A Wearable Health Monitor To Aid Parkinson Disease Treatment

by

Joshua A. Weaver

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2003

Author ....
Program in Media Arts and Sciences,
School of Architecture and Planning,
May 9, 2003

Certified by ....
Alex "Sandy" Pentland
Professor of Media Arts and Science
Thesis Supervisor

Accepted by ....
Andrew B. Lippman
Chairperson, Department Committee on Graduate Studies

A Wearable Health Monitor To Aid Parkinson Disease Treatment

by

Joshua A. Weaver

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on May 9, 2003, in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Science

Abstract

This thesis developed a wearable motion capture system to record Parkinson's patients performing daily activities during each of the three stages of their medication cycle. Five calibrated accelerometers continuously monitored motions of the subjects' torso, wrists and ankles, and stored the resulting data onto a low-cost compact flash (CF) memory card.

Five hours of data was recorded from a volunteer with PD wearing the motion recording system, along with the corresponding medication state rated by an observing physician. This data was divided into training and test sets, where one-quarter was reserved for testing.

A neural network demonstrated 85% correlation between data sampled from all five accelerometers to the dyskinetic medication state labelled by the physician. Noting inherent confusion between a sedentary patient with high dyskinesia and a properly medicated patient moving energetically, a second neural network was trained to identify periods of walking, with 75% correlation. Using this activity classifier to remove periods of walking increased the overall accuracy to 91%.

Thesis Supervisor: Alex "Sandy" Pentland
Title: Professor of Media Arts and Science

A Wearable Health Monitor To Aid Parkinson Disease Treatment

by

Joshua A. Weaver

Thesis Committee

Thesis Advisor ______________________

Alex "Sandy" Pentland
Toshiba Professor of Media Arts and Science
Human Design Group
M.I.T. Media Laboratory

Thesis Reader ______________________

Joseph A. Paradiso
Associate Professor of Media Arts and Science
Responsive Environments
M.I.T. Media Laboratory

Thesis Reader ______________________

Rosalind W. Picard
Associate Professor of Media Arts and Sciences
Affective Computing
M.I.T. Media Laboratory

Acknowledgments

Sandy, thank you for seven exciting years at the Media Lab and supporting me through my "extended" stay in the program. Roz and Joe, thank you for working with my down-to-the-wire scheduling and somehow still managing timely feedback. Linda, thanks for all the motivational emails – it's finally finished!

Rich, you've kept the wearable computing dream alive at MIT and made the BorgLab into a happy hacking zone. May EvilBØrg always watch over your projects.

Christine and Madleina, thank you for all your matlab advice and moral support. Sixteen hours straight hours in a computer cluster, now that's friendship!

Contents

Chapter 1

Introduction

The recent announcements that celebrities such as Michael J. Fox, Muhammad Ali, and Pope John Paul II suffer from Parkinson's Disease (PD) have drawn increasing public interest to the disease and the search for a cure. In truth, over a million people suffer from Parkinson's disease in the US, with approximately 50,000 new patients diagnosed each year. The National Institute of Neurological Disorders and Stroke estimates that the total cost of health care for Parkinson's patients exceeded $5.6 billion last year[1].

While Parkinson's is a chronic, progressive disease of the nervous system without a known cure, a variety of medications may provide dramatic relief from its symptoms. Effective treatment requires careful monitoring of the patient's symptoms to adjust dosage, typically through an observer hired to record the occurrence and frequency of movement impairments.

The goal of this thesis is to design an automated monitor to augment (and in some cases, replace) the human observer and aid physicians in fine-tuning medication dosage. As an added benefit, the same device could offer a standardized benchmark to evaluate new PD drugs and surgical procedures.

Chapter 2

Parkinson's Disease

Parkinson's disease is a progressive neurological disorder that results from degeneration of neurons in a region of the brain that controls movement (substantia nigra). This degeneration creates a shortage of the neurotransmitter dopamine, a chemical messenger responsible for transmitting signals between the substantia nigra and the next "messenger center" of the brain, the corpus striatum. Studies have shown that Parkinson's patients have a loss of 80 percent or more of dopamine-producing cells in the substantia nigra[2, 3, 4].

Normally, dopamine operates in a delicate balance with other neurotransmitters to help coordinate the millions of nerve and muscle cells involved in movement. Without enough dopamine, this balance is disrupted, causing the movement impairments that characterize the disease:

- Tremor of the hands, arms, legs and jaw, is a primary feature of Parkinson's disease. Classically, tremor appears while the individual is at rest and improves with intentional movement. The tremor often begins on one side of the body, frequently in one hand.
- Bradykinesia (slowness of movement) or akinesia (an inability to move, "freezing")
- Impaired balance and coordination, an unsteady walk with a shuffling gait, and a stooped posture.
- The severity of Parkinson's symptoms tends to worsen over time.

While no drug can stop the progression of PD, a variety of medications provide dramatic relief from its dehibilitating symptoms. These drugs work by stimulating the remaining cells in the substantia nigra to produce more dopamine (Levodopa drugs), or by inhibiting other neurotransmitters to restore chemical balance in the brain (anticholinergic drugs).

During early treatment, side effects from drug therapy are usually not a major problem. But as the disease progresses, the drugs work less evenly. As a result, many patients experience involuntary movements (dyskinesia), especially when the medication is having its peak effects. Waxing and waning of the response to the drug (wearing off effects) is also common, resulting in the reappearance of the characteristic symptoms of PD. Together, these effects form three phases of medication – The exhibition of classic PD symptoms (tremor, slow movement) when the medication has worn off (known as the "OFF" state[5]), normal movements free of tremor when the medication is balanced ("ON" state), and exaggerated involuntary movements when the medication is at highest concentration ("Dyskinesia").

Doctors work with patients to tailor a medication regimen to maximize the duration of the ON state, using direct observation of the medication cycle as the basis of adjustment. The goal of this thesis is to construct a wearable system to monitor this motion-based cycle and generate an automated daily log of a patient's medication state – see Figure 2-1 for a hypothetical output.

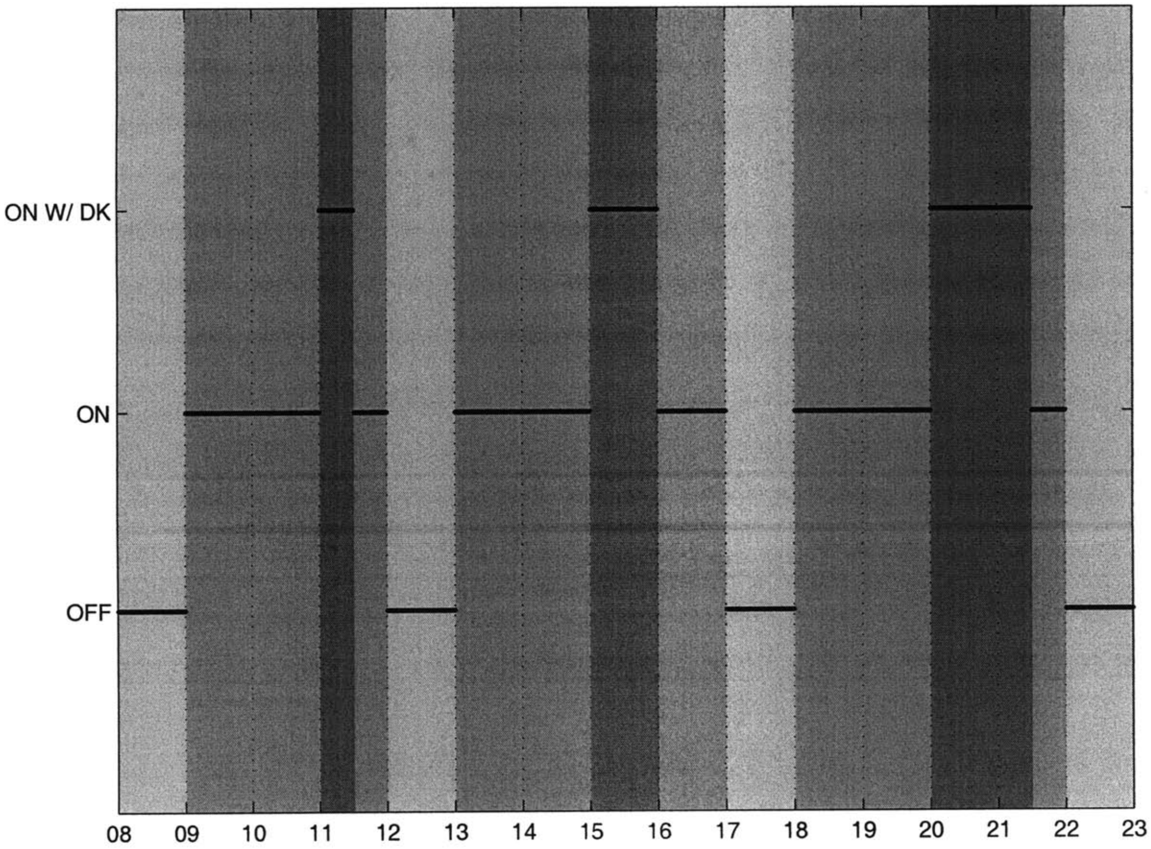


Figure 2-1: Desired system output (medication state vs. time of day)

Chapter 3

Hardware Design

This chapter discusses the hardware developed for this thesis, as well as the rational behind its design. While the initial goal was to design a generic accelerometer sensor for medical studies, the actual specifications of the project were tailored to measuring Parkinson's Disease.

After consulting with a Neurologist specializing in PD, it was determined that multiple sensors were needed, and would be attached to each wrist and ankle, with a fifth sensor worn around the waist to measure torso movement. Additionally, a minimal set of requirements was determined, ranked below in order of importance:

Rugged The sensors will be worn on potentially dyskentic patients having little or no control over their movements. All hardware must withstand basic physical abuse (cables snagging and pulling, boards being jostled and bumped, etc).

Networkable The system must support at least five sensors communicating over a shared data bus. Ideally, the low-level protocol would include some form of error detection or correction.

Light-Weight Since the device will interact mostly with elderly patients, the system should burden them with minimal additional weight.

Accurate The accelerometers must detect fine motions such as subtle tremors. Additionally, the system must sample the sensors at least at 30 times a second to guarantee accurate sampling of these events.

Easy to use The system might be loaned to doctors to collect data in their practice or patients to use while at home, so the fabricated hardware needs to be simple to assemble, test, and use.

3.1 Accelerometer Sensor Board

The sensor design used in the project is based off of accelerometer hardware previously fabricated for the MIThril wearable computing project[6]. This design was selected as a foundation due to its small footprint and portable, low-power architecture. The following describes the features of the new hardware:

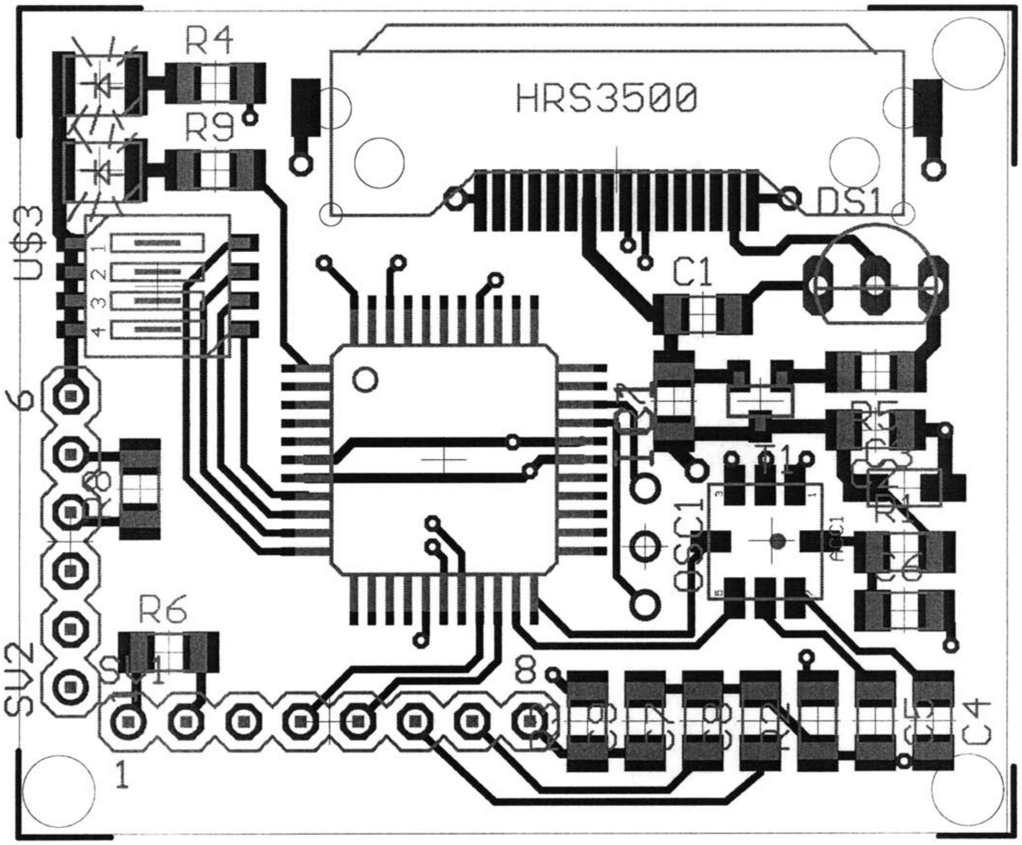


Figure 3-1: Prototype layout of Accelerometer sensor board

1. Standard MIThril header, passing regulated power, I2C data network, and DallSemi one-wire networks to the accelerometer board. The Hirose 3500 connector is rated for 10,000 insertion cycles and locks to resist physical strain.

2. The main Microcontroller was upgraded from a UltraViolet (UV) erasable PIC17C76 to a Flash-reprogramable PIC16F874. While the two processors offer a nearly identical feature-set, the newer 16F874 was included in a much smaller surface-mounted package, since Flash chips do not need physical removal for erasure and reprogramming. This decreased the overall size of the accelerometer board.

3. A four-toggle, micro-DIP switch was added to the design to allow easy selection of the device IDs uniquely identifying each device on the I2C network. Previously, each accelerometer's software included a hard-coded I2C ID, which required complete reprogramming to adjust sensor networks. Simply adjusting the DIP switches now allows up to eight devices to share the same I2C communication bus and provides visual confirmation of the current configuration.

4. The board layout uses two ADXL202E series accelerometers[7] mounted perpendicularly to give true three-axis measurements, which are sampled by the PIC16F874 Microcontroller at frequencies up to 67 hertz. Each accelerometer can measure both dynamic acceleration (vibration) and static acceleration (gravity) across a measurement range of 2g.

5. One of Dallas Semiconductor's DS2405 one-wire power switches was included to allow for additional power saving techniques. The external device utilizing the sensor boards can remotely toggle power power when motion data is not needed, extending battery life. Additionally, the one-wire device includes a globally unique 64-bit hardware ID which is useful for keeping track of specific devices used in clinical experiments.

- The accelerometer sampling algorithm was changed from constantly reading the ADXL202E to only reading on command and storing the most recent value. This new algorithm eliminates inconsistencies occurring when the sensor board published updated accelerometer values while previous values were still being read, and has the added benefit of synchronizing the recording frequency across multiple sensor boards.

- A calibration mode was added to the firmware to account for normal variations in the accelerometer manufacturing process. Before use, each accelerometer's axes were calibrated with respect to the earth's gravity to give a uniform response across all the hardware. These settings were saved in non-volatile memory inside each microcontroller.

The following sections describe these added features in greater detail and systematically works through major software design decisions. Please see figure 3-2 for an overview of the main routine.

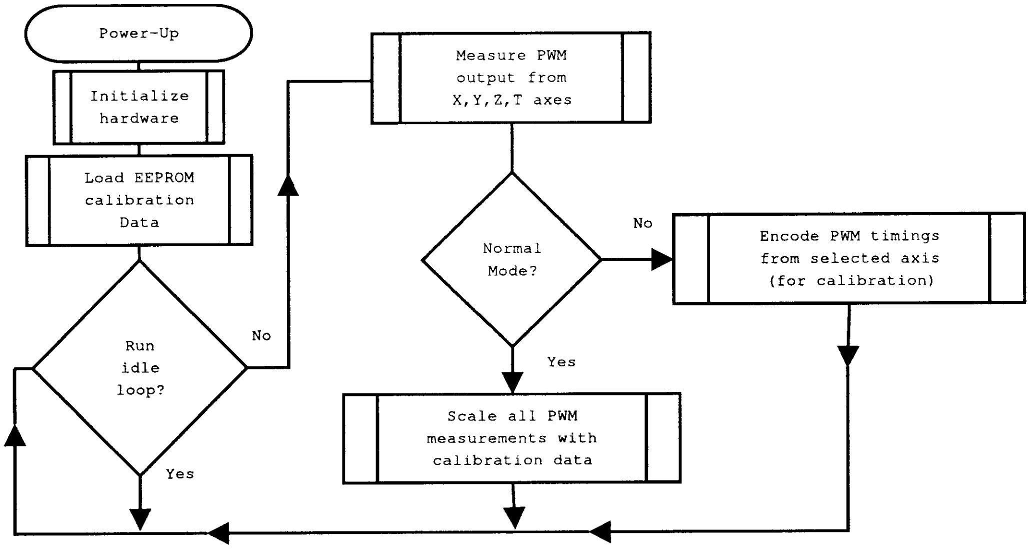

Power-Up
Initialize hardware
Load EEPROM calibration Data
Run idle loop?
No
Yes
Measure PWM output from X,Y,Z,T axes
Normal Mode?
No
Encode PWM timings from selected axis (for calibration)
Yes
Scale all PWM measurements with calibration data

Figure 3-2: Flowchart of accelerometer main routine

3.1.1 Initialization

Upon power-up, the PIC16F874 Microcontroller initializes its hardware interfaces and clears internal variables.

The code then examines the state of the 4-bit DIP switch to determine the sensor's address on the I2C communication bus. The value encoded in the switch is added to a base ID of "0xB0", used to distinguish the accelerometer sensors from other I2C devices used with the MIThril wearable. (The least-significant bit is currently dropped since the I2C protocol requires even hardware addresses; toggling the LSb represents a request for data from the I2C master device.)

Additionally, the Microcontroller is configured to respond to the non-specific "0x00" I2C address. Commands sent to this broadcast address are synchronously received by all devices on the I2C network.

Next, the PIC loads 20 bytes of calibration data from internal non-volatile EEPROM, checking to ensure critical values are within sane ranges. New microcontroller's EEPROM are arbitrarily initialized either high (0xFF) or low (0x00) when manufactured, and could potentially lock-up the device during the first boot, never allowing valid calibration data to be programmed over the I2C bus!

Finally, the PIC enables the interrupt routines handling I2C communication, and flashes an "I'm alive" blink sequence on the LED to signal successfully finishing the power-up initialization.

3.1.2 Data Acquisition

Once the initialization routine finishes, the PIC idles in a tight loop waiting for an external command signaling it to sample the four channels of accelerometer data. In this case, the I2C command "0x08" is caught by the interrupt routine, which in turn clears our idle loop and starts the PIC sampling data. Section 3.1.3 contains a detailed discussion of the interrupt routines.

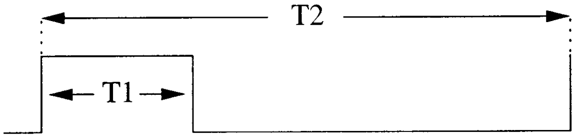


Figure 3-3: ADXL202E Duty Cycle output

The data from the ADXL202E accelerometers are encoded using a Pulse-Width Modulated (PWM) scheme, with a 50% duty cycle corresponding to 0g acting upon the device. The ratio between T1 and T2 changes in proportion to the amount of acceleration acting upon the device. With perfectly manufactured hardware, the acceleration can be calculated using the following formula:

$$A(g) = (T1/T2 - 0.5)/12.5\%$$

The PIC uses a simple polling loop and a 16-bit internal timer to measure the T1 and T2 values on the X-axis. The T2 period was set in hardware to last for one millisecond, yielding approximately 2500 counts per T2 cycle with our microprocessor running at 10MHz[1] and a resolution of 3.2mg (312 counts per g). Additionally, our polling loop algorithm's worst case running time is 2ms for a single axis.

The PIC continues this routine for the three remaining axes; the Y, Z, and redundant X-axis "T". The algorithm requires 8ms as the worst case runtime needed to time all four axes, though empirical testing shows a typical runtime of 6ms.

At this point, the execution branches based on an internal calibration mode. If the calibration mode is enabled, the firmware simply returns the T1 and T2 values for a selected axis. Each timing value is encoded as a unsigned long integer (representing the range 0-65535) and together they are

[1] Four clock cycles are required for each instruction cycle.

stored in the normal four-byte output of the system. Please refer to section 3.1.4 to see how these values are used to calculate the calibration constants loaded on startup.

Assuming we are instead in a normal execution mode, the PIC emulates floating-point math to convert the ±2g PWM data into values mapped across a signed byte [-128 to 127]. The aforementioned calibration data is used during these calculations to reduce the impact of slight differences in the fabrication of each accelerometer. Since this particular microcontroller does not contain a hardware floating point unit (FPU), these calculations take a significant amount of time. Almost seven milliseconds are spent simply number crunching.

Once these four values are calculated, the PIC disables interrupts to prevent data inconsistencies, publishes the temporary calculations for download, re-enables interrupts, and then returns to the beginning of the main routine to wait in the idle loop.

A worst-case total of 15ms are required to measure and calculate the calibrated accelerometer output, limiting our maximum sampling rate to 67 hertz.

3.1.3 I2C Interrupt Routine

This interrupt routine compliments the previously described data acquisition code by interacting with the external device controlling the sensor board. This routine is responsible for two major tasks: using the microcontroller's embedded communication hardware to implement the I2C protocol, and executing the higher-level commands transmitted to the board using I2C.

The generic I2C protocol specifies transactions consisting of an address byte (identifying particular hardware) followed by an arbitrary number of data bytes[8]. This thesis extends the generic protocol by designating the first non-address byte as a command to the firmware with optional subsequent data bytes. Table 3.1 lists the additional commands:

Table 3.1: I2C commands supported by sensor board

Command (hex)	Description
0x00	Toggle LED Off
0x01	Toggle LED On
0x02	Return X value on next read
0x03	Return Y value on next read
0x04	Return Z value on next read
0x05	Return T value on next read
0x06	Next byte selects calibration mode {normal, X,Y,Z,T}
0x07	Flashes subsequent 20 bytes of calibration data in EEPROM
0x08	Request for system to acquire accelerometer data

The interrupt routine is triggered by internal microcontroller hardware receiving a byte through valid I2C communication. Since the built-in hardware is blind to the higher-level aspects of the I2C protocol, the firmware must keep track of the communication state.

The interrupt routine first determines if the received hardware address byte is a request for data, signaled by setting the Least Significant bit of the address. If this byte indicates a read request, the current index of the cyclic output buffer is transmitted over I2C, and is subsequently updated to refer to the next location in the buffer. If instead there is no request for data, the received byte is our I2C address, so the I2C state is adjusted from NOTHING to ADDRESS_RECEIVED.

After verifying that a valid command byte has been received, the interrupt routine then either directly handles a simple request (such as turning on an LED) or updates the I2C state to reflect one of two multi-byte commands (CALIBRATION_MODE or CALIBRATION_DATA).

With CALIBRATION_MODE, the interrupt routine expects a single additional byte used to designate normal operation of the accelerometer (0x00) or to select the raw data of the specified axis be returned for calibration (0x01-0x04). In CALIBRATION_DATA, the routine programs the subsequent 20 bytes of calibration data into the microcontroller's EEPROM. These 20 bytes are further described in the following section, section 3.1.4.

After the required number of bytes have been read, the I2C state is reset to NOTHING, and the communication process begins anew.

3.1.4 ADXL202E Calibration

While Analog Devices holds the ADXL202E hardware to precise production tolerances, it is impossible to avoid slight variations in the manufacturing process, resulting in minor timing discrepancies between specific accelerometers. To correct this problem, the sensor boards were programmed to calibrate each accelerometer with respect to gravity to give a uniform response across all hardware. This section describes the external software used to interrogate the sensor board and generate the calibration values used during data acquisition. The following algorithm is a slight variant from an Analog Devices application note[9].

An external program collects multiple samples from each axis of the on-board accelerometers responding to ±1g, using the Earth's gravity as a reference. The program prompts the user to hold the accelerometer board in the proper orientations, collecting needed data at each stage. After eight orientations (±1g for four axes), the program uses the formulas described below to generate calibration constants, then flashes the new configuration data over the I2C link into on-board EEPROM.

These new calibration values are used once the sensor board reboots.

Three values are needed per axis to ensure calibrated output:

T2cal The averaged value of duty-cycle period (T2) during the calibration procedure.

Zcal The 0g value of duty-cycle output (T1) at the time of calibration, calculated using the following formula:

$$Zcal = \frac{T1max - T1min}{2}$$

K The scaling factor used to ensure the proper resolution (in bits) of the accelerometer calculation. To ensure ±2g is mapped to ±128 counts (to result in an 8-bit number), K is calculated by:

$$K = \frac{T2cal * 128}{T1max - T1min}$$

Since the ADXL202's duty cycle modulator uses the same reference for both axes, T2cal is averaged and only stored once per accelerometer. Zcal and K are calculated for each axis, resulting in 10 values: $[T2cal_1, T2cal_2, Zcal_X, Zcal_Y, Zcal_Z, Zcal_T, K_X, K_Y, K_Z, K_T]$. Two bytes are required to store each calibration constant, resulting in total of 20 bytes of data flashed into EEPROM.

Once the calibration constants are known, only two formulas are required to calculate acceleration from a T1 and T2 measurement:

$$Zactual = \frac{Zcal * T2}{T2cal}$$

This formula accounts for changes in T2 due to drift or jitter by using the averaged values from exposure to 2g of acceleration. Calibrated, scaled acceleration can then be calculated using:

$$Acceleration = \frac{K * (T1 - Zactual)}{T2}$$

3.2 Data-recording Hardware

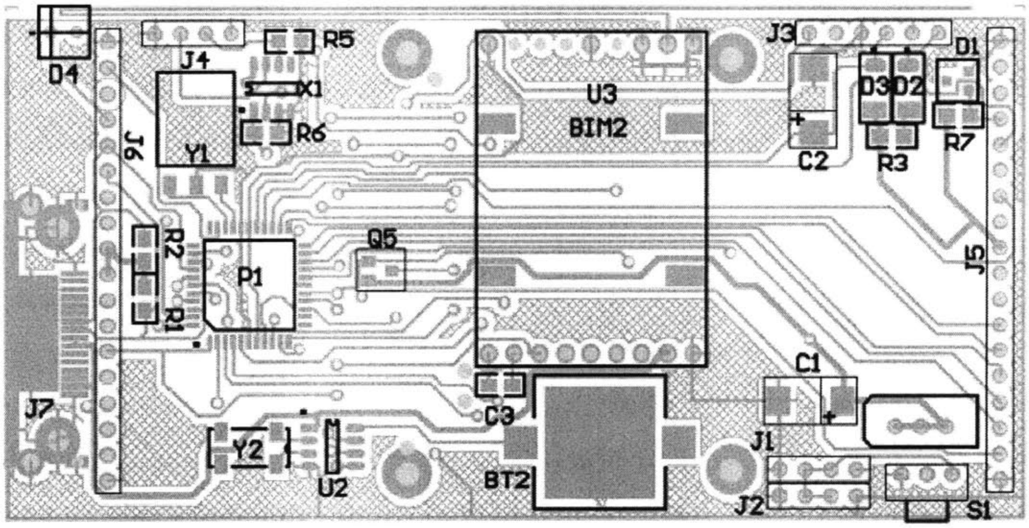


Figure 3-4: SAK hardware

Rather than using the full MIThril wearable, which includes multiple single-board computers, wireless networking hardware, and a head-mounted display, Vadim Gerasimov's "Every Sign of Life" (ESL) board[10] was chosen as a light-weight data recording system.

The main feature of this 4-inch by 2-inch board is a Microchip PIC16F877 microcontroller interfacing with a general purpose Compact Flash (CF) header. The embedded software allows data to be stored either on traditional Type-I CF cards or on the higher capacity IBM microdrives, allowing for up to one Gigabyte of storage.

Each series of measurements recorded by the ESL board is timestamped using a Maxim DS1302 timekeeping chip. This real-time clock provides full calendar information with second resolution and is supplementally powered by an lithium-ion backup battery capable of running the clock for five years. While the clock is accurate within a few minutes per month of operation, the ESL board allows the chip to be synchronized with a PC's clock via the serial port.

Finally, the ESL board can be expanded with a 2-way FM radio module or with custom daughter boards plugged into standardized headers.

Four rechargeable AAA batteries power the ESL board, and typically last for 36-hours of constant use. Recording at 50 hertz from 5 accelerometers – one on each limb, plus one as an ESL daughter board worn in a belt pouch – allows approximately 17 hours of data to be recorded onto an inexpensive 64 megabyte CF card.

3.2.1 ESL software Modifications

Vadim's original ESL code was modified to record data from the five of the accelerometer sensor boards described previously. The flowchart in Figure 3-5 illustrates the new data sampling routine:

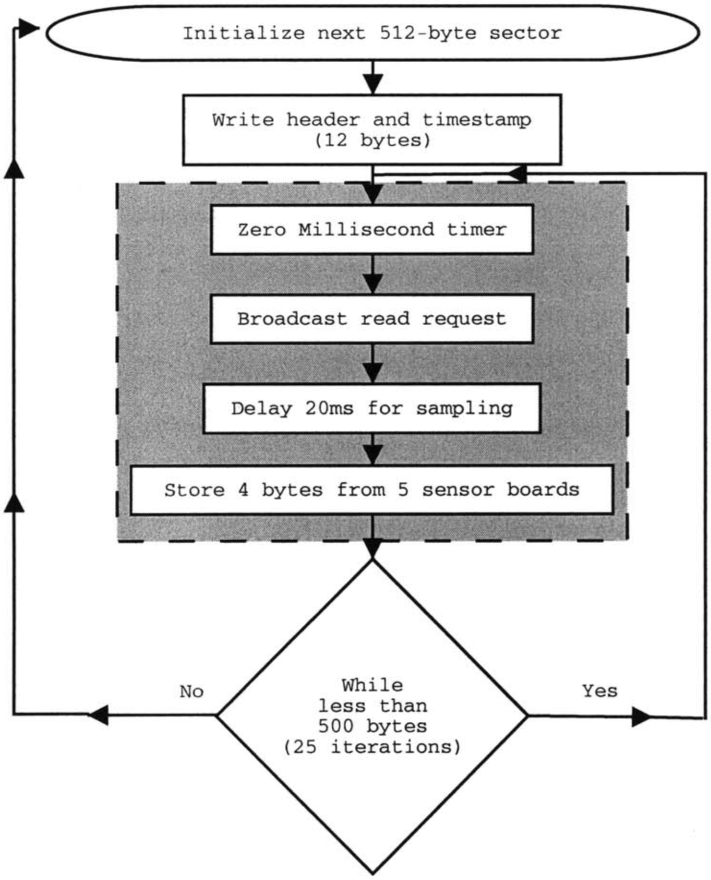

Initialize next 512-byte sector
Write header and timestamp (12 bytes)
Zero Millisecond timer
Broadcast read request
Delay 20ms for sampling
Store 4 bytes from 5 sensor boards
No
While less than 500 bytes (25 iterations)
Yes

Figure 3-5: Accelerometer sampling routine

The output is stored on the attached Compact Flash card as a single file in a FAT-16 filesystem. A secondary program running on a desktop computer converts the binary file into human readable format suitable for use with Matlab.

3.3 Packaging

Finding a way to attach the system to the human body was the final challenge in designing the hardware. The major tradeoff was between comfort during multiple-hour wear and a tight fit for a reliable coupling of accelerometer to limb. After much investigation, athletic wrist and elbow support equipment was found to be a ideal balance between comfort and snug fit. Velcro was used to attach the accelerometer boards to the athletic straps (Figure 3-6).

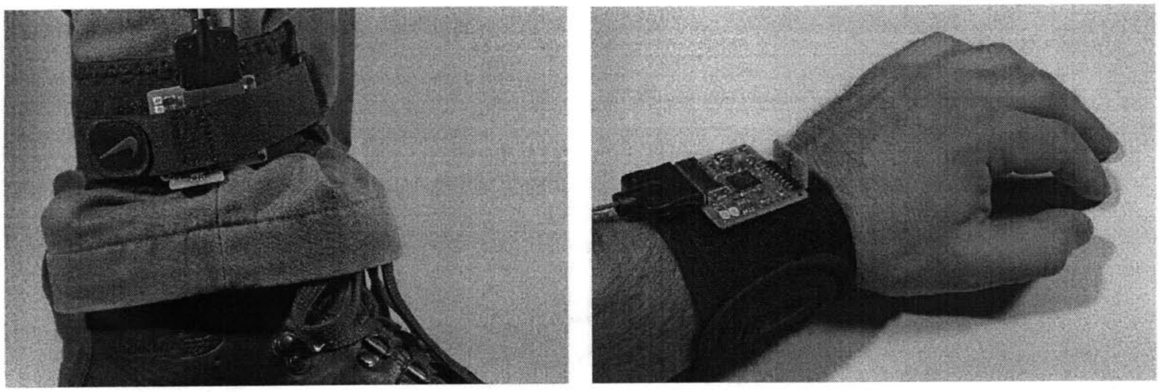


Figure 3-6: Velcro ankle and wrist mounting hardware

Running the wires between the components was also an issue. The amount of cabling was first reduced by constructing a custom wiring harness with light-weight, four-conductor cable. Cables for the upper body were kept out of the way by running them through the patient's shirt and out the sleeves. Wires for the ankle accelerometers were run over the patient's pants and held in place with strips of self-adhering ace bandage.

3.4 Testing

Individual accelerometer boards were fabricated, calibrated, and tested in isolation. These boards were then connected to the ESL board stack, tested on the bench, and worn around the Media Lab. The cables and locking connectors were tested to ensure simple snags would not result in dropped data[11, 12].

Chapter 4

Experiment

Two volunteers, located through Memorial Hospital's Parkinson Day Center , were instrumented with the data recording system while they performed common daily tasks (walking, sitting and reading quietly, and sitting in animated conversation). Particular care was taken to record each volunteer performing the same activities during the three phases of their medication (Off, On, Dyskinesia). For verification, each patient was filmed with a digital video camera synchronized to the system's real-time clock.

Additional care was taken to consistently place the accelerometer sensors in the same location on each patient. Even though the system could run for over 24 hours, batteries were replaced at the start of each data collection. Additionally, even though the system could record 17 hours of accelerometer data, it was powered down at a halfway point for the CF card to be removed and backed up onto a laptop.

Figure 4-1 displays all data recorded from the first patient, organized by accelerometer location.

The volunteer's medication state was rated every ten minutes by a trained physician observing the experiment. The physician filled out two separate test based on visual observation of the subject, classifying "On Vs Off" behavior separately from "Dyskinetic" motions.

Figure 4-2 displays the plotted observations for Subject 1.

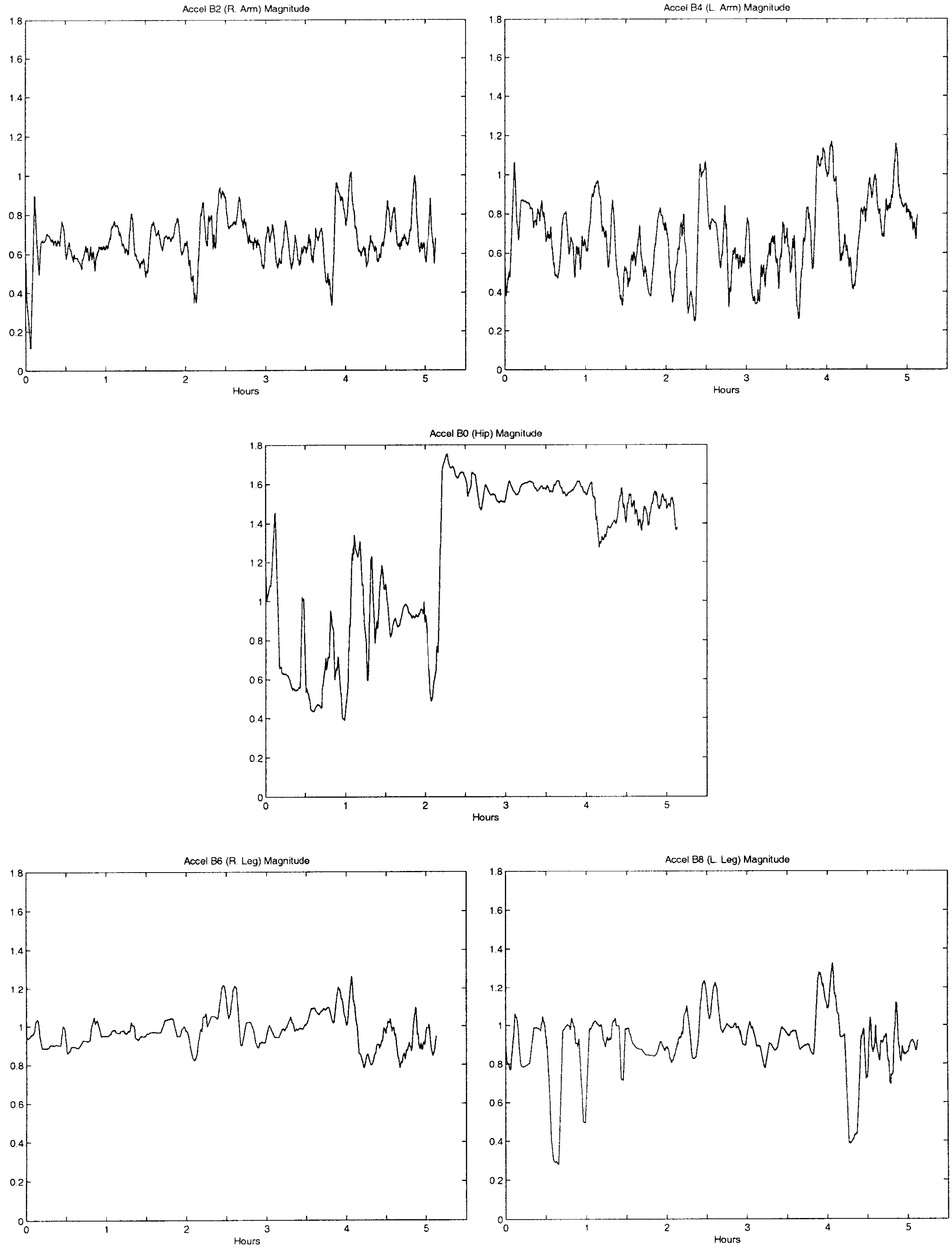


Figure 4-1: All accelerometer data recorded from Subject 1, smoothed and displayed by sensor location. Sensor B0 shifted two hours into data collection, resulting in the plot offset. The data from Subject 2 looks similar, though without the shift in B0.

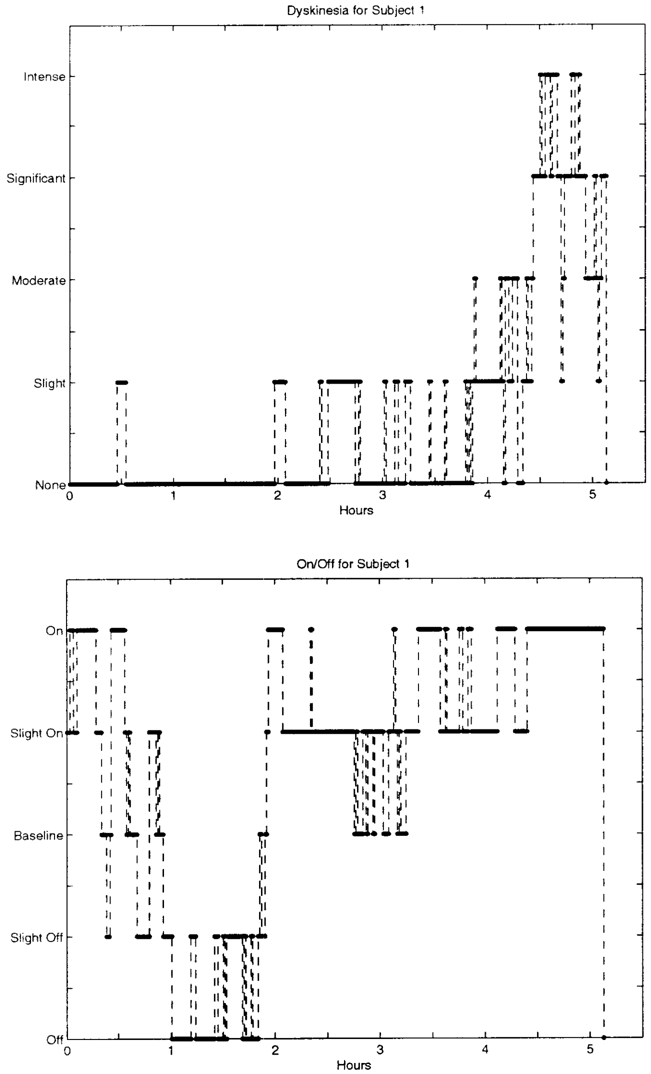


Figure 4-2: Physician's observations for Subject 1.

4.1 Analysis

The goal of this section is to analyze the accelerometer data to determine:

- If the system can detect the different phases of medication associated with Parkinson's Disease.
- Which accelerometer placement(s) on the body are best suited to this task.

Unfortunately, at the time of writing this thesis, the physician has only fully annotated the first subject's dataset. As a result, all analysis of the system is based on results from this patient.

4.1.1 Correlation in Matlab

Each accelerometer produced a four-component signal vector composed of unsigned bytes (0-255), sampled at approximately 40 Hz. A preprocessing step averaged the two redundant components of the four-axis accelerometer to produce a three-component vector. A script was written to compute the magnitude of the delta's of adjacent vectors, representing quality of motion as the strength of the change in acceleration.

The physician rated the patient's dyskinesia on a scale of 0-4, with 4 being the most dyskinetic.

Visual observation suggested a high correlation between the accelerometer data and physician's observations. Direct comparison proved difficult – the physician's observations were recorded every minute while the accelerometers sampled data at 40 hertz. An averaging filter with a three minute "window" was first run across the data sets before they were compared.

Figure 4-3 shows the superposition of the physician's observations of Dyskinesia overlaid on the smoothed magnitude of accelerometer B2, which was mounted on the subject's right arm.

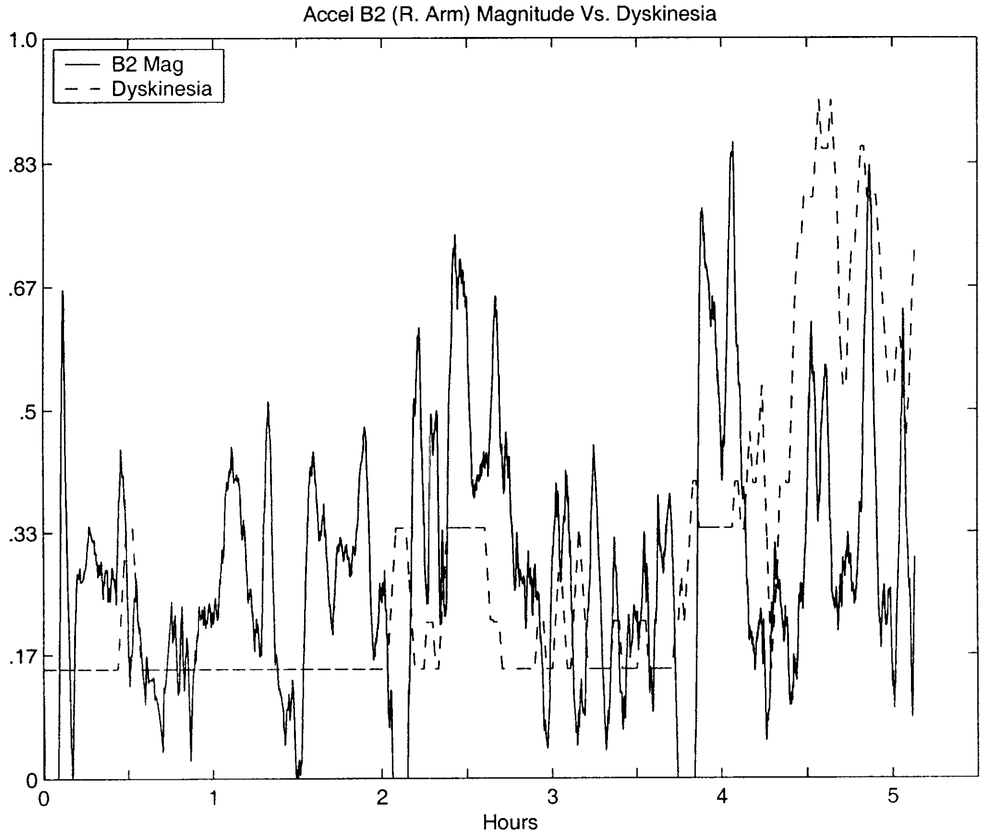


Figure 4-3: Right-arm accelerometer and dyskinesia data smoothed using three-minute window. There is an 82% correlation.

Initial analysis of the smoothed data showed correlation between the accelerometer outputs and dyskinesia as observed by the physician. Specifically, the highest correlation was found with the accelerometer on the right arm, which produced an 82% correlation.

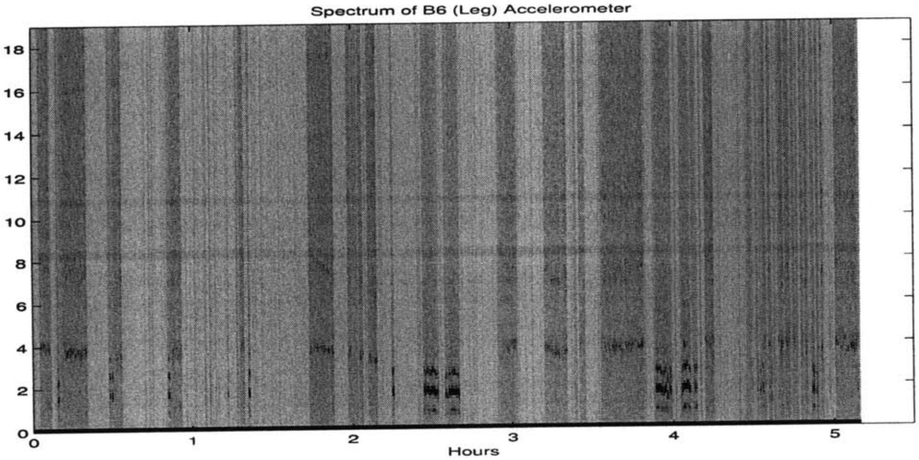
Unfortunately, this simple technique falsely assumes any period of rapid motion directly corresponds to dyskinesia. It would, for example, be unable to differentiate a sedentary patient with high dyskinesia from a properly medicated patient going for a walk.

There are two ways to deal with this problem:

- Constrain the experiment to only compare data recorded during similar tasks (walking, for example).
- Further analyze the accelerometer data to guess the patient's current activity from the constrained set of tasks.

4.1.2 Detecting Walking

In his paper, "Real-Time Motion Classification for Wearable Computing Applications"[13], Rich DeVaul used a single three-axis accelerometer to classify a range of user activity states (sitting, walking, running, biking, riding the subway). Following his methodology, the following is a power spectrum of the waist-mounted accelerometer:



This plot provides a visual means to pick out periods where the patient walked (vertical bars), suggesting that an automated system could perform the task. Inspired by this apparent correlation, a Neural Network was developed to autonomously identify periods of walking.

The video record was examined to determine the precise timestamps of walking activity, a task greatly simplified by the use of synchronized clocks on all devices. The regions of walking were flagged with a dataset parallel to the accelerometer data, and both datasets were subdivided to form training and test sets. Each dataset was subdivided into 5 second windows, with 3.75 seconds used for training, and 1.25 seconds used for testing. These numbers were chosen to allow a 3:1 training-to-test set ratio. Unfortunately, there was not enough data to form a separate "evaluation" data set, so all algorithms were evaluated against the test set.

Using the Matlab Neural Network Toolkit (nntool), several multi-layer neural networks (NNs) were created. The immense size of the data set (800,000 bytes per axis) severely limited the network topologies Matlab would simulate. In fact, Matlab would crash on any networks with greater than eight input nodes or more than a single neuron in the hidden layer. Correspondingly, all of the NNs used in this thesis included a hidden layer containing a single neuron, and an output layer also containing a single neuron.

Initial experiments with walking used three separate networks trained on logical sets of accelerometer data: both hands, both legs, and the hip. While these results were not ground-breaking (70%,71%,56% correlation), the corresponding weight vector of each NN was examined to find which inputs were the most important. Seven axes were found to be the most useful: x & y axes of the right arm, and right and left leg. Additionally, the z-axis of the left leg was deemed important.

A final NN was designed using these seven inputs (with 3/4*800,000 raw data points per axis), and trained against the training data based on periods of observed walking. After training, the NNs were run on the test data, and their output compared to the observed walking times in the test data.

Figure 4-4 shows the output of the best walking detector, which correctly classified walking 75% of the time.

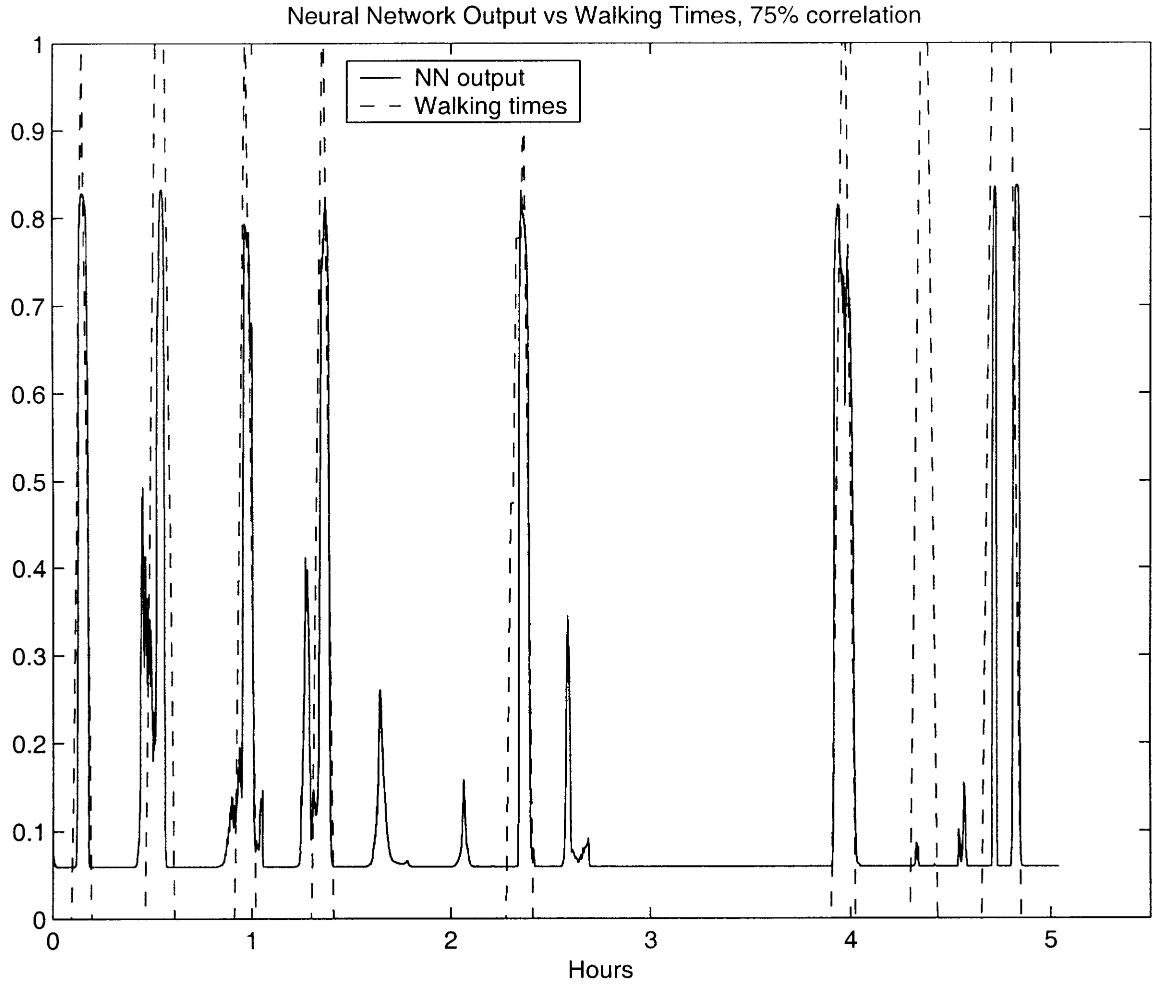


Figure 4-4: Output of neural network designed to autonomously identify regions of walking, plotted against corresponding regions of walking. There is a 75% correlation.

4.1.3 Detecting Dyskinesia.

As mentioned above, the initial analysis of the data determined an 82% correlation between the accelerometer data and observed dyskinesia. This high correlation suggested that an automated system could be used to determine dyskinesia based on the outputs of the accelerometers.

Following the methods used to determine regions of walking, the accelerometer output and the dyskinesia observations were subdivided into training and test sets. A NN was created in Matlab and trained on the output from all 5 accelerometers against the physician's dyskinesia observations. Testing this network revealed a 71% correlation between the NN output and dyskinesia.

The correlation of the raw data (82%) suggests that the NN should be able to do better. Looking at the physician's dyskinesia data provides some insight: the data includes measures of "slight," "moderate," "significant," and "intense" dyskinesia. Restricting this dataset to exclude "slight" dyskinesia and using it to train the NN has dramatic results; the correlation between this NN output and all observed dyskinesia (including "slight") is 85% (see Figure 4-5).

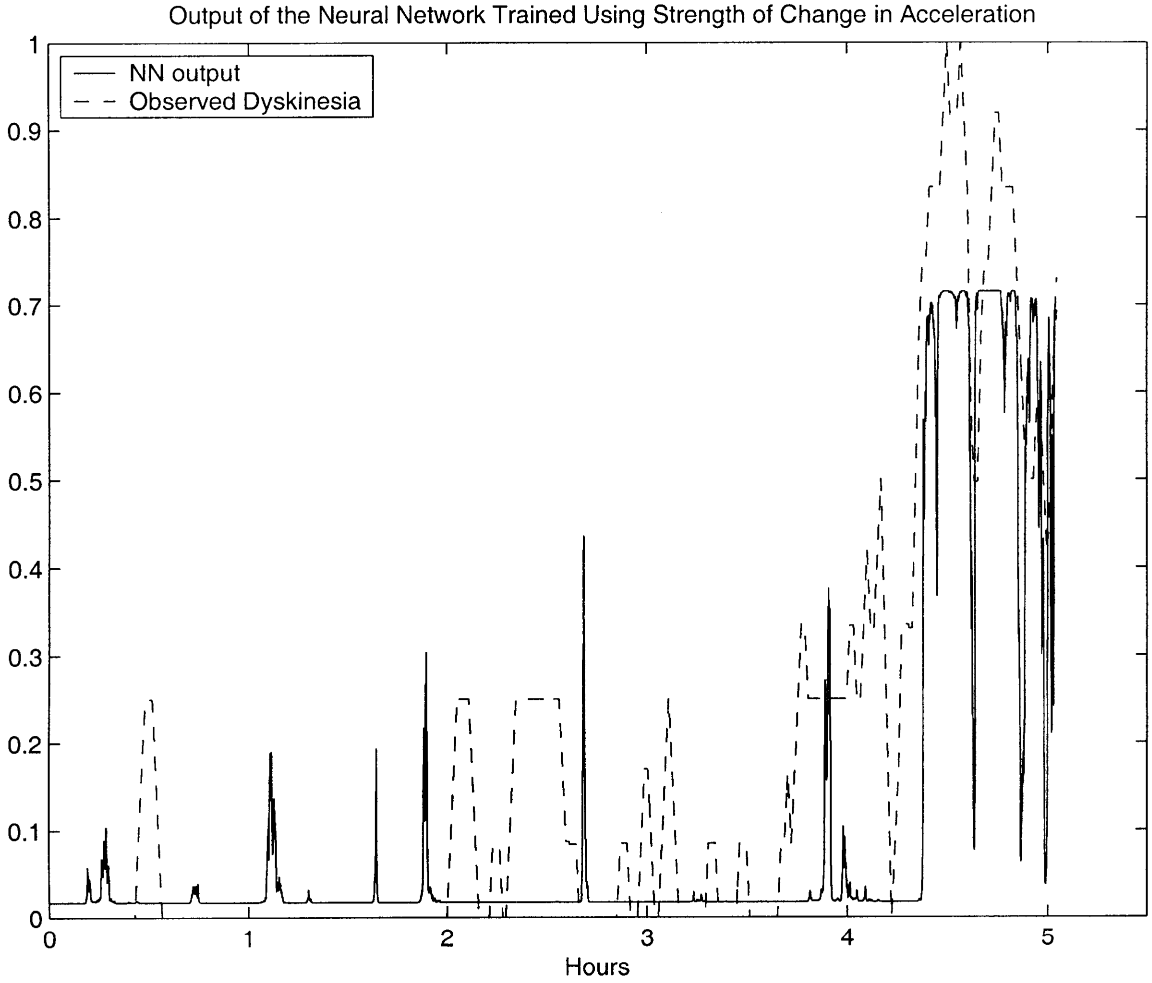


Figure 4-5: Output of a neural network, trained using only dyskinesia levels "moderate," "significant," and "intense" with all 5 accelerometer outputs, and plotted against all observed dyskinesia levels. There is an 85% correlation.

4.1.4 Improved Dyskinesia Detection

By combining the results of sections 4.1.2 and 4.1.3, dyskinesia can be predicted with even higher accuracy. Using the NN from section 4.1.2, it is possible to isolate periods of walking from the original dataset and remove them. Retraining the NN from section 4.1.3 on this data provides a correlation of 91% with dyskinesia (see Figure 4-6).

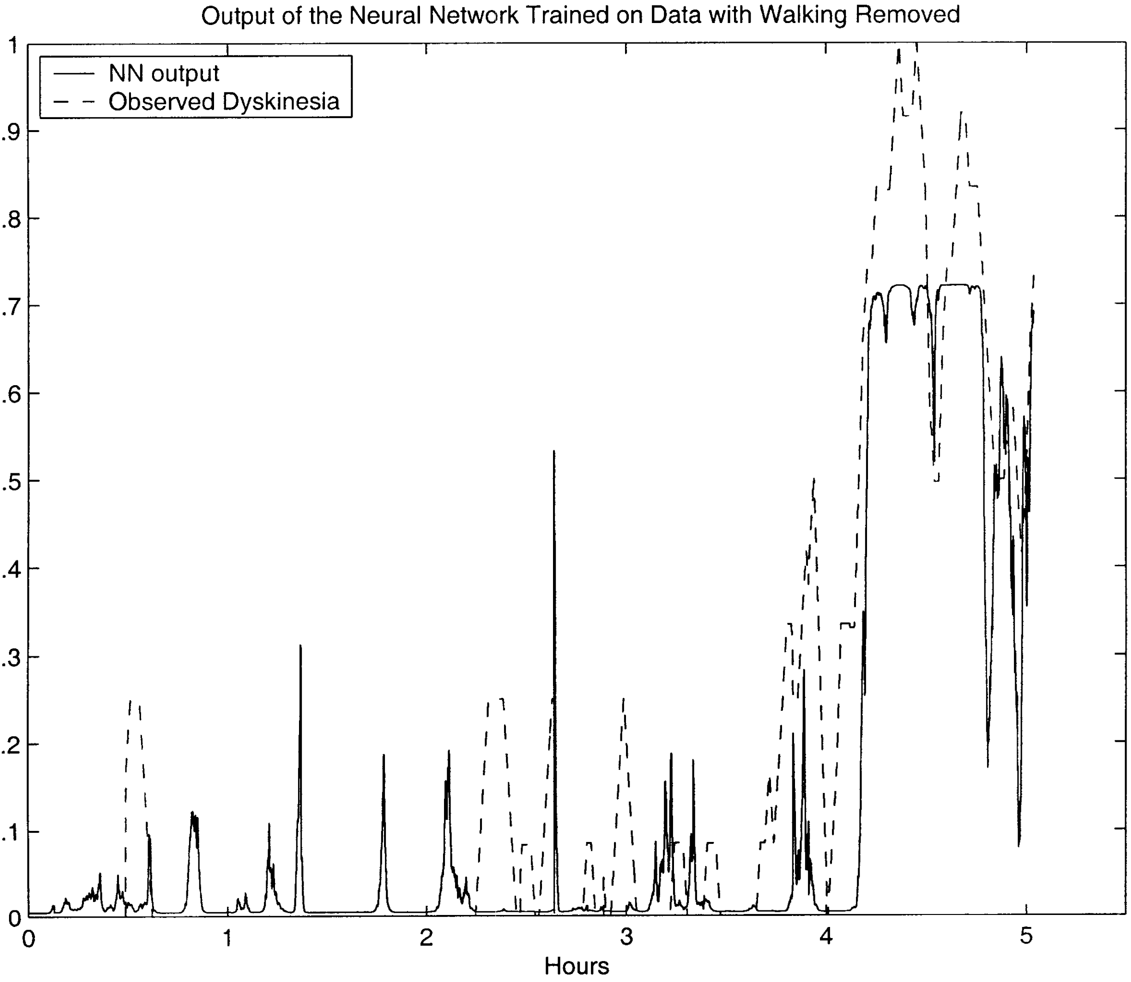


Figure 4-6: Output of a neural network, trained after automatically identifying and removing walking data. This NN was trained only using dyskinesia levels "moderate," "significant," and "intense" and all 5 accelerometer outputs, and is plotted against all dyskinesia levels. There is an 91% correlation.

Chapter 5

Conclusions

The main roadblock in the treatment of Parkinson's is proper dosage of the medications which provide relief from the symptoms. By automatically recording dyskinesia levels through different stages of the patient's medication cycle, doctors can more effectively adjust dosages to each patient's individual requirements. This thesis developed a way to automatically record and analyze such data.

The core hardware is a wearable medical data collection system, based off of low-power, calibrated accelerometer sensors. Five of these sensors were connected to a Compact-Flash based data recording board and successfully tested in an actual clinical setting. The system recorded ten hours of data from two volunteers with PD as they performed everyday activities.

This data was examined in the hopes of finding a relationship between the subjects' motions and their corresponding medication state, as rated by a physician observing the experiment. A preliminary test found 82% correlation between a single accelerometer and the Dyskinetic ratings, suggesting promise for in-depth analysis.

A neural network using all five accelerometers generated 85% correlation. This network was, however, limited by confusion between a sedentary patient with high dyskinesia and a properly medicated patient moving energetically. A second neural network was trained to identify periods of walking (the only vigorous activity in our data set) with 75% accuracy. Using this activity classifier to remove walking data increased our overall accuracy to 91%.

These results must be tempered by the recognition that they are based off a single data set, and the Neural Network that performed best on the test set was used (vs. testing the network on an entirely unseen set of data). However the recognition results are still significant, and suggest three important lessons for working with accelerometer based PD systems:

- Activity classification is an important aspect of the classification system.
- Multiple accelerometers across the body show significant improvement over a single accelerometer located anywhere on the body.
- The classification systems will be strongly tuned to each subject, as every person performs basic movements differently (ie, walking rhythms)

The automated monitor system developed in this thesis, combined with the artificial-intelligence based learning tools described, provide a way to successfully identify dyskinesia with significantly greater than random accuracy, achieving 91% in the patient studied here. This result has the potential to replace human observers in the monitoring of medication levels for Parkinson's patient. By removing the necessity of a human observer, more accurate data can be taken more frequently, hopefully allowing doctors to provide more effective treatments and dramatically improving the lives of people living with Parkinson's Disease.

Bibliography

[1] National Institute of Neurological Disorders and Stroke. Parkinson's Disease – Hope Through Research, July 2001.

[2] American Association of Neurological Surgeons. Parkinson's Disease, March 2000.

[3] Mayo Foundation for Medical Education and Research. What Is Parkinson's Disease, July 2002.

[4] National Institute of Neurological Disorders and Stroke. Parkinson's Disease Backgrounder.

[5] CD Marsden and JD Parkes. "On-off effects" in patients with Parkinson's disease on chronic levodopa therapy. *Lancet*, pages 292–6, Feb 1976.

[6] Rich DeVaul, Sandy Pentland, and Steve Schwartz. MIThril, The Next Generation Research Platform For Context Aware Wearable Computing. http://www.media.mit.edu/projects/wearables.

[7] Analog Devices Inc. http://www.analog.com/productSelection/pdf/ADXL202E_a.pdf.

[8] Phillips Semiconductors. The I^2C-Bus Specification, January 2000.

[9] Analog Devices. Using the ADXL202 Duty Cycle Output, 2002. Application Note AN-604.

[10] Vadim Gerasimov. Every Sign of Life. http://vadim.www.media.mit.edu/ESL/esl.html, Jan 2002.

[11] Matthew B. Debski. The SNAP! toolkit for developing sensor networks and its application to cross-country skiing. Master's thesis, Massachusetts Institute of Technology, MAY 2000.

[12] J. Rantanen, N. Alfthan, and J. Impio. Smart clothing for the arctic environment. *Fourth International Symposium on Wearable Computers*, pages 15–23, oct 2000.

[13] Rich DeVaul and Steve Dunn. Real-Time Motion Classification for Wearable Computing Applications. http://www.media.mit.edu/wearables/mithril/realtime.ps.gz, dec 2001.