April 13, 2005

Mr. Lakshman One
School of Engineering Science
Simon Fraser University
Burnaby, British Columbia
V5A 1S6

Re: ENSC 440:ThinQ Innovation Post-Mortem

Dear Mr. One:

Please find the attached document, *ThinQ Innovation Post-Mortem,* which briefly outlines the progress we made this semester in designing and implementing a P300 Brain-Computer Interface. As you know, we were successful in allowing able-bodied persons to spell with differing amounts of accuracy and time. Our most successful subject achieved 100% accuracy, communicating a symbol approximately every 15 seconds.

In the attached document we outline the obstacles we faced and give perspectives of team dynamics. Also, we briefly consider the team's future direction. Each member of the team then gives his personal opinion on the events of the past 3 months, outlining among other things new technical, professional and personal skills acquired.

ThinQ Innovation is comprised of five senior-level engineering students, each with unique strengths to complement the group and a common trait in determination. We are Jack Cha, Jae-Seok Jeon, Brian John, Min Seo and Jyh-Liang Yeh. ThinQ Innovation looks forward to consulting with you on any questions or concerns you may have. Should you have any questions, please feel free to contact us at ensc440-project@sfu.ca.

Best Regards,

Brian John
Project Lead

Enclosure: *ThinQ Innovation Post-Mortem*

ThinQ Innovation

Post- Mortem

Project Team Members: Jack, Cha
Jaeseok, Jeon
Brian Henry, John
Min, Seo
Jyh-Liang, Yeh

Contact: ensc440-project@sfu.ca

Submitted to: Mr. Lakshman One – ENSC 440
Mr. Mike Sjoerdsma – ENSC 305

Issued date: April 14, 2005

Revision: 1.0

Table of Contents

Table of Figures and Tables

1. Introduction

Since January 2005, ThinQ Innovation has been working on a P300-based brain-computer interface in an attempt to solve communication problems for the completely paralyzed or 'locked-in' patients. Recently, we got our project working and have had success in communicating.

2. Current State of the Device

Our P300 spelling device properly operates at a level of which all expectations were exceeded. Right now, the system is set up at Professor McDonald's lab in Psychology department and consists of a PC, a DAQ card, an amplifier, an electrode cap, few connectors and operating software. An additional PC was added into the network to provide dual screens and administrative controls.

All components are in excellent state where only the DAQ card and connectors belong to the group members, and the remaining equipment is due back to the generous lenders in Engineering and Psychology departments.

3. Results Achieved and Deviation from Expected

3.1. Overall System

The functionalities outlined in the functional requirements have all been met. More so, we have exceeded online system performance to 100% accuracy within 3 trials instead of 100% accuracy within 40 trials as defined in the functional specification. To the extent of our knowledge, our current prototype is the fastest online P300 Spelling Device to date.

3.2. Performance

According to the functional specification, our goal for offline data detection was to achieve 99% accuracy within 10 trials using less than or equal to 10 channels. We have met and exceed this requirement. Current prototype is able to achieve 100% accuracy within 3 trials in the best case (subject Jyh-Liang) and 90% accuracy within 15 trials in the worst case (subject Lucky) for both online and offline system. We have also implemented a word prediction driver, which displays a selection of matching words after the first letter is identified. As well, we are able to expand the database manually. Lastly, we have

implemented an option for saving models, training data as well as raw data directly from the DAQ onto offline files that characterize different individuals.

3.3. Input/Output/Physical System

We have met all of the input/output/physical system functional requirements as defined in the functional specification.

3.4. Deviations

We have exceeded our requirements by:

1. Improving offline accuracy/trials from 99% within 10 trials using 10 channels to 100% within 3 trials (best case) using 10 channels

2. Improving online accuracy/trials from 99% within 40 trials using 10 channels to 100% within 3 trials (best case) using 10 channels

3. Implementing word prediction driver with additional features such as manual ranking option, word addition option (adding into dictionary), and word deletion option (deleting a word from the database)

4. Implementing extra software functionalities such as option for saving not only model files but also training data and raw data acquired directly from DAQ

5. Extra functionalities include extra display monitor and DAQ waveform display

6. Implemented extra monitoring systems for hardware timers, API overheads, and memory usage

3.5. Testing

In our attempt to further improve our prototype, we have conducted a number of experiments. As a result, we were able to better characterize our prototype. We briefly outline the tests we have conducted.

3.5.1. SVM Implementations

Here, we outline the five approaches that were explored in implementing a multi-channel SVM algorithm for our P300 Spelling Device. The five approaches are: 1) averaging method; 2) three-dimensional array method; 3) concatenation method; 4) multi-model method and 5) weighted concatenation method.

We use two measures of performance in determining which of these three approaches yield best result; they are accuracy and time efficiency. We used the same sets of training data (2340 epochs) and testing data (1820 epochs) to ensure that accuracy and time efficiency of the five approaches are measured without bias.

Note that accessing *mexSVMClass* function (i.e. SVM software libraries) allows direct access to a variable, *ClassRate*, within which the percentage accuracy of the testing data set (1820 epochs) is computed and returned. Time efficiency is determined by using MatLab's profiling function.

All of the testing described in this document is carried out in MatLab. Therefore, it is essential that training and testing data are in *.mat* format. Also note that the rows (144) represent the number of attributes and the columns (training data: 2340; testing data: 1820) represent number of data epochs used in training/testing. Lastly, it is assumed that the training data have approximately equal number of P300 presence and absence. This is important since using a biased training data will lead to skewed distribution of data thereby negatively affecting the accuracy.

3.5.1.1. Averaging Method

The training and testing using averaging technique is slightly different from single-channel implementation in that, prior to either training or testing, all of the training/testing data are added and divided by the number of epochs (for training 2340; for testing 1820). The dimensions of resulting data are exactly the same as that of the single-channel implementation that is, training data is an array of 144-by-2340 and testing data is an array of 144-by-182.

Accuracy: 61.538% (tested with 1820 epochs)
Time Efficiency (Training): 8.45 seconds (2376MHz CPU)
Time Efficiency (Testing): 0.58 seconds (2376MHz CPU)

3.5.1.2. Three Dimensional Array Method

The motivation behind this approach is that combining one epoch from each of the 10 channels with one corresponding class label before training may provide more attributes. Having more attributes implies that SVM algorithm utilizes higher dimension feature space thereby giving higher degree of freedom to which it can separate each of the training data. Though theoretically sound, this approach was scrapped because it required complicated array manipulations to begin with, and it also was not compatible with the *mexSVMTrain* and *mexSVMClass* functions.

Accuracy: N/A
Time Efficiency (Training): N/A
Time Efficiency (Testing): N/A

3.5.1.3. Concatenation Method

The training and testing using concatenation technique is different from single-channel implementation in that, prior to either training or testing, all of the training/testing data are stacked on top of each other in such a way that the resulting training/testing data have 1440 attributes instead of 144. This is desirable since the number of attributes is ten-fold and SVM maps the training data into even higher dimension feature space where it has higher degree of freedom. It is notable that this principle is similar to that used in three-dimensional array approach.

Accuracy: 82.418% (tested with 1820 epochs)
Time Efficiency (Training): 41.94 seconds (2376MHz CPU)
Time Efficiency (Testing): 3.27 seconds (2376MHz CPU)

3.5.1.4. Multi-Model Method

In this approach, SVM makes no distinction from where the data is originated. Therefore, it operates exactly the same way as it were in single-channel implementation with 10 times the data. In other words, there will be 10 separate SVM engines; each classifies one of the 10 channel data. Obviously, both the training and testing takes much longer because of this fact, but the accuracy did not improve.

Accuracy: 65.22% (tested with 1820 epochs)
Time Efficiency (Training): 10+ minutes (2376MHz CPU)
Time Efficiency (Testing): 42.52 seconds (2376MHz CPU)

3.5.1.5. Weighted Concatenation Method

According to Mattias Kaper, the author of *BCI Competition 2003 – Data Set IIb: Support Vector Machines for the P300 Speller Paradigm*, Pz, P07 and P08 sites, on the scalp, exhibited strongest amplitude near 300ms after stimulus. He also suggests that strong amplitude at P07 and P08 sites cannot be conclusively determined to be P300 response-related. Nonetheless, it is worthwhile to make an effort to put more emphasis in the signals that exhibit stronger response.

From the accuracies obtained with varying weighting scheme, higher accuracy is observed when both training and testing have been weighted by different magnitudes (86.264% for weighted scheme vs. 82.418% for non-weighted scheme). Also, we observe that as the magnitude of weight term is increased, we see that accuracy peaks before beginning to decrease.

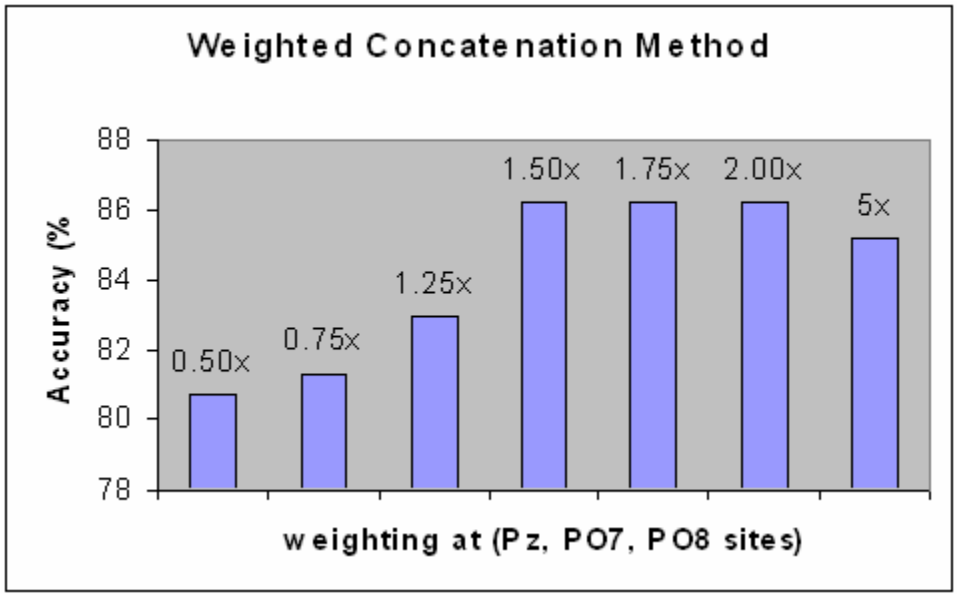


Figure 3.1: Letter accuracy for different concatenation weightings

Note that this approach improves upon concatenation method by approximately 4% with minimal increase in time efficiency. Therefore, weighting approach seems to yield the best result.

3.5.2. Accuracy across Subject

The following table 3.1 illustrates the accuracy observed across the subject. Note that significant changes have been made to the system after March 28th, 2005. The data shown in

the table shows that 100% accuracy is achieved within 3 trials (best case), especially after March 28th, 2005.

Table 3.1: Letter accuracy for different subjects

Date	Subject	# of Trials	Accuracy (letters)	Percentage
March 26	Jack	15	12/14	86%
March 28	Min	15	12/21	57%
March 30	Brian	15	19/19	100%
March 31st	Jyh-Liang	15	26/26	100%
April 2	Lucky	15	25/28	89%
March 31st	Jyh-Liang	3	13/13	100%
March 31st	Jyh-Liang	2	18/20	90%
March 31st	Jyh-Liang	1	10/21	48%

3.5.3. Accuracy over time

To evaluate whether the subject must train every time he/she uses the system, we have taken a previously-acquired training model and attempted to spell correct symbols. Table 3.2 illustrates our findings. The data suggests that accuracy over time is not affected by time. However, more rigorous testing over time is required to confirm this hypothesis.

Table 3.2: Letter accuracy over time (Subject: Jack Cha)

Date	Time Change from Model Made	# of Trials	Accuracy (letters)	Percentage
March 26	0	15	12/14	86%
March 28	2 days	15	19/19	100%
March 31	5 days	15	19/22	86%

3.5.4. Accuracy over Number of Trials

Having achieved 100% accuracy at 15 trials, we took the testing data and simulated offline using MatLab. Amazingly, for our best subject (Jyh-Liang), we achieved 100% accuracy after 3 trials as shown in figure xxx.x.

Subject: Jyh-Liang

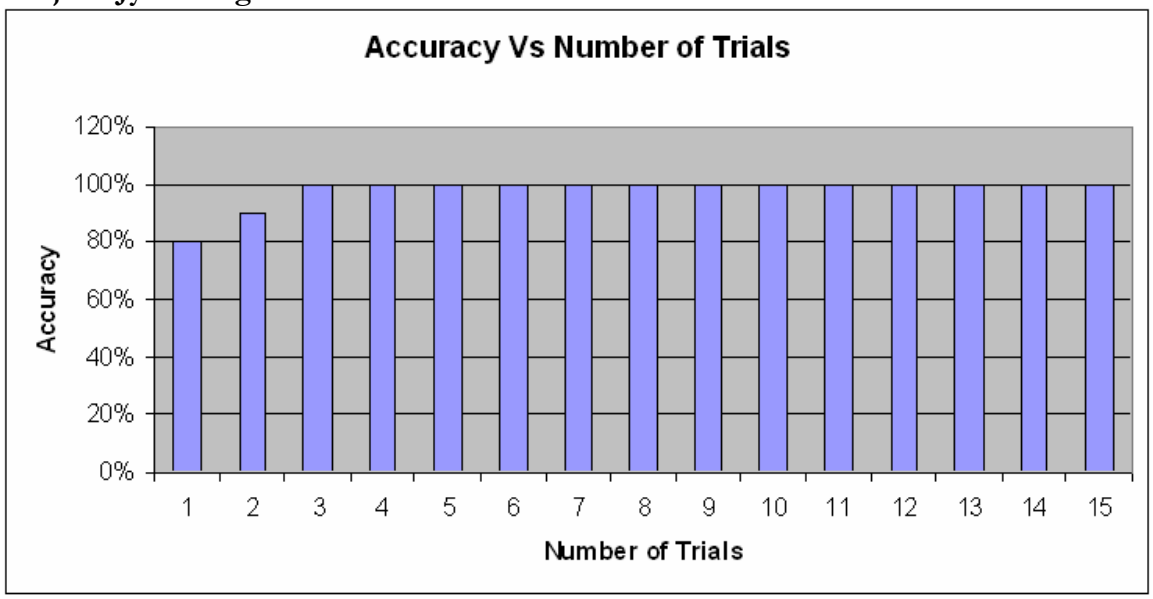


Figure 3.2: Letter accuracy for different number of trials (Subject: Jyh-Liang)

3.5.5. Accuracy vs. Model Size

In our attempt to determine the effect of training data size, we have conducted an offline experiment by taking subsets of training data and create various models based on them. Figure xxx.x shows that our system achieves 100% accuracy across the different model sizes. Unfortunately, this result is based on data taken from single subject (Jyh-Liang).

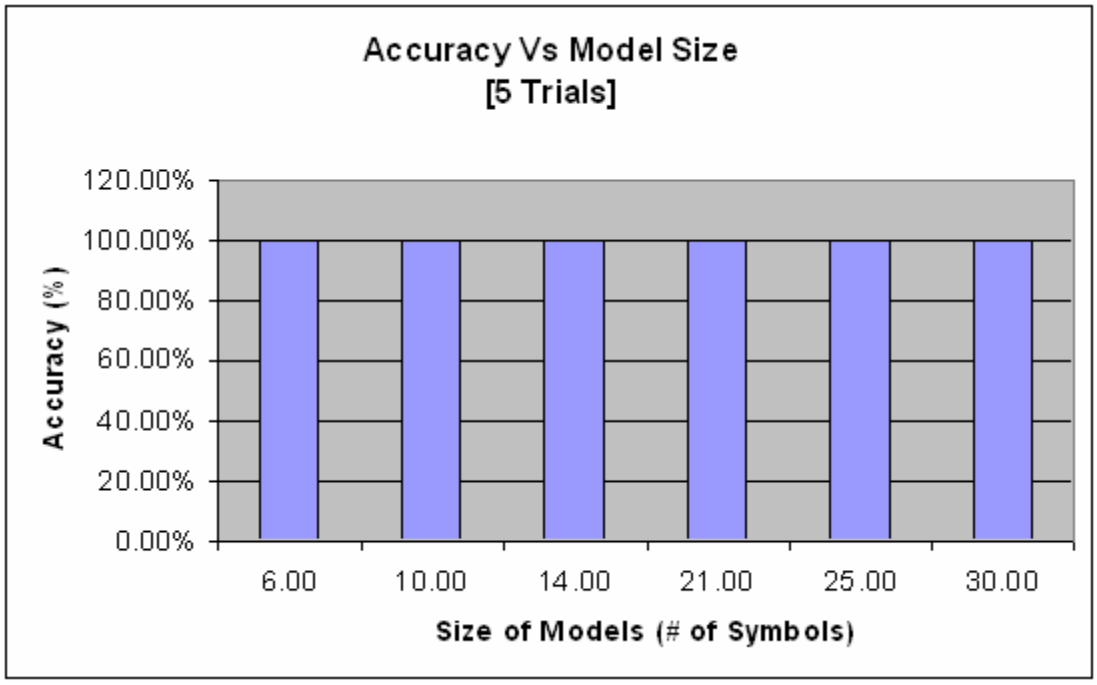


Figure 3.3: Letter accuracy for different training model sizes

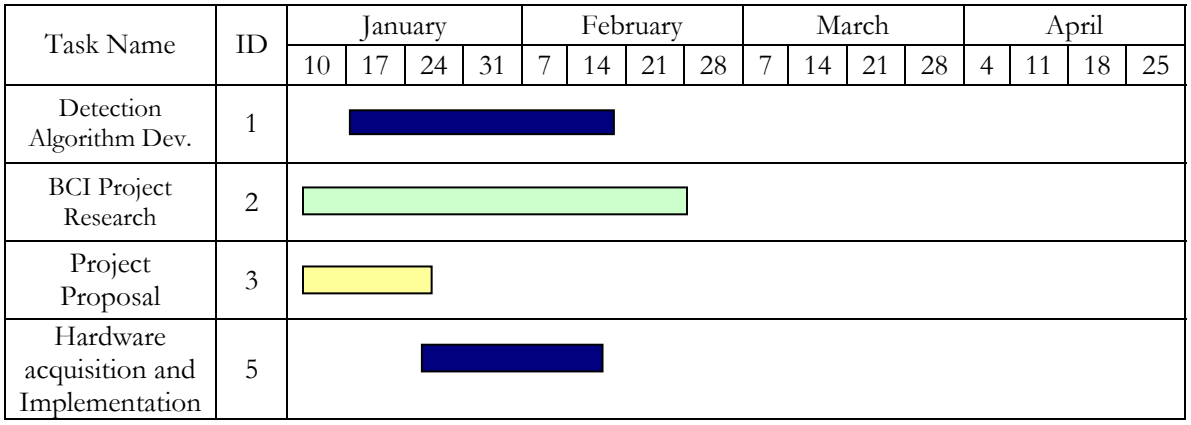
4. Budgetary and Scheduling Results

Below, in table 4.1, is our original budget and our actual budget. As can be seen from the table, we estimated a cost of $3250, but we ended up spending only $810. The only items that we failed to account for originally were the connectors that we used to connect the output of the amplifier to the input of the A/D. Some of the cost savings were due to generous equipment loans from Engineering and Psychology department.

Table 4.1: Estimated and actual costs of equipment

Equipment	Estimated Cost	Actual Cost
EEG Amplifer	Amplifier–On loan	On loan
Probes and Electrode Cap	$1200	$100 for probes, electrode cap on loan
PC Desktop System	$700	On loan
Video Card	$350	Used laptop as secondary monitor
Monitor	$200	On loan
Analog to Digital Converter Card	$800	$650
Connectors to interface A/D and amplifier	Not accounted for	$60
Total	**$3250**	**$810**

As for scheduling, we found in the middle of February that we had fallen behind schedule and that our schedule was no longer relevant. At this time we created and approved a new schedule, which for the most part we stuck with to the end. The only discrepancy between our revised schedule and the actual work done is perhaps with predictive text development, which happened much later than expected, and online system testing, which proceeded until a few days before our final demonstration. Figures 4.1 and 4.2 below show our orignal and revised schedules.



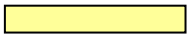
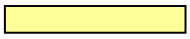
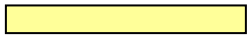
Control Software Development	6	
System Integration	7	
Prototype Demo	8	
Prototype Debugging	9	
Prototype Optimization	10	
Functional Specification	11	
Design Specification	12	
Group Demo & Presentation	13	
Post-Mortem	14	

Figure 4.1: Proposed schedule

Task Name	Length	Week ending March				April	
		7	14	21	28	4	5
Complete Offline software system	10						
Online system development	7						
Complete Hardware Acquisition	7						
Implement Online System	11						
Online System Testing (data collection)	7						
Design Specification	7						
Predictive text development	7						
Offline system testing/ playing with parameters	10						
Group Demo & Presentation	5						
Post-Mortem	14						

Figure 4.2: Revised schedule, approved on Feb 28/05

The main discrepancies in the two schedules came from development times. In the middle of the semester we took somewhat of a break to study for midterm exams and catch up on other course work. Also, when we revised our schedule, we were able to more accurately predict and quantify the remaining work that needed to be done.

5. Future Plans

We have four options to choose from. These are basically rough scenarios we brainstormed.

- **Patent**

Research the feasibility and patentability of our system for the purpose of, if possible, stopping others from implementing a similar or the same system as ours. We give a license to someone who wants to use our patent in the future.

- **Publication of Paper**

Organize and explain what we have done up until now by publishing a paper. We can help people in the biomedical field to be aware of the achievability of the world's fastest and most accurate BCI system (P300 system that we know of). By letting others implement a similar system as ours by following our results, we believe our accomplishments can bring better hope for disabled people.

- **Startup Company**

After further development and testing of our system to improve user-friendliness, features, reliability, and most importantly, economic cost, we can set up a venture company. At this stage, we also need to persuade Angel investors for necessary fund raising.

- **Other Applications**

We have learned that SVM is a very useful pattern recognition algorithm that can be geared towards other applications. We only implemented the algorithm in the context of P300 spelling devices. We believe that SVM can be applicative to other fields such as voice recognition, lie-detection, gaming, and communication networks.

- **Project Competitions**

We can attend various engineering competitions that are ongoing around the world. The upcoming event is IEEE and TELUS annual competition in June. With much certainty, we are planning to participate in the competition to prove that our BCI system is competitive and world shaking.

6. Inter-personal and Technical Experiences

Jack Cha

Through out this semester, working on the project has constantly exposed me to professional attitude and respect that each of our group member had toward each other. I believe that this enlightenment to be the fundamental source of our group's success. I learned that the group dynamic very much depends on the personalities of the members and on the method we use to solve both internal and external conflicts.

On the technical side, I was mainly responsible for the operation of data acquisition hardware and software that would be integrated to the main system. I mainly learned that writing source code should not focus on the number of lines of code itself but rather how my component will fit into the whole system. This would be done via thorough software engineering that must be done properly at the initial stage; otherwise you would waste literally hours changing your code at a near finishing stage.

Also, doing a presentation to a wide range of audience was an essential experience that gave insights on what and how to present your materials so you would not bore the audience to death but rather keep them keenly interested in your project.

But, the most valuable experience that I am walking away with is the professional attitude and methodology that an engineer should use in order to efficiently allocate resources on hands. I saw the real benefits that a task, which appeared as cumbersome at first sight, ended up saving time that we as a group would blindly put in without a second thought.

In conclusion, we had gained insights on the meaning of project management and how to do it professionally as well as the importance of keeping high group morals.

Jaeseok Jeon

Two weeks before we submitted our project proposal, we gathered in the pit to integrate each part assigned for the project proposal. One of the sections that I was responsible for was "Team and Company Organization". My opening sentence started with "Five brilliant engineering students at SFU came up with a world shattering idea…" Our great leader, Brian, revoked the word "world shattering", though I thought it was a good choice of word. The reason was simply because it was not proper and I agreed with him. Three months later, the choice of word has proven to be justified as it implies.

At the beginning of this semester, I did not expect that I would undertake parts of the software modules in our BCI system because of my inadequate coding capability compared to my hardware skills. More specifically, I did not have any background knowledge in SQL, database and dynamic linked libraries. In fact, I did not have any expertise in C++ programming. Through research on the fields that are relevant to our project and with a touch of motivation, I accumulated useful software knowledge and valuable software skills. For example, for the C++ version of SVM, which we eventually dropped after we have decided to use the Matlab version, helped me get more acquainted with C++ programming. The SQL database pertaining to the predictive text driver was started from scratch on my account, which adds one more area into my existing skills. In addition, I have become more familiar with Visual Basic. All I knew about the language was simple GUI constructions. During this semester, the cornerstone of our BCI system leaded me to apply object-oriented programming features of Visual Basic.

Each of my group members has done great job. They all did more than expected. Each had their area of talents that have contributed to our project. These personal abilities all added up contributed to a fruitful outcome at the end. Two main lessons that I learnt from group dynamics are that each should constructively listen to one another since each has a different perspective on a specific matter. This generates an atmosphere of mutual respect.

I do, however, have a personal regret. If I were able to go back before the beginning of this semester, it would like to have a better knowledge on the software languages used by spending time to learn before the start of the project. I believe this would save much time in implementing our system during this semester.

Brian John

This group was extremely fun to work with. Not only that, but in the end we exceeded all expectations concerning the project and achieved results that exceed any we've seen.

Over the course of the semester, I became much more familiar with brain-computer interfaces, improved upon my Matlab programming abilities, and practiced my management skills. Finding, organizing, (and sometimes) purchasing the equipment that we used, as well as finding our processing method, and setting direction for overall design were my main management tasks.

Conflicts in our group were minimal and when they did arise, they were often resolved using logic and (most of the time) without heated arguments. With all of us heading towards the same goal, our dynamic was healthy and fun. When decisions needed to be made on a design, we often discussed it and then promptly made a decision, providing no extra research needed to be done. When there was extra research required to make a decision, we did the research and came to a conclusion that way. At all times, the group demanded that when

decisions were to be made, that we made them based on previous results, logic, or preliminary results, not unfounded 'gut feelings', inklings or backwards thinking.

One thing that I've definitely learned throughout this project, is to be receptive to criticism and suggestions, but at the same time, to recognize unfounded, childish or malicious criticism and to ignore it. For instance, we often received criticism from uninformed people who seemed to know what they were talking about. Once we got to the root of the problem and gained more confidence as a group, however, we learned to ignore such comments as 'you'll never get it working', 'if it works it'll never work for more than one person', or 'your system is impossibly difficult to use and nobody will ever figure out how to use it.' On the other hand, other people, such as Lucky, were extremely helpful and offered genuinely good suggestions. He saw and used our system, offered real suggestions on how to improve it, and was encouraging throughout the project. Throughout the course of the semester, we received a lot of useful help and suggestions from many people. For this we are very grateful.

Min Seo

ThinQ Innovation was arguably one of the most well-structured groups I have had the pleasure to be a member of. Everybody in the group was diligent, dedicated and very competent. Each member brought unique set of skills, which made it easy to define specific roles. For example, Brian who has been our project lead, never lost sight of the big picture and provided the group with clear sense of direction while effectively delegating tasks according to individual strengths. Jyh-Liang, our chief programmer brought excellent programming aptitude and self-initiative. Jae-Seok and Jack took on tasks that required research and device-specific know-how but where they lacked in knowledge, they fully compensated through tireless effort and active learning. I had been mainly responsible for implementation and testing of MatLab interface to a C++ library used in signal processing. In the end, everyone fulfilled most of their responsibilities, if not all. We had virtually no group dynamic problem except for occasional arguments, which always pertained to our project.

From a technical perspective, I gained valuable experience in MatLab programming and ERP amplifiers. I also learned about issues involved in detecting brain signals, signal-to-noise, electrode connection and alternate signal processing algorithms that can be used in place of our current module.

Beyond technical skills, I gained, perhaps the most important asset that is my interest in the signal-processing algorithms. Throughout the semester, I have spent much of my leisure time studying a particular signal-processing algorithm, *Support Vector Machine*. I began to understand that the underlying principle of this algorithm could not only be used to detect P300 brain signal but also be used in other numerous areas including wireless

communication and most types of recognition algorithms. I plan to continue my studies in this area, as it has become my main area of interest over the course of this semester.

As it has been illustrated by our final demonstration, we have developed a very stable prototype P300 Spelling Device, fulfilling all of the functionalities outlined in functional specification. In some aspects, we even exceeded the requirements by notable margin. The only improvement that could have significantly changed our project would have been the time allotted for this semester. Although the prototype met the function requirements, there are still a number of experiments yet to be done to improve our system even further. My only regret is the time, or lack thereof.

Jyh-Liang Yeh

Reflecting back, this project is indisputably one of the best projects I had worked on within my undergraduate curriculum. Equally gratifying, I had the pleasure of working with 4 splendid colleagues who I have the utmost respect for. I admit that I was at first skeptical about the course and project, but the group along with their accomplishments has motivated me immensely and changed my point of view for the better.

I would like to thank our project leader, who has provided superior management and direction throughout the project. His dedication and knowledge would forever be a role model amongst us. As for SVM support group, I would like to express my gratitude for your continual commitment and contribution to the team. My appreciation also directs toward the hardware and software support group, who has poured countless hours and effort into the project. Nonetheless, the entire team has shown outstanding group dynamics. I learned that a healthy team consists not of members with similar traits and skills, but with different abilities that complement each other.

Moreover, I learned the importance of being attentive to others' opinions. In the case of our team, certain members have unorthodox way of solving problems. I believe that if we were not attentive to their suggestions, we would not have advanced this far. That is, we overcame many obstacles by granting their revolutionary ideas that were beyond conventional logic and wisdom. But, whenever we were sidetracked from our initial goals, the conventional thinkers stepped in and lead the team back on track. This yin-yang dynamics further demonstrate the complementary abilities of our team.

The experiences and skills gained from this project will definitely endure throughout my future career. I gained valuable skills in real-time programming, Matlab interfacing, and hardware driver implementation. Aside from the technical knowledge, one of the most valuable insights I gained was the importance of proper scheduling. This reflects an advice given by Lucky, which I treasure. He indicated that a realistic schedule is not one of sacrifices to accomplish milestones but one that includes other tasks unrelated to the project. For instance, we should not think of scenarios such as "if I saved time from watching T.V., I

may be able to fit this other task in". Rather, we should take account that "during this project duration, I need time for midterm exams, socializing, and nights out".

Furthermore, I learned to take lightly on criticisms from people even when they are of high stature. Instead, we should utilize these criticisms for constructive motivation, which would otherwise demerit and hinder our project. On the other hand, I realized that there are those, such as Lucky and Clayton, who are genuinely helpful from start to finish. Lastly, I discovered that the spices for success require the right mix of group members, enthusiasm and, least to say, the social occasions and luck.

7. Conclusion

In three months, ThinQ Innovation was able to successfully develop a P300-based brain-computer interface that exceeded all of our original expectations. All of our group members developed new skills and had a lot of fun working together. We hope to further our project in the future.