

Self-Sustaining Solar Powered Robot

Critical Design Review

March 5, 2003

Mason Drew
Mark Nolan
Wesley Varghese

Table of Contents

Problem Statement	3
Problem Objective	3
Design Constraints	3
Physical Robot	4
Circuitry	5
Programming	6
Design Process Update	7
Accomplishments and Obstacles	8
Updated Gantt Chart	8
Goals and Conclusion	8

Problem Statement

Robots are more and more becoming the solution to accomplishing difficult and/or mundane tasks. Though they prove useful and versatile for many situations, robots tend to be restricted by their energy source. Some robots, such as those common in factories, require being plugged into an outlet at all times. More mobile alternatives use batteries but have a drawback since batteries must be recharged or even replaced periodically. A last group of robots are solar powered. These are not limited to outlets or the maintenance issue of batteries but are rather confined by the fact that they must be in direct light at all times. Though all of these types of robots serve well to complete particular functions, they depend too much outside monitoring and assistance to operate for long periods of time.

Problem Objective

The objective of this project is to design and implement a mobile self-sustaining robot. The robot should be able to operate unassisted for long periods of time. To do this, the robot should be able to perform a task while monitoring its battery level. Once the robot senses that its battery has dropped below a specified level, it will stop its current task to seek a strong light source. The robot should be able to navigate to the light and stop while charging its battery. Once the battery is charged, the robot will leave the light source and continue to perform the task assigned to it.

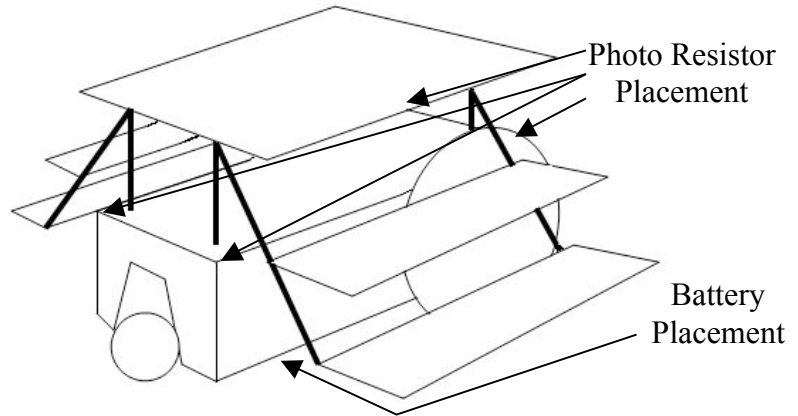
Design Constraints

It is vital, especially during the designing, experimenting, and implementing stages of the project, to keep in mind the constraints to provide a sense of feasibility and focus. The following constraints define the major aspects of the robot and play a key role in design process. First of all, the robot should be self-sustaining. It should be able to operate without any assistance from the outside world. This ability to work by itself can be broken down into three units: monitoring energy, seeking light and solar recharging. By monitoring energy, it should be able to know when it needs to find an energy source as well as when it is done charging. Light seeking includes finding the strongest light source and navigating to it for the purposes of charging or any other behavioral “instincts.” Finally, the robot should be able to charge by means of solar panels once the strongest light source is available. Inherently, we assume the robot to have mobility to the point where it can at least navigate on flat surfaces without obstacles.

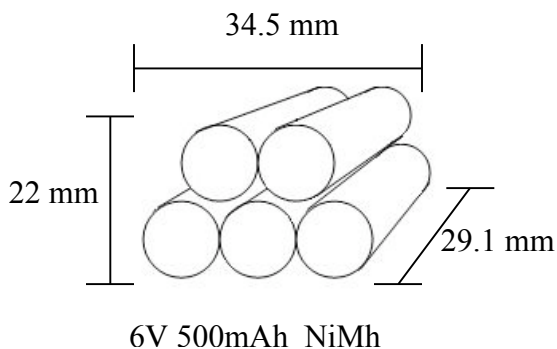
Some assumptions are must also be made to make this project feasible. One of them, as introduced in the mobility constraint, is the absence of obstacles blocking the robot from the light source. Strong contrast between bright light source and surrounding ambient light will also be necessary for the robot to seek an appropriate light source. Finally, we assume a strong light source will always be available. If time allows we plan to design a feature that makes the robot enter a low power “sleep” mode when a light source is not available. In sleep mode, the robot will evaluate the environment at a set interval until it senses a light source, then it will “wakeup” and perform as usual. With these constraints and assumptions in mind, we can now begin to design the actual solar powered robot.

Physical Robot

The physical makeup of the robot is the most basic level of design. The body of the Boe Bot is predetermined for the most part. It is therefore important to work around the limitations put in place by the existing robot parts when designing the solar robot to be both functional and energy efficient. One of the most important decisions, because of its size as well as the importance of positioning, was how to place arrangement of solar cells. Initial suggestions included a dome shape that would blanket the robot, which seemed more difficult to make worth in its usefulness, and just a flat layer over the robot, which would make the robot clumsy since it raised the center of gravity.



The final choice was a step ladder design (shown above right) that lowers the center of gravity while still allowing plenty of surface area to catch light. The design could easily be created using nothing more than some coat hangers and, if needed, some wire mesh. The placement of photo resistors on the four corners of the robot in this design would allow a good reading from all directions to indicate where light is located in the environment.

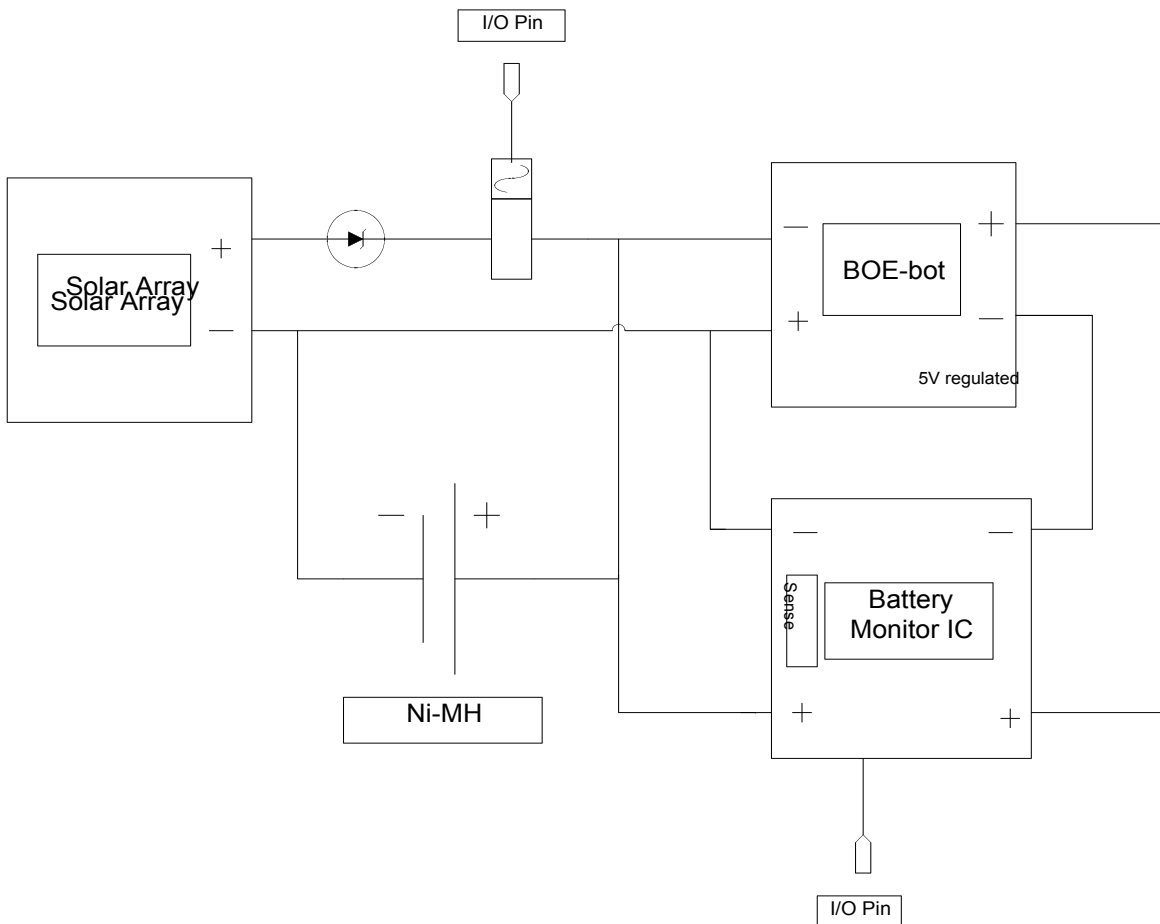


Aside from the solar cells and photo resistors, the arrangement of customized batteries and its placement were also an important issue. The first step in deciding this setup was determining how many battery cells would be needed to power the robot. Since the Boe Bot has the ability to regulate voltage for onboard components, like the microcontroller, the deciding factor

was the servos. These require 6 volts to work correctly, and thus an arrangement of 5 x 1.2 volt batteries was the ideal choice. Next came deciding the actual structure of the battery cell arrangement. The available mounting areas on the Boe Bot were the main consideration for this. Since the best (and intended) place to mount batteries was on the underside of the robot, an arrangement that would sit comfortably in this space was necessary. The measured usable dimensions of this area were 6 cm x 7 cm x 2.5 cm. After looking at battery measurements specifications on the website, we tried two arrangements that might work. The flat battery pack setup, with the cells laid next to each other, seemed feasible but could possibly have been a tight fit after mounting and attaching the batteries together. The pyramid setup (shown in the above left figure), fit within the required space and allowed enough tolerance for additional securing methods. With the batteries, solar cells, and photo resistors placed into the design, it is now time to begin looking into the logic that controls the solar power robot.

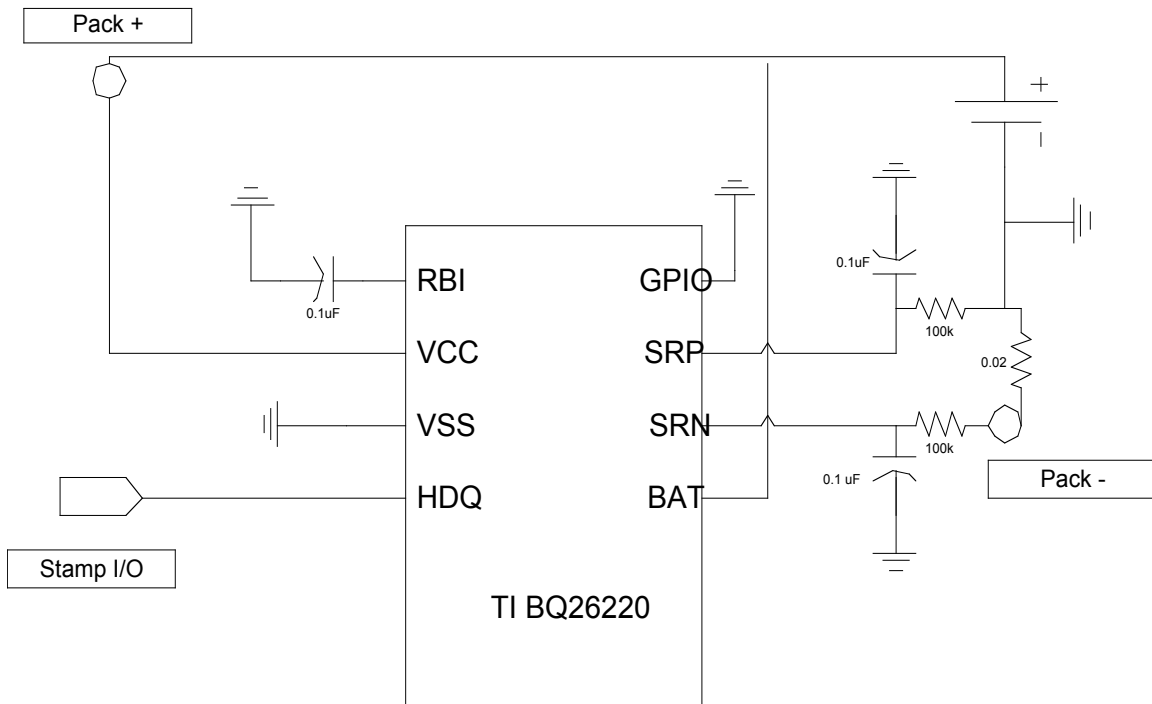
Circuitry

Circuitry that controls the functions of the robot is vital to success in this project. The circuit that will be used for operating the Boe Bot and charging the Ni-MH batteries is shown below. The charge on the batteries as well as the temperature of the batteries will be monitored using the battery monitoring IC (Texas Instruments bq26220). The BASIC Stamp2 will periodically poll the bq26220 and if the batteries are charged to capacity or if they become too hot the solar panels will be disconnected from the circuit via the relay switch. A diode is used to prevent current from flowing into the solar panels if they are not generating enough power. The nickel metal hydride batteries will source current for the Boe Bot and will be charged by the solar array if it is producing enough current.



The Texas Instruments bq26220 will be used to monitor the charge remaining on the batteries, the voltage across the terminals, and the temperature of the batteries. The monitoring circuit is presented in the diagram on the top of the next page. The IC uses a one pin output for asynchronous serial communication. One byte is written on the HDQ line and the format of that byte determines the bq26220's response. The chip uses two eight bit registers to store the value of the charge on the battery so two registers must be queried in order to retrieve the sixteen bit status of the battery. The capacitors are used to "hold-up" the voltage on the "sense" lines. Note that the incoming power source for the

bq26220 is the five volt output from the voltage regulator on the parallax board of education.



Programming

The code can be broken down into three main sections. Each of these sections will have their own specific duty that represents a function that the robot must perform to sustain its existence. The three sections are operate, seek light, and charge. These are broad generalities and the final code will probably include many more sub-sections of these.

The operate function (pseudocode is located to the right) is the actions that the robot will perform while it has adequate charge. This function is an infinite loop with a chance to break out of the loop if the battery level drops below an allotted threshold that we will determine. The first thing the robot will do when booted is run this function. Then it will start to perform its interesting behavior. After a few instructions of the interesting behavior are executed, the robot will poll the battery charge pin. If that number is less than the number allowed, then the robot needs to

```
operate:
  do interesting behavior
  .
  .
  .
  bc = get battery charge

  if bc < min_allotted
    goto seek_light

  goto operate
```

seek out a strong light source; therefore the seek_light function is called. If it determines that it has enough charge to continue, then the program loops back to the beginning of this function.

The seek_light function (pseudocode is located below left) is used to find an adequate

```
seek_light:

  if top_light_sensor >= enough_light
    goto charge

  for the number_of_sensors
    set control pins
    read input
  end for

  decide direction

  orient toward brightest light

  move toward light few inches

  goto seek_light
```

light source. The first thing this function will do is poll the photo resistor on the top of the robot. This sensor will give an accurate reading of the light directly hitting the solar panels on top. If this number is greater than or equal to a number that we will determine is adequate enough to charge the robot, then it will enter a charge state by calling the charge function. Otherwise, the robot will poll the photo resistors that are positioned directionally on the perimeter of the solar panels to get the direction of the strongest light. The robot will orient itself in that direction and move forward for a specified amount. After it has

moved forward the program will loop back to the beginning and start over by reading the light intensity off the top photo resistor.

The charge program (pseudocode is located to below right) is a very simple program to cause the robot to stand still and except the charge from the light source. It will pause for

approximately one minute, then check the battery charge level. If the charge level is not sufficient then it will loop back to the beginning and pause for another minute the recheck. If the robot determines that the charge was ample then it will break out of the charge loop by calling the operate function. By executing these three programs, the robot has essentially completed one entire "life-cycle".

```
charge:

  pause 1 minute

  get battery charge

  if battery charge >= max_threshold
    goto operate

  goto charge
```

In theory the robot should be able to operate forever as long as a strong light source is reachable by the robot.

Design Process Update

As with all projects, not all things go as expected. We have over the course of the past few weeks completed a number of tasks but not all that we originally planned.

Accomplishments and Obstacles

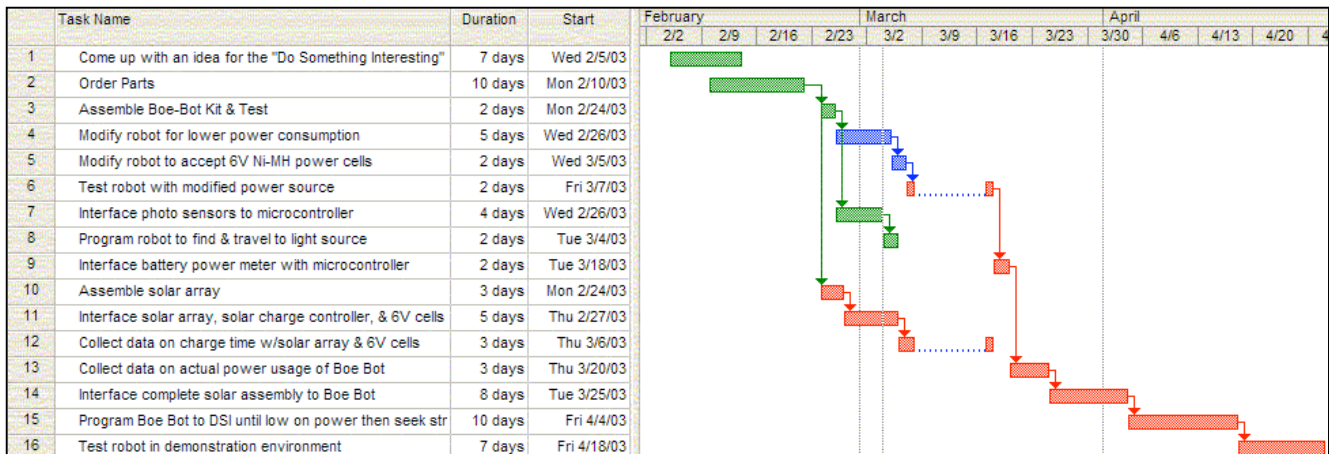
It is helpful to look at accomplishments to see if the project is going as expected. The following are specific tasks of our design process which have been successfully completed:

1. Developing interesting behavior.
2. Order individual parts individual from the specific companies.
3. Assemble Boe Bot kit and Test – build the basic robot and test mobility and circuitry.
4. Theoretically interface photo sensors to microcontroller via A/D converter.
5. Program robot to find and travel to light source – create an algorithm to sense the strongest light and navigate the robot toward it.
6. Interface battery power meter with microcontroller

There were also a number of issues that we encountered which must be resolved to make further progress. The following are specific tasks of our design process which have not been complete along with a short description of the obstacles preventing completion:

1. Assemble solar array – have not received solar cells.
2. Collect data on charge time with solar array and battery cells – neither part has arrived for testing purposes.
3. Testing Boe Bot proved functional except for unusual servo behavior.

Updated Gantt Chart



Green shows completed, blue shows in progress, red show pending.

Goals and Conclusion

A number of goals will help to direct our forward movement in the project. Since goals are not independent of the obstacles, a few of the following are obvious:

1. Obtain all parts – this specifically includes batteries and solar cells.
2. Modify and test robot for lower power consumption – Adjust the servo settings and reprogram microprocessor to consume less power to navigate robot.

3. Assemble solar array – assemble the solar array independent of robot to allow parallel testing. This also includes soldering the individual solar panels together.
4. Test parts specifics and tolerances – this includes checking our specific photo resistor behavior, checking power production of solar cells, calibrating servos.
5. Collect charge time and power usage data – Test the length of time to charge batteries with respect to different light sources as well as amount of power robot will consume will performing a specified task.
6. Interface complete solar power assembly to Boe-Bot – attach the completed add-ons to the robot and verify that it works.
7. Program Boe-Bot to do something interesting – Fully develop and program an interesting behavior for the robot to perform to consume energy.
8. Test robot in demonstration environment.

The design aspects of the robot are very much coming together. The theory behind building and programming the functions of the solar power robot are near completion. Remaining goals deal mostly with obtaining all the parts and physically combining what our theory outlines. The modified Gantt chart (located at the top of this page) shows how we have adapted our schedule to complete the project in an organized manner.

Note: Attached are Finite State Machine Diagram and Photo Resistor Testing Results.