**Autonomous Light Beacon Robot**

Project Requirements

Zachary Feuerstein

Kevin Hoffman

Xiaolin Zheng

1] Robot

- Our team must provide a single, self-contained autonomous robot to act as a participant in the game

- This robot must not have any information on beacon or obstacle locations before entering

- This robot must not communicate or receive any input or information from external

hardware sources

- All logical operations used to progress through the field and play the game must be internalized to a controller unit on the robotic platform

- The robot chassis must have some power source used to drive its components

- Our team robot should be capable of moving through the field in order to play the game effectively.

- Must have wheels and motors to drive them.

- Should have some logical algorithm for the controller to instruct it on how to move the motors to perform certain movement operations.

- The robot must be proportioned so as to fit within the maximum size of two feet long by two feet wide by one foot high.

- The robot must be small enough in size to navigate throughout the field without being permanently impeded by obstacles.

- The robot should be sufficiently tall in order to communicate with the beacon towers.

-The robot must operate safely

- Must be capable of completing the game without damage to the board or the other team’s platform.

- Robot should have internal logical checks to ensure that its operation won’t burn out or overstress its components.

- It must not use any Li-Ion or Li-Pol batteries to power the robot.

- Motors should remain off at the start of play, until the 2.5 ‘ by 2.5 ‘ pen is lifted at the game’s beginning.

2] Field

- Beacons

- Must be capable of assigning given team color to the robot at the start of the game

- Robot should have some latent memory element to store team status

- Robot should recognize the status of beacons’ possession

- Robot should possess some light sensing system for detecting different colors/wavelengths and intensities of light.

- Robot may want to recognize the difference between captured and uncaptured beacons.

- Robot should be capable of locating the beacons’ positions relative to itself, and moving to within communication distance of it.

- Controlling algorithm should have some method of assigning priority to beacons of different colors (e.g. unclaimed versus held by opponent) and deciding which to pursue

-Robot should be able to communicate with the light beacons once they are found.

- Robot must have some form of IR transmission and receiving hardware

- Robot should be programmed with correct protocols, based off those of the beacon towers, so that the towers will be receptive to its communication.

- Robot must claim the beacon to its team by instructing it to change its color to that of its team.

-Obstacles

- Robot should be capable of recognizing the presence of an obstacle.

- Must possess some sensory means of detecting the presence of an obstacle.

- Robot should navigate obstacles in order to communicate with the beacon towers

- Should have some logical algorithm for determining what sequence of movements to use to effectively bypass the obstacle.

3] GamePlay

- Robotic platform should be capable of playing the game for the full three minute period.

- Our team should implement some way to safely turn off the robot at the end of play.

- Must not intentionally interfere with the operation of the beacon towers, the game field, or the opposing team’s platform.

- Any unintentional interference caused by components of our robotic platform must be confirmed/discussed with an instructor.

----------------------------------------------------------------------------------------------------------------------------

**Autonomous Light Beacon Robot**

Subsystem Requirements

Zachary Feuerstein

Kevin Hoffman

Xiaolin Zheng

1) Controller Unit

- Must contain all high-level algorithms related to the control and operation of the robotic platform’s various functions

- Must take input from the sensor/communication board in order to effectively communicate with the beacon towers

- Shall be capable of storing sensory data and performing basic logical and arithmetic operations on said data

- Shall be capable of taking feedback from obstacle avoidance sensors and interpreting said feedback in order to control the movement and orientation of the platform

- Must possess some method of analog-to-digital and digital-to-analog conversion of input and output signals in order to interface properly with hardware

2) Sensor/Communication Board

- Must detect the location of beacon towers based on the color and/or intensity of radiating light

- Must receive a 38KHz-modulated infrared signal and communicate with the beacon towers via a Universal Asynchronous Receiver/Transmitter (UART) protocol at a baud rate of 300 symbols per second.

- Must transmit the capture signal, provided by the controller, in order to claim the beacon

- Must transmit within a one-second window between outgoing data packets from the tower

- Must detect the presence of obstacles on the playing field and provide relevant feedback to the controller based on location

3) Motor Actuation

- Shall receive control signals from controller and output relevant components as actuation signals

- Shall either restrict or allow the supply of voltage to the motors (e.g. enabled/disabled)

- Shall determine the direction of movement for the platform by taking control signals and using them to control the spinning direction of the individual motors (e.g. clockwise/counterclockwise)

- Shall provide feedback from the motors’ quadrature encoders to the controlling unit as input to movement algorithm

4) Power Management

- Must provide adequate power for all components on the robotic platform to operate safely

- Must provide a minimum of 5V for operation of the controller and motor actuation subsystems

- If necessary, should regulate certain voltage and current levels to allow for operation of certain components

-----------------------------

(From Moodle, project description)

Each team should create and enter a single self-contained autonomous robotic system, which will serve as

primary competing participant. While each team may only enter a single design, teams may ﬁnd it beneﬁcial to

create multiple “copies” of a design to aid in testing; and to provide a backup in case of electrical or mechanical

failure.

While the choice of a robotic platform is left to the teams, teams will be selected to ensure that at least

one member of each team owns a Digilent RDK platform in order to reduce costs to the student. To ensure

that the competition is fair to all entrants, the robot platform used in this year’s sophomore-level classes (the

Dagu Rover 5) cannot be used without prior instructor approval.

In addition to the requirements above, each robot must meet the following requirements:

max width: 2'

max length:

2'

Figure 1: Size limits

• To ensure that the robot ﬁts into each possible playing ﬁeld, the robot must

be less than two feet wide; and less than two feet long; measured from the

furthest produding points.

• To ensure that robots cannot interact with other competition areas, robots

must be less than one foot tall, measured from the highest point.

• All robots must be completely autonomous: that is, no human interaction

is allowed. Once the robot has entered the ﬁeld area, it cannot receive any

communications from outside of the ﬁeld area.

• Robots may not be designed to intentionally interfere with the gameplay

elements or opposing team. Unintentional interference may be acceptable—

for example, ultrasonic sensors may be employed; but note the possibility

of unintentional interference from the other team. If you suspect your robot

may produce (or receive) interference, ask the instructor if your design is

acceptable.

• All robots must be designed for safety; even under failure conditions. As a

safety precaution, your robot may not use lithium-based battery technologies

(e.g. Li-ion/Li-Pol); while these batteries oﬀer high energy densities, they

can pose a ﬁre hazard if incorrectly designed or used. If you are unsure as

to the safety of any design or technology, ask the instructor.

Problems We Will Face:

\* We need to have some way of determing a fixed displacement of movement

\* Could use the quadrature encoders

\* Zach has VHDL code from previous projects

\* When the board gets made: determine how fast we want our robot to move?

\* What risks are there in hitting a wall damaging our platform or a tower?

\* Robot needs to detect obstacles

\* Research sensors

\* Mechanical: Limit switches

\* Optical: Reflectivity sensors

\* Ultrasonic: Rangefinders

\* Probably want to use some combination of long and short range detection

\* Robot needs to avoid obstacles

\* Might come down to probability based on where we are in the game field

\* Might be able to avoid obstacles entirely based on tower position

\* Just something to think on for now

4) Robot needs to recognize the beacons

\* IR communication: Should probably do basic research

\* Need some way to detect the wavelength/intensity of light hitting our platform

5) Needs to communicate and receive information from the tower

\* How far does that IR signal travel?

\* How square do we need to be with the sensor?

\* How? What kind of information? How do we transmit our team info over?

6) Needs to detect the color/capture status of the beacons

7) Needs to decide some priority for capture

\* This’ll come down to pragmatism: Is it worth it based on how our platform operates?

8) Needs to (roughly) map the board

\* Can we give it info only on where it starts? i.e. the pen where it’s encased?

9) Power management, weight and hardware limitations

\* Cost benefit needs to happen once we know what components we need

Components:

1) ArduinoMega

2) IR sensor and receiver (1 each)

3) Luminosity/light sensors (4 to 8)

4) Robotic platform (Sophomore Design)

5) Working motors & a means to interpret/send information from/to the motors (Sophomore Design)

6) Some way to elevate the IR/PV/Light sensors (Questions about PCB mounting)

7) Contact sensors for obstacle avoidance (depending on size: 3 minimum)

8) Power management: Sensors vs. motors vs. controller

Details

\* 6' x 6' competition zone

\* Obstacles will not exceed 3' in either horizontal dimension

\* No obstacle will be placed between the robot and the closest beacon

\* 3 minute games

\* Captured beacons change on both sides

\* Can capture owned and unowned beacons alike

------------------------------

**Autonomous Light Beacon Robot**

Power Budget

Zachary Feuerstein

Kevin Hoffman

Xiaolin Zheng

Controller

Arduino Mega

- Voltage

o Minimum: 5V

o Maximum: 20V

o Optimal: 9V

- Current

o Per I/O Pin: 40 mA

o Per 3.3V Pin: 50 mA

Sensor/Communication Board

Light Sensors

1) CdS, PDV-P9005

- Voltage

o Maximum: 150V

- Continuous Power Dissipation

o Maximum: 125 mW/C⁰

- Resistance

o Dark Minimum: 2.5 MΩ

o Illuminated Minimum: 50 KΩ

o Illuminated Maximum: 94 KΩ

2) Photodiode, ISL29006

- Voltage

o Minimum: 1.8V

o Maximum: 3.6V

- Current

o Typical @ Dark: 220 nA

o Maximum @ Dark: 2.5 µA

o Typical @ 0 lux: 250 nA

o Typical @ 100 lux: 3.5 µA

o Typical @ 1000 lux: 27 µA

o Maximum @ 1000 lux: 35 µA

3) Photodiode, BH1603FVC

- Voltage

o Minimum: 2.4V

o Typical: 3.0V

o Maximum: 5.5V

- Current

o Minimum Supply: 51 µA

o Typical Supply: 74 µA

o Maximum Supply: 97 µA

o Maximum Output: 7.5 mA

- Power Dissipation

o Typical @ Room Temperature: 260 mW

4) Phototransistor, SFH 3710

- Voltage

o Typical Turn-On: 0.5V

o Typical Saturation: 5.5V

- Current

o Typical @ Collector: 20 mA

o Typical Dark Current: 3 nA

Contact Sensors

[Unavailable]

Ultrasound/Proximity

1) Ultrasonic Ranging, HC-SR04

- Voltage

o Typical: 5V

- Current

o Typical: 15 mA

IR Transmission/Receiving

[Unavailable]

Power Management

[Unavailable]

Motor Actuation

H-Bridge

<https://digilentinc.com/Data/Products/PMOD-HB5/PmodHB5_D_sch.pdf>

<https://digilentinc.com/Data/Products/PMOD-HB5/PmodHB5_RevD_rm.pdf>

Motors

- Voltage

o Minimum: 5V

o Maximum: 5V

- Current

------------------------------

Member Contribution Form

Monday, February 10th

-- All three present: Worked from 11 AM to 1 PM

- Requirements list and high-level brainstorming: Kevin, Xiaolin, Zachary

Wednesday, February 12th

-- All three present: Worked from 10:50 AM - 2:30 PM

- Soldering: Xiaolin, Zachary

- High-level conceptualizing and brainstorming: Kevin, Xiaolin

Thursday, Feburary 13th

-- One present: Worked from 2:00PM - 6:00PM

- Robot Assembly: Zachary

- Programming Controller to move motors: Zachary

Friday, February 14th

-- Two present: Kevin, Xiaolin. Worked from 1 PM to 5 PM

- Soldering: Kevin, Xiaolin

- Debugging and testing: Kevin, Xiaolin

Monday, February 17th

-- All three present: Worked from 11 AM to 1 PM, 4PM to 5:30

- Soldering: Xiaolin, Zachary

- Brainstorming: Kevin, Xiaolin, Zachary

- Block diagram creation, system analysis, subsystem requirements: Kevin, Xiaolin, Zachary

Wednesday, February 19th

-- Two present: Kevin, Xiaolin. Worked from 11 AM to 1:30 PM

- Beacon Board: LED characterization: Kevin, Xiaolin

Thursday , February 20th

-- One present: Zachary, Worked from 4:00PM to 6:00PM

- Programmed robot to move in a straight line

Friday, February 21st

-- Two present: Kevin, Xiaolin. Worked from 1 PM to 4 PM

- High-level algorithm brainstorming: Kevin, Xiaolin

- Parts shopping: Kevin, Xiaolin

Monday, February 24th

-- All three present: Worked from 2 PM to 6 PM

- Parts shopping: Light, ultrasound/proximity, contact sensors: Kevin, Xiaolin, Zachary

- Re-characterization of LEDs (due to previous error): Kevin, Xiaolin, Zachary

Tuesday, February 25th

-- One present: Xiaolin, Worked from 5 PM to 7:30 PM

- Circuit Implementation Research, Regulation Systems: Xiaolin

Wednesday, February 26th

-- All three present: Worked from 11 AM to 1 PM, 2 PM to

- Robot Characterization, Actuation: Zachary

- IR Characterization, implementation brainstorming: Kevin, Xiaolin

- Updating requirements documents, Block diagrams: Zachary

- Basic programming of motors through controller: Zachary

- Organizing and collecting datasheets: Kevin

-

-------------

DESIGN REVIEW REQUIREMENTS:

1) Requirements Documents and Block Diagram

- Block diagram needs completed parts for all chosen components

- Indicate all connections in and out of S/C board

- Identify our robot control platform (Arduino Mega)

- Subsystem requirements should be updated with more specific points

2) Any supporting documentation for our S/C and PCB

- Calculations supporting schematics

- Consider ideal and non-ideal properties (e.g. tolerances)

- Simulation results for analog hardware

- Bring any prototypes we have working to the review

- Power supply requirements and power budget

3) Any and all non-passive component datasheets

4) High level algorithms, flowcharts, state machines

- Show algorithms for overall gameplay as well as sub-tasks

- Obstacle avoidance, beacon-finding, etc.

5) Member contribution form

SCHEMATIC FOR PCB

Thing to consider:

1) Power management

- Diodes

- Current/voltage regulators (are these pre-fabricated?)

2) Sensors (where do we want them positioned on the robot?)

- IR transmitter

- IR receiver

- Light sensor (e.g. photovoltaic, electro-optical)

- Potentially any physical sensors (e.g. obstacle avoidance)

3) Headers to the microcontroller

- For power management and sensors

TASKS FOR NEXT WEEK

1) IR characterization

- Look at the waveforms going in and out of the IR hardware

- Set up a spare sensor and transmitter on a breadboard

- Try to capture the transmitted signal

- Try to claim the tower by transmitting the claim code back

2) Power Budget

- Figure out current draw of major components under load

- Use datasheets for our chosen S/C components

- Get a rough idea of net current draw

- Look up voltage/current regulating circuitry

3) Sensor characterization

- Look up types of sensors, confirm with course staff

- Buy sensors

- Buy a “breakout board” so we can test via breadboard

4) PCB Schematic

- Sensor circuitry for transmission and reception

- Circuitry to stabilize power (e.g. decoupling capacitors, signal amplification)

5) High-level algorithm

- All sub-system input, outputs, and state changes

- Encapsulate the entire program into one large algorithm

6) Risk assessment

- Robot needs to move around the field

- Obstacles (hit or stuck)

- Avoid it before we hit it

\* Ultrasound

- Need to receive an accurate signal

- What if we hit a corner?

\* Contact sensor

- Beacon tower luminosity

- We need to be within some range that the IR circuitry can communicate

- As long as we’re receiving, we can transmit

- If, after a few attempts, the color doesn’t change, move closer

- How do we tell if the light has been claimed?

- Lx reading from sensors

- Clearly identify sensor towers that can be claimed

- How do we detect a sensor tower’s location and direction?

- Multiple light sensors (4 in each cardinal direction)

- Difference between sensors on opposite sides of board used for positioning

- Do we have enough analog/digital I/O pins on the Mega?

- Minimum 3 digital

- Minimum 6 analog

- Maximum 10 analog

ULTRASOUND/PROXIMITY DETECTORS:

<http://www.newark.com/global-specialties/arx-ult/ultrasound-distance-detector-asuro/dp/97W8199?ost=distance+detector&categoryId=800000004028>

and this is a PIR montion sensor <http://www.adafruit.com/products/189?gclid=CJDUoKKV47wCFcEDOgodNSYAsw>

Ultrasonic

<http://www.amazon.com/SainSmart-HC-SR04-Ranging-Detector-Distance/dp/B004U8TOE6/ref=sr_1_2?ie=UTF8&qid=1393270046&sr=8-2&keywords=arduino+ultrasonic+sensor>

Reciept

<http://prntscr.com/2vice3>

<http://prntscr.com/2vikgy>

LIGHT SENSORS/PHOTODETECTORS

\* Illuminance vs. Irradiance

\* Effect of wide dynamic range on performance

\* Optical/Light? Transistor vs. CdS? Different applications of tech?

<http://www.intersil.com/content/dam/Intersil/documents/isl2/isl29023.pdf>

<http://rohmfs.rohm.com/en/products/databook/datasheet/ic/sensor/light/bh1603fvc-e.pdf>

CONTACT SENSORS

-----------------------------

PURCHASED PARTS:

PDV-P9005-ND

<http://www.digikey.com/product-detail/en/PDV-P9005/PDV-P9005-ND/480621>

DS: <http://advancedphotonix.com/wp-content/uploads/PDV-P9005.pdf>

475-1259-1-ND

<http://www.digikey.com/product-detail/en/SFH%205711-2%2F3-Z/475-1259-1-ND/1312602>

DS: Attachment

ISL29006IROZ-T7CT-ND

<http://www.digikey.com/product-detail/en/ISL29006IROZ-T7/ISL29006IROZ-T7CT-ND/1952907>

DS: <http://www.intersil.com/content/dam/Intersil/documents/isl2/isl29006-07-08.pdf>

BH1603FVC-TR

<http://www.digikey.com/product-detail/en/BH1603FVC-TR/BH1603FVCTR-ND/1950367?WT.srch=1&WT.medium=cpc&WT.mc_id=IQ62057691-VQ2-g-VQ6-31922847435-VQ15-1t1-VQ16-c>

DS: <http://rohmfs.rohm.com/en/products/databook/datasheet/ic/sensor/light/bh1603fvc-e.pdf>

Sainsmart HC-SR04

<http://www.sainsmart.com/ultrasonic-ranging-detector-mod-hc-sr04-distance-sensor.html>

DS: <http://www.micropik.com/PDF/HCSR04.pdf>

B00944TGIA - push button

<http://www.amazon.com/Push-Button-For-MINDS-i-products/dp/B00944TGIA>

------------------------------

Zach’s Github Username:

zfeuers1

Files stored under “Junior Design”