Distant Monitoring and Control for Mobile Robots Using Wireless Sensor Network

*** János Simon, ** Goran Martinović**

* Subotica Tech, Department of Informatics, simon@vts.su.ac.rs

** Faculty of Electrical Engineering, J.J. Strossmayer University of Osijek, goran.martinovic@etfos.hr

Abstract — This article presents a solution for distant monitoring and control of the Boe-bot mobile robot via internet with real time video stream from the robot's workspace. It has a web based user interface with possibility of managing the actuators on the robot. Also allows to log events on the system like acceleration, temperature, light etc.

Keywords – distant monitoring, Sun SPOT, WSN, Mobile robot, Web

1 Boe-bot

The Board Of Education is a complete, low-cost development platform. The Boe-Bot is a great tool with which to get started with robotics.

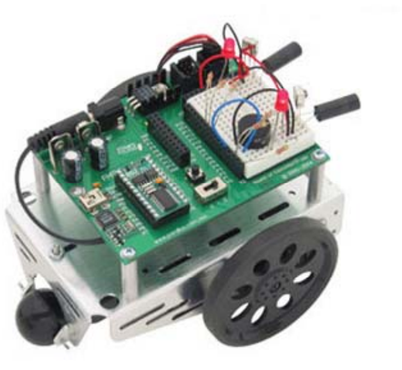

Figure 1. Assembled Boe-Bot

The WSN module makes it possible for the Boe-Bot robot's BASIC Stamp 2 microcontroller brain to communicate wirelessly with a web based user interface running on a nearby PC. The BASIC Stamp microcontroller runs a small PBASIC program that controls the Boe-Bot robot's servos and optionally monitors sensors while it communicates wirelessly with the web server.

2 Wireless Sensor Network (WSN)

A wireless sensor network (WSN) is a computer network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations. The development of wireless sensor networks was originally motivated by military applications such as battlefield surveillance. However, wireless sensor networks are now used in many civilian application areas, including environment and habitat monitoring, healthcare applications, home automation, and traffic control.

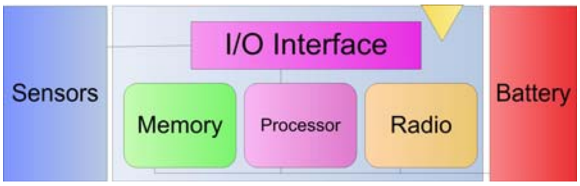

Figure 2. Sensor Node Architecture

In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and an energy source, usually a battery.

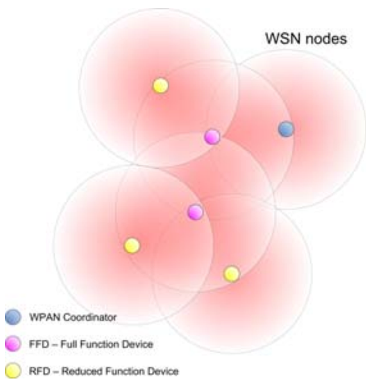

Figure 3. Typical wireless sensor network (WSN)

The size a single sensor node can vary from shoebox-sized nodes down to devices the size of grain of dust. The cost of sensor nodes is similarly variable, ranging from hundreds of dollars to a few cents, depending on the size of the sensor network and the complexity required of individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and bandwidth. In computer science, wireless sensor networks are an active research area with numerous workshops and conferences arranged each year.

3 Applications

The applications for WSNs are many and varied. They are used in commercial and industrial applications to monitor data that would be difficult or expensive to monitor using wired sensors. They could be deployed in wilderness areas, where they would remain for many years (monitoring some environmental variable) without the need to recharge/replace their power supplies. They could form a perimeter about a property and monitor the progression of intruders (passing information from one node to the next). There are a many uses for WSNs.

Typical applications of WSNs include monitoring, tracking, and controlling. Some of the specific applications are habitat monitoring, object tracking, nuclear reactor controlling, fire detection, traffic monitoring, etc. In a typical application, a WSN is scattered in a region where it is meant to collect data through its sensor nodes.

4 Wireless Radio

The CC2420 is housed in a 48pin quad leadless package (QLP or QFN) that is 7mm square. It is powered with +3.0V Vcc supply. The CC2420 has an internal 1.8V low drop out regulator for powering the internal RF and analog circuitry. It consumes 20ma during receive operation and 18ma for 0dBm transmit. The frequency generation uses an accurate 16MHz crystal with ±10ppm accuracy, ±10ppm stability and ±1ppm aging. The entire RF section is enclosed in an upper and lower RF shield and has modular FCC approval.

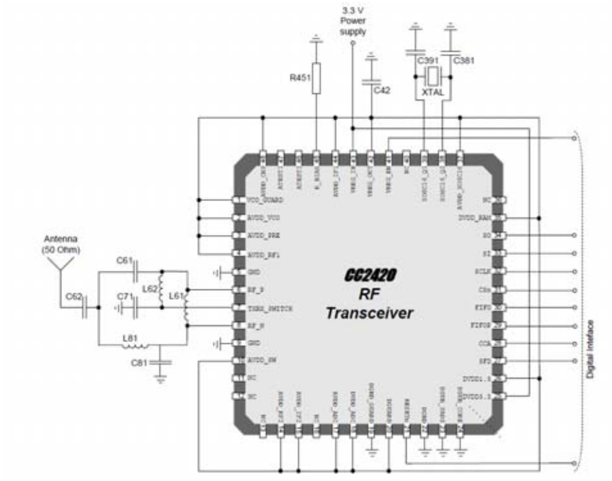

Figure 4. Typical application circuit for CC2420

The wireless network communications uses an integrated radio transceiver, the TI CC2420 (formerly ChipCon). The CC2420 is IEEE 802.15.4 compliant and operates in the 2.4GHz to 2.4835GHz ISM unlicensed bands. Regulations for these bands are covered by FCC CFR47 part 15 (USA), ETSI EN 300 328 and EN 300 440 class 2 device (Europe) and ARIB STD-T66 (Japan).

5 I/O pin Manipulation

The Atmega88 operates the 8 tricolor LEDs, the accelerometer configuration, and the following pins on the I/O header: I/O pins D0 through D3 can be set as either an output or input. The high current driver pins, H0 to H3, can only be used as an output. If configured as an output, the pin may be set hi, low, or toggled to its opposite state.

6 The Sun SPOT Emulator

Solarium includes an emulator capable of running a Sun SPOT application on your desktop computer. This allows for testing a program before deploying it to a real SPOT, or if a real SPOT is not available. Instead of a physical sensor board, Solarium displays a virtual SPOT with a control panel where you can set any of the potential sensor inputs (e.g. light level, temperature, digital pin inputs, analog input voltages, and accelerometer values). Your application can control the LEDs' color that is displayed in the virtual SPOT image, just like it would a real SPOT. You can click with the mouse on the push button switches in the virtual SPOT image to press and release the switches. Receiving and sending via the radio is also supported. Each virtual SPOT is assigned its own address and can broadcast or unicast to the other virtual SPOTs. If a shared base station is available a virtual SPOT can also interact over the radio with real SPOTs.

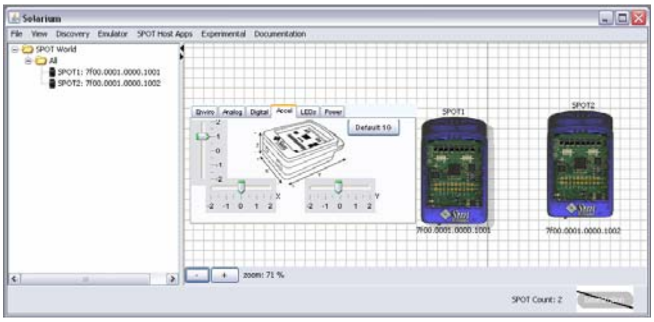

Figure 5. The Sun SPOT Emulator

Virtual SPOTs can communicate with each other by opening radio connections, both broadcast and point-to-point. Instead of using an actual radio these connections take place over regular and multicast sockets. When a base station SPOT is connected to the host computer and a shared base station is running, virtual SPOTs can also use it to communicate with real SPOTs using the base station's radio. The advantage of using a shared base station is that multiple host applications can then all access the radio. One disadvantage is that communication from a host application to a target SPOT takes two radio hops, in contrast to the one hop needed with a dedicated base station. Another disadvantage is that run-time manipulation of the base station SPOT's radio channel, pan id or output power is not currently possible. Each virtual SPOT has its own Squawk VM running in a separate process on the host computer. Each Squawk VM contains a complete host-side radio stack as part of the SPOT library, which allows the SPOT application to communicate with other SPOT applications running on the host computer, such as other virtual SPOTs, using sockets or real SPOTs via radio if a shared base station is running.

The current Solarium implementation is primarily an emulator since it actually runs a SPOT application in a Squawk VM, just like the VM on a real SPOT. Likewise radio interaction between virtual SPOTs is emulated with data sent via packets and streams from one (virtual) SPOT to another. Only the SPOT's interaction with the environment is simulated using a simple model where the user needs to explicitly set the current sensor values. Future versions may incorporate more simulation of SPOT properties like battery level or radio range.

7 Solution

In this project we have used SunSPOT-s to achieve remote control over a Boe-Bot. For this project we have used 2 SunSPOT-s from the kit (free range and base station module). SunSPOT's wireless protocol is Zigbee based protocol.

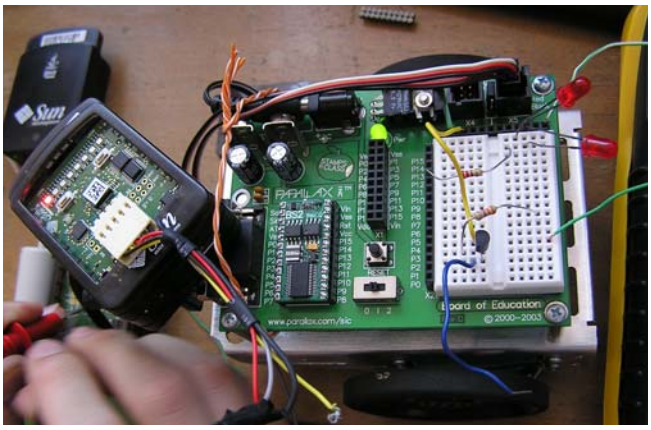

Figure 6. Process of development

We used the SunSPOT base station to read a file from the controlling computer and send its contents to the second free range SPOT. The second SunSPOT controls the motion of the mobile robot depending on the received data. The free range SunSPOT outputs control the speed of the wheels individually.

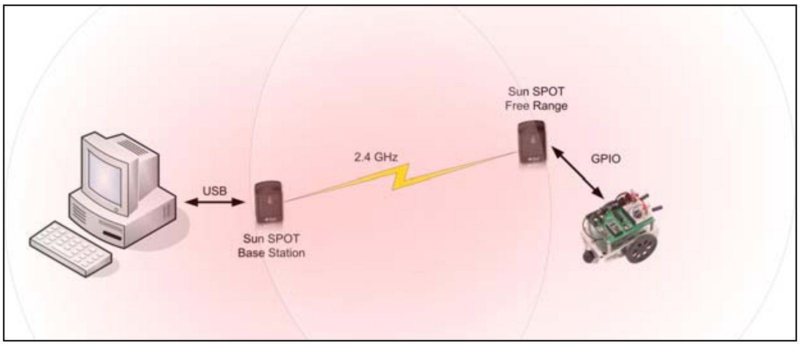

Figure 7. Connection of the system

The Hardware basically centers around Sun SPOT and DC Motors controlled by Basic Stamp. The Sun SPOT base station will send data to Sun SPOT on car which will drive the Basic Stamp controller to DC IO pins. The microcontroller will drive the Motors which will run the car.

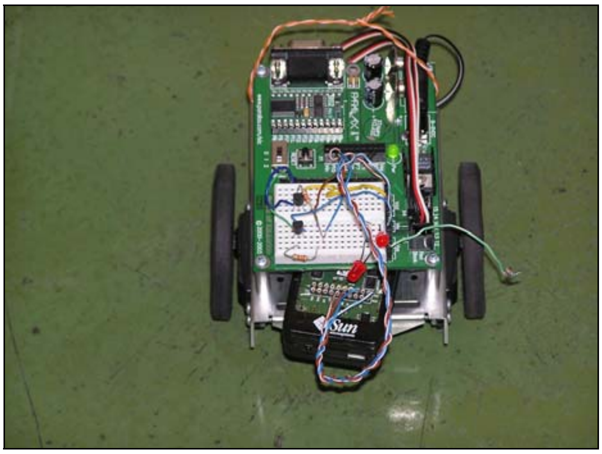

Figure 8. Boe-bot with SunSPOT mounted

Users can watch the real time videos provided by the web camera. With obstacles present in the unknown environment, the mobile robot reacts based on both the sensed information of the obstacles and the relative position of the target.

Figure 9. The web interface

In moving towards the target and avoiding obstacles, the mobile robot changes its orientation. When the obstacle in an unknown environment is very close, the mobile robot slows down and rapidly changes its orientation.

The web surface is simple and transparent, the mobile robot can be easily controlled. The image of the web cam can be seen on the internet, thus the user has a visual image of the mobile robot. For the display of the real time video streaming the web browser is also of vital importance, as it can be only accessed via the internet.

Conclusion

The field of wireless sensor networks offers a rich, multi-disciplinary area of research, in which a variety of tools and concepts can be employed to address a diverse set of applications. As such, many potentials of this field have been under study both in academia and in the industry. Robotics has come a long way, especially for mobile robots. In the past, mobile robots were controlled by heavy, large, and expensive computer systems that could not be carried and had to be linked via cable or wireless devices. Today, however, we can build small mobile robots with numerous actuators and sensors that are controlled by inexpensive, small, and light embedded computer systems that are carried on-board the robot. Building and programming a robot is a combination of mechanics, electronics, and problem solving. What you're about to learn while doing the activities and projects in this text will be relevant to "real world" applications that use robotic control, the only difference being the size and sophistication. The mechanical principles, example program listings, and circuits you will use are very similar to, and sometimes the same as, industrial applications developed by engineers.

References

[1] Sun™ Small Programmable Object Technology (Sun SPOT) Owner’s Manual Release 3.0, Sun Microsystems, Inc. 2007

[2] Sun Spot Developer’s Guide. Sun Microsystems. 2005

[3] Demo Sensor Board Library. Sun Microsystems. 2005

[4] Sergio Scaglia, „The Embedded Internet”, Addison-Wesley 2008

[5] James Gosling, „ The Java™ Language Specification Third Edition”, Addison-Wesley 2005

[6] Matijevics István, Simon János, „Advantages of Remote Greenhouse Laboratory for Distant Monitoring”, Proceedings of the Conference ICoSTAF 2008, pp 1-5, Szeged, Hungary, 2008

[7] Simon János, Matijevics István, „ Distant Monitoring And Control For Greenhouse Systems Via Internet”, Zbornik radova konferencije Yuinfo 2009, pp. 1-3, Kopaonik, Srbija, 2009

[8] Matijevics István, Simon János, „ Comparison of various wireless sensor networks and their implementation”, Proceedings of the Conference SIP 2009, pp 1-3, Pécs, Hungary, 2009