DEPARTMENT OF
COMPUTER SCIENCE
TEXAS A&M UNIVERSITY

CPSC 483: Senior Design

Project Proposal

Wireless Robot Control with a Personal Digital Assistant

Ta-Chang Chao

Quang Dang

Matt Dean

Brady Wied

February 5, 2003

Table of Contents

Summary

This proposal details the project plan for ‘Wireless Robot Control with a PDA.’ This will allow a user with a handheld PDA to control a robot and provide a framework for developing useful applications for this technology. The interface will allow the user to control the vehicle with a number of basic motor primitives, and also receive information from sensors on-board the vehicle. Our design will be able to operate in high radio frequency environments where wireless Ethernet designs may not operate properly, and Bluetooth technology is our communications choice because of its frequency hopping capability. Bluetooth also uses much less power than the wireless Ethernet counterpart and provides cost effective adapters for both the robot and the handheld device. Due to its fast processor and memory, our project uses the Compaq iPaq, with the Pocket PC operating system, for the handheld device. As previously mentioned, our design will provide a framework/API that can be used to build different applications with the functionality that we implement. Our particular project will demonstrate a mapping and navigation application.

Introduction

Remote control of robots is something that is increasingly utilized in research and production environments. Many wireless controls exist already, such as RF or IEEE 802.11, but these solutions sometimes have difficulty in noisy radio environments, can be costly, and are not very energy efficient. A better solution is needed that will allow use with a short range PDA in such an environment. Several problems could be solved by this approach. One example would be to allow corrective distance measuring between the robot's concept of its location and the actual position. This could be used to map some given territory. The solution must incorporate several features, including a mechanism to shut down the robot when communication is lost, a sophisticated information display on the PDA device, and an open source software development kit that could be used to develop further applications for this solution. The ability to produce a well-documented API for the PDA device will be crucial to the success of the project, and for others to build on what we have learned. We are not concerned with long-range robot communication, nor is this solution targeted to application capabilities that are beyond the resources a PDA can provide. The niche of this particular solution is for a cost effective robot that can perform these duties without much user education required.

Our information has been obtained from a variety of sources, including robot vendor web sites, Bluetooth specification books, Pocket PC websites, and interface module websites. In the following sections, we outline our solutions for solving the given problem, design constraints, alternative solutions, budget concerns, scheduling, and environmental/safety concerns.

Key Terms

ARIA (ActivMedia Robotics Interface Application):

An object-oriented API designed to communicate with ActivMedia robots.

Bluetooth:

A wireless communications protocol originally developed by Ericsson. The Bluetooth Special Interest Group (SIG) now controls the Bluetooth standard. While the standard is relatively new on the market, it has gained industry acceptance. Many companies have Bluetooth projects in the works.

Piconet:

A group of Bluetooth devices joined together in a short-range network. Each piconet is synchronized with each other's timing and frequency hopping sequence.

Protocol stack:

A layered set of functional units that will implement the Bluetooth protocol. Each layer in the stack has specific duties and responsibilities.

RFCOMM:

One of the layers of the Bluetooth protocol stack that provides RS-232 serial cable emulation.

Project Tasks

The process of creating an interface between a PDA and a mobile robot using Bluetooth technology requires several distinct tasks which are outlined below, in no particular order.

Task 1: Interface the robot to a personal computer using a direct serial connection to provide a test-bed for programming the operation and monitoring of the robot.

Task 2: Program the robot to interact with a map of a building, providing feedback indicating where it thinks that it is currently located in relation to the map.

Task 3: Program the PDA so that it is able to interact with the robot.

Task 4: Develop a user interface for the PDA to control and monitor the robot.

Task 5: Define a format for creating, viewing, and interacting with maps on the PDA.

Task 6: Link the PDA to the robot using Bluetooth communication devices.

Task 7: Implement safety features for the robot.

Task 1: Interface the robot to a personal computer using a direct serial connection to provide a test-bed for programming the operation and monitoring of the robot.

Although the final system that we are creating will use wireless communication to control the robot, much of the programming that will be done is independent of the communication method used. To facilitate the programming of the robot and validate that it is operable, we will connect it to a PC using a direct serial connection. The robot that we are going to use for the project comes with a C++ API called ARIA that we can use for interaction. This task will allow us to learn the basics of controlling the robot using the API and we will familiarize ourselves with the operation and limitations of the robot.

Task 2: Program the robot to interact with a map of a building, providing feedback indicating where it thinks that it is currently located in relation to the map.

Once initial control of the robot has been established and demonstrated, we can extend the functionality of the robot by programming it to interact with a map (such as a floor plan). This task will be closely coordinated with Task 5 because the format of the map is going to depend on what type of information the robot needs to have in order to interact with it. The robot will also need to be able to correct its estimate based on feedback from the user.

Task 3: Program the PDA so that it is able to interact with the robot.

One of the obstacles that we are going to face is the porting of our robot control application to the operating system of the PDA, which is likely going to be Pocket PC. The Pocket PC has several tools that will assist us, including an SDK with Visual C++. Initially we will probably program the PDA using a simulator, and interact directly with the robot using the serial connection between the robot and the computer (see Task 1). We will start off with basic commands to test the interaction between the PDA and the robot before creating a user interface in Task 4.

Task 4: Develop a user interface for the PDA to control and monitor the robot.

The majority of the interaction between the user of the robot and the robot itself will occur through the user interface of the PDA, so a good bit of attention will need to be paid to usability and design. The primary components of the interface will be a graphical joystick for control of the robot, a small graphic indicating the current status of each of the robot's sonar sensors, and a large window showing the current map that the robot is using, along with the location of the robot on the map. The map display will also allow the user to indicate where the robot actually is in relation to where it thinks it is.

Task 5: Create a format for creating, viewing, and interacting with maps on the PDA.

In order to validate our design and provide a useful tool for the robot, we will create an application for guiding the robot through a map. The application will indicate where the robot thinks it is and allow the user to provide correction to indicate where it actually is. This will necessitate the creation of a format for creating, viewing, and interacting with the maps using the PDA. The maps will need to indicate the path of the robot and allow for zooming and panning.

Task 6: Link the PDA to the robot using Bluetooth communication devices.

As the brain for the robot, the PDA will need to be able to communicate with the microcontroller to control and monitor the robot. To allow this to be done without a direct cable connection between the devices, we will interface them using Bluetooth. Hopefully we will be able to connect a Bluetooth to RS232 adapter directly to the robot and modify the ARIA API to use a virtual serial port created by the Pocket PC's integrated Bluetooth capabilities.

Task 7: Implement safety features for the robot.

We will implement several safety features for the robot such as an instant shut-off switch and automatic shutdown upon loss of connection.

Design Justification

Several choices were made during the specification of this project that warrant further justification. The first major decision was to use Bluetooth rather than other popular wireless communications systems such as WLAN (802.11). The second decision was to use a Pocket PC rather than a Palm PDA. We feel that both of these decisions are justified in the context of this project for the following reasons.

Bluetooth vs. 802.11

The market currently contains several devices and protocols for providing wireless connectivity, but the two most popular are Bluetooth and 802.11, also known as WLAN. We compared and analyzed both of these solutions based on several important characteristics: bandwidth, power, cost, range, and ease of interface.

Bandwidth:

Both 802.11 and Bluetooth devices are able to simultaneously transmit and receive amounts of data sufficient for our purposes, with data transfer speeds ranging from 0.5 to 1.5 megabits per second (Mbps) for Bluetooth and up to 54 Mbps for 802.11. Although WLAN devices are able to transmit significantly larger amounts of data than Bluetooth, the bandwidth necessary for our purposes is well within the range provided by Bluetooth. The bandwidth requirements therefore do not give preference to either type of device.

Power:

We next compared the power requirements of each device. This is important for two reasons. First, energy use always adds cost to a device. Secondly, the nature of the project dictates that the communication devices will need to operate on battery power, and battery life is an important consideration. The power requirements of 802.11 and Bluetooth vary depending on the implementation details and antenna powers for specific chipsets, but in all cases the power required by Bluetooth is ten to a hundred times less that that consumed by WLAN. Handheld devices such as PDAs that use WLAN components often require external batteries whereas Bluetooth chips can run in a phone, PDA, or other small device without quickly draining its battery power. Bluetooth also has a much more efficient idle state that allows it to conserve energy when it is not actively being used. The advantages of Bluetooth over 802.11 are quite conclusive in the context of power.

Cost:

Another important point on which WLAN and Bluetooth can be compared is the cost of the components necessary to provide communication between a handheld device and a robot. The cost for a Bluetooth transceiver varies depending on the specific implementation, but it ranges from approximately $10 to $50. Prices for WLAN chipsets are typically around $25. A quick search for adapters for PDAs such as those created for Pocket PCs cost $160 for Bluetooth and over $250 for 802.11. This leads to the conclusion that, though the chipsets for both types of devices are fairly similar, the adapters required to interact with them using a handheld device are more than twice as expensive for an 802.11 solution as they are for a comparable Bluetooth solution.

Range:

One of the main areas in which 802.11 and Bluetooth devices differ is in the physical range in which they can communicate. 802.11 transceivers can communicate at ranges of hundreds or even thousands of feet, but Bluetooth transceivers are only operable within ranges of approximately 30 feet. This is currently a major limitation of Bluetooth technology, but this distance is sufficient for the purpose of our project.

Ease of Interface:

A majority of the development effort to be expended on this project is likely to be consumed interfacing the communication components with the handheld device and the robot microcontroller. In this area, Bluetooth appears to have a significant advantage primarily due to the fact that it is specifically designed to operate on devices that are likely to use a microcontroller. Bluetooth is also geared towards PDAs, so the interface for the handheld device should be fairly standardized and have adequate APIs and other programming aides.

Conclusion:

In conclusion, the benefits provided by Bluetooth components seem to outweigh those found in 802.11 devices. Bluetooth has a significant advantage in the realm of cost, power use, and ease of interface, which, in the context of this project, WLAN cannot compensate for with its superior range and bandwidth. This project, therefore, will use Bluetooth technology to provide communication between the handheld device and the robot.

Pocket PC vs. Palm

The two most likely choices for the PDA to be used as the brain for the robot were Palm and Pocket PC (Compaq iPaq). The main areas in which these two options differ is in their processing power and memory.

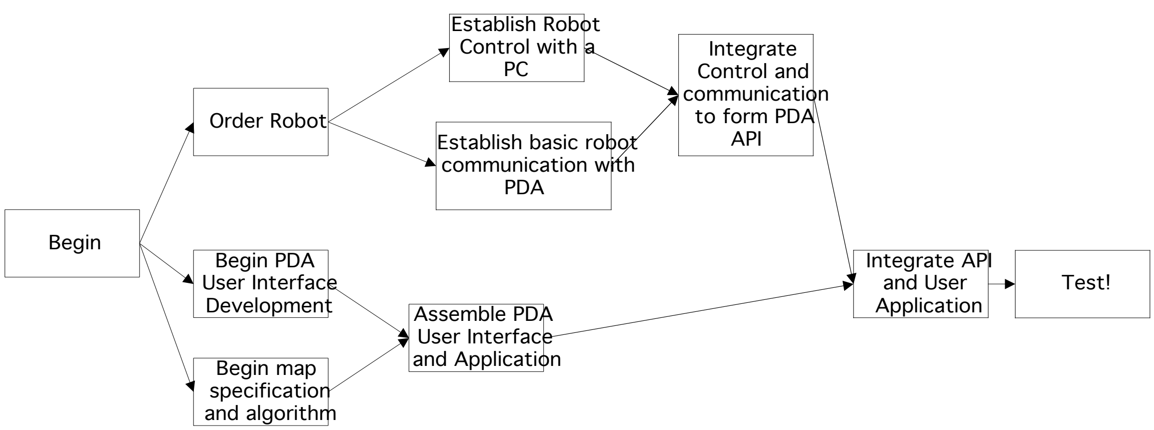
Processing Power

Differentiating between the Pocket PC and the Palm concerning their processing power is relatively easy. The Palm line of PDA's uses processors ranging from the 33 MHz Dragonball processor used in the m130 to the 144 MHz ARM processor in the Tungsten T. Compaq's iPaq, however, uses a 400 MHz ARM processor for superior processing power.

Memory

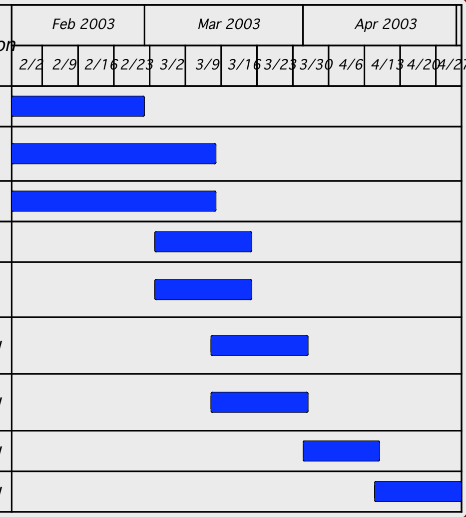
The difference between the Pocket PC and the Palm in regards to memory is also substantial. The Pocket PC comes with 64MB RAM and 48MB ROM, and the Palm has between 8MB and 16MB, depending on the model.

Project Schedule



PERT Chart 1

ID	Task Name	Start	Finish	Duration
1	Order Robots / Parts	2/3/2003	2/28/2003	4w
2	Begin PDA User Interface Development	2/3/2003	3/14/2003	6w
3	Begin map specification and algorithm	2/3/2003	3/14/2003	6w
4	Establish Robot Control with a PC	3/3/2003	3/21/2003	3w
5	Establish basic robot communication with PDA	3/3/2003	3/21/2003	3w
6	Assemble PDA User Interface and Application	3/14/2003	4/1/2003	2.6w
7	Integrate Control and Communication to form PDA API	3/14/2003	4/1/2003	2.6w
8	Integrate API and User Application	4/1/2003	4/15/2003	2.2w
9	Test	4/15/2003	5/1/2003	2.6w



GANTT Chart 1

Budget

IPAQ 3975 (including shipping)	$630
Bluetooth RS232 Adapter	$200
Total	**$830**

After much consideration of the methods of solution to implement the project and the available devices out there in the market, we finally come to a unanimous conclusion: iPAQ 3975 and the Bluetooth RS232 adapter.

IPAQ 3975 is a pocket PC with integrated Bluetooth wireless technology. It consists of a 400MHz Intel® XScale Processor, containing 64MB SDRAM with 48MB Flash ROM. The reason we choose this model instead of other models such as Palm m130 or Palm m500 is because it has a more powerful processor and much more memory that we figure it might be needed for use of future expansion and integration to more complex features. Also the Bluetooth wireless technology has been integrated with the iPAQ 3975, making it easy and convenient to use. We also consider using other available but cheaper models of the iPAQ such as iPAQ 1910 but unfortunately the Bluetooth wireless pack is not compatible with it.

On the robot side, we decide to use the Bluetooth RS232 adapter from LinTech. This Bluetooth transceiver features a Wireless V.24/RS232 connection to a PC, modem or other equipment with serial connection. The RS232 Adapter is compatible with any application using a serial cable. It is a Plug and Play product that is independent of the operating system environment.

Societal, Safety and Environmental Analysis

With the implementation of the wireless control robot using a Bluetooth enabled Personal Digital Assistant, many people from different walks of life would gain several benefits. For example, a geologist who wishes to collect some lava samples near volcanic areas can pull out his PDA and wirelessly control his robot to collect samples without carrying a heavy laptop; a bomb squad may use a PDA-controlled robot to disarm a bomb without having to go near it. With this technology, the useful applications are numerous and people in the society will be greatly benefited by it.

As a safety measure, our team is going to put an emergency stop button on the robot just in case the robot went out of human control. The emergency stop button will cut the power supply to the robot and cause it to stop. We are also going to have an emergency stop button on our PDA's GUI so that anyone controlling the robot can stop it at any time.

As an additional safety measure, the robot will regularly check to make sure that it is still communicating with the PDA and, if not, will shut itself off. This ensures that the robot will not be wandering around and cause harm to other people.

One concern that has been raised concerning the safety of Bluetooth was whether it produced significant amounts of harmful radiation. Because Bluetooth uses the frequency spectrum in the range of 2400 MHz to 2483.5 MHz, including the same natural frequency to excite water molecule in microwave ovens, it has been questioned whether it is harmful to use it around humans. However, a 1mW Bluetooth radio only emits $1/1{,}000{,}000^{th}$ the amount of power of a 1 KW microwave oven and the radiation is directed outward rather than focused inward so that it does not cause any significant harmful effect on humans in the operating vicinity.

A main environmental concern with the Bluetooth technology is whether it can co-exist under the same environment as the 802.11 Wireless LAN technology. According to a study done by Forrester Research, the two technologies can co-exist with minimal problems. Bluetooth devices operate in the open 2.4 GHz band.

Although other devices also use this band, Bluetooth counters this problem by employing a fast frequency-hopping scheme together with error protection and correction.

For security, Bluetooth employs a high speed, pseudo-random frequency-hopping algorithm. This makes it very difficult to listen in on a Bluetooth connection. It also uses a strong contemporary cipher algorithm called SAFER+, which generates 128-bit cipher keys from a 128-bit plain text input for link encryption and authentication.

Team Qualifications

We believe our team has the skill and experience to do this project. Each of us brings to the team talents and strengths that, when combined, help us to work effectively and efficiently.

Ta-Chang Chao has taken a CPSC 462 Microcomputer System class. The organization and structure of that class is similar to CPSC 483 Computer System Design where the majority of the class time is dedicated to one big semester project. In the 462 class, he was involved in writing programs in C that interact with other team members' and other groups' programs to read sensory data on a Microsoft microcontroller board and display it on the monitor screen. The scope of that project is similar to what the team is doing this semester where we want to be able to control the robot and transmit its sensor data back to the PDA's screen.

Quang Dang also took CPSC 462 Microcomputer System class and worked on the same kind of project that Ta-Chang Chao mentioned above. He has been writing in C++ since doing projects for the CPSC 211 and CPSC 311 classes.

Matt Dean is an experienced software developer who has been involved in several large-scale design projects, primarily dealing with information architecture issues and Internet application development. He has taken several courses at Texas A&M which should add insight into this project, including Real Time Systems, Software Engineering, and Operating Systems. He is also currently taking a course in Microcomputer Systems. For the past three years he has owned a software development firm, Trabian Technology, and has been project manager for several successful products such as the Texas A&M Corps of Cadets website (http://www.aggiecorps.org) and the Aggie Band site (http://band.tamu.edu). These past experiences should be adequate preparation for design and implementation of this project.

Brady Wied has completed several related projects. He has interned with Neiman Marcus, Silicon Graphics Inc, and the accounting firm Whitley, Penn, and Co. He completed CPSC 462 last semester and dealt with embedded systems on the Motorola Coldfire platform. He has managed a software engineering project and also has developed and led a conference registration system for COSGA (Conference on Student Government Associations). He is currently working on Java Servlets and web-based programming in his other courses.