Complete System Level Block Diagrams

Photovoltaic Martian Bugs

Adam Jackson and Matt Travis

Hardware Subsystems

The Photovoltaic Martian Bugs have the following hardware subsystems. Each subsystem is listed with its inputs and outputs, along with an operational description for each mode. Refer to Figure 1 for the hardware organization and Figure 2 for the hardware flowchart.

Subsystem: Light level sensor
Inputs: Photons
Outputs: Light level signal
Operation:

The bug is equipped with two or more light level sensors. Each of these sensors are mounted in a different position on the bug, each facing a different direction. In all modes, including the "off" modes, these sensors generate a signal indicating the current light level. This signal is used to determine whether the light level is sufficient to continue operation. In Forage mode, the microcontroller compares the light signals from the sensors and walks towards the region with more light. If the light level exceeds some threshold, the bug stops foraging and begins transmitting an infrared signal for other bugs to home in on.

Subsystem: Infrared transceiver
Inputs: Infrared data
Outputs: Infrared data
Operation:

The bug is equipped with two or more infrared transceivers. These are used for communication between bugs. In the Forage mode, the bug listens to the IR transceiver. If it receives a transmission from another bug indicating a good light source, the bug enters Tracking mode. In the Tracking mode, the microcontroller compares the relative strength of the received signals to determine the location of the bug sending the transmissions. To determine location a more than one transceiver is required. If the bug receives a signal on its left receiver it will turn left and move forward. The tracking bug continues to execute these moves and turns to converge on the position of the sending bug. If a signal is detected on more than one receiver the strongest of the signals will be chosen. In Transmission mode, the microcontroller sends out an IR signal on the transceiver to alert other bugs in the area of a strong light source. It waits for some period of time to allow the other bugs to converge on the signal, and then returns to Random-Walk mode. In all other modes, the IR transceivers are powered off.

Subsystem: Photovoltaic Cells
Inputs: Photons
Outputs: Power
Operation:

In all modes, the photovoltaic cells convert photons to electric current. This current is used to charge the backup battery, move the bug, and process the various other input signals the bug receives.

Subsystem: Battery
Inputs: Trickle-charge current
Outputs: Back-up current
Operation:

In all modes, the battery supplies the operational current needed to run the bug. It acts as a voltage source in parallel with the photovoltaic cells. The battery is not the primary power source. It is only intended to supplement the power from the solar cells during high load or low light conditions.

Subsystem: Collision sensors
Inputs: Hard objects
Outputs: Collision signal
Operation:

In all the moving modes (Tracking, Forage, Random-Walk), the collision sensors detect collisions with other objects. The microcontroller uses this signal to correct the path of the bug to avoid the object. In all other modes the collisions sensors have no effect.

Subsystem: Motor and power electronics
Inputs: Motion signal
Outputs: Kinetic energy
Operation:

In all the moving modes (Tracking, Forage, Random-Walk), the microcontroller sends motion signals to the motor to move the bug. The bug is capable of forward and reverse motion and can turn. In all the other modes this subsystem does nothing.

Subsystem: User I/O
Inputs: Button presses
Outputs: Blinking LEDs
Operation:

In all modes, the LEDs blink to indicate the current mode of operation. The user can change the operating mode by pressing one of the buttons (Sleep or Power).

Subsystem: Microcontroller
Inputs: Light level, collision signal, button presses, infrared data
Outputs: LED signals, motion signal, infrared data
Operation:

The microcontroller determines the current mode from the various inputs. Based on the current mode, the microcontroller generates the desired output signals. This is described in more detail in the next section.

Software Subsystem

The following is a breakdown of each of the software subsystems. The overall software flow chart seen in Figure 3 shows the interconnections between the software modules and how the system moves from state to state.

Subsystem: Random Walk Mode
Inputs: Light level data, collision signals, button presses
Outputs: Motion
Operation:

In random walk mode the bug will move a random distance X then turn in a random direction Y. The distances and direction will not be entirely random but will have enough combinations to have the appearance of randomness. This mode will also monitor the light level being input from the light sensors. If the level drops too far it will enter into Forage mode. If the bug encounters an object it will stop, back up, and turn a specified direction away from the object, and continue.

Subsystem: Forage Mode
Inputs: Light level, collision signal, button presses, infrared data
Outputs: Motion
Operation:

In Forage mode the bug attempts to find a stronger light source. In this mode its movements are no longer random but are based upon the inputs of the light sensors. If one light sensor produces a higher light level than the other the bug will turn in the direction of that sensor and continue forward. If the light levels are the same on both sensors the bug will continue to move straight forward. The bug will move in this manner until the light sensors read adequate light levels. This mode maintains the same procedure for encountering objects in the random walk mode. This mode also accepts signals from an infrared receiver. These signals indicate that another bug has found light and the receiving bug will attempt to locate the transmitting bug. If no signals are received and no light can be found after a certain amount of time the bug will enter sleep mode.

Subsystem: Tracking Mode
Inputs: Light level, collision signal, button presses, infrared data
Outputs: Motion, status LED
Operation:

If the bug is in Forage mode and looking for light it may receive an infrared signal from a second bug. This signal will tell the bug that there is light in a general direction. The receiving bug will detect which IR receiver has the strongest signal and turn in that direction and continue. The bug will continue doing this until it reaches and adequate light level or it loses the IR signal. If it loses the signal, it will return to forage mode and continue.

Subsystem: Transmission Mode
Inputs: Light level, button presses
Outputs: LED signals, infrared data
Operation:

When a bug finds a light source that is adequate to exit forage mode it will stop moving and begin to pulse out data on the IR transmitter. These signals indicate to other bugs that there is light at its location. Bugs receiving this signal will then attempt to converge on its location.

Subsystem: Sleep Mode
Inputs: Light level, button presses
Outputs: LED signals
Operation:

In sleep mode the bug does nothing besides wait. The bug waits for light to return or for an input from the user. While in this mode the solar cells collect what light they can to charge the battery and run the status LED.

Subsystem: Death Mode
Inputs: Photons
Outputs: None
Operation:

If the bug is in standby mode too long it has the possibility of depleting all of its energy sources. The system requires a very minimal amount of power in standby but if such power is not available the bug will not be able to continue operation. If this occurs the bug will essentially die. The bug will stay in this mode until human intervention revives it from this mode.

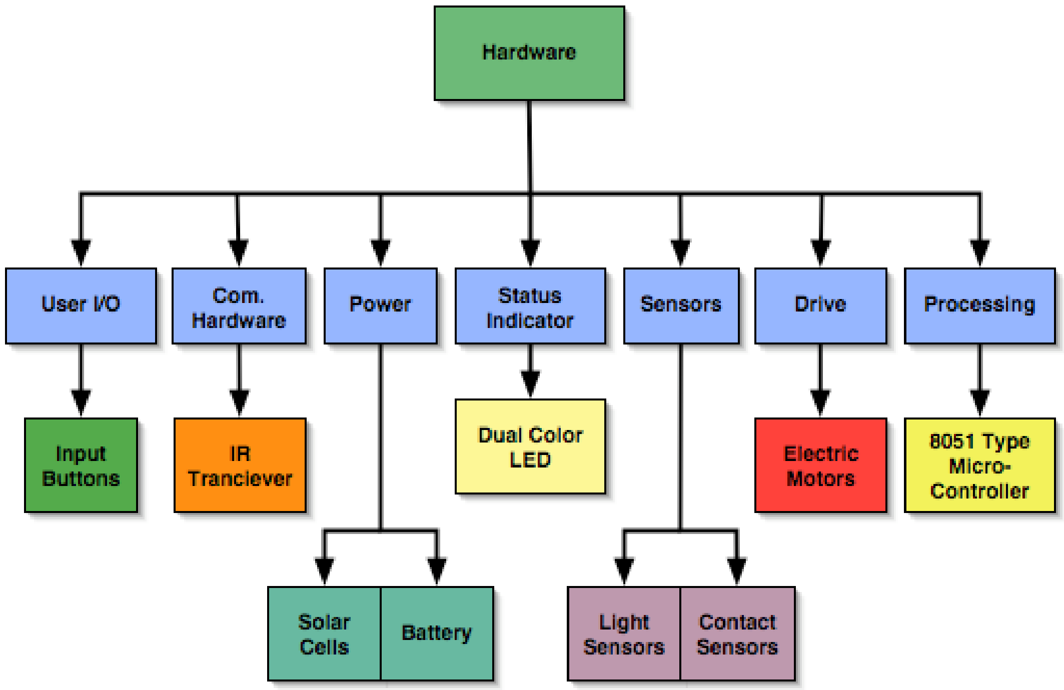

Figure 1: Hardware organizational chart

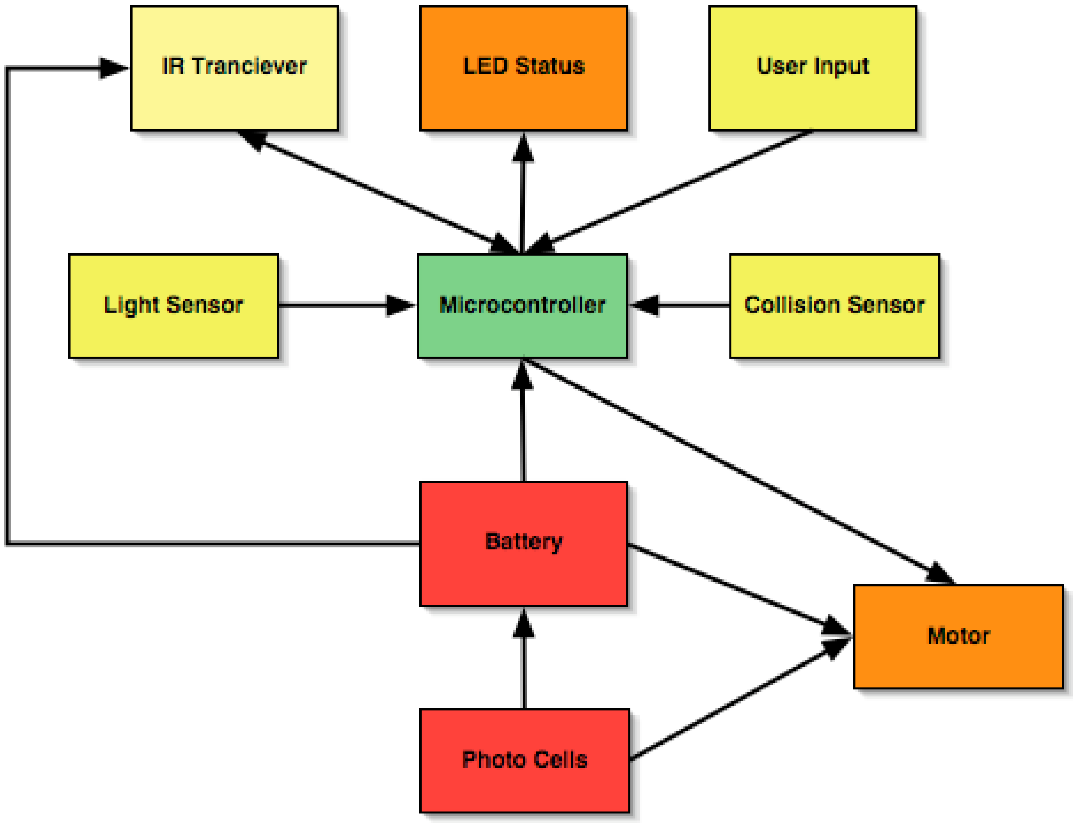

Figure 2: Hardware flowchart

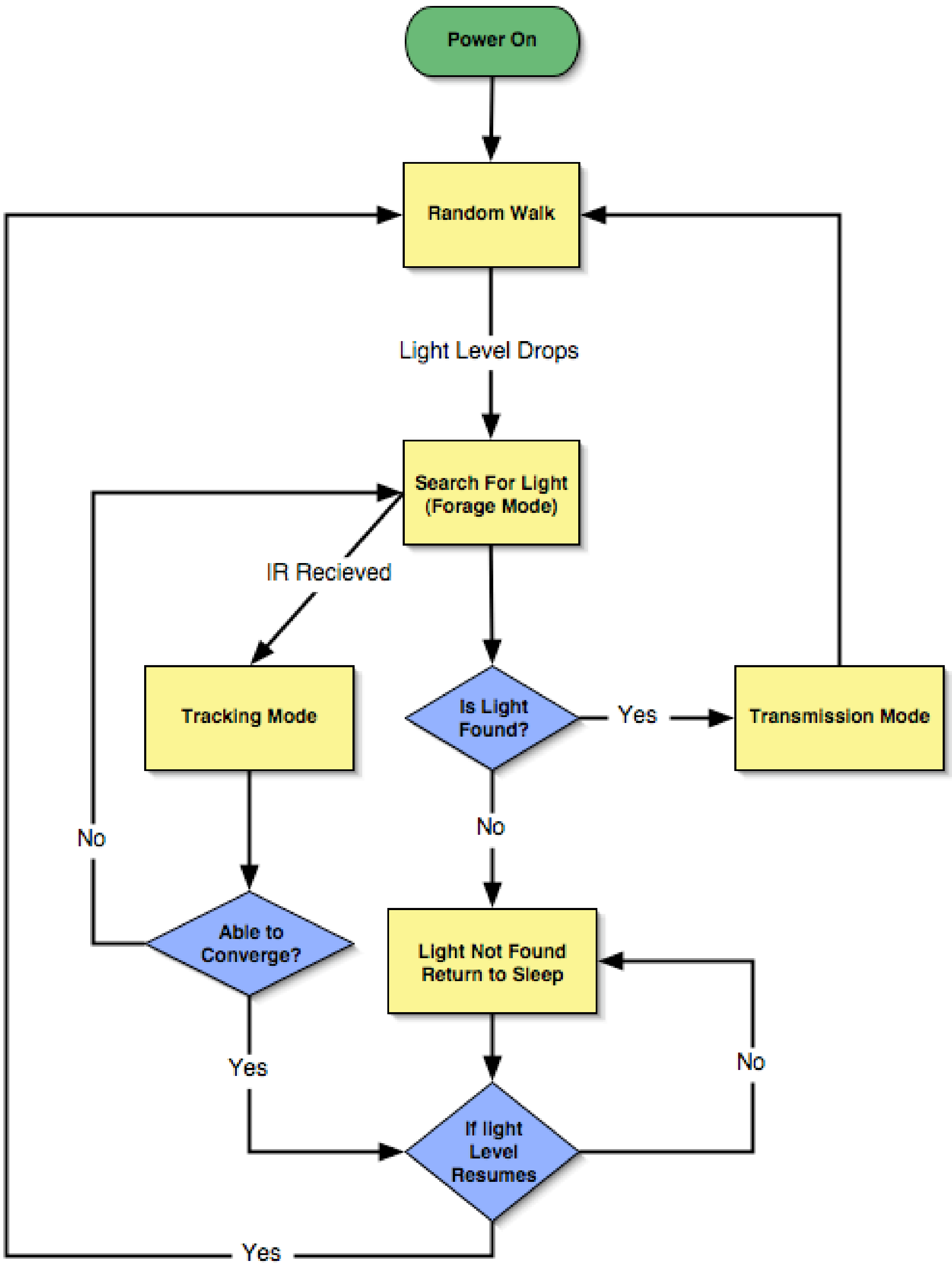

Figure 3: Software flowchart