magnetic encoder module. **a** Side view and **b** back view



where ϕ is the estimated angular position, ψ_i is the angular distance between i th sensor and the ‘top’ of the robot (IR-1 in Fig. 4). \hat{s}_i , $i \in \{1, 2, 3, 4, 5\}$ is the translated IR intensity from sensor i .

Since the proximity sensors have a narrow viewing angle (approximately 45°), there are blind spots between the sensors near the edge of the robot. However, those proximity sensors were deployed to act as the bumper for collision detection with walls and other robots, therefore to provide an accurate range & bearing function for the Mona, it needs an extension module covering 360° around the robot with longer sensing radius (> 15 cm). This long-range sensing module is currently under development.

Mona’s battery voltage is also monitored using an ADC from the μC and a voltage divider including two resistors. This value helps the robot to keep the voltage of battery in a safe range (> 3 V) and also monitors the charging process. This is very important as the motors’ velocity is a function of the battery voltage as shown in the motor’s model (1). It’s also a key parameter in swarm scenarios which consider power management system e.g. perpetual swarm robotics [17].

3.5 Communication

As with the sensing systems, there are two sets of communication modules. The first is the on-board internal communications bus and the second is external communications add-on boards (such as Wi-Fi, RF or Bluetooth). Figure 7 illustrates specified communication terminals on Mona’s main board. Mona supports three serial communication methods including:

- USART: used to send and receive data via USB (FTDI module). It is mainly for the Arduino programming link, but it can also be used to communicate with other add-on boards such as a Raspberry Pi or a bio-inspired vision system [42].
- I²C: is a general purpose communication link which can be used by external modules which support I²C protocol e.g. temperature sensor.

- SPI: is mainly used for RF communication module (e.g. NRF24L01 2.4 GHz). It is also used for programming of μC using the ISP method.

The IR proximity sensors can be used for face-to-face inter-robot communication in case of multi-robot scenario [43], however they do not provide high quality or fast communication due to the distance limitation of the utilised modules (approximately 5 ± 1 cm). Therefore, Mona requires an external module to provide inter-robot communication, distance estimation, and bearing covering 360° around the robot, similar to the module deployed by the e-puck robot [44].

3.6 Robot Modularity

To make Mona a low-cost versatile platform, it must be flexible enough to support other modules which have been developed for research and education. As illustrated in Fig. 7, the J2-SPI connector was considered to be a general purpose SPI connector, however the main reason was to attach a NRF24L01 module which is a widely used low-cost RF module with Arduino boards.

Due to the limited processing power of the main μC , Mona has been designed to interface with other more powerful processing modules such as Raspberry Pi Zero² to extend its applications.

A breakout board (Fig. 3b) has been developed which supports: i) Raspberry Pi Zero, ii) XBee module, and iii) NRF24L01. The board is mounted on top of the main platform and is able to communicate with the main platform using the internal communications buses.

To study the possibility of controlling Mona using ROS commands, a breakout board has been made that supports a Teensy 3.2 module.³ Additionally a WiFi module (WINC1500⁴) was attached on top of Mona and

²www.raspberrypi.org

³www.pjrc.com

⁴www.microchip.com

It is worth mentioning that the robot can be programmed with any software platform that supports ISP, hence it is not only limited to Arduino.

4.2 Robot Programming

Mona has been developed based on an AVR RISC microcontroller (ATmega328P). The architecture of the robot allows its connection to Arduino-based platforms via a USB cable. However, it is possible to use any programming language which was developed for AVR μ Cs including C, C++, Java, Pascal, Basic, and Assembly.

Due to the Arduino platform's popularity amongst undergraduate and graduate students and the open-source nature of the Arduino project, lab activities were therefore conducted with C programming. Students are given an Arduino Sketch file which can contain as much or as little of the underpinning code as is required (Arduino sketch is available at: <https://github.com/MonaRobot/Mona-Arduino>).

For example, the sketch could contain all of the functions to access the peripherals and actuation systems and the students only have to write the main loop, calling these pre-written functions. Alternatively students could be asked to write the functions themselves.

The purpose behind using the sketch was to give students the experience of programming the robots at a relatively low level, in the manner used by the researchers. The aim is also to use readily available and supported interfaces. All of the underlying code will be accessible on Git-Hub. A custom GUI is being considered, however there are challenges with regards long-term support of such an interface should new versions of the robot be developed.

The use of Arduino sketches has been successfully implemented in a number of lab sessions in the MRAS course using not only Mona, but also just an Arduino with an IMU.

4.3 Simulation Software

Mona has been modelled in Stage [47] for the study of bio-inspired aggregation scenarios [48]. Stage is an open-source software platform that simulates a group of mobile robots in a 2D environment.

5 Experiments

The robot was evaluated for its suitability for teaching and research in two phases. The first phase considered the core functionality of the robot with regards to the characterisation of the motors and sensors. The second phase evaluated the suitability for a range of motion planning algorithms, including traditional open- and closed-loop controls and modern swarm control techniques.

5.1 System Characterisation

The quantitative experiments were performed to evaluate the performance of the hardware design and focused on two areas:

- An analysis of the actuation method and the characterisation of the utilised motors by extracting the macroscopic model's (1) parameters (α_m and β_m).
- An evaluation of the sensor system for obstacle detection and the estimation of the model's (3) parameters (α and β). Additionally, the accuracy of the relative angular position presented by Eq. 4 was evaluated.

These experiments characterise the performance of the robot and can also act as an excellent introductory lab for students. This could be either as stand-alone exercises on actuation or sensing, or as part of a robotics systems performance lab.

5.2 Motion Planning

In this set of experiments, a constant speed for both left and right motors was selected. The error, $e(t)$, is the differences between the current ($PV_{l,r}$) and desired (set point, $SP_{l,r}$) rotational speed of the motors. The error term is caused by two aspects of the design: i) the motors heterogeneity and ii) physical asymmetry.

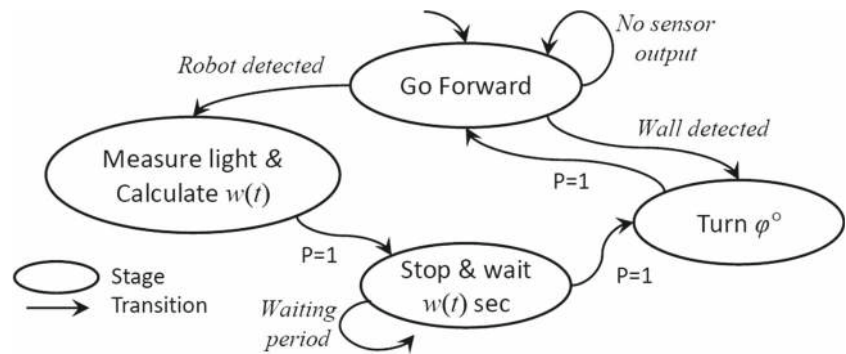
Experiments were conducted in for open- and closed-loop control at different motors' speeds of $N_{l,r} \in \{60, 70, 80\}$ rpm. In the case of open-loop control, the left and right motors' set points were assigned to the PWM pulse generators and the error was observed using the feedback from wheels' encoders, $PV_{l,r}$.

In the closed-loop control experiment, a proportional controller was selected as the robot does not have rapid changes in its trajectory. The goal was therefore to identify, compare and compensate the motion error due to motors' manufacturing heterogeneity and physical error parameter at different forward speeds of $N_{l,r} \in \{60, 70, 80\}$ rpm. The proportional controller calculates a proportional output, P_{out} , over a period of δ_t using the following equation:

$$P_{out} = K_p e(t) + P_0 \quad (5)$$

where $K_p \in \{0.05, 0.15, 0.25\}$ is the proportional gain, $e(t)$ is the error of the system, $e(t) = SP - PV$, over period of $\delta_t = 200$ ms and P_0 is the output of controller when there is no error, which is the PWM value. It is worth mentioning that the Eq. 5 controls the left and right motors independently using the separate feedback values from the motor encoders. The preliminary experiments showed that the effect of different time frames (δ_t) was

Fig. 8 Controller of the performed swarm robotic scenario at MRAS Lab



not significant. However, increase in the timer size may increase fluctuations over time.

5.3 MRAS Lab Activity: Swarm Control

One of the benefits of using the Mona robots for teaching and research is that multi-agent systems can be investigated. The work presented in [17] showed how their capabilities for research, however it is important that this functionality can be translated into the teaching domain.

A lab exercise was developed based on the bio-inspired BEECLUST aggregation algorithm [49] and experimental setup presented in [48]. The BEECLUST, a state-of-the-art swarm aggregation algorithm, was chosen due to its simplicity in implementation and programming. To perform the swarm aggregation, all the robots follow a similar control mechanism which is shown in Fig. 8. A light source (cue) was provided by the desk lamp as a gradient cue that placed in one side of the arena. The robots followed the aggregation mechanism to find the optimal part of the arena (local optimum).

In the lab session, groups of students were given an individual Mona and required to programme it to search for a light source. They had to do some limited robot characterisation (ambient light intensity and open-loop motor characterisation) before running a basic motion planning algorithm. An Arduino sketch was provided with the basic interface functions pre-written.

Once the students completed the individual robot exercise, they had to modify their code to implement

the BEECLUST algorithm. Multiple groups were then combined so that five robots were placed in an arena at the same time. The behaviour of the robots was then recorded.

The deliverables expected from each group were: i) adjust the waiting time and find appropriate values for the parameters, ii) record the aggregation time when 4 robots are aggregated within the cue zone, iii) repeat experiments 10 times and calculate median and standard error rate, iv) submit a short video showing an aggregation of 5 robots, v) report any issues and unexpected outcomes.

6 Results

6.1 Motor Characterisation

The model's (1) parameters were extracted from an experiment using an adjustable power supply. Figure 9a shows the rotational speed, N , of the motors as a function of its motors' voltage. Motors showed a linear relationship as voltages varied from 0.5 to 5 V. Results on 40 motors showed that all motors followed the macroscopic model with similar $\alpha_m = 28.07 \pm 2$ and $\beta_m = -6.2 \pm 0.5$.

In the second step, the reactions of the robot's motors in different PWM and motion directions were evaluated. Figure 9b shows a linear relationship between speed of motors and the PWM values in both forward and reverse directions.

Fig. 9 **a** Motors' rotational speed as a function of voltage, and model parameters (α_m and β_m , in Eq. 1). **b** Speed of the robot forward and backward as a function of PWM value

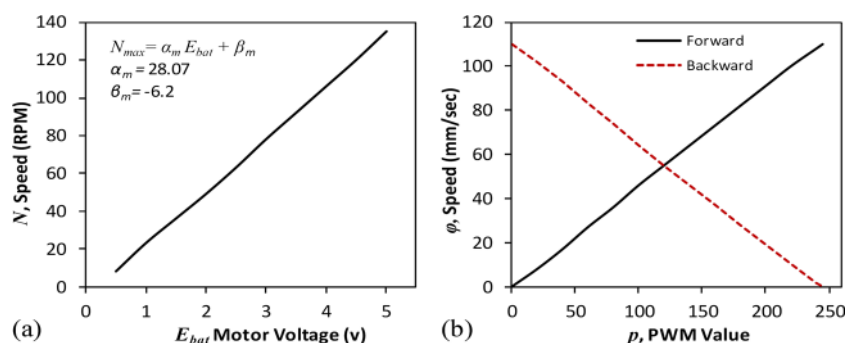
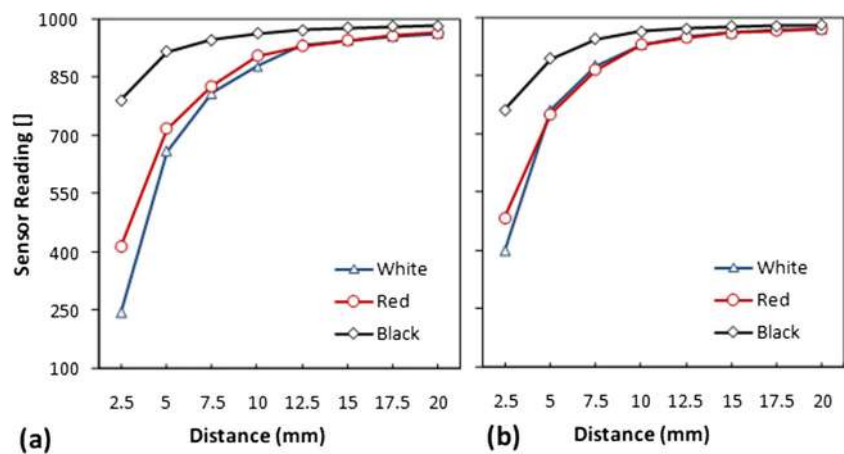


Fig. 10 Sensor readings from IR short-range proximity sensor reflected from **a** 20 mm wide and **b** 10 mm wide obstacles with different colours (white, red, and black) which was faced towards the sensors. The lower sensor reading shows a higher amount of IR



6.2 Sensor Characterisation

Figure 10 illustrates the recorded values from the IR proximity sensors which were reflected from two different sizes of obstacles with 10 mm and 20 mm widths. The effects of obstacles' colours have also been studied.

The results showed that an increase in size of the obstacle improves the distance estimation function by increasing the amount of reflected IR from the surface. However, this reflection was not similar for experiments with different colours. As shown in the results, the white colour obstacle reflected higher amount of IR than the darker obstacles.

Table 3 shows the extracted model parameter (γ and κ , Eq. 3) for the individual IR sensors. As discussed previously, γ relies on several parameters including the reflectivity coefficient for the obstacle, output power of the emitters, and sensitivity of the receiver. The results revealed that the obstacles' widths and colours directly impacted the γ value of the model (3) in every configuration. Similar results have been reported in [41]; white white colour obstacle had the highest reflection and black colour obstacle had the lowest reflectivity.

On the other hand, κ , which that relies on the emitters' power and ambient light, was not affected by the different experimental configurations. Since the experiments were conducted at the same time under similar conditions (lighting and IR emitter power source), the results had approximately similar κ value of 3.

In the next step, we tested the neighbouring robots detection which is an important issue in collaborative robotic systems. The neighbour robot was placed at three different angles $\{-45^\circ, 0^\circ, 45^\circ\}$ from the centre of the observed robot and the model (4) in different distances (5.6 to 11.3 cm) from the centre of the observed robot was tested (Fig. 11). Since the utilised sensors are very short range and narrow angle, the readings for longer distances did not fit the bearing model with a high precision. Therefore, to

increase the bearing accuracy using a low-cost solution, weights (gain) for each sensor were added to the Eq. 4:

$$\phi = \text{atan} \left(\frac{\sum_{i=1}^5 \eta_i \hat{s}_i \sin(\psi_i)}{\sum_{i=1}^5 \eta_i \hat{s}_i \cos(\psi_i)} \right), \quad (6)$$

where η_i is the gain value for the i th sensor which are extracted using model fitting on the recorded values from the preliminary experiments. Therefore, the extracted gain value for sensors (Left, Front-Left, Front, Front-Right, and Right) are $\eta \in \{2.01, 0.95, 0.65, 0.65, 0.75, 2.20\}$, respectively. The values show that, the side sensors need to have higher gain which was expected due to the narrow viewing angle of the sensors.

6.3 Motion Planning

Figure 12 shows the median of the observed results of motion control experiments with open- and closed-loop controllers. Samples were collected every 100 ms during 100 sec of the experiment at different motors' rotational speeds, $N_{l,r} \in \{30, 60, 90\}$ rpm. Each set of experiments were repeated 10 times for 5 different Mona robots.

The results showed that an increase in speed of the motors results in increasing the precision of the motors rotational speed in open-loop control scenario by reducing the error rate from 4.4% (at 60 rpm) to 2.2% (at 80 rpm).

Table 3 Model parameters for IR sensor in different configurations

Obstacle's width	Parameter	Obstacle colour		
		White	Red	Black
1 cm	γ	11.7	9.9	3.8
	κ	3.10	3.07	3.15
2 cm	γ	14.5	11.1	4.5
	κ	3.06	3.02	3.15

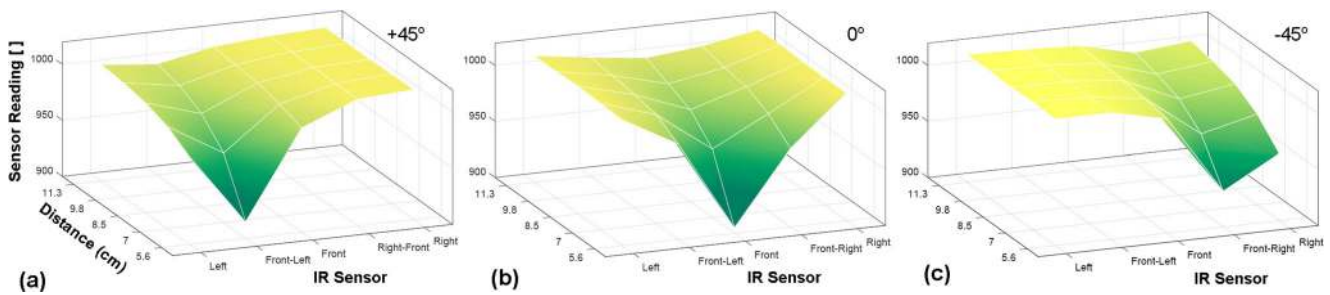


Fig. 11 Independent readings from received IR that emitted from another robot in different distances (5.6 to 11.3 mm) at angular location of **a** $+45^\circ$, **b** 0° , and **c** -45°

This was because of the nominal voltage of the motors which is 6 V, however the maximum voltage that a battery can provide is 4.2 V.

The proportional control that was used to compensate motion error demonstrated a significant improvement in motion planning. The results revealed that an increase in proportional gain, K_p , increases the performance of the system in all configurations. However, it contained fluctuation in error rate with lower speed (60 rpm) than with faster rotational speed.

Therefore, with $\delta t = 100$ ms, the proportional gain of $K_p = 0.25$ demonstrated precise error compensation with median error rate of $\{1.15\%, 0.94\%, 0.8\%\}$ at $N \in \{60, 70, 80\}$ rpm, respectively. We must mention that the closed-loop motion control could be more optimised by using a different time frame or applying a PI or a PID controller, which is beyond the scope of this paper.

6.4 MRAS Lab Activity: Swarm Control

The MRAS Swarm Control lab session was run for 3 h with 30 students. The first phase of the lab focused on characterising Mona's functions and hands-on programming and was successfully completed during the allocated 45 min.

The groups were able to measure and draw the light map of the arena and the observed results showed that the light

intensity reading was between 800 to 1000 (dimensionless, 10 bits ADC). Therefore, ambient light (assumed 800) was subtracted from the future calculations.

In the second phase of the lab, groups were expected to run the BEECLUST algorithm. The students were able to understand that the ambient light (the offset) must be subtracted from the light reading to avoid unexpectedly long waiting times. As expected, the groups observed variance in their aggregation times, primarily due to their arena position and light interference caused by the room lighting. The median of aggregation time was 200 s with ± 50 s tolerance, which is normal for experiments with low population size as reported in [37, 50]. A future extension of the lab could be to run the scenario with different population sizes to observe the effect on the aggregation time.

It is worth noting that there were two issues during the experiments. The first was that the light sources (desk lamps) contained some IR, despite using cold white colour lamps. This led to the robots performing unexpected turns due to received IR from light that was mistaken as an obstacle.

The second issue which was that several collisions happened when a robot reached the other robot from back side. In this case, the robot couldn't receive any reflected IR using its front proximity sensors. To solve the issue, future robots will have a white colour case that reflect the IR, hence it is considered as an obstacle.

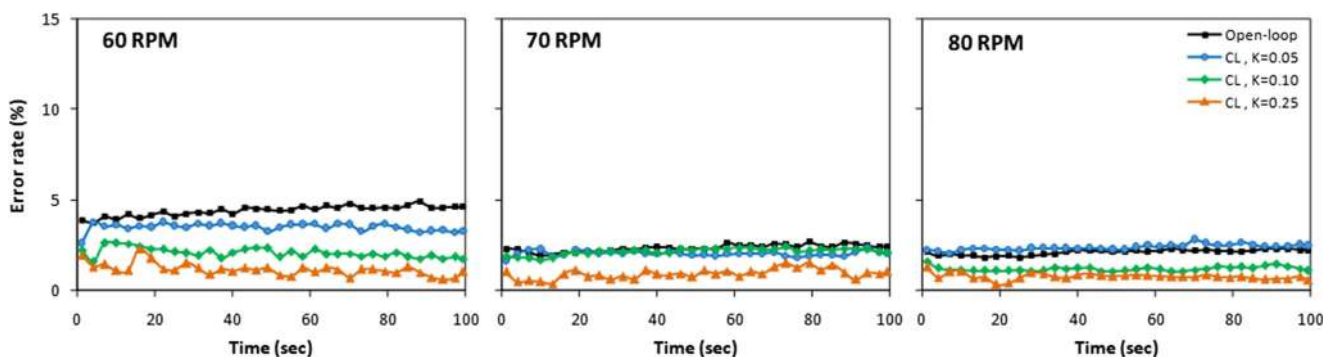


Fig. 12 Median of error rate of the robot's motion during 100 s forward motion using open-loop and closed-loop (CL) controls in different wheels' rotational speeds ($N_{l,r} \in \{60, 70, 80\}$ rpm.) with three proportional gains of $K_p \in \{0.05, 0.15, 0.25\}$

7 Conclusion

This paper has presented an open-source and low-cost mobile robot platform that has been developed for education and research purposes. An affordable mobile robot, Mona, can be programmed using off-the-shelf open-source programming platforms (e.g. Arduino). The robot's hardware functionality including: sensory systems, actuation, communication, and power management has been assessed and characterised. Control systems for single and swarm robotic scenarios have been implemented to show the abilities for both teaching and research. Mona is currently being used for two different tasks: i) teaching resource for autonomous system and robotics as an educational platform and ii) study on long-term evolutionary algorithms as a platform for research.

The next stage of development for the Mona robot is the design of a series of add-on modules which will allow additional functionality. As shown in the [Appendix](#), a number of these modules are already being developed, including one which allows the robots to be controlled using ROS. It is hoped that this unique capability (for a robot of this size) will make the robots more attractive for both teaching and research. The open-source repository of hardware and software will be improved as well as all the systems being available commercially.

From a research perspective, the Monas are currently being used to explore fault tolerant control of multi-agent systems, pheromone communication based swarm behaviour and human-robot-interactions using mixed-reality interfaces.

Acknowledgements This work was supported by the EPSRC (Project No. EP/P01366X/1 and EP/P018505/1), Innovate UK (Project No. KTP 009811), CONACyT and the National Nuclear Laboratory.

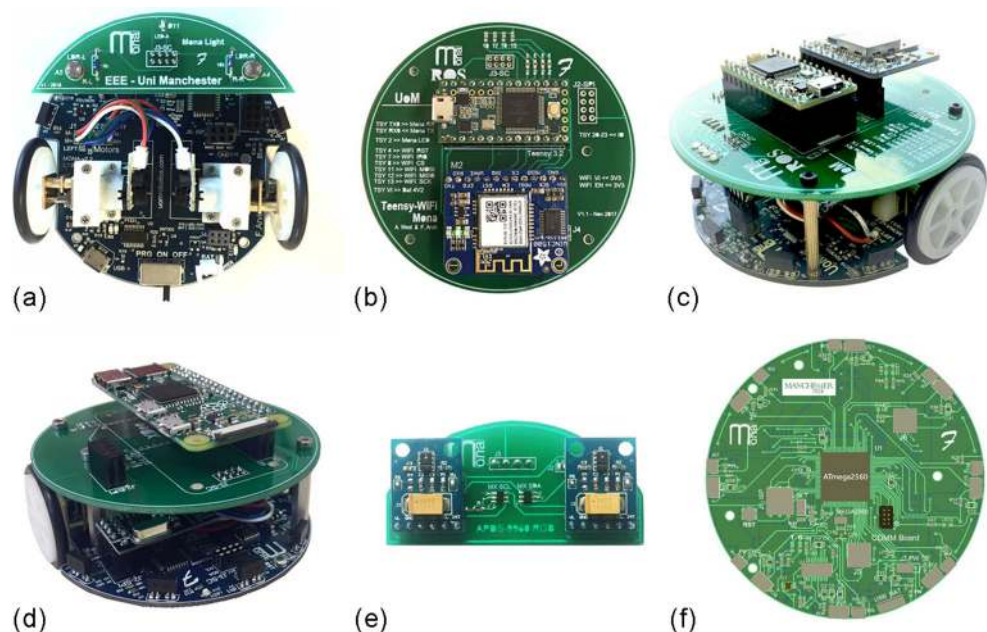
Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix

There are several modules which have been developed as the extension modules for Mona. They expand Mona's applications in research studies. Figure 13 shows the Mona robots' modules.

Figure 13a shows Mona that was equipped with a light sensing module. This module was used in MRAS lab activity and it was used for study on bio-inspired swarm aggregation scenario preseted in [48]. The second module shown in Fig. 13b is ROS communication board [45]. The module has been developed to study the feasibility of using ROS as the communication protocol for Mona, Fig. 13c. The ROS module contains a Teensy 3.2 board, a WiFi module and 4 LEDs. The next module shown in Fig. 13d which is a breakout board has been developed to connects a Raspberry Pi-0 board and Xbee module to the Mona robot. The plan was to use the Raspberry Pi board to add an image processing module and other functions which requires a fast and strong processing unit. Figure 13e shows

Fig. 13 Mona's extension modules: **a** light module including two LDRs, **b** Mona's ROS communication module, **c** Mona is equipped with a ROS module, **d** Raspberry Pi Zero module, **e** colour sensing and ambient light intensity module, and **f** design of inter-robot communication module



the module which was developed to read RGB colours with Mona robot. The module communicate using I2C ports. It has two APDS-9960 RGB and Gesture sensors, which was developed for use with Arduino boards. The next module which is shown in Fig. 13f is a communication module which has been developed for inter-robot short-range communication.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Merdan, M., Lepuschitz, W., Koppensteiner, G., Balogh, R. (eds.): *Robotics in Education : Research and Practices for Robotics STEM Education*. Springer, Berlin (2017)
2. Jojoa, E.M.J., Bravo, E.C., Cortes, E.B.B.: Tool for experimenting with concepts of mobile robotics as applied to children's education. *IEEE Trans. Educ.* **53**(1), 88–95 (2010)
3. Chaudhary, V., Agrawal, V., Sureka, P., Sureka, A.: An experience report on teaching programming and computational thinking to elementary level children using lego robotics education kit. In: *IEEE Eighth International Conference on Technology for Education*, pp. 38–41 (2016)
4. Scott, M.J., Counsell, S., Lauria, S., Swift, S., Tucker, A., Shepperd, M., Ghinea, G.: Enhancing practice and achievement in introductory programming with a robot olympics. *IEEE Trans. Educ.* **58**(4), 249–254 (2015)
5. Wang, D., Chen, J., Liu, L.: Discussion of robot application laboratory construction. *International Journal of Education and Learning* **5**(1), 1–12 (2016)
6. Conti, D., Nuovo, S.D., Buono, S., Nuovo, A.D.: Robots in education and care of children with developmental disabilities: a study on acceptance by experienced and future professionals. *Int. J. Soc. Robot.* **9**(1), 51–62 (2016)
7. Scilliano, B., Khatib, O.: *Springer Handbook of Robotics*. Springer, Berlin (2008)
8. Sadanand, R., Joshi, R.P., Chittawadigi, R.G., Saha, S.K.: Virtual experiments for integrated teaching and learning of robot mechanics using RoboAnalyzer. In: *CAD/CAM, Robotics and Factories of the Future*, pp. 59–68. Springer (2016)
9. Calvo, I., Cabanes, I., Quesada, J., Barambones, O.: A multi-disciplinary pbl approach for teaching industrial informatics and robotics in engineering. *IEEE Trans. Educ.* **61**(1), 21–28 (2018)
10. Felder, R.M., Spurlin, J.: Applications, reliability and validity of the index of learning styles. *Int. J. Eng. Educ.* **21**(1), 103–112 (2005)
11. Felder, R.M., Soloman, B.A., et al.: Learning styles and strategies. At <http://www.engr.ncsu.edu/learningstyles/ilsweb.html> (2000)
12. Rivera, J.H.: Science-based laboratory comprehension: an examination of effective practices within traditional, online and blended learning environments. *Open Learning: The Journal of Open, Distance and e-Learning* **31**(3), 209–218 (2016)
13. Nguyen, K.A., DeMonbrun, R.M., Borrego, M.J., Prince, M.J., Husman, J., Finelli, C.J., Shekhar, P., Henderson, C., Waters, C.: The variation of nontraditional teaching methods across 17 undergraduate engineering classrooms. In: *2017 ASEE Annual Conference & Exposition* (2017)
14. Spolaôr, N., Benitti, F.B.: Robotics applications grounded in learning theories on tertiary education: a systematic review. *Comput. Educ.* **112**, 97–107 (2017)
15. Cielniak, G., Bellotto, N., Duckett, T.: Integrating mobile robotics and vision with undergraduate computer science. *IEEE Trans. Educ.* **56**(1), 48–53 (2013)
16. Ortiz, O.O., Franco, J.A.P., Garau, P.M.A., Martin, R.H.: Innovative mobile robot method: improving the learning of programming languages in engineering degrees. *IEEE Trans. Educ.* **60**(2), 143–148 (2017)
17. Arvin, F., Watson, S., Turgut, A.E., Espinosa, J., Krajník, T., Lennox, B.: Perpetual robot swarm: long-term autonomy of mobile robots using on-the-fly inductive charging. *J. Intell. Robot. Syst.* (2017). <https://doi.org/10.1007/s10846-017-0673-8>
18. Matthews, D., Dang, S., Christodoulou, L., Chen, H., Hawit, Y.: Embedded systems project: innovative autonomous line-following buggy design and implementation. In: *Annual International Conference on Emerging Research Areas: Magnetism, Machines and Drives*, pp. 1–5 (2014)
19. Surendran, A., Mija, S.J.: Sliding mode controller for robust trajectory tracking using haptic robot. In: *IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems*, pp. 1–6 (2016)
20. Michieletto, S., Tosello, E., Pagello, E., Menegatti, E.: Teaching humanoid robotics by means of human teleoperation through rgb-d sensors. *Robot. Auton. Syst.* **75**, 671–678 (2016)
21. Browne, A.F., Conrad, J.M.: A versatile approach for teaching autonomous robot control to multi-disciplinary undergraduate and graduate students. In: *IEEE Access* (2017)
22. Kofinas, N., Orfanoudakis, E., Lagoudakis, M.G.: Complete analytical forward and inverse kinematics for the nao humanoid robot. *J. Intell. Robot. Syst.* **77**(2), 251–264 (2015)
23. Mondada, F., Bonani, M., Raemy, X., Pugh, J., Cianci, C., Klapotcz, A., Magnenat, S., Zufferey, J.C., Floreano, D., Martinoli, A.: The e-puck, a robot designed for education in engineering. In: *Proceedings of the 9th Conference on Autonomous Robot Systems and Competition*, vol. 1, pp. 59–65 (2009)
24. Riedo, F., Chevalier, M., Magnenat, S., Mondada, F.: Thymio II, a robot that grows wiser with children. In: *IEEE Workshop on Advanced Robotics and its Social Impacts*, pp. 187–193 (2013)
25. Gyebe, E., Hanheide, M., Cielniak, G., et al.: Affordable mobile robotic platforms for teaching computer science at African Universities. In: *6th International Conference on Robotics in Education* (2015)
26. Afari, E., Khine, M.S.: Robotics as an educational tool: impact of lego mindstorms. *International Journal of Information and Education Technology* **7**(6), 437–442 (2017)
27. Álvarez, A., Larrañaga, M.: Experiences incorporating lego mindstorms robots in the basic programming syllabus: lessons learned. *J. Intell. Robot. Syst.* **81**(1), 117–129 (2016)
28. Alkilabi, M.H.M., Narayan, A., Tuci, E.: Cooperative object transport with a swarm of e-puck robots: robustness and scalability of evolved collective strategies. *Swarm Intell.* **11**, 185–209 (2017)
29. Chovanec, M., Čechovič, L., Mandák, L.: AERIS—Robots Laboratory with Dynamic Environment. In: *Robotics in Education*, pp. 169–180. Springer International Publishing (2017)
30. López-Rodríguez, F.M., Cuesta, F.: Andruino-a1: low-cost educational mobile robot based on android and arduino. *J. Intell. Robot. Syst.* **81**(1), 63–76 (2016)
31. Arvin, F., Murray, J., Zhang, C., Yue, S.: Colias: an autonomous micro robot for swarm robotic applications. *Int. J. Adv. Robot. Syst.* **11**(113), 1–10 (2014)
32. Szymanski, M., Breitling, T., Seyfried, J., Wörn, H.: Distributed shortest-path finding by a micro-robot swarm. In: *International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 404–411. Springer, Berlin (2006)
33. Soares, J.M., Navarro, I., Martinoli, A.: The Khepera IV mobile robot: performance evaluation, sensory data and software toolbox. In: *Robot 2015: Second Iberian Robotics Conference*, pp. 767–781. Springer International Publishing, Cham (2016)

34. Bonani, M., Longchamp, V., Magnenat, S., Rétornaz, P., Burnier, D., Roulet, G., Vaussard, F., Bleuler, H., Mondada, F.: The marxbot, a miniature mobile robot opening new perspectives for the collective-robotic research. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4187–4193 (2010)
35. Yu, J., Han, S.D., Tang, W.N., Rus, D.: A portable, 3d-printing enabled multi-vehicle platform for robotics research and education. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1475–1480 (2017)
36. McLurkin, J., Rykowski, J., John, M., Kaseman, Q., Lynch, A.J.: Using multi-robot systems for engineering education: teaching and outreach with large numbers of an advanced, low-cost robot. *IEEE Trans. Educ.* **56**(1), 24–33 (2013)
37. Arvin, F., Turgut, A.E., Krajník, T., Yue, S.: Investigation of cue-based aggregation in static and dynamic environments with a mobile robot swarm. *Adapt. Behav.* **24**(2), 102–118 (2016)
38. Arvin, F., Bekravi, M.: Encoderless position estimation and error correction techniques for miniature mobile robots. *Turk. J. Electr. Eng. Comput. Sci.* **21**, 1631–1645 (2013)
39. Arvin, F., Murray, J.C., Shi, L., Zhang, C., Yue, S.: Development of an autonomous micro robot for swarm robotics. In: IEEE International Conference on Mechatronics and Automation, pp. 635–640 (2014)
40. Arvin, F., Samsudin, K., Ramli, A.R.: Development of a miniature robot for swarm robotic application. *International Journal of Computer and Electrical Engineering* **1**, 436–442 (2009)
41. Benet, G., Blanes, F., Simó, J.E., Pérez, P.: Using infrared sensors for distance measurement in mobile robots. *Robot. Auton. Syst.* **40**(4), 255–266 (2002)
42. Hu, C., Arvin, F., Xiong, C., Yue, S.: A bio-inspired embedded vision system for autonomous micro-robots: the LGMD case. *IEEE Transactions on Cognitive and Developmental Systems* **9**(3), 241–254 (2016)
43. Arvin, F., Samsudin, K., Ramli, A.R.: Development of IR-based short-range communication techniques for swarm robot applications. *Advances in Electrical and Computer Engineering* **10**(4), 61–68 (2010)
44. Gutiérrez, A., Campo, A., Dorigo, M., Amor, D., Magdalena, L., Monasterio-Huelin, F.: An open localization and local communication embodied sensor. *Sensors* **8**(11), 7545–7563 (2008)
45. West, A., Arvin, F., Martin, H., Watson, S., Lennox, B.: ROS integration for miniature mobile robots. In: *Towards Autonomous Robotic Systems (TAROS)* (2018)
46. Banzi, M., Shiloh, M.: *Getting Started with Arduino: the Open Source Electronics Prototyping Platform*. Maker Media, Inc, San Francisco (2014)
47. Vaughan, R.: Massively multi-robot simulation in stage. *Swarm Intell.* **2**(2), 189–208 (2008)
48. Ramroop, S., Arvin, F., Watson, S., Carrasco-Gomez, J., Lennox, B.: A bio-inspired aggregation with robot swarm using real and simulated mobile robots. In: *Towards Autonomous Robotic Systems (TAROS)* (2018)
49. Schmickl, T., Thenius, R., Moeslinger, C., Radspieler, G., Kernbach, S., Szymanski, M., Crailsheim, K.: Get in touch: cooperative decision making based on robot-to-robot collisions. *Auton. Agent. Multi-Agent Syst.* **18**(1), 133–155 (2009)
50. Arvin, F., Turgut, A.E., Bazyari, F., Arian, K.B., Bellotto, N., Yue, S.: Cue-based aggregation with a mobile robot swarm: a novel fuzzy-based method. *Adapt. Behav.* **22**(3), 189–206 (2014)

Farshad Arvin is a Post Doctoral Research Associate at School of Electrical and Electronic Engineering at the University of Manchester, UK (since July 2015). He received his BSc degree in Computer Engineering in 2004, MSc degree in Computer Systems Engineering in 2010, and PhD in Computer Science in 2015. He was a Research Assistant at the Computational Intelligence Laboratory (CIL) at the University of Lincoln, UK. He was awarded a Marie Curie Fellowship to be involved in the FP7-EYE2E and LIVCODE EU projects during his PhD study. Farshad visited several leading institutes including the Institute of Microelectronics at Tsinghua University in Beijing and recently, he visited the Italian Institute of Technology (iit) in Genoa as the visiting research scholar. His research interests include swarm robotics, autonomous systems, and collective robotics. He has developed several swarm robotic platforms including Colias and Mona. Farshad is a Chartered Engineer (CEng) and member of IEEE and MIET.

Jose Espinosa is a Mechatronic Engineer from the National and Autonomous University of Mexico (UNAM, Mexico City). Currently, he is a PhD student at the University of Manchester Robotics Group. His research interests are Autonomous exploration systems, underwater localisation of robotics platforms and swarm robotics.

Benjamin Bird is a PhD researcher in the department of EEE, University of Manchester. He received his Master of Philosophy in Robotics in 2015, and his Bachelors of Electronic and Electrical Engineering in 2014. His research interests include but are not limited to; Robotics, Computer Vision, Embedded Systems, Embedded Systems for Aerospace and Commercial Space Flight.

Andrew West is a Post Doctoral Research Associate at the School of Electrical and Electronic Engineering at The University of Manchester. His current research is focused on the robotic deployment of scientific instruments for inspection of hazardous environments, such as those found in the nuclear industry. He has expertise in instrument design and data analysis, with a background in experimental plasma physics.

Simon Watson is a Lecturer in Robotic Systems at the School of Electrical and Electronic Engineering at the University of Manchester. He obtained his MEng in Mechatronic Engineering in 2008 and his PhD in 2012, both from the University of Manchester. His research focus is on mobile robots for the exploration and characterisation of hazardous and extreme environments and active areas of research include novel platform design, communications and localisation, sensing and navigation and multi-level control. His current research portfolio includes developing robots for the nuclear industry (for the Sellafield and Fukushima sites) and power generation (offshore wind).

Barry Lennox is Professor of Applied Control and Nuclear Engineering Decommissioning in the School of Electrical and Electronic Engineering at The University of Manchester and is the Research Director of the Dalton Cumbrian Facility. He is an expert in applied control systems and their use in process operations and robotics and has considerable experience in transferring leading edge technology into industry.