A simple practical approach to a wireless data acquisition board

Smilen Dimitrov
Medialogy
Aalborg University Copenhagen
Lautrupvang 15
2750, Ballerup, DK
smilen@smilen.net

Stefania Serafin
Medialogy
Aalborg University Copenhagen
Lautrupvang 15
2750, Ballerup, DK
sts@media.aau.dk

ABSTRACT
In this paper, we describe the design of a novel wireless acquisition board, its advantages and characteristics, and its applications to NIME.

Keywords
Interface, data acquisition, wireless, interaction.

1. INTRODUCTION

The recent development in human-computer interaction research shows a demand for customized design of electronic sensor systems, used in ubiquitous computing, wearable interfaces, music and dance performances etc. This implies that the interaction designer should consider applicable sensors, in addition to programming the interpretation of their signals in a high-level software platform. Often, the problem occurs beyond this scope of interaction design – being about which hardware/software interface platform to use, to make the sensor signals available as altering variable values in the desired software platform. Market acquisition or custom building of such interface platforms requires acquaintance with data-acquisition systems from an engineering perspective, which is not covered in the academic scope of traditional human computer interaction educations. Such trends are becoming more and more addressed academically (see, for example, [1] [2]), however they require introduction to a broad theoretical base. In this paper, we describe the concept of a simple prototype implementation of a wireless data acquisition board, which physically illustrates theoretical concepts and their systematic binding, which might assist a student of HCI in gaining practical insight.

Custom development of a data acquisition system for interaction applications is made easier (if not possible) by the continuous market development, and availability of inexpensive ICs that are functional units with relatively simple electronic interfaces. In this case, the central interest is in microcontrollers, which both contain on board memory (and can be easily reprogrammed), and different built-in functional units: multiple input A/D converter, an universal asynchronous receiver transmitter (UART) for serial communication, radio transmitter etc. In addition, the market offers a myriad of different devices aimed at establishing a digital wireless radio link, with a different range of possibilities – from simple modulation of a radio wave, up to full implementation of network communication protocols.

In this paper, we focus on implementing the (almost) simplest possible board allowing a multichannel, wireless data acquisition, aimed at interaction processing usage. A data acquisition board commonly accepts a conditioned analog voltage signal, hence it can be used with a variety of sensor systems – as long as the sensed values are mapped to voltage that fits the defined input range of the board. The important point is that any microcontroller that features a multiple channel A/D converter, can be programmed to act as a data acquisition hardware in a relatively simple manner (as long as there is a high-level software IDE with a serial code routine, as is mostly the case).

Our intention is to illustrate a simple data acquisition system involving a single microcontroller circuit that communicates the acquisition data to a PC via serial. The traditional PC asynchronous serial interface allows the possibility to establish a simplex communication line using only one transmission line; that is, a single digital voltage signal. This then allows that the simplest modes of radio wave modulation (AM/FM or the binary versions ASK/FSK) can be implemented to establish a wireless link, using tuned/coupled modulator+transmitter and receiver+demodulator IC devices. Finally, we obtain a prototype system of two hardware devices, a transmitter and a receiver, and a corresponding software driver.

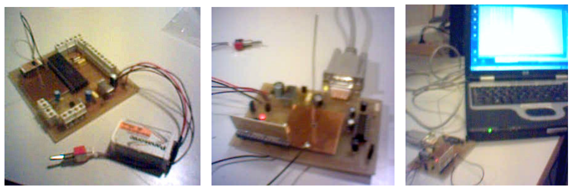

Figure 1. Transmitter device (left), receiver device (center), and receiver connected to a PC (right).

The academic value is first and foremost in the simplicity: all circuits involve only a single side printed circuit board design and basic etching and soldering skills. The design revolves around usage of a microcontroller (in this case a PIC16F74) as a data acquisition hardware that creates an asynchronous serial code, transmitted through a coupled pair of radio devices - digital FM transmitter and receiver ICs (in this case RTFQ1 and RRFQ1[1]), and a MAX-232 chip for conversion of TTL into RS-232 levels

[1] www.rfsolutions.co.uk

used by the PC. A serial driver/decoder is provided in Max/MSP[2], coded fully through its native objects. The simplicity then allows direct relationship to engineering concepts: the particular microcontroller architecture relates to the basic structure of a multichannel data acquisition system. The serial code generated, in both TTL and RS-232 levels, is observable on an oscilloscope, and directly relates to the theoretical basics of asynchronous serial. Furthermore, the system can be seen as a simplex communication line, and we ourselves define the communication protocol, and code and decode it, illustrating basic telecommunication concepts – without however going further into electrically more complex concepts of handshaking. Involving a wireless radio link illustrates basic introduction to radio waves, antennas and propagation – however, most of the complicated RF issues can be avoided thanks to the functionality of the IC devices; only a ground plane and stable power source need to be considered in this particular design. Finally, this requires but four ICs and several basic electronic components (resistors, transistors, capacitors) to implement, all easily accessible in common educational facilities for electronic hardware – and usually regularly covered in a budget of an educational institution, thus minimizing additional expenses to running the implementation of the system as a student exercise (even for private purchase, the total current price of components required should not be over 70-100 Euro).

The simple design and quick implementation, the direct illustration of diverse basic theoretical concepts in a single system, and the relatively low expenses related to building process, are the general points that make the implementation of such a prototype level wireless data acquisition system applicable as an course exercise, executed say in a one week workshop format, and aimed at students of interaction design.

2. RELATED WORK AND INSPIRATION

We consider such device to be like Teleo Intro board by Making Things[3] or the Kroonde wireless data acquisition system by La Kitchen[4], which are marketed for use with Max/MSP by Cycling 74, a software development platform commonly used for customized interaction processing. Such devices are characterized by digitizing of multiple input channels, with sampling rates with an upper bound of around 1 KHz per channel, sampling resolutions spanning from 1 to 16 bit per channel sample, some form of physical connection to a PC (USB, Ethernet) and a driver software, which eventually is exposed as interface as a Max/MSP object – allowing further interaction programming in this software platform. Most commonly, each channel to which a conditioned analog sensor voltage is applied (commonly in the range of 0V-5V) is exposed as an integer numeric value, provided through a corresponding outlet of the driver interface object in Max/MSP; and refreshed at the effective sampling period per channel.

Similar functionality is also found in other systems. Besides these commercial market products, HCI research also discusses custom-built data acquisition interfaces, such as Atomic Pro [4] or WiSe Box [5], which too have a similar functional interface (although aimed at other interaction uses, such as MIDI or OSC from CNMAT [7] for music instrument hardware).

The Teleo Intro board implements this functionality through a PIC microcontroller, which samples the four analog inputs on the Teleo, and communicates the results to a PC via USB connection. In hardware, the analog inputs on the Teleo Intro are connectors with screws, and in addition, +5V and ground connectors are offered. This allows that simple sensors (such as a potentiometer) can be attached directly to the board, mechanically instead of through soldering, which makes it quite accessible for education uses - and inspired the same solution for our prototype system as well.

3. THE PROPOSED DESIGN

The system design revolves around usage of a PIC16F74 microcontroller, coupled digital transmitter/receiver IC's RTFQ1 and RRFQ1 (operating at 868 MHz) and a MAX-232 line signal converter ICs. This results with two circuits – a transmitter and receiver circuit, where the central issue is to program the PIC to behave as a data acquisition hardware. The system is completed with a Max/MSP decoder/driver.

Power sections are implemented through voltage regulator ICs, and power connectors are 9V battery connectors, allowing both battery and standard adapter power-up.

3.1 PIC data acquisition program

The PIC program is an extremely simple endless loop, which simply activates the A/D conversion on an analog input, and stores the results on a corresponding index in an array; the process is repeated for all available analog inputs. Based on this array, a 35 bytes string is produced, which represents a single update word – and sent via serial, which involves generating an TTL voltage signal on one of the microcontroller pins. A pseudo-code looks like the following:

```
Initialize PIC

loop forever

  erase out array
  read ADC 1
  store ADC 1 value in out array
  read ADC 2
  store ADC 2 value in out array
  ...
  read ADC 8
  store ADC 8 value in out array

  format string based on out array
  send string based on out array

end loop
```

This program involves knowledge of only a few registers in the PIC and can be very easily written in a high-level software. We used a BASIC interpreter, which almost follows the pseudocode structure directly. The serial uses a software bit-banging routine, instead of using the hardware UART, thus it can use any I/O pin of the microcontroller for serial code output.

[2] www.cycling74.com

[3] www.makingthings.com

[4] www.la-kitchen.fr

Using a Microchip PIC microcontroller allows for usage of a free integrated development environment for writing and compiling a given PIC program. However, a PIC also allows for usage of the classic David Tait [6] hardware serial programmer, as well as variety of freeware software programmers, for programming the microcontroller. The hardware programmer used here is also custom-built version of the Tait programmer, and in this case can also be integrated in the educational exercise concept, as it not only discusses basic microcontroller programming, but also usage of the parallel port of a PC as a programmable power supply, and synchronous serial communication.

3.2 Transmitter circuit

The transmitter circuit hosts the power supply section, the PIC and the transmitter IC RTFQ1, and is powered by a 9V battery. The antenna is a 9 cm long wire, and a ground plane is implemented by a separate non-etched PCB. The dimensions of the unit are 88 x 94 mm, the antenna can be folded and the unit can be worn in a pouch or a pocket.

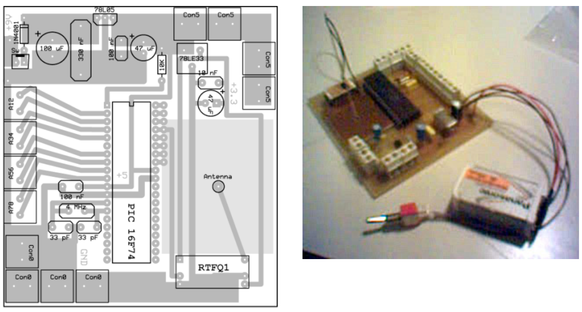

Figure 2. PCB Layout and implementation of the transmitter circuit

The PCB layout on Figure 2 shows that only 15 components are used for the circuit, the rest of it is connectors.

3.3 Receiver circuit

The receiver circuit hosts the power supply section, the receiver IC RRFQ1, and the line level converter IC MAX-232; as well as a DB-9 connector for serial connection to a PC. The dimensions of the unit are 94 x 64 mm, and is powered by a standard AC/DC adapter set at 9V.

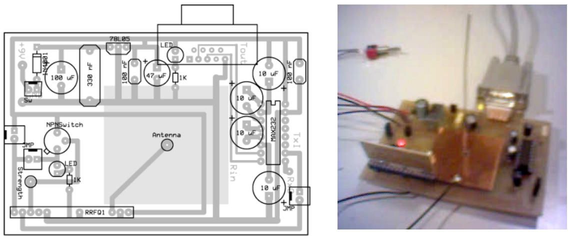

Figure 3. PCB Layout and implementation of the receiver circuit

The device also features two LED diodes – one to indicate power up, and one turns on whenever the transmitter is turned on (which is made possible by a RF signal strength measurement pin on the RRFQ1 IC).

The electric design is again simple, however, it doesn't take into account that when the transmitter is not operating, the receiver IC generates in essence a random digital voltage. Such a signal usually causes errors in a PC, hence there has to be a procedure where the transmitter is turned on first, and then the receiver.

3.1 Max/MSP decoder/driver

The usage of the synchronous serial signal in Max/MSP involves

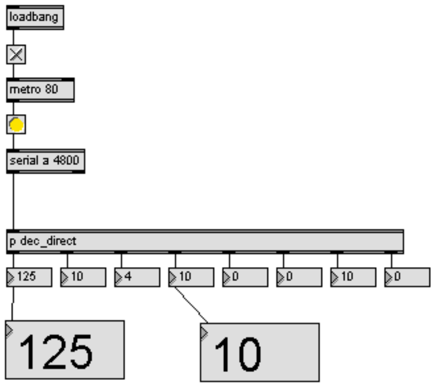

Figure 4. A sample Max/MSP patch demonstrating use of the decoder

the native Max object *serial*, which provides the characters received through the serial port of the PC as a consecutive sequence of integers (ASCII codes). From the original 35 bytes word, we need to extract the 8 packed values that represent the latest sensor values acquired by the transmitter, which is performed by a custom Max decoder.

As Figure 4 demonstrates, the decoding functionality is encapsulated in a subpatcher object call *p dec_direct*, which accepts the incoming character stream as an input, and provides the extracted sensor channel values as integers through the corresponding outlets, which is again a principle that the Teleo uses as well for its interface object.

The decoder itself is implemented by native Max objects, which allow basic understanding of a decoding process through a graphic programming environment. Although this is useful from an educational perspective, where such representation might be closer to the student than a serial driver in C, it may be so doing so at expense of quality – as Max patches using this particular driver in its current version, crash quite often.

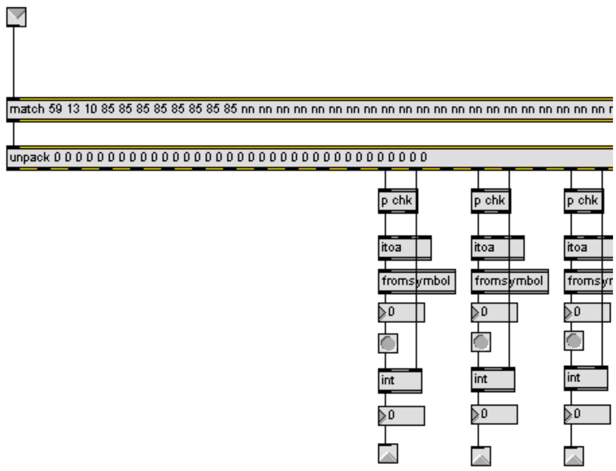

Figure 5. The Max/MSP serial decoder implemented through native Max objects

4. FINAL REMARKS

Concerning the first version of the prototype, lot of things can still be improved. Advantages of the prototype are the fact that it provides 8-channel, 8 bit per channel wireless data-acquisition, and a single 9V battery provides up to 20 hours of transmitter operation.

However, even at this level certain measures can be undertaken that allow a more reliable operation of the system – usage of onboard UART on the PIC; compacting the redundant 35 byte codeword; implementing a gating circuit at the receiver that opens only when transmitter is sending, improving possible RF construction issues, and increasing the communication speed to the maximum offered by RRFQ1/RTFQ1 which is 9600 bps. A reliable and slightly faster circuit can then already be applied in any custom interaction design that could also implement a Teleo or a Kroonde as interaction data acquisition boards.

In the sense of actual usage, the exact same concept can be further engineered, by using other microcontroller devices (possibly with built in wireless transmitters) and/or other wireless ICs (possibly ones that use a network communication protocol), which of course increases the design and implementation complexity.

However, as the system currently is, its simplicity is an excellent opportunity to physically illustrate the most basic engineering issues in data acquisition for interaction design to students of HCI.

This device has already been successfully used in several interaction design and music projects, where students appreciated mostly the wireless feature which enhanced motion possibilities in a virtual reality context [3].

Having the opportunity to build such a system and use it, gives the students practical insight in electronics, telecommunications and software issues, an insight that would assist the HCI student both in evaluating different data acquisition systems on the market for a given interaction task, and give a starting point if a building of a custom data acquisition system is required.

Detailed circuit diagrams and instructions on how to assemble this board are available at www.imi.aau.dk/~sts/board.html.

5. REFERENCES

[1] Putnam, W., Knapp, R. B., "Input/Data Acquisition System Design for Human Computer Interfacing", online course notes, Stanford University, 1996 http://ccrma.stanford.edu/CCRMA/Courses/252/sensors/sensors.html

[2] Cook, P. R., "Real-Time Computing For Human Computer Interfacing", online lecture notes, Princeton University, http://soundlab.cs.princeton.edu/learning/tutorials/RealTime/realtime.html

[3] Nordahl, Rolf. Auditory rendering of self-induced motion in virtual reality. M.Sc. project report, Dept. of Medialogy, Aalborg University Copenhagen, 2005.

[4] Fléty, E. AtoMIC Pro : a Multiple Sensor Acquisition Device. Proc. NIME 2002.

[5] Fléty, E. The WiSe Box : a Multi-performer Wireless Sensor Interface using WiFi and OSC. Proc. NIME 2005.

[6] David Tait homepage: http://www.people.man.ac.uk/~mbhstdj/piclinks.html

[7] Freed, A. and Wright, M. Open SoundControl: A New Protocol for Communicating with Sound Synthesizers. In *Proc. of ICMC 1997.*