codeBLUE: A Bluetooth[1]
Interactive Dance Club System

Dennis Hromin, Michael Chladil, Natalie Vanatta, David Naumann

Dept of Computer Science,

Stevens Inst of Technology, Hoboken, NJ 07030

Farooq Anjum, Ravi Jain

Applied Research, Telcordia Technologies

Morristown, NJ 07960

CS Report 2002-1

Abstract

Collaborative music creation can be one of the most powerful forms of human communication. Playing in private improvisational sessions or at public concerts exhilarates musicians, due to the exchanges of emotional and intellectual information inherent in such creative settings. While many people express themselves musically by whistling and humming, relatively few have enough confidence in their musical abilities to collaborate openly with others. An interactive dance club system acts as a simplified musical interface. By transforming simple dance movements into musical modifications, untrained performers can collaborate in ways previously reserved for seasoned professionals.

This report presents the creation of a Bluetooth wireless dance club system, called codeBLUE. Although this system incorporates aspects of previous interactive dance and music systems, its wireless implementation based on Bluetooth technology sets it apart, as practical viability in a realistic dance club setting becomes possible.

codeBLUE: A Bluetooth
Interactive Dance Club System

Dennis Hromin, Michael Chladil, Natalie Vanatta, David Naumann
Dept of Computer Science,
Stevens Inst of Technology, Hoboken, NJ 07030

Farooq Anjum, Ravi Jain
Applied Research, Telcordia Technologies
Morristown, NJ 07960

30 MARCH 2002

Abstract

Bluetooth is a low-cost, low-power wireless technology designed primarily as a short-range cable replacement. Its applications include, for example, "cutting the cord" between portable stereos and headphones. This paper examines the use of Bluetooth for a collaborative music creation application where the low cost and small dimensions of Bluetooth technology are critical.

Collaborative music creation can be one of the most powerful forms of human communication. Playing in private improvisational sessions or at public concerts exhilarates musicians, due to the exchanges of emotional and intellectual information inherent in such creative settings. While many people express themselves musically by whistling and humming, relatively few have enough confidence in their musical abilities to collaborate openly with others. An interactive dance club system acts as a simplified musical interface. By transforming simple dance movements into musical modifications, untrained performers can collaborate in ways previously reserved for seasoned professionals.

This paper presents the creation of a Bluetooth wireless dance club system, called codeBLUE. Although this system incorporates aspects of previous interactive dance and music systems, its wireless implementation based on Bluetooth technology sets it apart, as practical viability in a realistic dance club setting becomes possible.

Dancers using the codeBLUE system wear clothing incorporating Bluetooth-enabled sensors that measure and transmit information about the dancers' movements to a Bluetooth access point positioned in the demonstration area, which in turn forwards the information to a control system. The system software transforms the simple dance movements into musical modifications in real time, altering the melodic, rhythmic, and dynamic properties of the music stream in terms of MIDI parameters. A configuration console allows the DJ to modify the effects that each type of sensor produces, providing him or her yet another channel of creativity and keeping the codeBLUE experience fresh for participants.

The paper describes the architecture, design, and hardware and software implementation of the codeBLUE proof-of-concept prototype. The system has been successfully demonstrated to a live audience.

---

1. INTRODUCTION

Collaborative music creation can be one of the most powerful forms of human communication. Playing in private improvisational sessions or at public concerts exhilarates musicians due to the exchanges of emotional and intellectual information inherent in such creative settings. While many people express themselves musically by whistling and humming, relatively few have enough confidence in their musical abilities to collaborate openly with others.

This paper explores whether emerging Bluetooth technology [Bluespec, Bluebook01] can be used to provide a means for allowing untrained performers to share in the joy of musical collaboration and interaction. The idea is to develop a wireless interactive dance club environment that presents users with a simplified and intuitive musical interface. By transforming simple dance movements into musical modifications, untrained performers can collaborate in ways previously reserved for seasoned professionals. (A glance at the popularity of karaoke and sing-alongs demonstrates the powerful attraction of informal and amateur musical collaborations.) This paper presents the architecture, design, and hardware and software implementation of a prototype Bluetooth wireless dance club system, called codeBLUE. Our objective is to explore the practical viability of using Bluetooth wireless technology to expand and realize the opportunities for interactive musical collaboration. While previous research projects [e.g. Paradiso99] have also considered interactive dance club applications, we believe that Bluetooth technology, with its low cost, low power, and small dimensions, has the potential to render musical collaboration through dance for tens and possibly hundreds of participants a reality.

Participants in the codeBLUE dance club environment interact and collaborate musically by dancing. Sensors with Bluetooth interfaces embedded in articles of clothing transmit information about participants' actions to ceiling-mounted access points. The access points communicate with a computer running a special software application that modulates its musical output in response to incoming sensor data. The musical output is then sent to music synthesizers and lighting controllers.

Example scenario. Alice, Bob and Carol are at *Club codeBLUE*, and have been outfitted with various wireless sensors. The DJ is playing a continuous selection of dance music. A motion sensor attached to Alice's sleeve allows her to modify the volume of the drumbeat on channel 1 when she moves her arm in a certain way; a pressure sensor attached to Bob's shoes allows him to raise or lower the pitch of certain instruments on channel 2 by tap dancing; and a force sensor in Carol's palm allows her to modify certain lighting effects by clapping. Like other participants, Alice, Bob and Carol are provided detailed instructions about the effects of each sensor by the doormen or DJ; alternatively they are allowed to discover them completely by experimentation, or something in between. In any case, as participants learn and interact the quality of their collaboration becomes more interesting. Participants wearing sensors that cause the same effects discover each other over the course of a session and experiment by synchronizing their movements. (A simple analog is the way spectators at a sporting event collaborate to form "the wave".) Alternatively, only a subset of the attendees at the dance club – chosen randomly, or by raffle, etc. -- is provided sensors. Thus the system facilitates social as well as musical interactions. The DJ can modify the effects that each type of sensor produces, possibly in accordance with the characteristics of the music (disco, rock, new wave, etc.), or over the course of a session, providing him or her yet another channel of creativity and keeping the codeBLUE experience fresh for participants. Finally, regardless of formal musical training, Alice, Bob and Carol can create music together and sample the exhilaration that trained musicians experience in collaborative performance.

The codeBLUE scenario immediately raises several constraints. The weight and form factor of a wearable device has to be such as to make it unobtrusive during dancing or movement. The device has to be rugged enough to withstand use in a dance club. Battery life has to be long enough not only to last a dance session but also to minimize the cost and nuisance of replacement. Finally, the cost of the device itself has to be sufficiently low to warrant its use in a dance club environment. Sensors meeting these requirements are available in the market today. This fact along with the use of Bluetooth technology allows us to meet the requirements. To a large extent, codeBLUE rests on the promise of light, low-power, inexpensive Bluetooth technology.

Our contributions in this paper are as follows. We propose a novel application domain for Bluetooth wireless technology, namely human collaboration, and its instantiation for a wireless dance club environment. We also investigate the practical viability of applying Bluetooth for this class of applications. In the following section we describe the system model and architecture. In sections 2 and 3 we describe the hardware and software design respectively. In section 4 we briefly discuss the cost and usability features of codeBLUE, as well as related work. Finally we end with conclusions and suggestions for further work.

The codeBLUE prototype system has been implemented and successfully demonstrated before a live audience in an auditorium with state-of-the-art theatrical lighting and sound equipment to create an interactive musical environment. Two versions of the prototype have been installed, one at Stevens and one at Telcordia. If this paper is accepted a demonstration could be provided at the conference.

1 SYSTEM MODEL AND ARCHITECTURE

The core components for implementing project codeBLUE include the various sensors for data acquisition, the hardware and low-level software components for wireless communications, and the high-level software application for manipulating live music streams and lighting effects in response to acquired sensor data.

A block diagram of the system architecture is shown in Figure 1. Battery-operated wireless sensor modules are worn by participants and communicate sensor data (pressure, temperature, etc.) to wireless access points over Bluetooth links. Each sensor module consists of a body sensor, a microcontroller, and a Bluetooth module. The sensor module acts as a Bluetooth slave and the wireless access point as a Bluetooth master. An access point has power as well as wired LAN connections and is typically mounted on the dance club ceiling. The access point relays the sensor data packet over an Ethernet LAN to the codeBLUE Controller computer. The Controller interfaces to a standard Musical Instrument Digital Interface (MIDI) format file player for sound effects. It also interfaces via MIDI to a standard DMX lighting controller to produce visual lighting effects for stand-alone and intelligent lighting instruments.. The codeBLUE Controller application, called *CBPlayer*, contains a data filter that modifies the MIDI stream in response to sensor data from the access point. It also offers the DJ or system administrator a console and GUI to configure the music playlist as well as the effect that each sensor has, and other program parameters.

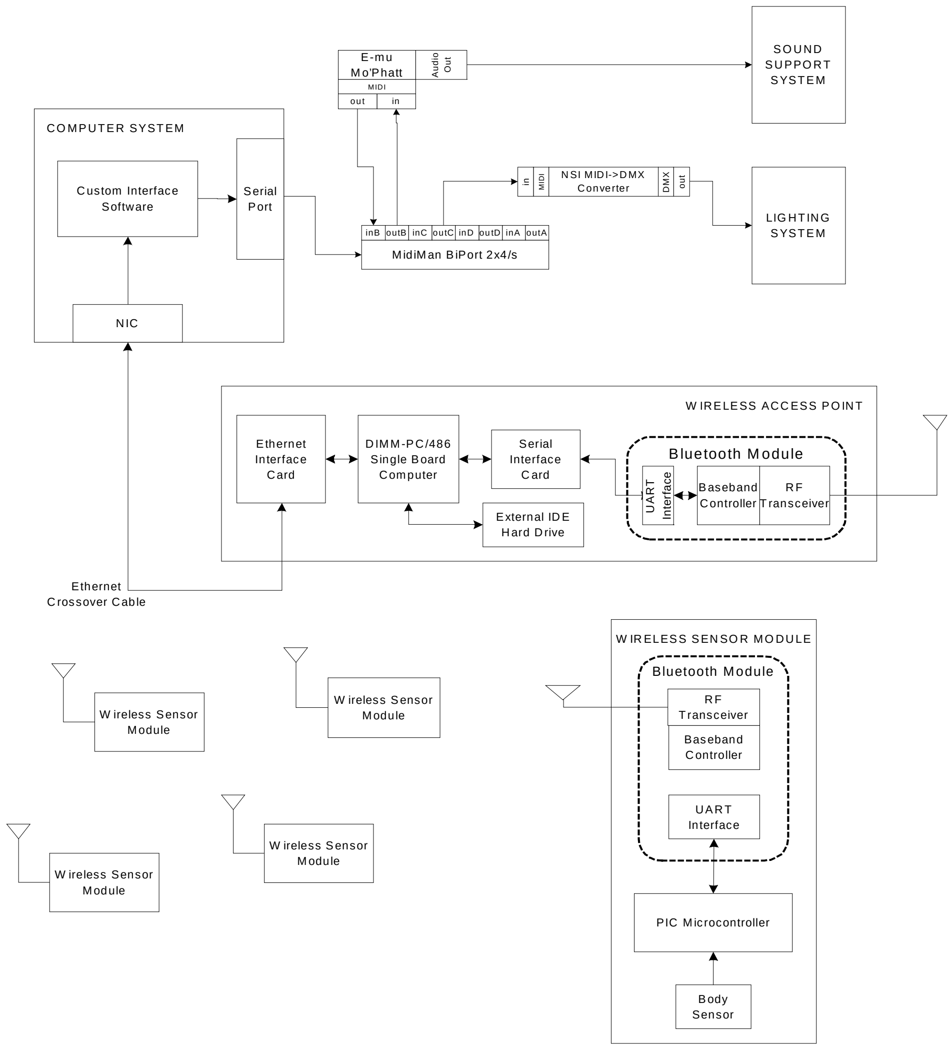

Figure 1: Block diagram of codeBLUE system architecture

MIDI format files are binary data files containing instructions that tell electronic instruments how to play a song – from the notes and their durations to dynamic and pedal markings – much like musical scores. Standard MIDI files were chosen as best suited to the codeBLUE application, even though there are a number of standard audio file types to choose from (.wav, aiff, MP3, .au.) There are several advantages to generating sound with a MIDI synthesizer rather than using sampled audio from a disk or CD-ROM. The first advantage is storage space. Data files used to store digitally sampled audio in PCM format (such as .wav files) tend to be quite large. A four-minute song sampled at 44.1khz in stereo requires about forty megabytes of storage space, or ten megabytes per minute. MIDI files, on the other hand, typically consume ten kilobytes per minute. The second reason codeBLUE utilizes MIDI lies in the interactivity requirement. Since MIDI files are merely instructions about how to play a song, music stored in them can be easily modified in real time.

Making these transformations in an audio file are extremely processor-intensive, and, in the case of rhythm, sometimes not even possible.

2 HARDWARE DESIGN AND IMPLEMENTATION

In this section we provide an overview of the hardware design and implementation. Schematics and other details are available in [Hromin01]. The hardware consists largely of three components: the sensor module interfaced with a Bluetooth module, the wireless access point and the codeBLUE controller interfaced with sound/lighting hardware.

2.1 Bluetooth modules

Bluetooth is a low power, low cost short range wireless communication system operating in the 2.4GHz ISM band [Bluespec,Bluebook01]. It enables small portable devices to connect to each other and communicate in an ad-hoc fashion with nominal speeds of up to 1 Mbps. Every Bluetooth device has a unique device address, and functions either in the slave mode or in the master mode during communication. A set of communicating Bluetooth devices form a group called a piconet, which has (exactly) one device operating as a master and up to seven devices actively functioning as slaves at any given time. Each of these seven slaves has an address allocated by the master called the active address.

The devices in a piconet use frequency hopping to communicate. The unique Bluetooth device address of the master and its clock determine the frequency-hopping pattern used within a piconet. Thus, the frequency hopping sequences used in different piconets will be different, helping to avoid interference. When slaves connect to a piconet, they obtain the device address and the clock of the master, and thus are synchronized to the master. Time is divided into equal-sized *slots* of length 625 microseconds, and TDMA is used to mediate access to the piconet amongst slaves. Time Division Duplex is used between the master and each slave. The master is responsible for determining which slave is going to transmit and when. The master always transmits in the even-numbered slots and a slave transmits in the odd-numbered slots; a slave can transmit only if the master has addressed it in the previous odd numbered slot. In order to keep up the alternation between even and odd slots, packets must occupy an odd number of slots; hence each packet can span 1,3 or 5 slots and is transmitted at a single frequency. After each packet is transmitted the devices retune their radio to a different frequency resulting in hopping from frequency to frequency. Bluetooth also allows various power saving modes, details of which can be found in [Bluespec,Bluebook01].

Bluetooth technology was integrated into both the wireless sensors as well as the wireless access points as part of the project. The Bluetooth module from Ericsson was used in the prototype, and includes the radio, baseband, and firmware units. As per the specification, the Bluetooth radio unit operates in the unlicensed 2.4 GHz Industrial, Scientific and Medical (ISM) band, using Frequency Hopping Spread Spectrum (FHSS) to avoid interference. Only Asynchronous Connectionless (ACL) Bluetooth links, designed for data packets, are used in the codeBLUE system.

2.2 Wireless sensor module

The wireless sensor module is a key part of the hardware design, as this is where form factor, weight, size, power and cost considerations are most severe.

In use, specialized sensors are placed on participants' bodies and clothing to measure hand motion, finger pressure, temperature, positioning, etc. The function of each sensor is to output an analog voltage proportional to its measurement function. The sensor module is a package that contains a microcontroller, the Bluetooth wireless transceiver module, and up to five analog sensors. The analog data from the sensor reading is digitized using an analog to digital converter. The microcontroller appends the sensor data to a Bluetooth protocol packet and sends it to the wireless transceiver. From the point of view of the Bluetooth protocol, the sensor module operates as a slave, and the wireless access point operates as a master.

Sensor Name	**Parameter**	**Sensing Method**	**Range**	**Size (mm)**
G-Force	Single-axis acceleration	Piezo-electric	+/- 2.5G	56 x 38 x 19
Reach-Close	Proximity	IR reflection	80 cm	13 x 14 x 30
Bend	Flex angle	Piezo-resistive	0 - 130^{o}	110 x 7 x 0.5
Touch	Contact pressure	Force-sensitive resistor	100g - 10Kg	38 x 38 x 0.5
Touch-Mini	Contact pressure	Force-sensitive resistor	100g - 10Kg	19 (circle) x 0.5
Hot	Temperature	Zener	-40 -100 C	6 x 10 x 8

Light	Ambient light	Photo- resistor	180^o acceptance	6 x 10 x 5
Flash	Illumination	Photo-transistor	400 - 1100 nm	286 x 10 x 8

Table 1: Sensors used in the codeBLUE prototype

The sensors used in the codeBLUE prototype are listed in Table 1. All sensors operate at 5V and output a variable voltage 0 - 5V. The *ReachClose* sensor measures infrared light reflections to measure a distance to an object and is insensitive to ambient light variations. The *GForce* sensor measures acceleration and deceleration along a single axis. To measure the flex angle of a specific body part such as the wrists, elbows, and knees, a *Bend* sensor is used. The *Flash* sensor is used to detect rapid lights changes or flashes (such as strobe lights at a club) using a phototransistor. The *Light* sensor senses variations in ambient light using a photoresistor. The temperature on an individual's body or the room can be measured using the *Hot* sensor. Finally, the *Touch* sensor provides a useful way to measure the pressure of a handclap or foot stomp.

The 8-bit microcontroller on the sensor module, the PIC 16F876, was chosen for low cost and simplicity, and also as it has five internal 10-bit A/D converters that easily interface with the sensors. It has a maximum clock speed of 20MHz, 364 bytes of user RAM and 8 KB of program ROM. With careful programming this was found to be sufficient for our application.

The sensor module circuit operates as soon as power is switched on; the program within the PIC immediately begins to run. After general initialization of registers, the Bluetooth module is reset and initialized. The PIC then waits hardware interrupts indicating data on its serial I/O pins. After a connection is established to the access point, the digitized reading is taken and sent out.

The final wireless sensor prototype module consists of two small PCB boards, one containing a DC-DC power converter (MAX710), a RS232 level converter (MAX3223), and the PIC microcontroller, while the other contained the Bluetooth module with a two-wire serial interface between the two boards. The prototype module is sufficiently small and light to be worn comfortably on the belt (see Figure 3).

2.3 Wireless access point

The design of the wireless access point is oriented towards minimizing cost, size and power requirements. It is based on a single-board x86 processor, namely an AMD 486 66MHz processor board called the DIMM-PC, running Linux. Available on board is 16MB of user RAM and 16MB of CompactFlash for program storage. An IBM 340 MB microdrive is used to store the Linux OS and user program for interfacing with the Bluetooth module. The access point runs the Bluetooth protocol stack, operating as a piconet master, to receive data from the wireless sensors. An application on the access point connects to wireless sensor devices, and runs in a loop converting the sensor data packets to UDP packets for transmission over Ethernet to the codeBLUE Controller.

2.4 Controller and Sound/Light system

The remaining hardware in the codeBLUE system consists of the codeBLUE Controller and sound/light hardware. The codeBLUE Controller is a Windows PC.

The prototype system uses the following music hardware: the MidiMan Biport 2x4s, the E-mu Mo' Phatt Urban Dance Synth, an Alesis QS6 keyboard and a Mackie 1402VLZ-pro audio mixer. The MidiMan is a external device that takes the place of a sound card with MIDI playback capabilities in the computer. An external unit is needed due to the connection interface with the synthesizer and DMX lighting controller. The MidiMan converts the serial data in to MIDI data and routes to the appropriate midi port. The EMU Mo' Phatt is the MIDI synthesizer for the system. The Alesis keyboard was used to compose the demonstration music. The Mackie mixer was used to mix and set a preset sound level for the output channels of the MIDI synthesizer.

3 SOFTWARE DESIGN AND IMPLEMENTATION

The protocol software stacks for the codeBLUE system are shown in **Figure 2**. The software components for implementing the codeBLUE system consist of two parts. The first is the low-level software running on the Bluetooth wireless access network, i.e., wireless sensor modules and access points. The second is the *CBPlayer* application running on the codeBLUE Controller PC.

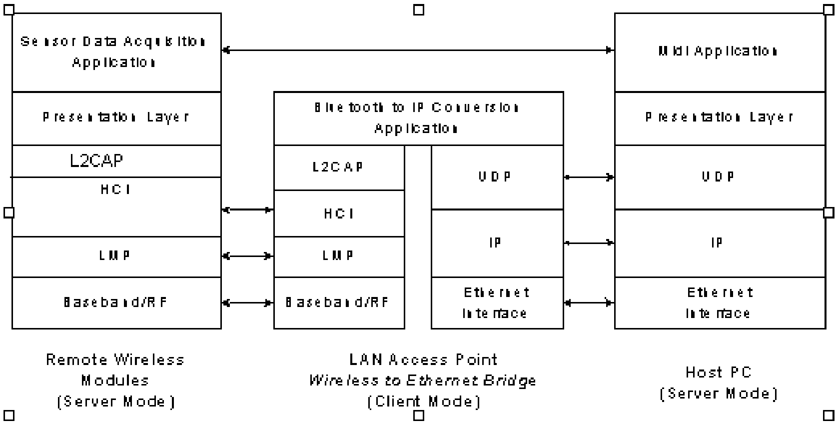

Figure 2: Protocol software stacks on the wireless sensor module (Remote wireless modules), the access point, and the codeBLUE Controller PC (Host PC)

3.1 Bluetooth network applications

The Bluetooth interfaces on both the wireless sensor module as well as the wireless access point require software to control the hardware and manage all connection and data traffic.

The sensor module application is a slave from the point of view of the Bluetooth protocol. After initialization the program instructs the Bluetooth module to enter the Pagescan mode where it listens to Page packets from the master, i.e., the access point. It waits until a connection is established with *BT2IP*, the Bluetooth-to-IP converter running on the access point, and periodically sends it sensor data. The sensor data is sent as a Bluetooth packet and includes a sensor ID value and the sensor data value consisting of the 7-bit analog reading from the corresponding sensor.

The BT2IP application on the access point establishes connections with sensor modules by initializing the Bluetooth module and then instructing it to enter the Page mode. It subsequently waits for sensor data and relays it over the Ethernet using UDP.

To save memory, processing time as well as implementation time we do not implement the IP protocol, or all the layers of the Bluetooth stack. Higher Layers like RFCOMM, OBEX, TCS etc.

[Bluespec] are not necessary for transmitting simple sensor data packets and are not implemented. Thus the wireless sensor and the access point communicate using only L2CAP packets.

3.2 CBPlayer application

The CBPlayer application running on the codeBLUE Controller is the brain of the Interactive Dance Club System, connecting the dancers' movements to the music and essentially creating a virtual musical instrument that multiple dancers can simultaneously play to create music.

CBPlayer consists of a module to receive sensor data from access point(s), a standard MIDI file player, a *Data Filter* that modifies the MIDI stream in response to sensor data, and a DJ's configuration interface called the *Configurator* for defining the correspondence between sensors and musical outputs along with other program parameters.

3.2.1 Data Filter

As sensor data arrives, special algorithms or "musical agents" in the Data Filter perform operations on the current stream of MIDI data flowing from the MIDI file player. In general, an algorithm in the Data Filter can take two actions in response to sensor inputs. It can insert MIDI events in the MIDI output stream, or it can modify MIDI events in the stream that are from the song track.

The exact action taken by each algorithm is something that musical experts, musicians and DJs develop with experience. At present the codeBLUE system contains algorithms developed by the project team and that are under experimentation. In general each algorithm (e.g. to modify drumbeat) has a minimum and maximum value parameter (e.g. minimum rate, maximum rate) that can only be changed at system initialization. This is so that modifications to the music do not accidentally exceed acceptable bounds in normal operation e.g. the volume stays within tolerable limits.

In general, the algorithms in the Data Filter must be such that the dancer-initiated modifications to the music are subtle enough to keep the music from becoming inaccessible, but at the same time responsive enough to keep the interest of the participants. The Synesthesia system [Synesthesia98], described later, lists "Ten Commandments" of audience-input music synthesis, such as "This isn't MTV, so a scene cutting 4 times a second won't do. Smooth movement that matches the musical pace works well" and "Every gesture the player makes to the input devices needs to produce a visible and an auditory change, and needs to be repeatable". Designing algorithms and rules for codeBLUE is not the focus of the present paper, but is a fertile and enjoyable area of investigation for dancers and musicians.

3.2.2 Configurator

The Configurator not only allows use of the system in different installations by specifying basic parameters appropriately, but also augments creative possibilities for new interactive performances. For instance, by including a configuration system, an elbow bend sensor doesn't always have to be mapped to rhythm modification. Since the cutting edge of electronic music is constantly changing, owners of codeBLUE systems will want to reconfigure their sensor/music interactivity. With MIDI as the underlying communication and control system, additional outboard music hardware could easily be added to a system. By reconfiguring the mappings, a new MIDI-controlled resonance filter could be linked with the same elbow bend sensor that previously performed rhythm modification.

The Configurator GUI allows the DJ to both see and change how the mappings of sensor to algorithm to MIDI channel occur. The Configurator can read in and display all the current mappings of the system and is able to access all interfaced MIDI devices (external and internal devices), all active sensors, and all of the available Data Filter algorithms. The Configurator also monitors restrictions on the system, namely, that one sensor cannot be mapped to multiple channels or algorithms.

4 DISCUSSION

The key challenges of codeBLUE lie in two areas: the system cost and usability, as well as converting dance to music in a meaningful and aesthetically pleasing manner. The codeBLUE prototype focuses on the former, i.e., how newly available wireless technology can be exploited to make the application viable, and this is discussed below. The latter is addressed by the CBPlayer application, which allows the DJ and trained musicians to explore the dance to-music mapping, and is an ongoing effort. It was found that the latency between a participant manipulating a sensor and observing the result in the musical output was acceptable.

4.1 Cost and usability

The cost of the system can be broken down into four main components: wireless sensor modules, access points, the codeBLUE controller, and the sound/light equipment. In the following we discuss the costs of the prototype system. (All prices below are in US $.)

Five sensor modules were prototyped, with each costing about $500 for the Bluetooth module and $100 for the remaining components. With the expected drop in Bluetooth device prices to about $5-$10, and volume production, the cost would drop greatly. A club could invest in a relatively modest quantity of the devices, letting patrons borrow them (e.g. in exchange for IDs, as is done for other equipment like darts and pool cues). Even with some loss and breakage, the enhancement of the user experience could be well worth the expense.

The access point as implemented in the prototype is about $1000, and is relatively expensive for its function. This is mainly due to the miniaturized components used, such as the IBM microdrive. It appears likely that now PDAs with Bluetooth interfaces and capable of running Linux are easily available they might be a more cost-effective approach for prototyping, dropping the cost to a few hundred dollars. Bluetooth piconets are relatively small (nominally 10m radius), and so several access points would be required in a dance club. Each piconet can support at most seven simultaneously active slaves, possibly leading to more access points being required; however, smart scheduling techniques could expand this limit [Adamou01].

The codeBLUE controller software is a simple PC, and would suffice for handling data from hundreds of sensors. The cost of the sound and light hardware is about $4000, including about $2500 for the DMX lighting system.

The system implementation took roughly 24 person-months of effort over about 6 months, including hardware and software design, implementation, testing, installation, demonstration and documentation.

The key component of usability is the wireless sensor module. Implemented in prototype, the module measures about 7 cm x 5 cm x 3 cm (without the sensor itself) and weighs about 100 g. Since not all components were of surface-mount type, the size of the wireless sensor module custom PCB is larger than it would be otherwise. Using a one-chip Bluetooth module and the Li-Ion polymer technology battery will reduce the size substantially – we estimate by a factor of four -- making the module unobtrusive. Most of the sensors themselves also have a relatively small form factor (see Table 1.) Even with the prototype implementation users did not find the sensor module awkward or cumbersome.

4.2 Short-term improvements

At the time of implementation Bluetooth modules implementing the full Bluetooth version 1.1 specification were not commercially available, so there is no support in the prototype for master/slave switching and role discovery features, which would make the system more configurable and flexible.

Replacing the wireless access point with a PDA seems like a natural place to improve price, form factor and portability of the prototype.

Currently the CBPlayer only allows a sensor to manipulate one environmental resource at a time. It would be interesting to allow one sensor to change multiple environmental aspects, and see at what point the usability degrades. Similarly the Data Filter algorithms contain minimum and maximum parameter values for MIDI output that are fixed at configuration time, and it would be interesting to make these variable by the DJ.

4.3 Related Work

We survey some interesting recent work in the area of music collaboration technology and compare it with the codeBLUE system.

The Brain Opera [Machover96] is a three-movement composition that merges the performance of Dr. Machover's score with audience input. Prior to the performance, audience members explore and manipulate a variety of musical interfaces in the theater lobby to create a repository of improvised musical material. Throughout the subsequent concert, the opera's performers accessed material in the repository and blended it with the framework of Dr. Machover's composition. Thus no two performances of the Brain Opera are similar, and unlike traditional orchestral compositions where performances differ only in interpretation, the Brain Opera is composed anew each time it was performed. Ongoing work in this research group is exploring hyperinstruments [Machover00] as well as other physical and tactile musical interfaces.

The Interactive Dance Club [Synesthesia98] was a step forward from Brain Opera in that it allowed participants to control aspects of the sound, lighting, and imagery within the club environment in real time. Participants manipulated physical objects (floor pressure tiles, tires, etc.), which modified sound and light effects.

CosTune [Nishimoto01] is a networked wearable musical instrument consisting of several clothing-mounted sensors that communicate using an IEEE 802.11 wireless LAN in ad hoc mode. Users exchange song components with other CosTunes and servers located at various places. In Cybershoe [Paradiso98], shoes contain electronic sensors that transmit the movement of the feet to a base station via fixed-frequency FM wireless links. The base station contains the mappings from the movements to music.

Music in Motion [Ng00, Ng01] is another attempt aimed at creating an intuitive and non-intrusive audio-visual performance interface. This framework though is different from our work and the other works described earlier in that it uses input from a video camera. The video camera is used to track visual movements of the scene under inspection. Recognized movements are then used to trigger musical events based on the mapping functions used.

While codeBLUE shares the goal of collaborative music with these projects, it differs primarily on account of the use of a frequency hopping, low power and low-cost wireless technology, namely Bluetooth. Brain Opera, hyperinstruments and the IDC do not use wireless technology. Unlike codeBLUE, in CosTune there is no notion of a shared dance club environment that is playing a single song that all users interact upon, so it is likely to be harder to use with many participants. Also, the ad hoc nature of the network may make collaboration harder due to connectivity, delay and hidden terminal problems, and implementation using 802.11 technology may raise the cost. Unlike Cybershoe, the codeBLUE system is not limited by the fixed-frequency allocation so can scale to more participants, and the use of standard Bluetooth low-cost technology is preferable. The codeBLUE prototype also incorporates different types of sensors and affects sound as well as light, and allows the DJ to modify the dance-to-music mapping.

5 CONCLUSION

The codeBLUE prototype was demonstrated in the DeBaun Auditorium at Stevens Institute of Technology. Three team members each wore a sensor module connected with one sensor, with each mapped to different algorithms for light control, filter sweep, and initiating a sound sample. Music was specially composed by one of the team members for the demonstration. The experience showed that for a short demonstration to a general audience (composed of many technology experts!), it is more effective to use a familiar piece of music so that the dance-to-music mappings can be more easily observed. Audience participation at the end of the demonstration was enthusiastic.

Currently the codeBLUE prototype is being ported from C++ to Java to allow for better portability and maintainability, and to allow for a web interface. The latter is particularly interesting as it could allow a DJ to collaborate with other clubs or operate from a remote location.

The Bluetooth technology was found to be very satisfactory for this application, despite the fact that very early versions of hardware and drivers were used in the prototype. Bluetooth potentially opens the door for practical realization of many innovative applications that, like codeBLUE, can enrich human experience.

REFERENCES

[Adamou01] M. Adamou, F. Anjum, and R. Jain, Helpers and Sleepers: Master assisted auxiliary piconet formation in Bluetooth, submitted for publication, 2001.

[Bluespec] Bluetooth specification, http://www.Bluetooth.com

[Bluebook01] B. Miller and C. Bisdikian, *Bluetooth Revealed*, Prentice Hall, NJ 2001.

[Hromin01] D. Hromin et al: CodeBLUE, A Wireless Interactive Dance Club System, Stevens Inst of Tech., Technical Report 2001.

[Machover96] T. Machover, Brain Opera. http:// brainop.media.mit.edu

[Machover00] T. Machover, Hyperinstruments, http://www.media.mit.edu/hyperins

[Nishimoto01] K. Nishimoto, T. Maekawa, Y. Tada, K. Mase and R. Nakatsu: Networked Wearable Musical Instruments Will Bring A New Musical Culture, *Proc. 5th Intl. Symp. on Wearable Computers (ISWC2001)*, pp.55-62, 2001.

[Ng01] K.C. Ng., Music via Motion: Trans-Domain Mapping of Motion and Sound, *Proc. Intl. Workshop on Human Supervision and Control in Engineering and Music*, Kassel, Germany, September 2001.

[Ng00] K. C. Ng, S. Popat, E. Stefani, B. Ong, D. Cooper, and J. Smith-Autard, Music via Motion: Interactions between Choreography and Music, *Proc. 10th Intl. Symp. on Electronic Art (ISEA2000)*, Paris, France Dec 2000.

[Paradiso99] J. Paradiso, E. Hu and K. Hsiao: The CyberShoe: A Wireless Multisensor Interface for a Dancer's Feet, *Proc. Intl. Dance and Technology 99*, Tempe AZ, Feb 26-28 1999.

[Synesthesia98] Interactive Dance Club. *SIGGRAPH*, June 1998, http://www.elkabong.com/IDC/idc.html.

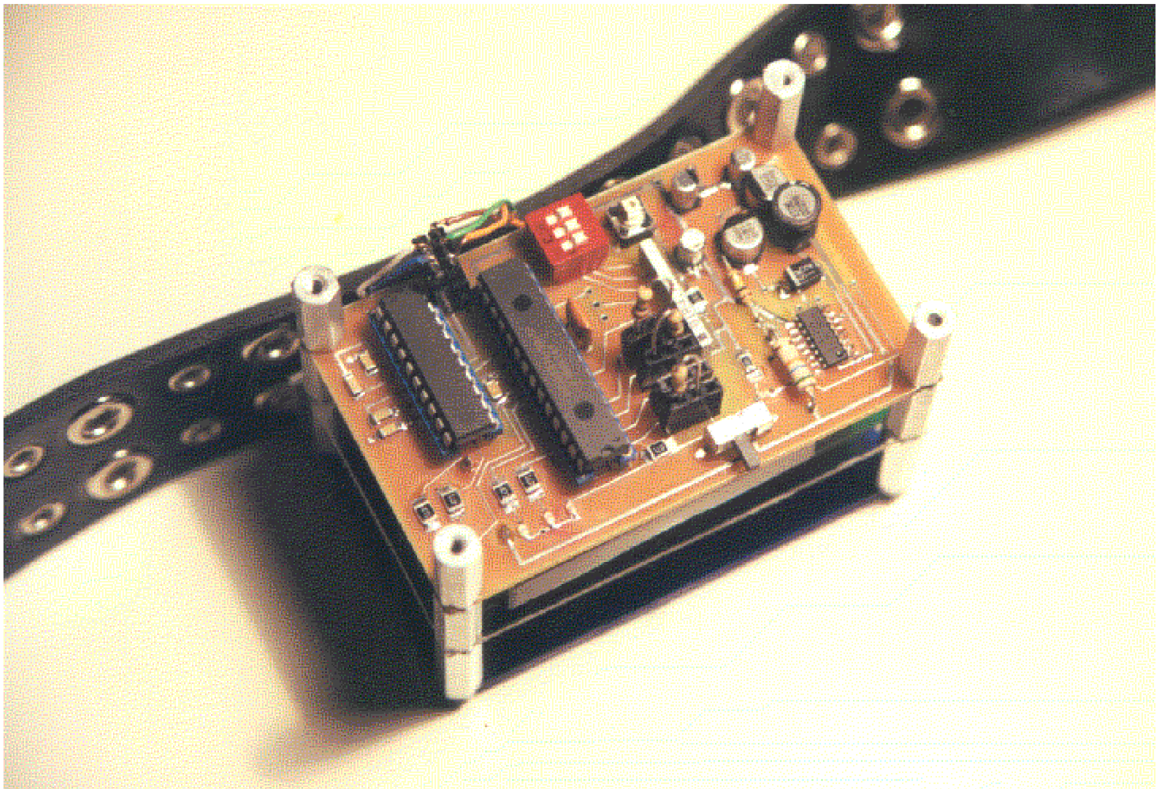

Figure 3: Photograph of the wireless sensor prototype unit