RoboCup Rescue RMRC 2019 Team Description Paper r.2 RoboBlues

Info
Team Name: RoboBlues
Team Institution: The Hill School
Country: USA
Team Members:
Eric Cardoza
Priyanka Dondeti
Amanda Grippo
Max Moore (will not attend)
Michael Schiavone
Tim Youndt
Team Leader: Timothy E. Jump
Email: tjump@thehill.org
Team URL: TeamRoboBlues

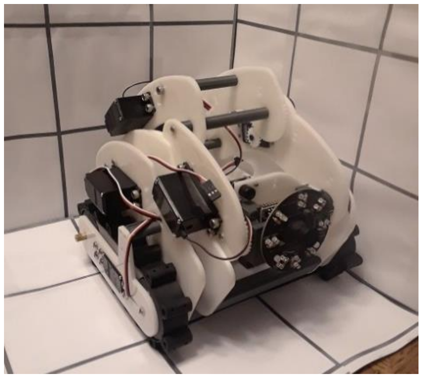


Fig. 1. Working iteration of RoboBlues RMRC Robot (as of June 6, 2019). Lines are spaced 10cm.

Abstract— Team RoboBlues is the enrichment component of The Hill School's Engineering program (Hill Engineering). The goal of Hill Engineering is to give students conceptual understanding in four engineering foundations (mechanical, electrical/electro-mechanical, design and control). Students that show extreme promise during their classroom development of small scale rescue robots are invited to be members of team RoboBlues. Ultimately, our participation at RoboCup is an opportunity to demonstrate and integrate advanced engineering concepts and learning, both to and from Hill Engineering students, and to participate and expose students to a true research project preparing them to be competitive applicants for research positions as they enter their university studies. We also hope to produce functional products that can be implemented in higher level research projects and be commercially viable.

Index Terms—RoboCup Rescue RMRC, Team Description Paper, Small Form-Factor.

I. Introduction

HILL ENGINEERING follows the Engineering[3] [1] curriculum which reflects the DHS/NIST/ASTM [2] test methods for responder robots. In the beginning levels of Hill Engineering the focus is on developing small form factor, advanced mobility, intuitively controlled (teleoperated) mechanical transport systems. In the advanced levels of Hill Engineering the focus shifts to sensor integration, data acquisition and interpretation, and autonomous control. In all levels of Hill Engineering, students work in teams of two to explore various robot design iterations with the best platform(s) selected for entry into RoboCup Rescue RMRC. In general, the RMRC entry will be a single robot incorporating significant elements from several competing designs. In Hill Engineering we use LEGO to prototype ideas, then integrate laser cut and 3D printed parts across multiple design iterations for our advanced robot platforms. When appropriate, we have parts machined at a local shop. The electrical/electro-mechanical components utilized are non-LEGO (with the exception of LEGO M-motors). There are also various servo motors available, and the control components are those available through Engineering[3]. Our inaugural entry into RMRC focuses on delivery of a small form factor robot (< 25cm) with wireless, teleoperative control that can traverse the RMRC terrain elements; employ a manipulator to perform touch and rotation tasks; and machine vision functions to track motion, identify color and detect QR codes.

II. System Description - Robot

A. Hardware

1) Drive System and Chassis

We incorporate a tread drive system on the RoboBlues robot, with flippers used to lift the front or back of the robot to allow traversing of terrain elements. We use Parallax continuous rotation servos to drive the treads, and to control flipper position. The chassis is an assembly of laser cut Acetyl plates and 3D printed parts in ABS Plus.

2) Control

Control of the RoboBlues robot is managed through the Enginbeering[3] RCM (Robot Control Module) Control Stack which is an assembly of five different PCBs:

- RCM-Cam (camera module)
- RCM-Comm (radio module)
- RCM-BFin (CPU)

- RCMx1 (control interface)
- RCM-PBB (power buss)

Fig. 2. E3 RCM-Control Stack.

3) Power

We use an 8-AA cell, 2000mAh NiMH rechargeable battery pack onboard the robot. Full charge output is 9.6V.

Fig. 3.E3 Battery Pack

4) Sensors

Teleoperative navigation is managed through visual data streamed through the OmniVision OV7725 camera and our MainController vision interface. Although not a part of our RoboCup 2019 entry, we are investigating autonomous image processing options that run as a background service and aid in obstacle avoidance and path planning.

Fig. 4. E3 RCM-Cam with OmniVision OV7725 camera.

We are also investigating autonomous navigation through the inclusion of a perimeter detection system that will direct the robot around obstacles, through doorways, paths, etc. We are experimenting with Sharp digital and analog IR sensors and short range Maxbotix ultrasonic sensors as detection devices.

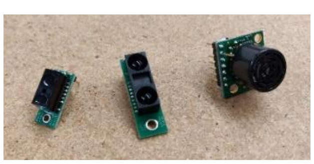


Fig. 5. Sharp digital IR (left), Sharp analog IR (center), Maxbotix Ultrasonic (right).

Integrating the image data through the camera and the perimeter detection data we hope to deploy advanced navigation and localization routines in 2020 that manage the skewed data from sensors when the robot is traveling across the undulating terrain elements in the test suites.

We do have an Adafruit 9-DOF IMU that gives us orientation data, and hope to use this to aid in our autonomous navigation process.

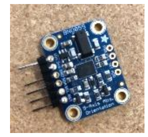


Fig. 6. Adafruit BNO055 9-DOF IMU.

A final element to aid in navigation is the inclusion of a transponder function that detects whether or not the robot is communicating with the command computer. If communication is broken there is an "emergency stop" code embedded in the robot firmware that will immediately set all outputs to "off". Next step in our development is to have the robot autonomously navigate/retrace its path until communication is reestablished.

In addition to using the camera for navigation, we can also detect motion, color (blob detection), along with identification of visual information from QR tags (still pending).

We have no thermal imaging through our OV camera, but we have thermal detection through our E3 Thermal sensor that uses a Melexis MLX90614ESF-BCI-000-TU sensing element to read object temperature.

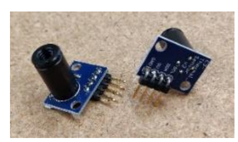


Fig. 7. E3 Thermal sensor with Melexis sensing element.

We have CO_2 detection utilizing a CO2Meter ExplorIR-W 5% sensor.

Fig. 8. CO2Meter ExplorIR-W 5% sensor.

5) Manipulator

Our inaugural manipulator has 5 DOF. There is camera tilt, rotator, rotator tilt, main arm tilt and rotator arm tilt. Panning motion is handled by rotating the full robot.

The camera is located on the main arm and on an alternate axis than the rotator arm. The rotator arm is hinged mid-main arm. Together, this should help with visual positioning of the rotator.

We have US Digital E4T miniature shaft encoders at every joint to report the angular position of each.

Fig. 9. US Digital E4T miniature shaft encoders.

We needed an interface to connect the encoders to our RCMx1 so we hybridized SparkFun Qwiic_Twist LED boards by removing the potentiometer knob and adding headers to meet the need. *Firmware edit added to the Qwiic_Twist GitHub repository.

Fig. 10. SparkFun Qwiic_Twist LED boards.

B. Software

1) Low Level Control

Low level control onboard the robot is managed through the E3 RCM-BFin main CPU and the E3 RCMx1 control interface. The RCM-BFin manages control code and communication with the RCMx1. The RCMx1 includes a PIC processor and manages all the I/O port functions on the RCMx1. Internal control operates across a 5V, I2C serial bus (SPI and UART also available).

The ports/registers of the RCMx1 handle various functions:

- A-D (H-Bridge Motor Outputs at Battery V)
- 1 – 8 (PWM/RC Servo and GPIO at 5V)
- 9 – 22 (Analog Input and GPIO at 5V)
- 23 – 28 (Digital GPIO at 5V)

H-Bridge and PWM/RC Servo (pulse timing from 1.0ms - 2.0ms) motor control equate to an 8-bit protocol/decimal range from 0 – 255 (hex range from 0x00 – 0xFF).

- 0 (0x00) = power off
- 1 (0x01) = full speed CW
- 128 (0x80) = stop/brake
- 255 (0xFF) = full speed CCW

Ports 1 – 22 as GPIO utilize a value of 0 or 1. Ports 23 – 28 utilize a value of 0 or the 8-bit value equated to the port:

- Port 23 = 0 or 1
- Port 24 = 0 or 2
- Port 25 = 0 or 4
- Port 26 = 0 or 8
- Port 27 = 0 or 16
- Port 28 = 0 or 32

Ports 9 – 22 as Analog Inputs utilize a 12-bit value range from 0 (0V) to 4095 (5V).

2) Communication Protocol

Communication is handled using the client server model, where the client software on the PC requests data or sends commands to the server running on the robot. The protocol used is a custom TCP/IP command/response binary protocol utilizing packet formatting so that command/data integrity can be guaranteed.

Commands can be sent from the client to the server requesting that the server push out certain types of data to the client at predetermined rates without the client requesting more data (streaming). This is used in cases like video frames or certain sensor data.

Video frames (jpeg) are captured from the robot camera, compressed, and sent to the client for display. Sensor or other data from the server can be retrieved by the client by individual commands, or can be streamed to the client at certain intervals.

Many commands do not request data from the server but simply request robot state changes (i.e. changing outputs, uploading new programs).

3) Control Protocol

Automated control protocols can be managed locally at the robot or remotely at the command center. Robot managed control consists of low level autonomous elements such as navigation (wall following, controlled turns, obstacle avoidance), and preliminary victim detection (thermal signatures, CO2 signatures, motion and sound detection). High level control elements are managed at the command center with data streamed from the robot (localization and mapping, path planning, symbol interpretation), or through a hybrid exchange between the robot and command center.

Human control protocols are managed through the command center.

C. Communication

RoboBlues uses integrated multi-frequency radios (802.11 a/b/g/n) that allow for expanded flexibility to meet the requirements of different locations (for RoboCup Rescue RMRC we will be utilizing 5GHz-802.11a). The radios maximize performance on congested frequencies.

We use an embedded Lantronix XPico radio on the robot:

- E3 RCM-Comm carrier board
- Lantronix XPico 240 Embedded IOT GW, 802.11a/b/g/n, Eth (dual U.FL connectors)
- 2.4/5GHz 3dBi Omni Swivel Antennae (U.FL connector)

SSID: RRL-RMRC RoboBlues

D. Human-Robot Interface

Control for the RoboBlues robot incorporates both teleoperation and autonomous functions. Teleoperation is managed by the Engineering[3] MainController custom control interface built on National Instruments LabVIEW software.

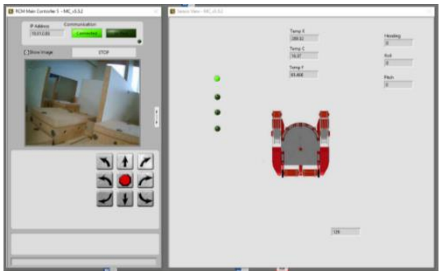


Fig. 11. MainController control interface.

From our MainController.vi we simply input the robot IP address and connect. We can then drive motors, monitor sensors, monitor video, load and run code.

Code for the RoboBlues robot is written in picoC. [3]

We have a semi-functional interactive mode in our firmware that allows for simultaneous running of autonomous functions and teleoperative controls, but it still needs more work.

III. System Description – Command Center

A. Hardware

1) Power Base

The Power Base functions as both the electrical distribution center and support base for the Command Center. The electrical core of the Power Base is a 100-240VAC-5A Input/15VDC-22A output power supply. The power supply feeds a DC/DC distribution/battery back-up module that manages power output to components from either the power supply or installed rechargeable LiFEPO4 battery. Also integrated into the Power Base are voltage regulators for supplying set voltage to components that don’t accept the 15VDC output of the power supply (i.e. monitors only accept 12VDC).

A fixture in the center of the Power Base accepts a multi-rail support post to which all other command center components attach.

2) Computers

There are two fanless, embedded PCs attached to the Command Center. Two computers allows for simultaneous running of multiple operating systems, or multiple apps across different hardware to minimize hardware sharing across high demand apps and/or differentiated robot control processes.

3) Monitors

Each of the computers (above) has an independent 19 inch monitor.

4) Access Point

The Command Center access point uses MikroTik components:

- MikroTik RouterBOARD RB/433AH
- MikroTik Radio R52Hn 802.11a/b/g/n 320mW mini PCI card (dual MMCX connectors)
- MikroTik 2.4/5GHz 3dBi Omni Swivel Antennae (MMCX connector)

SSID: RRL-RMRC RoboBlues

5) Peripherals

- Keyboard
- Mouse

B. Software

1) Operating System(s)

We use Microsoft Windows for our primary robot control interface. We can run either MS Windows or Linux/Ubuntu/ROS on our second computer, depending on the applications we choose to run.

2) Applications

Our primary robot control interface (MainController) is built on National Instruments LabVIEW. The MainController.vi is open source and can be run on LabVIEW 2014 and newer. We can also generate a stand-alone exe file of MainController for anyone to run without LabVIEW.

In our explorations with image processing, our initial approach is to leverage the National Instruments Vision Development Module as it can interface with LabVIEW. If we encounter significant difficulties with this approach, we will explore alternate applications.

Fig. 12. RoboBlues Command Center.

IV. Application

A. Set-up and Break-Down

The robot should ship fully assembled. Set-up is installation of a fresh battery and activate the power switch. Break-down is power off the robot and remove the battery.

Set-up of the command center is to install the battery into the Power Base (quick connect cable, six screws to attach the top); attach the support rail (three screws); slide on the dual-computer stack, access point and monitors (tighten the thumbscrew clamps for each); attach the keyboard (hook-n-loop) and mouse; connect the wiring harness; power up.

Break-down of the command center and robot is to reverse the above processes.

B. Experiments

We've developed a hybrid RMRC testing field that includes the RMRC terrain elements (dexterity elements to be added; gravel ramps separate) within a 165cm^2, two-story simulated building. The test field is a permanent fixture in our lab and student teams have open access to the field for testing their robots and determining the viability of all systems related to teleoperaton, navigation, mapping, etc.

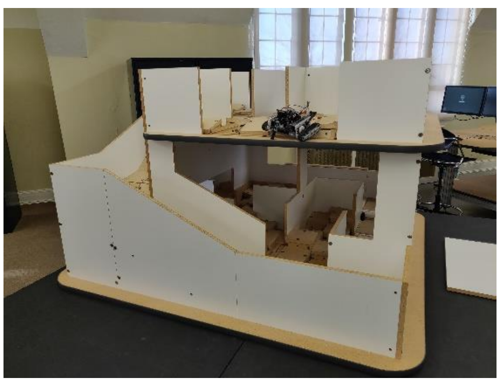


Fig. 13. Test Field; Full View.

Fig. 14. Test Field; Cross Views.

C. Application in the Field

Primary to our focus is small form-factor. We seek to demonstrate how significant research can be accomplished across both mechanical and control systems development, and at a minimal expense, without the complications associated with full scale platforms and testing suites. This can pave the way for more researchers to enter the field by minimizing financial and fabrication barriers to entry.

After development at the small form-factor, all elements can be more easily scaled up as the need arises.

V. Conclusion

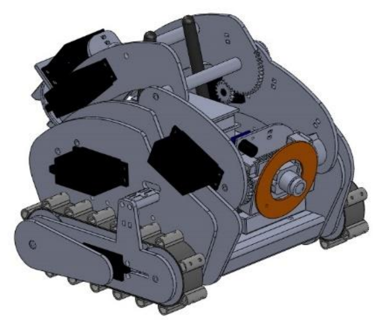
For The Hill School, this will be the first year of participating in a formal RoboCup RMRC event. We hope to learn how participation in RoboCup RMRC helps enrich our Hill Engineering program and our students. Looking to the future, we hope RoboCup RMRC can be a long-term option to add to our student learning trajectory.

Appendix A
Team members and Their Contributions

Each E3/E4 Student/Senior in Hill Engineering participates across all of the following areas. Specific technical roles for team members:

- Robot Locomotion/Treads – P. Dondeti
- Robot Locomotion/Drive – A. Grippo
- Control – M. Schiavone
- Image Processing – T. Youndt
- Robotic Arm – E. Cardoza
- Operations - All

Appendix B
CAD Drawing/Prototype



APPENDIX C
LISTS

A. Systems List

TABLE IA
MOBILITY SYSTEM

Attribute	Value
Locomotion	Tracked
System Weight	1.7 kg
Typical Operation Size	24 x 22 x 18 cm
Startup Time (ready for full operation)	20 ms
Battery Capacity	2000 mAh
Battery Endurance (idle/ normal/ heavy load)	240 / 120 / 60 min
Battery Charge Time (80% / 100%)	30 / 45 min
Maximum Speed	25 cm/s
Payload Capacity	untested
Any other interesting attribute	Flipper Arms
Cost	1000 USD

TABLE IIA
MANIPULATOR SYSTEM

Attribute	Value
Locomotion	ConRot Servo/Gearbox
System Weight	0.6 Kg
Maximum Operation Height	18 cm (above chassis)
Maximum Operation Reach	18 cm (rear of chassis)
Payload (at full extend)	untested
Power Consumption (idle/ typical/ max)	untested
Any other interesting attribute	Mid-Hinged Forearm
Cost	$200 USD

TABLE IIIA
OPERATOR STATION

Attribute	Value
Power Base	Var.
Computer 1	Windows
Computer 2	Windows/ROS
Monitors (2)	19in
Access Point	802.11a
Keyboard/Mouse	Var.
System Weight	10 kg
Typical Operation Size	45 x 55 x 66 cm
Startup Time (ready for full operation)	20 min
Battery Power	122.8 Wh
Battery Endurance (idle/ normal/ heavy load)	(est.) 11 / 9 / 7 h
Battery Charge Time (80% / 100%)	(est.) 0.75 / 1 h
Cost	2500 USD

TABLE IVA
SUPPORT COMPONENTS

Attribute	Value
Battery Chargers (2)	0.5 kg
Extra Batteries (4)	1 kg
Power Strip	100-240V
Extension Cord	7.6 m
International Plug Adapters	Var.
Cost	200 USD

TABLE VA
TRANSPORTATION AND SET-UP/ROBOT

Attribute	Value
System Weight w/Transportation Case *Includes Robot, Operator Station (simple) and Support Components	< 23 kg
Transportation size	63.5 x 45.7 x 53.3 cm
Unpack and assembly time	10 min
Cost	700 USD

TABLE VIA
TRANSPORTATION AND SET-UP/COMMAND CENTER

Attribute	Value
System Weight w/Transportation Case *Includes Robot, Operator Station (simple) and Support Components	< 23 kg
Transportation size	63.5 x 45.7 x 53.3 cm
Unpack and assembly time	20 min
Cost	1800 USD

B. Hardware Components List

TABLE IB
ROBOT/MOBILITY COMPONENTS

Part	Brand & Model	Unit Price	Num.
Continuous Rotation Servo Motors	Parallax 900-00008	15 USD	8
Modular Tread	E3	15 USD	32

TABLE IIB
ROBOT/STRUCTURAL COMPONENTS

Part	Brand & Model	Unit Price	Num.
Acetyl Plate (0.126”) 12”x24”	Generic	30 USD /Sheet	Var.
3D Print ABS Plus	Stratasys P430	250 USD /Cartridge	Var.
8-32 Screws Var. Lengths	Generic	Var. /Length	Var.
4-40 Screws/Standoffs Var. Lengths	Generic	Var. /Length	Var.
0-80 Screws/Standoffs Var. Lengths	Generic	Var. /Length	Var.
Splined Shaft (Axles)	Grob 250-20-2-464 Naval Brass	18 USD/Ft	2

TABLE IIIB
ROBOT/CONTROL COMPONENTS

Part	Brand & Model	Unit Price	Num.
RCM-Control Stack RCM-BFin RCM-Cam RCM-Comm RCM-x1 RCM-PBB	E3	850 USD	1
Battery	E3-Bat	20 USD	1
IMU	Adafruit/	35 USD	1
Shaft Encoder	US Digital E4T	22 USD	5
Shaft Encoder	US Digital E8T	32 USD	1
Encoder Interface (modified)	SparkFun Qwiic_Twist	20 USD	6
Thermal	E3-Thermal	46 USD	1
CO2	CO2Meter ExplorIR-W 5%	110 USD	1

TABLE IVB
OPERATOR STATION COMPONENTS

Part	Brand & Model	Unit Price	Num.
Computers	Intel NUC i5	800 USD	2
Monitors	Insignia (BestBuy)	100 USD	2
Access Point	MikroTik RouterBoard RB/433AH	65 USD	1
	MikroTik Radio R52Hn	40 USD	1
Keyboard	TBD	15 USD	1
Mouse	TBD	15 USD	1

TABLE VB
SUPPORT COMPONENTS

Part	Brand & Model	Unit Price	Num.
Battery Chargers	3PN3020MP-NH-TAM	25 USD	2
Extra Batteries	E3-Bat	20 USD	4
Power Strip	Generic	30 USD	1
Extension Cord	Generic	30 USD	1
Intl. Plug Adapters	Generic	20 USD	1

C. Software List

TABLE IC
SOFTWARE LIST

Name	Version	License	Usage
Windows	7	KMS Client	OS
NI LabVIEW	2014	NI VLP	Control OS
SolidWorks	2018	SW PDM	CAD
Notepad++	7.5.9	GNU/GPL	Code Dev
picoC		BSD	Code Dev
E3 MainController	5.3.2	GNU/GPL	Control

D. Open Source Lists

TABLE ID
OPEN SOURCE LIST/HARDWARE

Name
E3 RCM-BFin
E3 RCM-Cam
E3 RCM-Comm
E3 RCMx1
E3 RCM-PBB
E3 Touch
E3 Thermal
Robot 1

TABLE IID
OPEN SOURCE LIST/SOFTWARE

Name
RCM-BFin Firmware
RCMx1 Firmware
E3 MainController
SparkFun Qwiic_Twist Firmware

ACKNOLEDGEMENTS

Brian Schmalz, Schmalzhaus LLC. – Control Firmware, Hardware and Coding Support
Michael Schultz – MainController Support
Jim Yost – Command Center Hardware Support

REFERENCES

[1] T. Jump, “Engineering^3”, 2015.
[2] A. Jacoff, E. Messina, H-M. Huang, A.-M. Virts, A. Downs, R. Norcross, R. Sheh, “Standard Test Methods for Response Robots” E54.08.01.
[3] Z. Saleeba, “picoC”, 2009