A DISTRIBUTED, LOW-POWER TELEMETRY SYSTEM FOR SOLAR RACE CAR APPLICATIONS

E. S. Tuomey and G. Velasquez, Graduate Students
S. Slade, K. Bunker, E. Reyes and T. Yousefnejad, Undergraduate Students
L.C. Schooley, Faculty Advisor
The University of Arizona

ABSTRACT

This student paper was produced as part of the team design competition in the University of Arizona course ECE 485, Radiowaves and Telemetry. It describes the design of a telemetry system for the University of Arizona's Daedalus solar car. This is a distributed, low-power, telemetry-on-demand system that solves many of the problems typically encountered in this specialized telemetry application. The topology of the distributed microcontroller system is shown, as are optimal command and data packet structures. Also featured is a high-gain, low profile antenna system designed specifically for the solar car. Additionally, a customized chase car operator interface is illustrated.

KEY WORDS

Solar Car, Distributed, Low-power, Telemetering Systems

INTRODUCTION

The ultimate goal of any competitor in a solar car race is to cross the finish line at the same time the last electron is drained from the batteries. Having done this means you've run the most efficient race possible. To accomplish this task, the race team must monitor battery usage in all components of the solar car system and project remaining power at race-end. Of course, all of this must be done in a weight, space and power-efficient manner. To this end, the University of Arizona Daedalus solar car project requires a fairly unique telemetry system.

Data is requested from the sensors on the solar car through a distributed network of microcontrollers and passed back to a laptop computer on the chase car. The computer then uses a formula based upon battery drain, weight, and aerodynamic drag of the solar car to predict remaining power at race-end.

Based upon these calculations, an operator in the chase car can communicate to the solar car driver the most efficient driving parameters.

COMMAND SYSTEM

In contrast to the ‘normal’ telemetry system, the Daedalus telemetry system does not have a command processor on the payload. Instead, the payload command system is completely distributed, with each sensor ‘box’ having its own microcontroller. A distributed command system is not a radically new idea; in fact it is identical to the computer networks that most everyone is familiar with. The distributed command system has several advantages over the conventional command processor:

The microcontrollers draw less than 1 µA typical standby current
The microcontrollers require no cache or off-chip RAM memory
The microcontrollers are programmed with only 33 instructions (no large assembly code)
The redundancy allows the system to operate if any of the microcontrollers fail
The microcontrollers are efficient and require little overhead processing

The distributed organization of the telemetry system allows each box to survive and operate independently regardless of the status of the other boxes. This has a distinct advantage over the dedicated command processor, which controls all data gathering.

Figure 1 illustrates the sensor boxes and their relationship to the Phillips I^2C data bus. The vertical stripes on the bus represent data. The request for data is received via the Radio Modem and put on the bus (a request) the appropriate microcontroller then polls the data and bursts it out onto the bus for transmission via the radio modem (a send).

As mentioned above, the Daedalus telemetry system does not stream all of the data at all times. Rather, the system provides “telemetry-on-demand.” The 13 microcontrollers are initially set to SLEEP and stay that way until the chase car requests data or the WATCHDOG timer times out. The WATCHDOG timer is a built-in function of the microcontroller used in the system (see the next section), that activates the microcontroller by pulling an exception after a period of unexpected delay. The telemetry-on-demand functionality allows the chase car to slow or stop data gathering at any time for any consideration, or to increase the data gathering on any transducer for any reason.

SYSTEM BUS AND PACKET CONSTRUCTION

Each of the aforementioned sensor boxes is equipped with a Microchip 8-bit CMOS Microcontroller. The data packet is constructed upon request at the specified location and then a data transfer is initiated on the system bus. The system bus has been chosen for maximum flexibility. This versatility allows the user to monitor the status of a particular sensor at any one of the 13 locations without having to transmit data from all 13 locations.

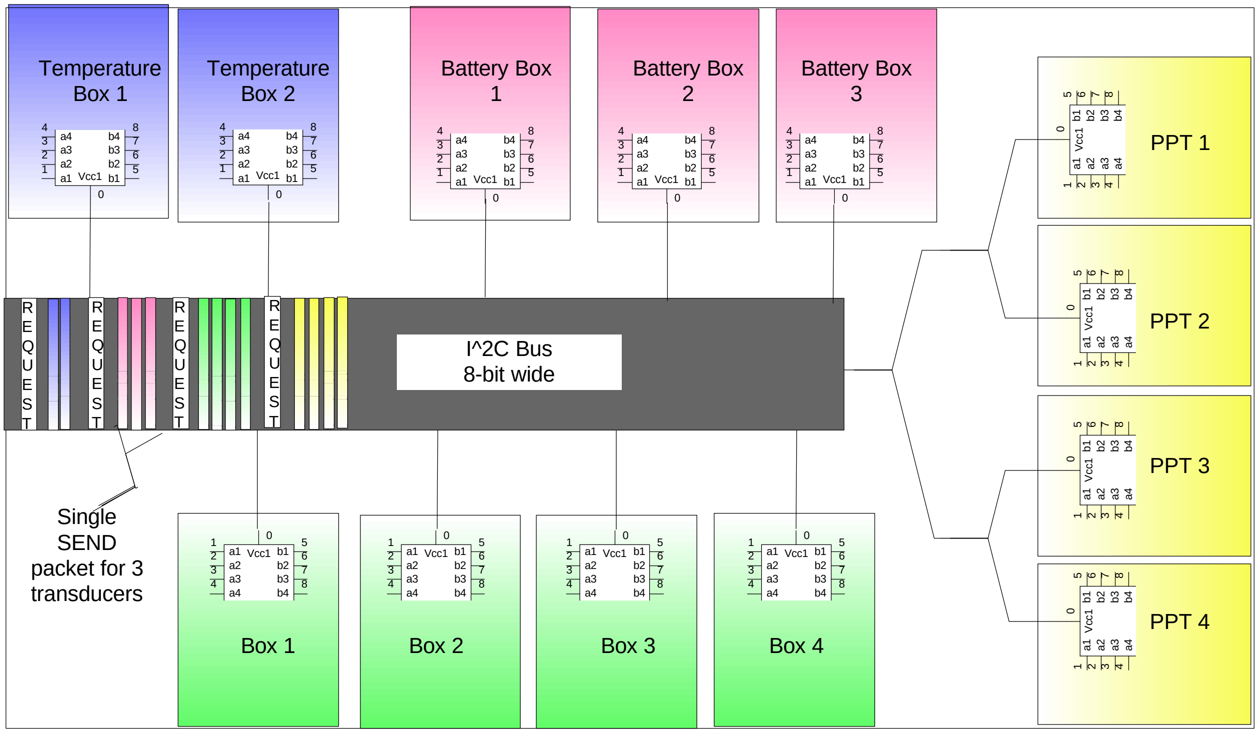


Figure 1: Sensor Boxes and the I^2C Bus.

The I^2C bus is a 2-line, bi-directional communication system between different devices. The design requires equal access to any of thirteen microcontrollers. The I^2C bus and the microcontrollers provide an efficient on-board data transmission system between the modulating device and the sensors.

There are several features of the I^2C bus that make it suitable for the solar car application. The bus consists of two lines; the serial data line (SDA) and the serial clock line (SCL). Stable data on the SDA line is valid during the HIGH period of the clock. Therefore, the data and the clock must be synchronized. The I^2C bus is a multi-master bus including collision detection and arbitration to prevent data corruption if more than one device initiates data transfer. The master, the microcontroller in control of the bus, generates the data and clock. Each master generates its own clock signal when transferring data on the bus.

An uplink connection is established when one of the sensor microcontrollers is the master and the modulating device is the slave. A downlink connection is established when the modulating device is the master and one of the sensor microcontrollers is the slave. The bus is capable of transferring 8-bit data at up to 100 kbits/sec in standard mode. This is appropriate for the 8-bit microcontrollers in the system. The packet structure has been designed so that the operator can monitor each sensor in the system. A Request Packet is generated at the operator interface and submitted via the RF link to the I^2C bus. The specified microcontroller then generates a Send Packet with the specified sensor data.

The Request Packet is used to request data from all sensors of a particular type at a specific box (see Figure 2). This packet is 4 bytes long. The first byte is the size of the packet. The ID byte contains the 8-bit bus address of the particular microprocessor desired.

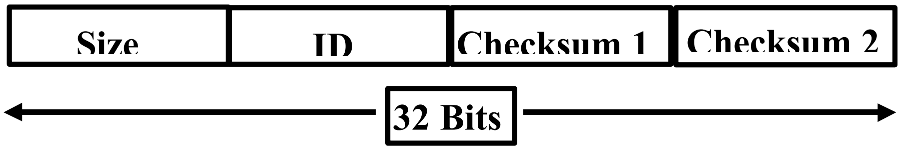


Figure 2: Request packet structure.

The Send Packet (Figure 3) is the response to the Request Packet. It is generated by the microcontroller once all of the data is collected. The data for each channel is a single byte containing the sensors current value. The total overhead is 4 bytes. A request to any location, excluding Box 1, would have a minimum Send Packet size of 6 bytes and a maximum size of 10 bytes. A request to Box 1 would have a minimum Send Packet size of 7 bytes and a maximum size of 19 bytes.

Figure 3: Send packet structure.

SENSORS

Exactly 62 sensors are distributed about the car. As can be seen from Table 1, temperature, voltage, strain, acceleration, and peak power are measured at various places on the car. Although one could easily choose to monitor many other system parameters, the designer must take care. Since the main approach is to let the chase car's computer model determine driving style, few parameters actually need to be sampled. Even more important is the fact that since power drain is at a premium, only the minimum number of components should actually implemented. The designer must weigh the benefits of added information against the additional drain on the solar car's system

	Number of Sensors	Bits per Sensor	Type
Temp Box 1	4	8	Array Temp
Temp Box 2	4	8	Array Temp
Battery Box 1	6	8	Battery Temp, Voltage
Battery Box 2	6	8	Battery Temp, Voltage
Battery Box 3	6	8	Battery Temp, Voltage
Box 1	12	10	DC-DC Converters: Voltage, Current, Temperature; Pack Battery: Voltage, Current; Motor Current
Box 2	4	8	Driver Temperature
Box 3	3	8	Strain gauges
Box 4	3	8	Modem: Battery Voltage and Current, Temperature
PPT 1	3	8	PPT 1 Voltage, Current, Temp
PPT 2	3	8	PPT 2 Voltage, Current, Temp

	Number of Sensors	Bits per Sensor	Type
PPT 3	3	8	PPT 3 Voltage, Current, Temp
PPT 4	3	8	PPT 4 Voltage, Current, Temp
Main Strut	2	8	Accelerometer

Table 1: Sensor descriptions and locations.

RF LINK

The radio frequency link has historically imposed many difficulties upon solar car race teams. Daedalus' original telemetry system has been plagued with signal-loss due to the high electromagnetic interference levels from the motor and poor gain of the monopole antenna being employed. A new lightweight and aerodynamic antenna has been designed which has four times the gain of the existing monopole. Additionally, a wireless modem with higher transmit power, lower standby power draw, and a special sleep mode further increases the link margin.

Because of the special relationship between the solar car and the chase car, it is safe to assume that the chase car is always in the rear hemisphere behind the solar car. This assumption is further safeguarded by the telemetry-on-demand organization of the telemetry system. If, by some strange arrangement, the chase car does temporarily end up in the forward hemisphere of the solar car, the request for data can be repeated upon return to the rear hemisphere. A more directive antenna design has the advantage of increasing antenna (and thus RF link) efficiency since the antenna is no longer wasting half of its power by radiating away from the receive antenna on the chase car.

A microstrip patch array antenna was designed for this application (see Figure 4). The microstrip patch architecture makes the antenna low profile and lightweight. This allows placement of the antenna upon the surface of the solar car with negligible effect on aerodynamics. The array geometry allows steering of the beam towards the chase car by implementing a progressive phase difference of the signal fed into adjacent patches. As can be seen in Figure 5, the gain is directed largely towards the rear of the solar car with its peak at approximately $\theta = -50°$. Very little of the antenna's energy is directed in the front hemisphere ($\theta > 0°$). The resulting gain is 8.4 dB, approximately 4.3 times more gain than the standard monopole.

An additional improvement in the RF link design was the addition of a more powerful and efficient radio modem. This modem has nearly 3 times the output power and a special sleep mode, which draws only 14 mA. Link budget calculations made with the new antenna/modem specifications showed that if the new antenna design is placed upon the solar car only, the resulting margin is nearly 6 dB better than the original design. If the new antenna is also added to the chase car, a 15 dB increase over the present configuration is realized.

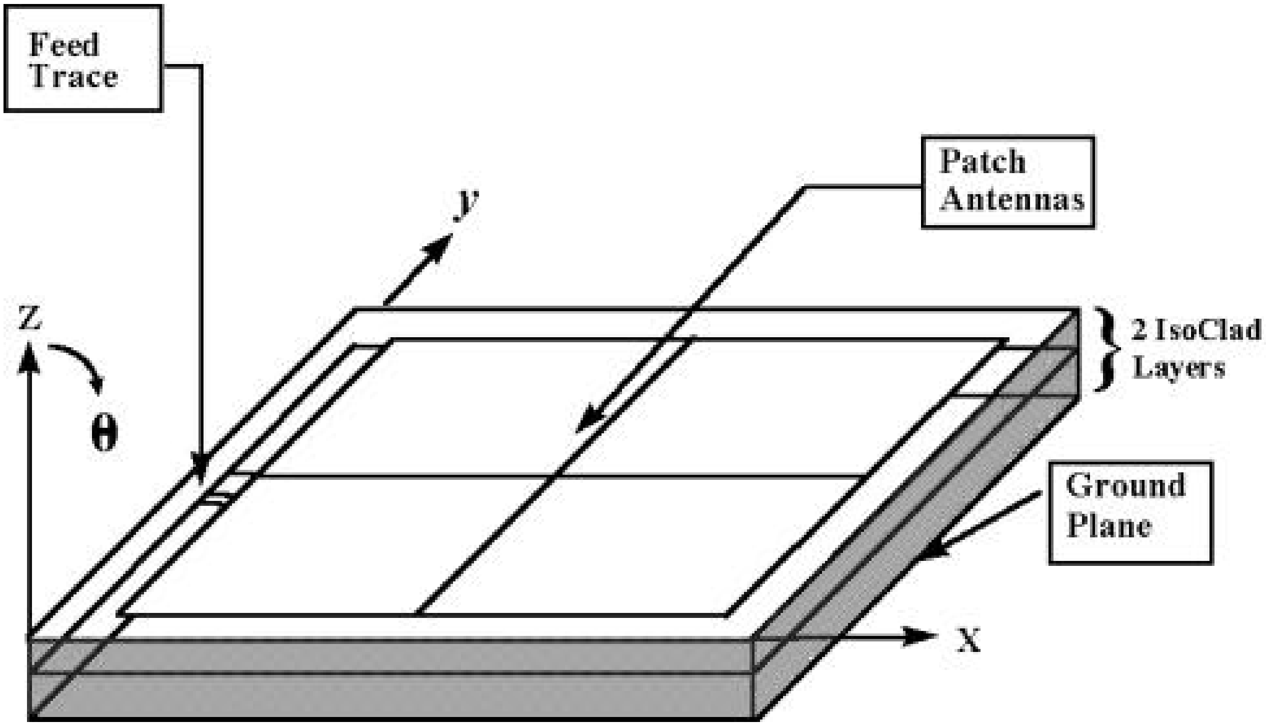


Figure 4: Microstrip Patch Antenna

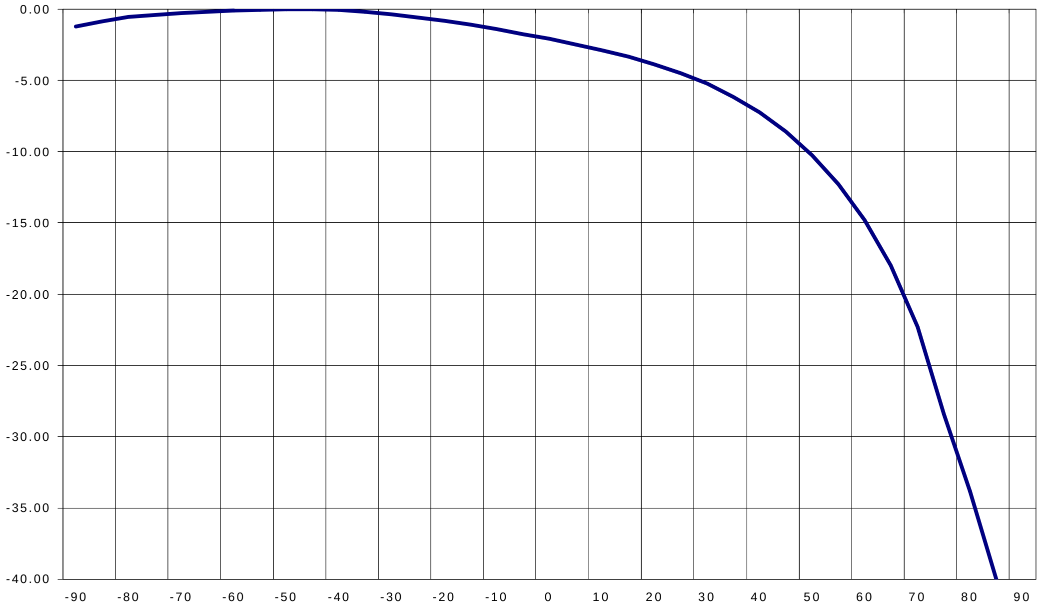


Figure 5: E-Theta plane normalized gain cut of antenna.

OPERATOR INTERFACE

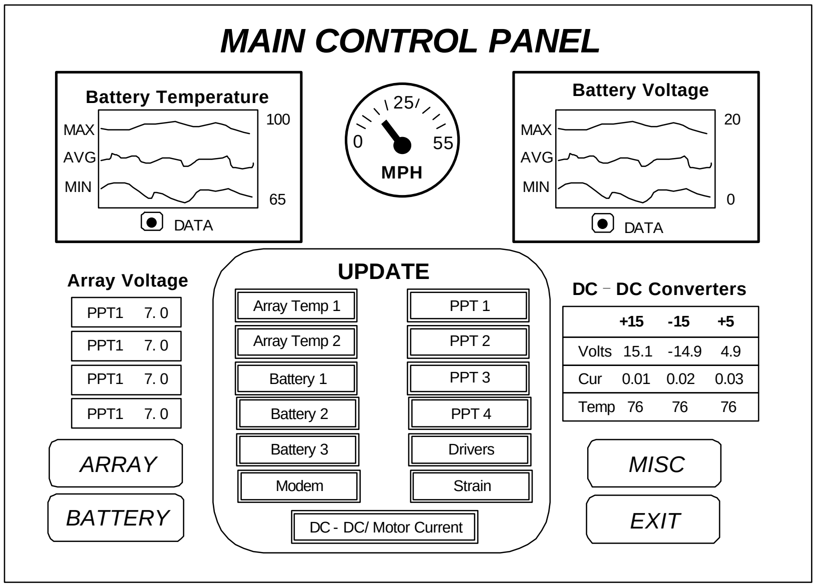
A telemetry processor is used to process the real time data that is coming from the solar car. This processor is also used as the ground station command processor. Since the processor is located in the chase car a portable laptop was also chosen. The laptop system is composed of a Pentium® II –300MHz with 64Mb of RAM and 4.3 Gig Hard Drive. The system would also run the Microsoft® Windows NT 4.0 operating system. The system can be run from AC power, battery, or a cigarette lighter.

LabView® 5.1, by National Instruments, is a software package written specifically for viewing and interpreting data in real time or not. The user interface can be customized to provide screens specific to your data needs. Data can be viewed in the form of tables, graphs, gauges and more. LabView also has the ability to send commands to the solar car.

The operator interface, in LabView®, was setup into four areas; the main control panel, array, battery and miscellaneous displays. Each display is customized to display information about its category. The Main Control Panel is the primary display (Figure 6). The control panel displays information that is crucial to the success of the solar car's operation. The display contains a histogram of the batteries average temperature and voltage, as well as the maximum and minimum temperatures and voltage of the individual batteries. A speedometer is displayed to correlate with possible changes in voltage or current. Array voltages of the four PPTs are displayed as well as voltage, current and temperature of the DC-DC converters. Each line of data coming in from a sensor has a maximum and minimum value associated with it that is defined in LabView and a audible alarm is programmed to be triggered if the values extend past these limits, thereby eliminating clutter on the main display. If more info is desired on the batteries or arrays then the user can be taken to that display at the touch of a button. Another key feature that is found on the main control panel is the ability to request an update in data from any box of sensors. At the touch of a button, a command is sent to a particular box of sensors requesting a reading. Also, throughout the interface, all histograms have a data button that will display past data values.

The Array display contains more detailed information about the array's condition. This detailed information includes sensor readings from each individual sensor relating to arrays. Through the histograms, one can view current conditions and past history of array temperature, voltage, current, and PPT temperature. This information can be used to verify that all arrays are functioning similarly or notice if one (or more) may be malfunctioning.

The Battery display provides a more detailed representation of data collected on the batteries. Similar to the array display, one can view the battery temperature or voltage for each of the individual batteries. This, as with the array display, can be used to observe if the batteries are functioning similarly or if one (or more) are malfunctioning. One can also view the battery pack current and voltage data.



PPT1 7. 0
PPT1 7. 0
PPT1 7. 0
PPT1 7. 0

	+15	-15	+5
Volts	15.1	-14.9	4.9
Cur	0.01	0.02	0.03
Temp	76	76	76

Figure 6: Main control panel.

The miscellaneous display contains information about all other areas. Displayed are gauges showing strain amounts from the three sensors, a table showing modem's temperature, current and voltage, driver temperature and a histogram of motor current. As mentioned before, if one of the sensor values falls out of predefined limits on any of these sub-displays, an audible alarm will sound to alert the operator. Also, every display has button located near the bottom of the display that allow the operator to easily switch between displays.

CONCLUSION

The telemetry system for the University of Arizona's Daedalus solar car has several features which could prove beneficial in many other telemetry applications. The telemetry-on-demand nature of this system is much more power-efficient than a traditional continuous stream system. The distributed topology of the microprocessors enables system redundancy and flexibility. A more directive microstrip patch array antenna system increases link robustness while being more aerodynamic. A customized operator interface enables easy analysis of system status and quick response to degrading conditions.

ACKNOWLEDGEMENTS

The authors would like to thank the designer of the Daedalus solar car's original telemetry system, Rai Ramadorai, for all of his help and patience. Beginning our design work with such a well-conceived system was a pleasure. Thanks also to Dr. Larry Schooley for guiding us during this challenging, educational, and enlightening project.

REFERENCES

1. Horan, Stephen, *PCM Telemetering Systems*, CRC Press, Boca Raton, FL.
2. Pozar, David M., *Microwave Engineering*, John Wiley & Sons, NY, NY.
3. Balanis, Constantine A., *Antenna Theory*, John Wiley & Sons, NY, NY.
4. *I^2C Bus specification sheet*, Phillips Inc.
5. *8-bit CMOS Microcontroller specification sheet*, Microchip Inc.
6. *LabView 5.1 Manual*, National Instruments Inc.