

**ERPs MEASUREMENT USING NEW VISUAL
STIMULI COMPARING WITH STANDARD METHOD**




**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF ENGINEERING
(BIOMEDICAL ENGINEERING)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2004**

**ISBN 974-04-5557-3
COPYRIGHT OF MAHIDOL UNIVERSITY**

Thesis

Entitled

**ERPs MEASUREMENT USING NEW VISUAL
STIMULI COMPARING WITH STANDARD METHOD**



Manosh Suparang

Mr. Manosh Suparang
Candidate

Prakob Pooviboonsuk

Asst. Prof. Dr. Prakob Pooviboonsuk,
Ph.D. (Psychopharmacology)
Major-advisor

Chatchai Neatpisarnvanit

Asst. Prof. Chatchai Neatpisarnvanit,
Ph.D. (Electrical Engineering)
Co-advisor

Rassmidara Hoonsawat

Assoc. Prof. Rassmidara Hoonsawat,
Ph.D.
Dean
Faculty of Graduate Studies

Theeraporn Rubcumintara

Asst. Prof. Theeraporn Rubcumintara,
Ph.D. (Materials Engineering & Science)
Chair
Master of Engineering
Programme in Biomedical Engineering
Faculty of Engineering

**ERPs MEASUREMENT USING NEW VISUAL
STIMULI COMPARING WITH STANDARD METHOD**

was submitted to the Faculty of Graduate Studies, Mahidol University
for the Degree of Master of Engineering (Biomedical Engineering)

on

12 Oct, 2004

Manosh Suparang

Mr. Manosh Suparang
Candidate

Prakob Pooviboonsuk

Asst.Prof.Dr. Prakob Pooviboonsuk,
Ph.D. (Psychopharmacology)
Chair

Chatchai Neatpisarnvanit

Asst.Prof. Chatchai Neatpisarnvanit,
Ph.D. (Electrical Engineering)
Thesis Defence Committee

Udom Tipayamontri

Asst.Prof. Udom Tipayamontri,
Ph.D. (Physiology)
Thesis Defence Committee

Orawan Silpakit

Mrs.Orawan Silpakit
MD.
Thesis Defence Committee

Rassmidara Hoonsawat

Assoc. Prof. Rassmidara Hoonsawat,
Ph.D.
Dean
Faculty of Graduate Studies
Mahidol University

Piya Rattanasuwan

Asst. Prof. Piya Rattanasuwan,
B.Eng. (Civil Engineering)
Dean
Faculty of Engineering
Mahidol University

ACKNOWLEDGEMENTS

Special Thanks to Asst. Prof. Dr. Prakorb Pooviboonsuk, my advisor, who always gave me a lot of help in the knowledge of the Event Related Potentials (ERPs) and the support for the Medelec Sapphire Premiere machine for recording the ERPs signal. A special grateful thanks to Asst. Prof. Chatchai Neatpisarnvanit, my co-advisor, who gave me a lot of help in signal processing technique for solving ERPs' signal. Another special grateful thanks to Dr. Orawan Silpakit, my external advisor, who suggested me about the statistic analysis in ERPs' component. And the next person is Asst. Prof. Udom Tipayamontri for dedicating his worthwhile time for teaching me how to write this thesis.

I would like to express my sincerity and appreciation to Mr. Tharis Hincharanundana for the ERPs' signal recording and advice me many useful thing. Moreover, it is my pleasure to express my thankfulness to Biomedical staff, Miss Saisunee Ulis and Miss Vipada Chuairaksa, for an aid in documentations, advices, and helps in everything. Futhermore, I would like to thank my friends for their willpower.

Finally, I would like to express my sincere gratitude and heartfelt appreciation to my beloved family, grandfather, grandmother, father, mother, and my sister for encourage me in everything.

Manosh Suparang.

ERPS MEASUREMENT USING NEW VISUAL STIMULI COMPARING WITH STANDARD METHOD.**MANOSH SUPARANG 4238198 EGBE/M****M.Eng.(BIOMEDICAL ENGINEERING)****THESIS ADVISORS: PRAKOB POOVIBOONSUK, Ph.D., CHATCHAI NEATPISARNVANIT, Ph.D.****ABSTRACT**

Brain potentials have been widely used for studying brain activity associated with higher mental functions in humans. Event Related potentials (ERPs), one of the brain potentials are electric fields that arise from neural activity associated with the processes of perception and cognition. One of the important parts of ERPs is P300 component. P300 ERPs are complex waves which occur between the latencies of about 300-400 msec. The P300 appears to be related to cognitive processes and is also linked to memory.

The aim of the research was to compare ERPs, especially P300, using new visual stimuli versus the standard method. A visual stimulus was presented in the Medelec Sapphire Premiere machine, and it is presented as the black and white checkerboard picture. The new visual stimuli which are color picture were presented on the PC monitor by a program written in C language, however, the picture cycle was still controlled by the same Medelec Sapphire Premiere machine. After both tests, the result was analyzed using paired t test and energy density of signal.

The results from the paired t test comparing both P300 amplitude and P300 latency showed that there was no significant difference between new visual stimuli and the standard stimuli. The results pointed out that the new visual stimulus would be used as process interchangeable with the standard method. The result from energy density of signal also pointed out that the new pictures made the brain's memory consume more energy than did the standard pictures.

**KEY WORDS : EVENT RELATED POTENTIALS/ MEMORY/ P300ERPS/
PAIRED T-TEST/ ENERGY DENSITY****118 P. ISBN 974-04-5557-3**

การวัดค่า ERPs โดยใช้ภาพกระตุ้นแบบใหม่เปรียบเทียบกับวิธีมาตรฐาน (ERPS MEASUREMENT USING NEW VISUAL STIMULI COMPARING WITH STANDARD METHOD.)

มาโนช สุพะรัง 4238198 EGBE/M

วศ.ม. (วิศวกรรมชีวการแพทย์)

คณะกรรมการควบคุมวิทยานิพนธ์ : ประกอบ ผู้วิบูลย์สุข, Ph.D., ฉัตรชัย เนตรพิศาลวนิช, Ph.D.

บทคัดย่อ

มีการศึกษาศักย์ไฟฟ้าของสมองอย่างกว้างขวางเนื่องจากศักย์ไฟฟ้าที่เกิดขึ้นจะสะท้อนถึงสติปัญญาชั้นสูงของมนุษย์ หนึ่งในนั้นคือ ศักย์ไฟฟ้าที่เกี่ยวกับเหตุการณ์ (ERPs) ERPs คือสัญญาณไฟฟ้าที่เกิดขึ้นในขณะที่เซลล์ประสาททำงานเกี่ยวกับการรับรู้และจดจำ ส่วนที่สำคัญส่วนหนึ่งของ ERPs คือ P300 ซึ่งเป็นลูกคลื่นที่เกิดในช่วงเวลาที่ 300 ถึง 400 มิลลิวินาที พบว่า P300 นี้เชื่อมโยงกับกระบวนการในการรับรู้ซึ่งจะเชื่อมโยงไปยังความจำ

ในงานวิจัยนี้ได้ทำการเปรียบเทียบการวัด P300 ERPs ด้วยการมองภาพ โดยวิธีมาตรฐานและวิธีใหม่ ซึ่งวิธีมาตรฐานนี้ ภาพที่ใช้ทดสอบ จะเป็นภาพตารางหมากรุก สีขาว-ดำ ที่เกิดจากเครื่อง Medelec Sapphire Premiere ในขณะที่วิธีใหม่นี้ ภาพที่เกิดขึ้น จะเป็นภาพสีที่ถูกควบคุมโดยโปรแกรมคอมพิวเตอร์ที่เขียนด้วยภาษาซี แต่ช่วงเวลาของการเกิดภาพยังคงถูกควบคุมด้วยเครื่อง Medelec Sapphire Premiere หลังจากนั้นผลจากการทดลองจะถูกนำไปวิเคราะห์ด้วยวิธีการทางสถิติอันได้แก่ การเปรียบเทียบด้วยวิธี Paired t test, และการหาความหนาแน่นของพลังงาน (Energy density of signal)

ผลจากการศึกษาพบว่า การเปรียบเทียบด้วยวิธี Paired t test ทั้งขนาดของ P300 (P300 Amplitude) และ ช่วงเวลาการเกิด P300 (P300 Latency) ให้ผลที่ไม่แตกต่างกัน ดังนั้นจึงสรุปได้ว่าวิธีใหม่สามารถใช้วัด P300 ERPs เทียบเท่าวิธีมาตรฐาน อีกทั้งเมื่อเปรียบเทียบความหนาแน่นของพลังงานจากพื้นที่ได้กราฟ พบว่า วิธีใหม่มีพื้นที่ได้กราฟมากกว่า ซึ่งแสดงเป็นนัยว่า วิธีใหม่ทำให้สมองต้องใช้พลังงานในการทำงานมากกว่าวิธีมาตรฐาน

118 หน้า. ISBN 974-04-5557-3

CONTENTS

	Page
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
LIST OF TABLES	viii
LIST OF FIGURES	ix
CHAPTER I INTRODUCTION	
1.1 Introduction	1
1.2 Objectives	3
1.3 Expected Result	3
CHAPTER II LITERATURE REVIEWS	
2.1 Basic visual pathway	4
2.2 Memory and the brain	6
2.3 Working memory and the frontal lobe	10
2.4 Event-related brain Potentials	12
2.5 Deriving Event-Related Potentials	14
2.6 The relation between memory and P300	16
2.7 Psychological meaning of P300 and Applications	17
2.8 How to elicit P300	19
2.9 P300 Measurement and Identification	20
CHAPTER III MATERIALS AND METHODS	
3.1 Subjects	22
3.2 Experiment	22
3.2.1 Getting ready	22
3.2.2 ERPs recording	22
3.2.3 Method of testing	24
3.2.4 Communication between Medelec machine and PC	27
3.2.5 C program	28
3.3 Data Acquisition	28

CONTENTS (Cont.)

	Page
3.4 Data Analysis	30
3.4.1. The Paired t-Test	31
3.4.2. The total energy in the signal	32
3.4.3 Correlation	33
CHAPTER IV RESULTS	
4.1 Experiment flowchart	34
4.2 Output signal	35
4.3 Delay times	36
4.4 Result of P300 amplitude and latency	38
4.5 Paired-t test	39
4.6 Energy density of signal	40
4.7 Correlation	41
CHAPTER V DISCUSSION AND CONCLUSION	45
REFERENCES	49
APPENDIX	52
BIOGRAPHY	118

LIST OF TABLES

Table	Page
3.1 The truth table of parallel port is used to display picture.	28
4.1 P300 amplitude and latency from subjects are tested with new picture (portrait) and standard picture (checkerboard).	38
4.2 Energy density of signal from subjects is tested with new picture (portrait) and standard picture (checkerboard).	40
4.3 Correlation in energy density of signal at Cz location.	42
4.4 Correlation in energy density of signal at Cz location.	42



LIST OF FIGURES

Figure	Page
FIGURE 2.1 The primary visual pathway 1.	5
FIGURE 2.2 The primary visual pathway 2.	5
FIGURE 2.3 Medial view of cerebrum.	7
FIGURE 2.4 In an experiment, the subject was presented with a stack of cards on each of which all stimuli were presented, but in different random orders.	12
FIGURE 2.5 Schematic representation of the operations involves in the recording of event-related brain potentials.	14
FIGURE 3.1 The International 10-20 system of electrode placement.	23
FIGURE 3.2 Channel instruments for scalp electrodes from Medelec machine.	23
FIGURE 3.3 Schematic diagram illustrating data acquisition from standard pictures.	24
FIGURE 3.4 Schematic diagram illustrates picture on monitor from Medelec video output.	26
FIGURE 3.5 Schematic diagram illustrates signal to parallel port.	27
FIGURE 3.6 Schematic diagram illustrates C programming.	28
FIGURE 3.7 The signal occurs from new picture at position of Fz and Cz respectively	29
FIGURE 3.8 The signal occurs from standard picture at position of Fz and Cz respectively.	30
FIGURE 4.1 The flowchart shows subjects experiment with television (standard pictures) and Computer (new pictures).	34
FIGURE 4.2 Target signal is generated from Medelec Sapphire Premiere machine.	35

LIST OF FIGURES (CONT.)

Figure	Page
FIGURE 4.3 Non-Target signal is generated from Medelec Sapphire Premiere machine.	36
FIGURE 4.4 Target and Non-target delay times.	37



CHAPTER I

INTRODUCTION

1.1 General Background

Brain potentials have been widely used for studying brain activity associated with higher mental functions in humans, in both of continuous EEG and averaged Event-Related potentials (ERPs). In addition, ERPs are time-locked electric fields that arise from elaborate patterns of synchronous neural activity associated with the processes of perception and cognition (1). The ERPs are recorded from the human scalp and are used for evaluate certain perceptual and cognitive processes, both in normal persons and in patients who have abnormalities in these functions.

Hillyard (2) mentioned three main goals for studying ERPs in relation to cognitive processes. The first is to identify ERP components that are related to specific intermediate stages of information processing and clarify the structure of complex cognitive processes. The second is to understand the neural bases of cognitive activities, and ERP is one of the few techniques available for study of the neurophysiology of cognitive processing. The third major goal of the ERP research is in the area of clinical application: to use ERP measures to specify which processes are deficient in patients with psychopathology, neurological disease, disorders of learning and attention, dementia, and other cognitive deficits. From the above mentions one can say that, ERPs are valuable tool for studying the mass neural activity generated during cognitive processes.

One of the important parts of ERPs is P300 component. This component is named by its polarity (p for positive) and latency after stimulus onset (3). P300 ERP is a large (10-20 μv) positive-going wave complex which occurs between the latencies of about 300-400 msec. It is elicited when the subject attends to a train of regularly occurring stimuli in order to discriminate occasional "Target" that differs from the standard ("Non-Target") stimuli by some physical characteristic, such as the

frequency of the tone (4). The P300 appears to be related to cognitive processes and is also linked to memory that occurs after incoming information has been initially analyzed or closely tied to moment of decisions, when the subject classifies the stimulus (5, 6). Consequently, it is useful for assessment of cognitive ability in normal subjects as well as in patients with neurologic and psychiatric diseases.

Neurophysiology provides many methods for assessing age-related changes in central nervous system (CNS) function. In particular, the P300 event-related potentials (ERPs) has been considered to study about the age-related cognitive dysfunction because it reflects attention and memory processes. One of age-related cognitive dysfunctions is complicated in dementing illness which is a disorder of higher cognitive function involving memory, personality, judgment, and intelligence, therefore it often causes the slowness of thought processes and behavioral responses. Many methods are discovered to diagnose the dementing illness but the most effective one is Visual P300 ERPs, whereas the P300 latencies were found to be a long procedure for demented subjects.

Nowadays, at the Siriraj Hospital, there is an equipment for recording the Visual P300 ERP; called the Medelec Sapphire Premiere machine. It is an equipment that is generally operated for recording P300 ERP in both the methods of auditory and visual stimuli. A visual stimulus has been already presented in the Medelec Sapphire Premiere machine; as a matter of fact, it is presented as the checkerboard picture. Not only that's because of the convenient to take pictures, but it maybe also the links to a little subject's memory. Consequently, the new picture that maybe links more properly the subject's memory than the checkerboard picture is presented. The new picture will be appeared on the PC monitor with a program written in C language; however, the picture cycle is still controlled by the same the Medelec Sapphire Premiere machine as the checkerboard picture

Thus, the both of the methods' results are compared with the paired t-test. I am entirely interested in the possibility of using new picture despite of the checkerboard picture because of its easier installment in the PC. Furthermore, the specific picture can be setup to other kinds of patients who need diagnosis and treatment. Next, Parseval's relation is used to compute the signal's energy density. The hypothesis is

that the brain should be using more energy to analyze complex color pictures compared to energy used to do so in analyzing the easy checkerboard black & white color pictures.

1.2 Objectives

This thesis proposes new visual stimuli, which are written by C program. The objectives are as followed.

Objective 1

To compare the ERPs resulted from new visual stimuli and the standard method. Are the two small sets of quantitative data related?

Objective 2

To examine the total energy density of brain signal with Parseval's relation. Are the new visual stimuli and standard method related?

1.3 Expected Result

As described above the new pictures are written in C program to support ERP recording. The new picture is presented to the subjects which they have to discriminate between Target and Non-Target picture. Ideally, the latency of P300 ERPs from new picture should be equal to the one from standard picture; however, there might be some differences. The differences maybe from the new picture which is harder to discriminate than the checkerboard picture. Moreover, the delay time from this program and the circuit interface may involve. Although all of the above make a delay of time to the new picture, the prominent frequency that contains in the memory can be as valuable as the old one.

CHAPTER II

LITERATURE REVIEWS

This chapter explains the basic visual pathway, the relation between brain and memory and the main topic in Event Related Potentials (ERPs). The procedures used to derive ERPs and the relation between memory and P300 including psychological meaning of P300 and applications.

2.1 Basic visual pathway

Visual perception begins in the retina. Light entering the cornea is projected onto the back of the eye, where it is converted into an electrical signal by a specialized sensory organ, the retina. The retina contains two types of photoreceptors, rod and cones. Cones are responsible for day vision but rods mediate night vision. Rods are exquisitely sensitive to light and therefore function well in the dim light that is present at dark or a night, when most stimuli are too weak to excite the cones. The information leaves the eye by way of the optic nerve, and there is a partial crossing of axons at the optic chiasm. That is, the fibers from the nasal half of each retina crosses to the opposite side of the brain, but the axons from ganglion cells in the temporal hemiretinas do not cross.

After the optic chiasm, the axons are bundled through the optic tract. In this arrangement the axons from the left half of each retina (the temporal hemiretina of the left eye and the nasal hemiretina of the right eye) project in the left optic tract, which thus carries a complete representation of the right hemifield of vision. Fibers from the right half of each retina (the nasal hemiretina of the left eye and the temporal hemiretina of the right eye) project in the right optic tract, which carries a complete representation of the left hemifield of vision.

Then, the optic tract wraps around the midbrain to get to the lateral geniculate nucleus (LGN), the principal subcortical structure that carries visual information to the cerebral cortex. From there, the LGN axons fan out through the deep white matter of the brain as the optic radiations, which will ultimately travel to primary visual cortex

(Brodmann's area 17), at the back of the brain for further processing necessary for perception (7, 8, 9).

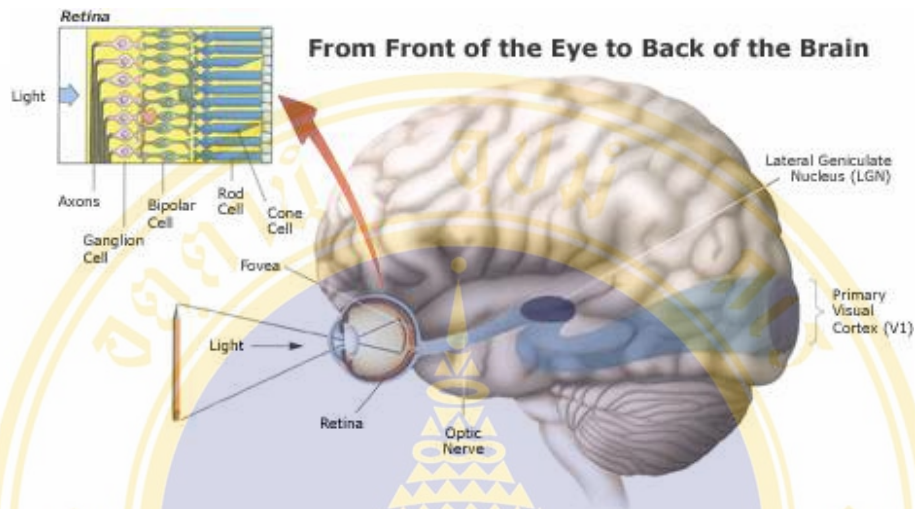


FIGURE 2.1 The primary visual pathway 1 (10).

► **The Primary Visual Pathway**

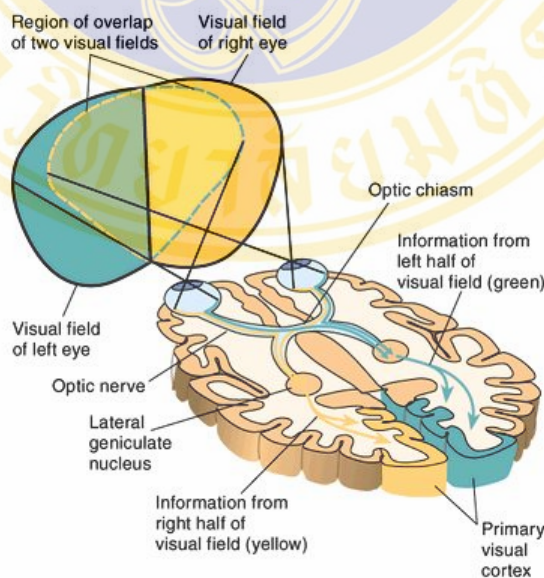


FIGURE 2.2 The primary visual pathway 2 (10).

2.2 Memory and the brain

A primary function of the brain is to coordinate all of the nervous activity of the body. The brain receives incoming nerve impulses from the eyes, ears, and other sensory organs, analyzes information in the light of past experience, and is responsible for making the necessary adjustments or movements. In addition, the brain is the seat of man's consciousness, memory, reasoning, and intelligence.

2.2.1 Where is memory stored in the brain?

In the early 20th century, American neuropsychologist Karl Lashley (1890-1958) made the first systematic effort to determine memory's location in the brain. After training rats to traverse a maze, he surgically removed various amounts of their cerebral cortex. Lashley found that no single cortical locus could be removed that would prevent the rats from relearning the maze. However, he also discovered that the more cortical tissue he removed, the more difficulty the rats had in relearning the path. These results were originally interpreted to mean that memory has no particular site in the brain but is instead distributed evenly throughout the cortex.

Later experiments, he suggest that the rats' surgically induced learning impairments resulted from the progressive removal of specialized cortical areas used to analyze and process maze-related information, including olfactory, visual, and spatial details. It is now believed that the brain stores information in the same cortical areas that are used for the initial processing and analysis of incoming data. Thus, memory exists not in a single spot but within many specific brain regions.

2.2.2 Brain systems specialized for memory functions

Although no single area in the brain is responsible for the existence of long-term memory, different aspects of this process are particularly dependent on certain regions. These sites include the hippocampus (and related structures in the medial temporal lobe and diencephalon), for declarative memory. The amygdala relates to emotional memory, that is, feelings of fear, anxiety, and pleasure that are based on memory of past experiences. The caudate nucleus and putamen relate to habit learning and, the cerebellum, for classical conditioning.

In addition, the frontal lobes appear to play an important part, together with the hippocampus, in acquiring and retaining source knowledge, that is, the where, when, and how information of episodic memory. The frontal lobes also play a role in implementing strategies and carrying out planned activity (11).

2.2.3 Dissociable memory systems in human

Evidence from psychology and neuroscience points to at least five major systems in human memory: episodic memory, semantic memory, the perceptual representation system, procedural memory, and working memory.

Episodic memory

Episodic memory is the explicit recollection of incidents that occurred at a particular time and place in one's personal past. Damage to the medial or inner parts of temporal lobes, including the hippocampal formation, greatly impairs the acquisition of new episodic memories. Patients with organic amnesic syndromes resulting from damage to the medial temporal region invariably have serious impairments of episodic memory: they are unable to remember ongoing events in their day-to-day lives.

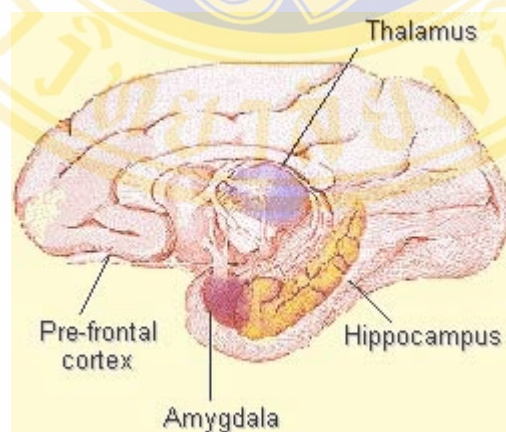


FIGURE 2.3 Medial view of cerebrum (12).

Regions within the prefrontal cortex play a key role in episodic memory. Although patients with selective damage to prefrontal regions do not develop a profound amnesia for recent events, they have great difficulty remembering when and

where recent events occurred. Damage to the frontal lobes can also produce striking distortions of episodic memory in which patients claim to remember events that never occurred.

Semantic memory

Semantic memory refers to general knowledge of facts and concepts that is not linked to any particular time and place. For example, episodic memory is critical for remembering a specific visit to the city of Bangkok, whereas semantic memory is important for knowing that Bangkok is the capital of Thailand. The acquisition of new semantic memories depends on the integrity of the medial temporal lobe. For instance, amnesic patients have great difficulty acquiring new vocabulary and factual knowledge, although they can acquire large amounts of new semantic knowledge when that information is presented repeatedly. Whereas the outputs of episodic memory contribute to an experience of remembering the contextual details of a past event, the outputs of semantic memory contribute to an experience of knowing, or being familiar with a fact or association. Amnesic patients make fewer remember and fewer know judgments about recent events than nonamnesic control subjects, a phenomenon that suggests they have impairments in both episodic and semantic memory.

A patient with damage to the anterior sectors of the temporal lobe appeared to remember difficult in specific past episodes, but had great difficulty understanding the meaning of common words and had lost knowledge of historical events. Similarly, elderly patients characterized by the syndrome of semantic dementia have an extremely impoverished knowledge of the properties of specific objects, yet show generally intact episodic memory.

Evidence from neuroimaging suggests that the anterior and lateral regions of the temporal lobe, particularly in the left hemisphere, roles in the semantic memory system.

Perceptual representation system (PRS)

PRS plays in important role in the identification of words and objects on the basis of their form and structure. PRS can be classified into at least three major

subsystems. A visual word form subsystem that handles information about physical and orthographic features. An auditory word form system that handles phonological and acoustic information. And, a structural description subsystem that handles information about the relations between parts of an object that specifies its global form and structure. Moreover, PRS also appears to play a prominent role in the phenomenon of priming. It is demonstrated as an improvement in the ability to identify the partial item following the recent exposure to the entire item. Priming appears to operate unconsciously, in the sense that people can exhibit effects of priming under conditions in which they lack explicit memory for having studied a word or object. Most amnesic patients exhibit intact priming across a wide variety of tasks materials and situations, including visual word and non-word priming, auditory word priming, and priming of novel visual objects and patterns. These findings indicate that priming does not depend on the medial temporal structures that mediate explicit remembering and are consistent with the idea that the posterior cortical regions that constitute the PRS are involved in priming

Procedural memory

Procedural memory refers to the acquisition of skills and habits, that is, “knowing how” rather than “knowing that”. Procedural memories are acquired gradually through repetitive practice. Studies of amnesic patients have revealed that even patients with a profound inability to explicitly remember past experiences can gradually acquire new perceptual, motor, and cognitive skills, habits that are involved in classification and categorization, and implicit knowledge of sequences or grammatical rules. These show that the acquisition of procedural knowledge does not depend on the medial temporal lobe structures that are damaged in amnesic patients. Moreover, neuroimaging evidence suggests that cerebellum is involved in some forms of procedural memory: patients with cerebellum damage have great difficulty learning to execute sequences of movements.

Working memory

The memory system considered so far are all concerned with long-term retention spanning time periods of minutes, hours, weeks, and years. In contrast,

working memory is concerned with short-term memory retention, operating over periods of second. Working memory is used to hold information on-line in the service of such basic cognitive activities as comprehending, reasoning, and problem solving. Some researcher has explained the working memory consists of three components: a central executive and two slave subsystems that support it. One subsystem, called the phonological loop, allows rehearsal or recycling of small amount of speech-based information, but is not necessary for entering information into long-term memory. The second slave subsystem of working memory is the visuospatial sketch pad. It is thought to be involved in the short-term retention of visual and spatial information. Evidence from brain-damaged patients and PET scan suggests that a variety of regions in the right hemisphere, including the visual association cortex, inferior parietal lobule, and inferior prefrontal cortex, are important components of the visuospatial sketch pad (13).

2.3 Working memory and the frontal lobe

Defining the role of the frontal cortex in memory is a complex issue that requires consideration of both the types of mnemonic process and the precise regions of the frontal cortex involved. In addition to its more direct involvement in certain aspects of memory, the frontal cortex carries out a variety of executive functions concerned with the organization of behavior. It is inevitable that these executive functions will find expression in memory as much as in other aspects of cognitive functioning. The frontal cortex is also a large part of the cerebral cortex and comprises many different architectonic areas that have distinct connections with other cortical and subcortical regions of the brain. These widely differing connections reflect functional differences that must be addressed in any attempt to deal with the function of the frontal cortex. Input to specific part of the prefrontal cortex originates from the visual prestriate cortex, the auditory and the visual temporal association cortex, and the posterior parietal cortex. Most of these connections are reciprocal, indicating that the prefrontal cortex can both influence and be influenced by posterior cortical area that are involve in the processing of visual, auditory, and somatosensory information. In addition, the frontal cortex is strongly linked with regions of the medial temporal lobe critical for mnemonic processing via two pathways. One is a large fiber system

running from the dorsolateral and dorsomedial frontal cortex and the cingulate cortex to the presubiculum of the hippocampal complex. The other is an another association system that links the caudal orbitofrontal cortex with the entorhinal and perirhinal cortex.

2.3.1 Damage to the prefrontal cortex causes memory impairments in human

Although performance on several standard tests of verbal and nonverbal memory can be normal in patients with prefrontal cortical damage, mnemonic performance can be severely impaired in certain circumstances. Mostly on short-term memory tasks that require comparisons among a small set of recurring stimuli or judgments of the relative recency of stimuli. These tests are very similar in requirements to those of the classical delayed-response tasks, where a few stimuli recur all the time and the main requirement is to discriminate their most recent occurrence from several earlier ones.

Patients with dorsolateral prefrontal lesions have major impairments monitoring events within working memory. These impairments are clearly illustrated in the self-ordered working memory task in which the subject must remember the precise serial order in which events have occurred. The patients are presented with a set of stimuli and are required to select a different stimulus on each trial until all the stimuli are selected. From the moment they start responding, the subjects must keep track of the stimuli that have already been selected and compare them with the stimuli that remain to be selected.

From lesion studies in monkeys have shown that these monitoring processes within working memory depend on the middorsal part of the lateral frontal cortex (i.e., dorsal area 46 and area 9). Monkeys with such lesions can remember stimuli, as demonstrated by normal performance on recognition memory tests (13).

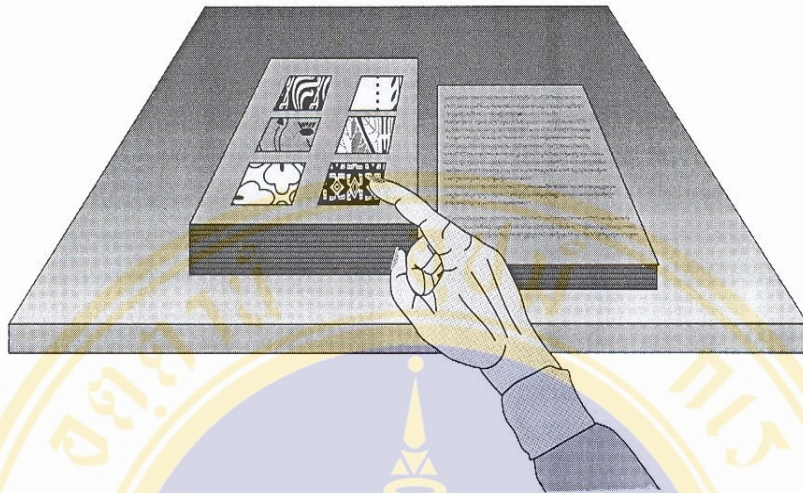


FIGURE 2.4 In an experiment, the subject was presented with a stack of cards on each of which all stimuli were presented, but in different random orders. The subject had to select one of the stimuli, point to it, and then turn to the next card to select another stimulus and continue like this until all stimuli had been selected. In subsequent testing the subject was required to recognize which object had been selected on each card from a set of equally familiar alternatives. The subject was therefore required to monitor which objects had been selected on each card from a set of equally familiar alternatives. In the example illustrated, the stimuli are abstract designs. Patients with dorsolateral frontal lesions were impaired in this self-ordered monitoring task (13).

2.4 Event-related brain Potentials – a method to record human memory

The 100 billion or so neurons in the human brain communicate by generating small electrochemical signals. If the probes from an instrument for measuring electrical energy (such as a voltmeter) are placed near such a brain cell it will register a voltage change whenever the neuron is active. Neurons can be active or generate potentials up to several hundred times per second. These electrical potentials are relatively small and cannot be monitored individually because there are a lot of neurons and neighboring neurons frequently are active close together in time. Since

1929, it has been known that groups of neurons can be monitored with probes (called electrodes) placed on the scalp outside the brain.

The on-going electrical activity of the brain measured from scalp electrodes is called the electroencephalogram or EEG. The EEG is present from before birth until death. In fact, in some places death is defined by the absence of an EEG. The EEG is usually monitored using a device called a polygraph and is displayed as continuous changes in voltage over time. In a typical EEG session a number of electrodes are attached to the scalp of the subject by glue or more commonly these days by wearing an elastic cap. The electrodes are then plugged into the polygraph and the EEG is displayed on a moving sheet of paper or on a computer screen. The EEG is used clinically to help diagnose brain disorders such as epilepsy and sleep disturbances. It gives relatively good information about the general mental state of the individual. Changes in state are associated with a change in the amplitude and frequency distribution of the EEG. For example, alpha waves are 7 - 12 Hz relatively large amplitude EEG waves that are associated with a relaxed but awake state, while beta waves are 13 - 20 Hz waves of lower amplitude than alpha, which are associated with a more alert or active mental state. Unfortunately, while the raw EEG can distinguish between such gross changes in state, it has not proven to be specific or sensitive enough to distinguish between more fine-grained changes in mental activity. For example, the EEG looks very much the same whether a person is reading, listening to music or watching TV. Therefore, raw EEG has been of limited use in studying moment-by-moment human cognitive activity.

Buried within the EEG is a signal which is more revealing about information processing in the brain. This signal can be obtained by time-locking the recording of the EEG to the onset of events such as a person reading a word on a computer screen, listening to a musical note played on an instrument, or viewing a picture in a magazine. The resulting activity is called event-related potentials (ERPs) (3). The ERPs are transient series of voltage oscillations that occurs in response to the occurrence of a discrete event. This temporal relationship between the ERP and an event is what discriminates the ERP from the ongoing electroencephalogram (EEG) activity. The ERP, like EEG, is a multivariate measure; however, unlike EEG, the ERPs are broken

down into a series of time rather than frequency domains. Moreover, ERPs are considered as manifestations of specific psychological processes (14).

2.5 Deriving Event-Related Potentials

The procedures used to derive ERPs compose of amplifiers and filters as same as obtaining EEG (see Figure 2.5).

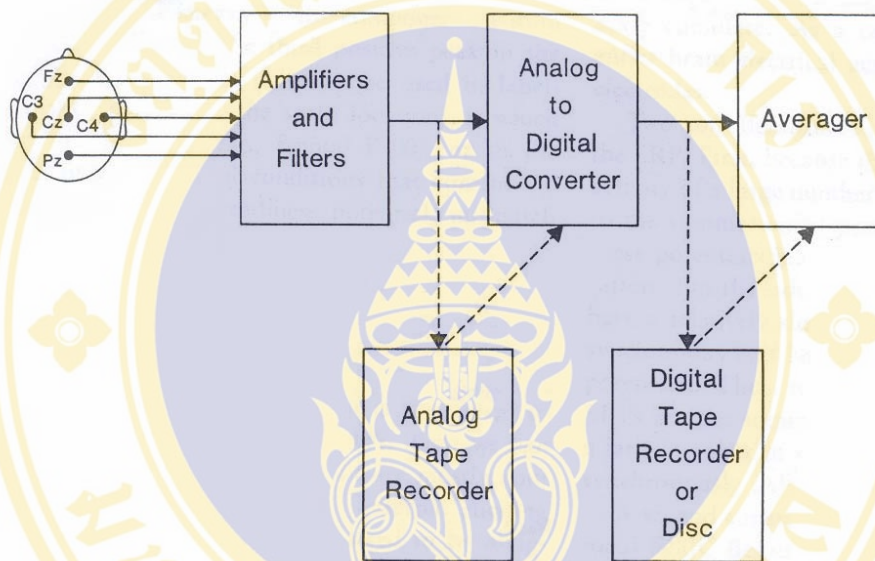


FIGURE 2.5 Schematic representation of the operations involves in the recording of event-related brain potentials. From left to right: (a) Top view of the head, indicating the placements of five electrodes (Fz, C3, Cz, C4, and Pz) from which EEG is recorded – note that other locations are also frequently used. (b) The EEG signal is then transferred to an amplifying and filtering system. (c) The amplified and filtered signal may be stored temporarily on an analog magnetic tape. (d) The analog signal is then converted into a digital signal by sampling the potential at a high frequency (usually at least 100 Hz) by an analog-to-digital converter. (e) The digitally transformed signal may be stored on a digital storage device (magnetic type or disk). (f) Finally, ERPs are exerted form the digitized EEG signal via point-by-point averaging across a large sample of trials (more than 20) (15)

Electrodes are attached to the scalp at various locations and connected to amplifiers. The recording locations are usually chosen according to the International 10-20 system. The outputs of the amplifiers are converted by an analog to digital converter. The potentials are sampled at a frequency ranging from 100 to 10,000 Hz (cycles per second) and are recorded for subsequent analysis.

The ERPs signal is smaller (a few microvolts) than EEG (about 50 μV). Thus, the analysis generally begins with a procedure to increase the discrimination of the “signal” from the “noise” (background EEG). There are several procedures that have been supported to increase the signal-to-noise ratio, including filtering, and averaging.

2.5.1 Filtering

This involves the attenuation of noise whose frequency differs from the signal. For example, most endogenous components have frequencies of between 0.5 Hz and 20 Hz. Thus, at the time of recording (or later, at the time of analysis), analog or digital filters can be used to attenuate activity outside this frequency range. Great care should be taken in the selection of filters. The amplitude and latency of an ERP component (as well as the general ERP waveform) will be distorted if the band-pass of the filter excludes frequencies of interest.

2.5.2 Averaging

This involves the summation of a series of EEG epochs (or trials), each of that is time-locked to the event of interest. These EEG epochs are assumed to be the product of two sources: (i) the ERP, and (ii) other voltage fluctuations that are not time-locked to the event. Because of other fluctuations are random with respect to the event; consequently, they should average to zero and so leave the time-locked ERP both visible and measurable. If (a) the ERP signals are constant over trials, (b) the noise is random across trials, and (c) the ERP signals are independent of the background noise, then the signal-to-noise ratio will be increased by the square root of the number of trials included in the average.

Because ERPs are measured between two locations, which vary as a function of (a) the electrode site at which they are recorded and (b) the reference electrode used. Spatial (topographic) distribution is regarded as an important discriminative

characteristic of the ERPs. Therefore, positive and negative peaks in the ERPs are generally described in terms of their characteristic scalp distribution, their polarity, and their latency.

The labels given to the peaks of an ERP waveform usually include description of polarity and latency. For example, P300 refers to a positive peak with a latency of 300 msec. A similar labeling system involves a description of polarity (P and N) followed by a number that perform of orderly latency of the component. Within this system, “P3” refers to the third positive peak in the waveform. Other description for refers peaks reference used the scalp locations at which the potential is maximal (e.g., frontal P300) (15).

2.6 The relation between memory and P300

After 30 years have been passing on the research of P300, there is no evident to confirm brain activity underlying the scalp. There is an only estimate that P300 may result from the summation of activity in cortical and possibly subcortical areas. Some evidence shows that P300 may generate from medial-temporal lobe because the lesions of this area are affected to the P300 amplitude.

Some researcher reported that P300 amplitude is sensitive to stimulus probability that provided to subject's task. If the events occur while the subject is performing another task, then the infrequent event does not elicit the P300. In addition, P300 can be elicited by diversity of stimuli – as long as the subject is able to classify them unambiguously. In research on P300 latency found that it may reflect stimulus evaluation or categorization time. There is observation from correlation between P300 latency and reaction time which show that it is higher when subjects are interested in accuracy rather than the decision speed. Furthermore, if distinguishing types are more difficult, P300 latency will become longer. However, it appears that P300 latency depend on the process of stimulus evaluation and categorization rather than current response. From the mention above, some researcher propose that P300 may be related to the updating of models of the environment or context in working memory. This updating depends on the current event and has implicated for the processing of the response to future events.

At the present, ERPs have been recorded in paradigms testing the recognition of a previously presented item. Some of these early studies employed the Sternberg memory search paradigm. Subjects are presented with a short list of items to be memorized and asked to indicate that each item belongs to the memory set. The most of this research focused on variations in the latency of P300 as a function of the number of items in the memory set. It was observed that positive response (Yes) was elicited with large P300 than negative response (No) (15).

2.7 Psychological meaning of P300 and Applications

In the study, it is examined for the visual recognition memory in a group of eight hypoxic brain-injured patients and 24 age-matched controls by visual oddball task, that can be summarized as follows. First, P300 was prolonged and displayed different scalp topography in the patients compared with the control. Second, the target P300 in control group was the largest one at the parietal recording sites but more frontal scalp distribution in increasing age subject's. Third, in both recognition conditions the reaction times and accuracy were degraded in the patients compared with the controls (16).

K.Hirata et al. studied in 26 patients with dementia of Alzheimer's type (DAT) and 12 age-matched normal subjects by auditory oddball stimulation. He found that the P300 latencies were prolonged and amplitudes decreased in DAT group were compared with control groups. Moreover, the reaction time was slower in the DAT group than in the control group (17).

In the study cognition processing used for paradigm test elicited the attention related P300 component. The 17 patients with dementia of the Alzheimer type were compared with age matched controls. Researcher found that P300 latencies represent an index of stimulus evaluation time which increasing in the reaction time. This is defected in the organization of motor response (18).

Some researcher studied the effect of interstimulus interval on visual P300 in Parkinson's disease. They measured ERPs and reaction time in 38 non-demented patients with Parkinson's disease and 24 healthy elderly subjects. They found that P300 latency at Cz and Pz delayed after rare target stimuli in patients with Parkinson's

disease. Moreover, the reaction time was prolonged in patients compared with the normal control (19).

A study from Alzheimer and multi-infarct dementia patients by Ryuji Neshige, Geoff Barrett, and Hiroshi Shibasaki. They indicated that P300 latency was greater than the means values from the normal age, although they are mild or moderate for these diseases. Some study suggested that P300 latency was significantly prolonged in more than half of 15 demented patients.

In addition, they summarized that P300 component has been associated with a variety of cognitive processes. The P300 latency is considered to reflect the speed of neural events underlying perception and discrimination of the target or rare stimulus; matching with that particular information which against stimulus categories in memory; and making an appropriate decision whether to response it or not. Thus to slow neural processing involved in stimulus evaluation or decision-making functions is postulated to underlie the associated impairment of P300 latency. In the other words, P300 latency appears to be strongly associated with attention and short term memory but not with general intelligence and reasoning (20).

P300 amplitude is believed to index brain activity that is “required in the maintenance of working memory” when the stimulus environment is updated. P300 amplitude is also proportional to the attention in a given task and has been associated with superior memory performance. P300 amplitude can be measure the CNS activity that reflects the processing of incoming information and manifest in term of memory.

P300 latency is considered for measure the speed classification and this is unrelated to response selection processes. This property makes the P300 is a valuable tool for estimate cognitive function because its latency is an index of the processing time required before response production. Not only that, it is a sensitive to temporal measurement of neural activity underlying the processes of attention allocation and immediate memory. P300 latency is negative correlated with mental function in normal subjects, with shorter latencies associated with superior cognitive performance. The neuropsychological tests that are best correlated with P300 latency that how rapidly subjects can allocate and maintain attention resources. This indicates that if P300 latency increases, cognitive capability will decreases because of dementing illness. Therefore, P300 latency is directly associated with cognitive capability in both

normal and patient populations. Indeed, P300 in the clinical usefulness must be consideration in the control of previously ignored stimulus characteristics, task factors, and inter-subject variability (21).

2.8 How to elicit P300

The P300 component often is elicited using a simple discrimination task, this is called oddball paradigm.

In this task, there are two types of random stimuli with one of the two is infrequent appearance. In the above figure shows auditory stimuli that consist of two different tones, with the target tone presented less frequently than non-target tone. Subject must to distinguish the two tones by responding to the target only (e.g., mentally counting, pressing a button). This procedure most often used in the clinical setting.

Although, this oddball is a simple task, in clinical application must to consider the component of stimulus characteristics, task factors, subject variables, electrophysiological recording methods, and measurement procedures. Because these components effect to P300 ERPs that are elicited from clinical population, the appropriate procedure is considered. Unfortunately, at the present has not standard method to performance. However, in many clinical setting for measuring P300 assign some parameter respect to stimulus and task variables.

For auditory paradigms, the tone's frequency of 1,000 Hz and 2,000 Hz are suitable frequency for used because these are frequency's range in the speech of human. Furthermore, stimulus intensity of 70 dBL is suitable intensity for used also because this is generally of stimulus intensity of human in the speech. Auditory stimulus intensity should be constant for avoid subject's confusion because P300 amplitude and latency measures with variations in intensity.

For visual paradigms, the picture does not fixed assign but should not be make subjects confuse. Furthermore, a stimulus interval is also affected to P300. The time for presentation should long enough for subject's easy discrimination (e.g., 50-100 ms).

In addition, P300 amplitude is affected to the probability of target's appearance. Target stimulus is set at 20% and non-target stimulus is set at 80% produce large P300 amplitude when compare with more or less than target stimulus. Not only that,

interstimulus interval (ISI) is important factor. ISI should be set 2-3s because this is a suitable time interval for most subjects; in addition, this time interval generates robust component amplitudes. Short ISI ($< 2s$) should not used since this task is very difficulty for elder subjects. Longer ISI (6-10 s) will produce large P300 components but, this time interval can reduce the influence of probability on target amplitude. However, long ISI is suitability for patients who have slowly decision making.

Moreover, a critical electrophysiological parameter is the recording band-pass or time constant. The high-pass filter should be as low as possible because the P300 wave has a relatively low frequency (~ 3 Hz). A setting of 0.01 Hz is used in most experimental work because it does not distort the slow wave activity that contributes to the P300 component. However, high-pass filters of 0.1 or 0.5 Hz can be used, as these setting will produce acceptable amplitude and latency values. A low-pass setting of 30 Hz can remove the higher frequencies, including 60-cycle activity.

At least at the frontal (Fz), central (Cz), and parietal (Pz) midline electrode sites at the scalp distribution, should be used for elicited the P300 peak. Scalp distribution data are especially important for elderly and many patient populations, in whom distinct peaks may not be clearly present. The active electrodes should be in a monopolar configuration referenced to link A1/A2 (i.e., earlobes or mastoids) or another site such as the nose tip. The ground is usually placed on the forehead.

The minimum number of single trials required to produce stable P300 amplitudes has been demonstrated empirically to be ≥ 20 . Furthermore, because peak latency generally changes little in each trial; consequently, the rejection of some epochs because they contain artifact will not affect this measure greatly.

2.9 P300 Measurement and Identification

The P300 is measured the quantity of amplitude (size) and latency (timing). The P300 component can be identified as the largest positive-going peak of the ERP waveform occurring after the earlier, exogenous components (N100, P200, N200) within a given latency window (e.g., 250-500 ms, this range can vary depending on stimulus, task, and subject variables). Amplitude (μv) is defined as the voltage difference between the component peak and a prestimulus baseline. Latency (ms) is defined as the time from stimulus onset to the point of maximum positive amplitude

within the latency window. In addition, P300 scalp distribution is defined as the change in component amplitude across the Fz, Cz, and Pz locations. Amplitude increases and latency decreases from anterior to posterior sites. Scalp distributions have effect to the variation in amplitude patterns that refer to information about P300 neural generators. Moreover, these morphologic and timing characteristics are the most important factors to be used in identifying a subject's P300 component measures (21).

Conclusion

The ERPs are one of the main tools available for cognitive neuroscientists. They are tools that are appropriate for investigating the function of the human brain. Especially, P300 ERPs components are related to cognitive function; as a result, they are useful for assessment of cognitive ability in both normal subjects and patients with neurologic and psychiatric disease.

CHAPTER III

MATERIALS AND METHODS

This chapter clearly describes the method of recording brain memory signal from subjects including getting ready, preparation and methods of testing. The paired t test is used to compare two set of subjects' signal. Moreover, Parseval's relation is used to calculate energy density of subject's signal for support idea that brain uses more energy in complicated picture than less detail picture.

3.1 Subjects

All 30 healthy participants were recruited from Siriraj hospital and various faculties in Siriraj (25 women and 5 men, aged 18 to 30 years). They are firstly informed with the general aims of the investigation. Subjects who passed medical interview are examined to make sure that they have no history of neurological, psychological, and visionary problems.

3.2 Experiment

3.2.1 Getting ready

Subjects are instructed to comfortably seat in a chair and rest their head and arms on appropriate position rest. Their demo graphic data are collected, including: age, weight, height, education, occupation, medical history (included past illness, surgical history, and accident). They are instructed to concentrate at a picture during testing for a reduce of noise signal.

3.2.2 ERPs recording

The silver disc electrode placed at Fz, and Cz (international 10-20 system), and Ground is applied. Reference electrode is applied to mastoid (see figure 3.1).

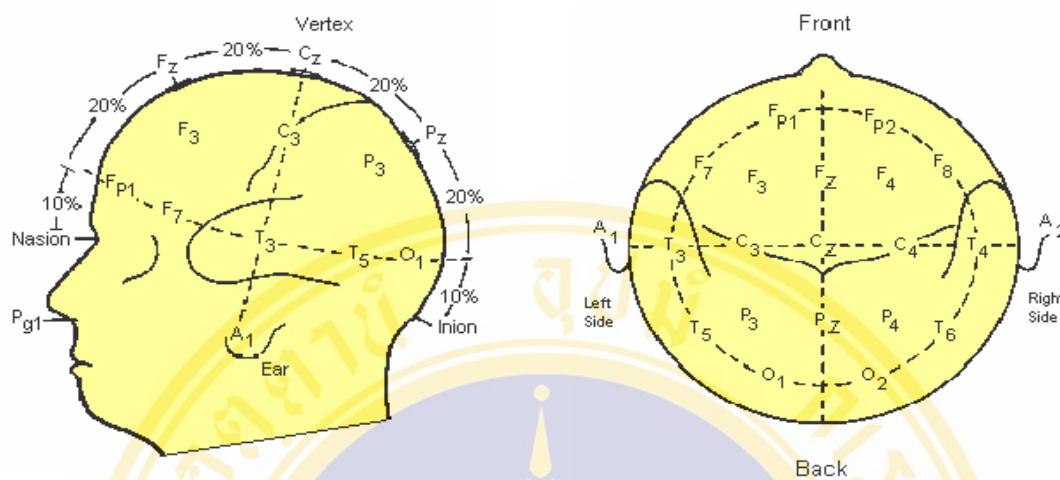


FIGURE 3.1 The International 10-20 system of electrode placement.

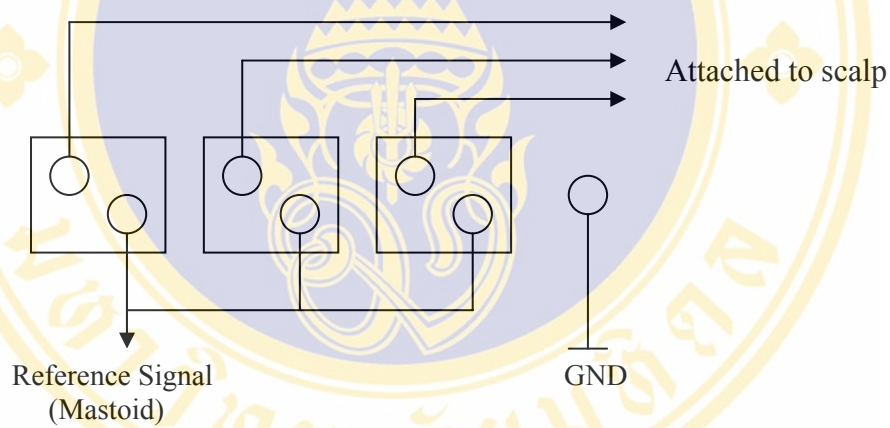


FIGURE 3.2 Channel instruments for scalp electrodes from Medelec machine.

The electrodes are then filled-up with conductive jelly, in order to reduce the impedance electrodes at skin contact. The electrical activity is amplified with the built-in amplifier in the Medelec Sapphire Premiere machine. The amplifiers are set for a high-frequency cut-off of 50 Hz (12 dB/octave roll-off) and a low-frequency cut-off of 0.1 Hz (6 dB/octave roll-off). The Amplifiers sensitivity is set at 50 uV/division.

The data of subjects are transferred from the scalp electrodes through Medelec Sapphire Premiere machine amplifiers and then through A-D converter (Analog to digital converter). Sampling rate for all signals is 20 us (50 kHz). ERPs are recoded by the Medelec machine and are displayed on the monitor.

3.2.3 Method of testing

After the subjects are already attached with the electrode already, they are explained step by step until they understand how to perform the experimental task.

The beginning of the methods at the oddball paradigm is applied to the subjects with records ERPs. For Oddball paradigm, there are two types of pictures in this experiment. One is a standard picture (checkerboard picture), and another is the new picture (view picture).

First, subjects are tested with the standard pictures; it is shown on the TV by the Medelec Sapphire Premiere machine, which generates signal pass video output port. There are two sizes of standard picture (one is big, and another is smaller than the former one). The bigger checkerboard refers to the non-target picture and the smaller one refers to the target picture. Checkerboard picture is presented in random sequence when small checkerboard appears subjects which must be counted silently and remembered in their mind.

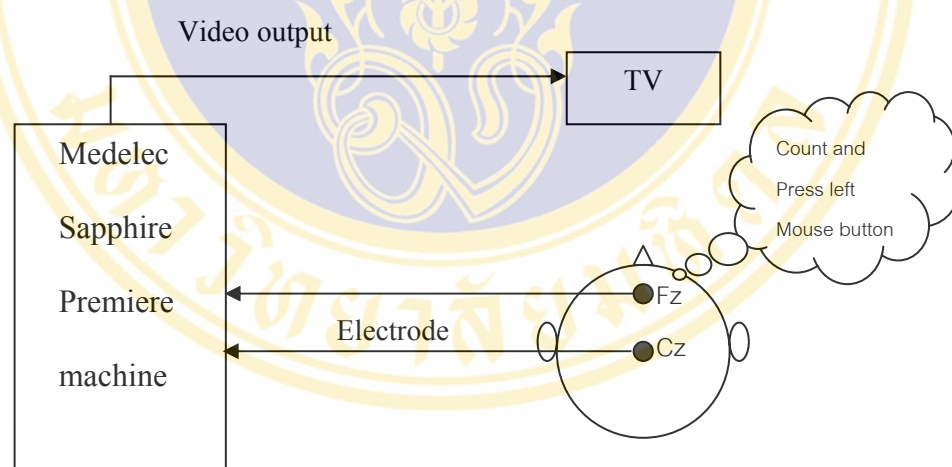


FIGURE 3.3 Schematic diagram illustrating data acquisition from standard pictures.

During the test, there are 64 small sized checkerboard pictures which can only represent the target pictures. However, the total of the pictures are presented around 220-240, including target pictures.

That is because the P300 ERPs appears more dominantly for the target stimulus than the non-target about 300 ms after stimulus. However, the appearance on P300 not only depends on the mentality of the subjects such as their habituation, fatigue, judgement and recognition, but also is influenced by background waves. These cause the fluctuation of amplitude and latency. Consequently it is difficult to precisely estimate the event potential from a single trial response. Therefore, waveform of ERP is usually estimated by summing up a large number of responses to reduce the background wave and noise components. Then maximum amplitude of each component such as P300, and its latency at the maximum point are measured (23).

Duration of picture is defined 10 seconds until the test finishes. The pictures are presented randomly, so subject hardly forecast a sequence of pictures. In each time that target picture which appears on the Medelec Sapphire Premiere machine will record ERPs signal. When next target appears, the medelec will sum up with this ERPs signal before, the ERPs signal and repeat it until the last target picture. Finally, there is only one ERPs signal representing of each subject.

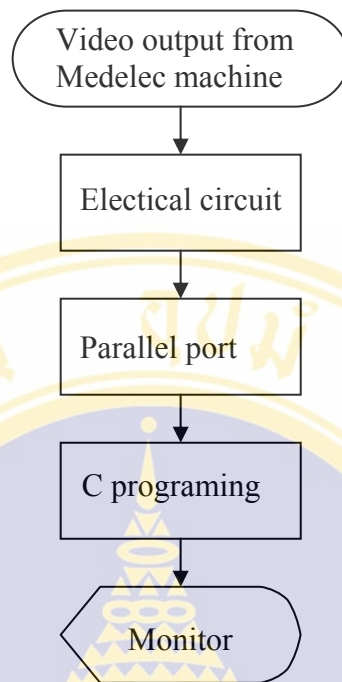


FIGURE 3.4 Schematic diagram illustrates picture on monitor from Medelec video output.

Second, subjects are tested with new pictures, the method and the process are as same as the standard method but the new pictures are shown on the PC monitor. The big checkerboard is replacing by view picture (animal, river, aircraft etc.) and small checkerboard is replacing by human picture (man and woman). The sequence of the picture is controlled by the signal that is sent from the Medelec Sapphire Premiere machine. This signal from video output is sent through electronic circuits before to the parallel port. From this, the coming signal is checked by C program for showing picture on the PC monitor.

Pictures are presented in the center of the monitor screen during the test. Subjects must discriminate the picture on monitor by pressing mouse button. Mouse is fixed at the appropriate position so that subjects do not need to move their arm to press it (because arm movement produces noise signal that can disturb ERPs signal). If subjects think that the picture is the target picture, they do press left mouse button. But if they think that the picture is the non-target picture, they do not press mouse button any more. By the end of the test the program will have shown the result from the pressing mouse button.

3.2.4 Communication between Medelec machine and PC

The most important part in this experiment is about the synchronization between the display pictures on monitor and the record visual P300 ERPs. At the behind of Medelec Sapphire Premiere machine, there is a video output used for a connect to TV to display checkerboard picture. At this video output, the signal is converted to the digital signal by comparator circuits and sent through pin 13 and 12 of parallel port to computer (See the schematic diagram of the electrical circuit in appendix A).

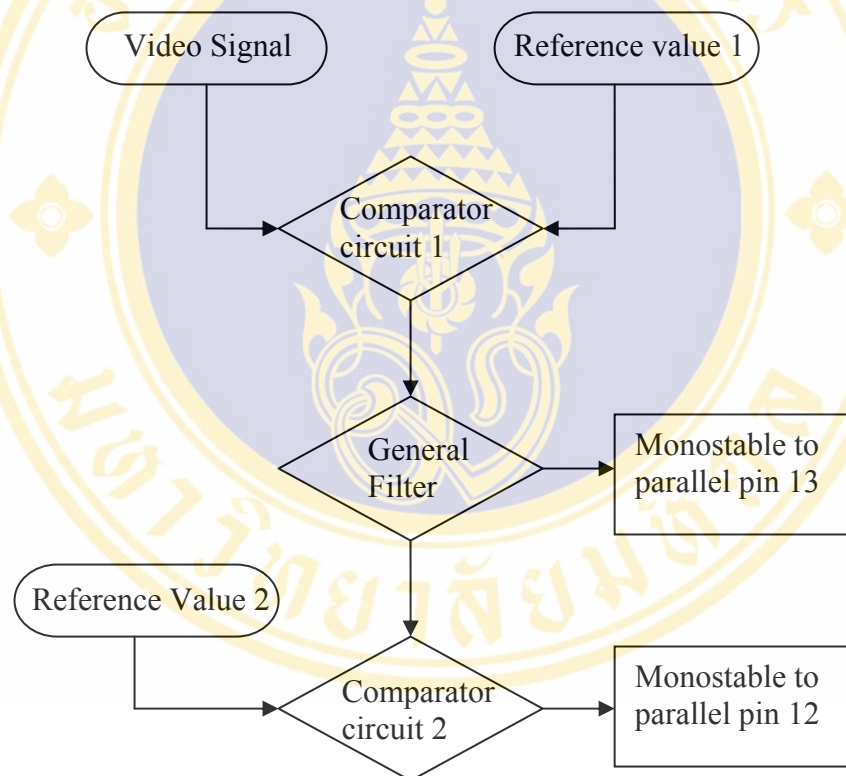


FIGURE 3.5 Schematic diagram illustrates signal to parallel port.

Parallel port		Picture
Pin 13	Pin 12	
0	0	Not change
0	1	Not change
1	0	Target
1	1	Non-target

Table 3.1 The truth table of parallel port is used to display picture

3.2.5 C program

C program control the picture displaying on the monitor. From the above truth table, picture is changed if pin 4 has signal coming. Addition, signal that coming to pin 2 control target or non target picture (See appendix C).

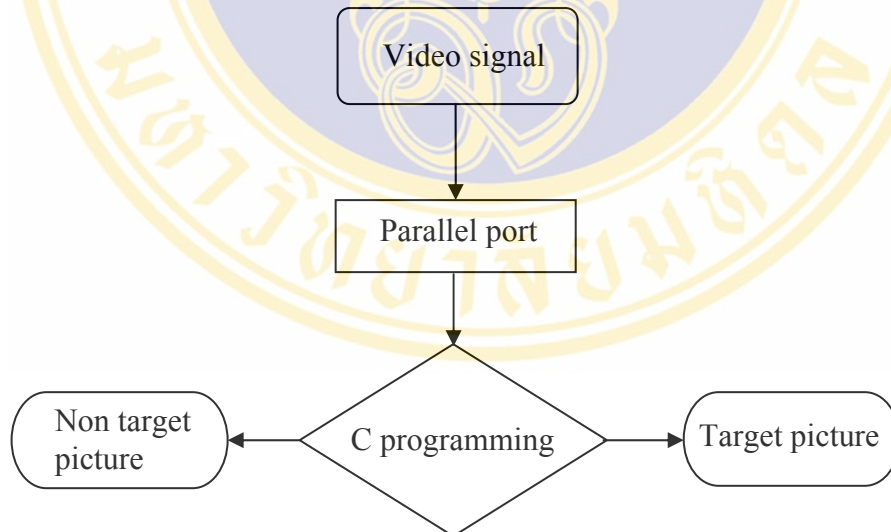


FIGURE 3.6 Schematic diagram illustrates C programming.

3.3 Data Acquisition

The raw data from Medelec machine is recorded in a form of trace data (name.D00). This data is converted into an ASCII format by READMFF (name.001-

name.002 is automatically set to show the store numbers). After this data, it can be plotted a graph by using the matlab program.

This is an example of raw data from one subject. The groups are divided into two, the first one gives a signal at Fz and Cz which occurs from testing with new pictures, when the second one gives a signal at Fz and Cz which occurs at the same location, but testing with standard pictures.

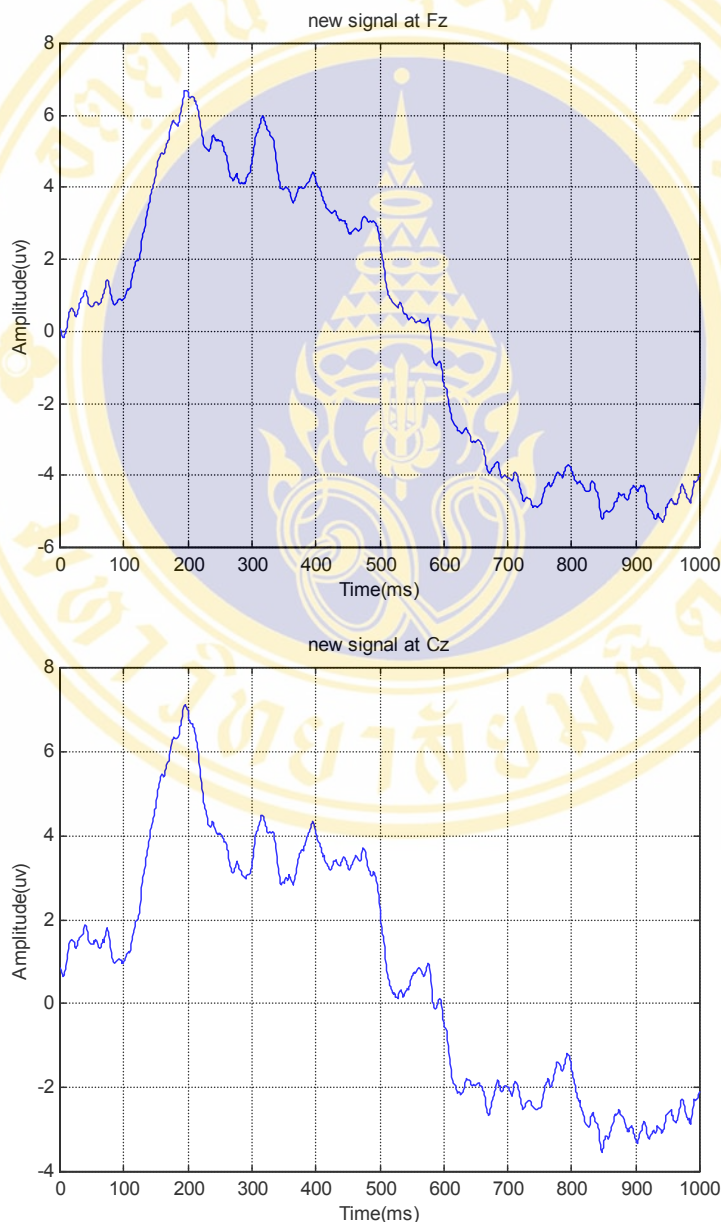


FIGURE 3.7 The signal occurs from new picture at position of Fz and Cz respectively.

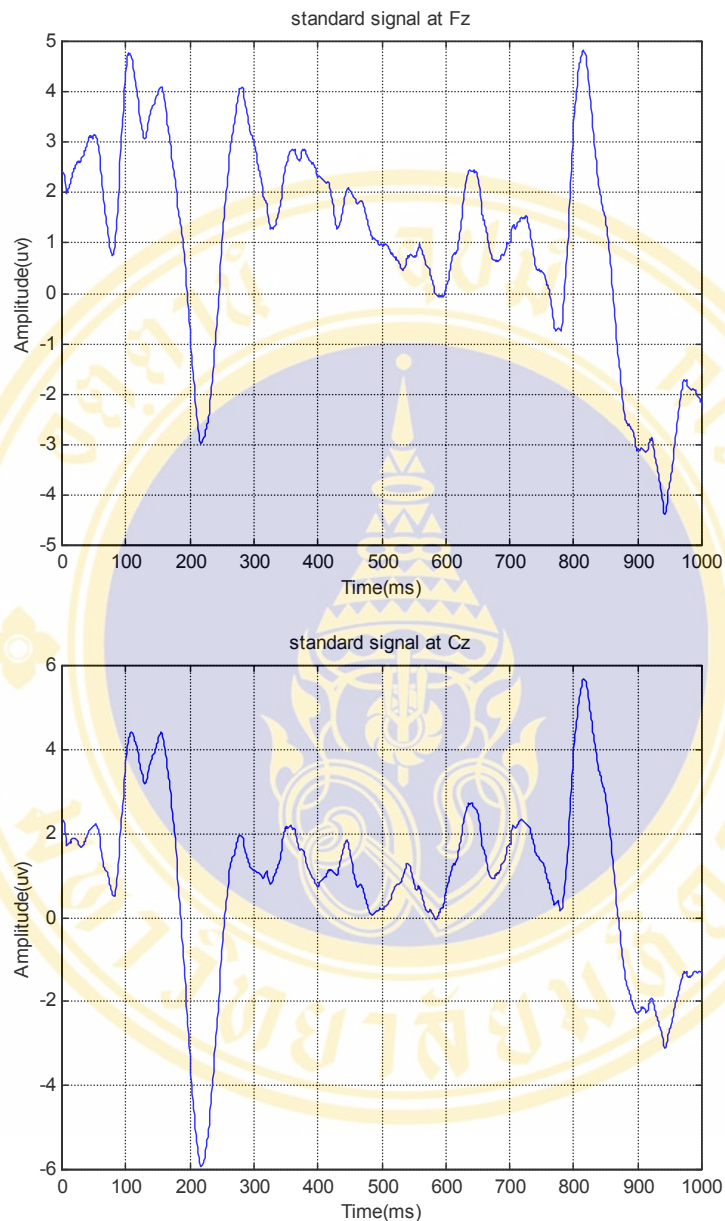


FIGURE 3.8 The signal occurs from standard picture at position of Fz and Cz respectively.

3.4 Data Analysis

In this experiment, the numbers of points in each data set are the same, and they are organized in pairs, in which there is a definite relationship between each pair of data point. Because the data in each sample set are related in a special way, so the paired t test is used to compare two small sets of quantitative data.

3.4.1. The Paired t-Test

Subject's signal P300 amplitude and latency are important factors that can interpret the state of subject memory. The new method should indicate the significance of amplitude and latency in the same way of P300 as the standard method, consequently, paired t test is the statistic method that's an appropriate tool.

The latency of P300 is measured in interval of 250-400 ms of the signal. The highest peak in this interval is P300 amplitude. Then the paired t-test is used to test hypothesis. The data from this experiment which is collected in pairs is recorded the latency and amplitude of P300 on each method. The test procedure would then consist of analyzing the differences between value on each method. If there is no difference between value, then the mean of the differences should be zero. So, it can written

$$H_0: \mu_D = 0$$

$$H_1: \mu_D \neq 0$$

The objective of the decision is that the hypothesis is true or false. The procedure for decision making is called test of hypothesis. In test of hypothesis, random variation data from interested population is used. If this data is consistent with the hypothesis, then this can be concluded that the hypothesis is true: however, if this information is inconsistent with the hypothesis, this can be concluded that the hypothesis is false.

The appropriate test statistic for above equation is

$$t_0 = \frac{\bar{D}}{S_D / \sqrt{n}}$$

$$\bar{D} = \frac{\sum_{j=1}^n D_j}{n}$$

$$S_D^2 = \frac{\sum_{j=1}^n D_j^2 - \left[\left(\sum_{j=1}^n D_j \right)^2 / n \right]}{n - 1}$$

Where \bar{D} and S_D^2 are the sample mean and variance of the differences. We would reject $H_0: \mu_D = 0$ (implying that $\mu_1 \neq \mu_2$) if $t_0 > t_{\alpha/2, n-1}$ or if $t_0 < -t_{\alpha/2, n-1}$. One-sided alternatives would be treated similarly (22).

3.4.2. The total energy in the signal

If $x(t)$ and $X(j\omega)$ are a Fourier transform pair, then

$$\int_{-\infty}^{+\infty} |x(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |X(j\omega)|^2 d\omega \quad \text{----- (1)}$$

This expression, referred to as Parseval's relation, follows from direct application of the Fourier transform. Specifically,

$$\begin{aligned} \int_{-\infty}^{+\infty} |x(t)|^2 dt &= \int_{-\infty}^{+\infty} x(t)x^*(t) dt \\ &= \int_{-\infty}^{+\infty} x(t) \left[\frac{1}{2\pi} \int_{-\infty}^{+\infty} X^*(j\omega) e^{-j\omega t} d\omega \right] dt \end{aligned}$$

Reversing the order of integration gives

$$\int_{-\infty}^{+\infty} |x(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X^*(j\omega) \left[\int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt \right] d\omega$$

The bracketed term is simply the Fourier transform of $x(t)$; thus,

$$\int_{-\infty}^{+\infty} |x(t)|^2 dt = \frac{1}{2\pi} \int_{-\infty}^{+\infty} |X(j\omega)|^2 d\omega$$

The term on the left-hand side of eq. (1) is the total energy in the signal $x(t)$. Parseval's relation is said that this total energy may be determined either by computing the energy per unit time ($|x(t)|^2$) and integrating over all time or by computing the energy per unit frequency ($|X(j\omega)|^2 / 2\pi$) and integrating over all frequencies. For this reason, $|X(j\omega)|^2$ is often referred to as the energy-density spectrum of the signal $x(t)$.

3.4.3. Correlation

The correlation is one of the most common and most useful statistics. A correlation is a single number that describes the degree of relationship between two variables. The degree of association is measured by a correlation coefficient, denoted by r .

The correlation coefficient may take on any value between plus and minus one.

$$-1.00 \leq r \leq +1.00$$

The sign of the correlation coefficient (+, -) defines the direction of the relationship, either positive or negative. A positive correlation coefficient means that as the value of one variable increases, the value of the other variable increases; as one decreases the other decreases. A negative correlation coefficient indicates that as one variable increases, the other decreases, and vice-versa.

The coefficient of determination (r^2) is a variable that describes the appropriate of correlation equation in data. If data's point closes the linear correlation, this is a good linear correlation. Consequently, the r^2 will high value.

The equation is

$$r^2 = \frac{SS_{\text{explained}}}{SS_{\text{total}}}$$

$$SS_{\text{explained}} = b^2 \left[\sum x_i^2 - \frac{(\sum x_i)^2}{n} \right]$$

$$SS_{\text{total}} = \sum y_i^2 - \frac{(\sum y_i)^2}{n}$$

$$b = \frac{n \sum xy - (\sum x)(\sum y)}{n \sum x^2 - (\sum \bar{x})^2}$$

CHAPTER IV

RESULTS

This chapter describes delay time, paired t test result, and energy density of signal. Delay time section describes the different signal pass through TV and PC monitor, whereas paired t test section describes the relation between standard method and new method. Finally, energy density of signal describes the brain's energy use during test both types of pictures.

4.1 Experiment flowchart

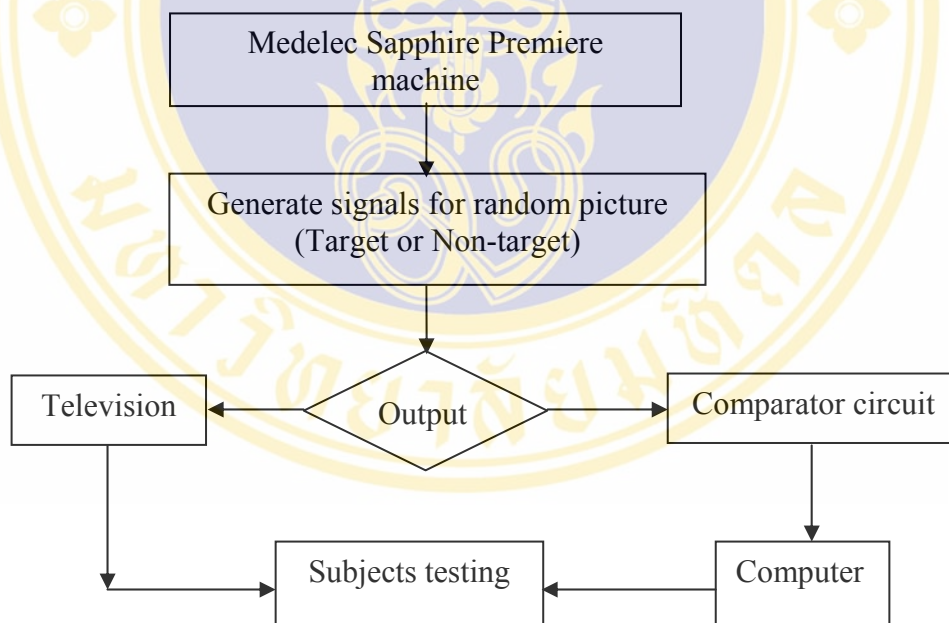


Figure 4.1 The flowchart shows subjects experiment with television (standard pictures) and Computer (new pictures)

4.2 Output signal

From the above flowchart, The Medelec Sapphire Premiere machine generates two frequencies of signal level for control checkerboard picture display. One is high frequency signal for display target picture and another is less frequency signal for display non-target picture (see figure 4.2 and 4.3). These signals are sent through the television for a display of checkerboard picture for standard testing. However, for compared testing, the two frequencies signals are passed comparator circuits. These signals are discriminated between high and low level before passing through computer. Then, the computer program receives signal and display new picture on the monitor.

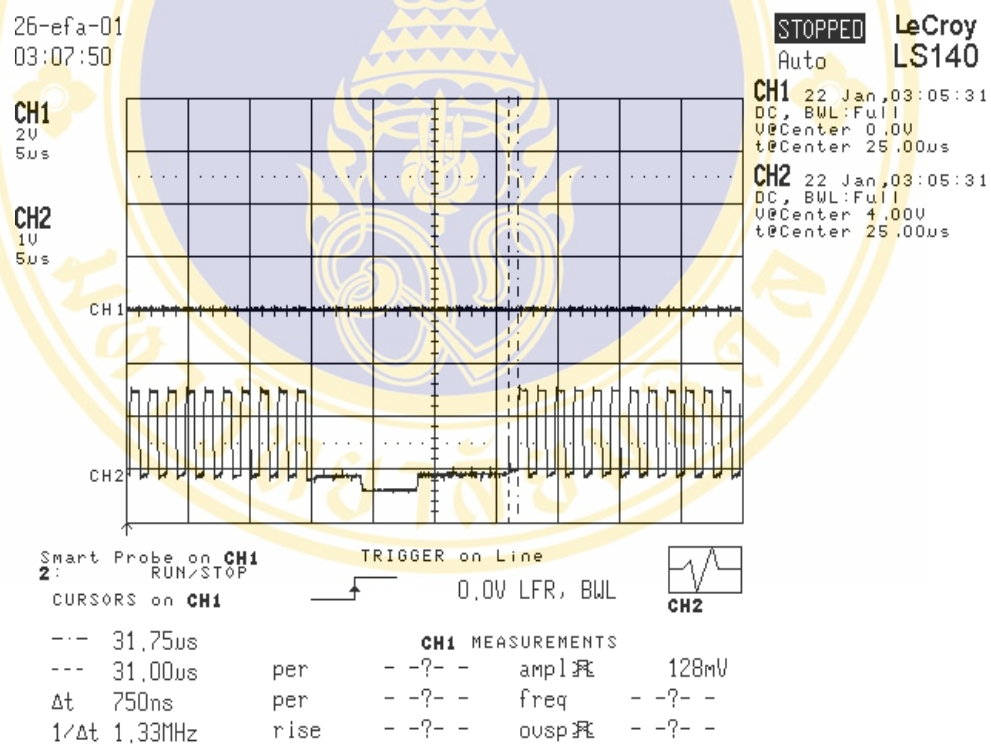


Figure 4.2 target signal is generated from Medelec Sapphire Premiere machine

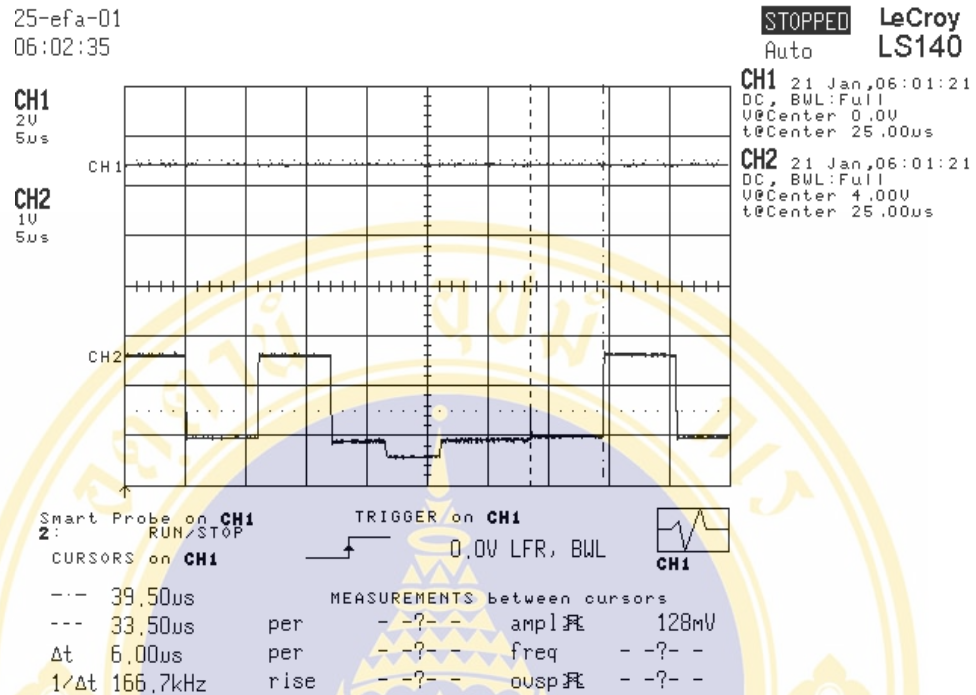


Figure 4.3 Non-Target signal is generated from Medelec Sapphire Premiere machine

4.3 Delay times

Because the Medelec Sapphire Premiere machine generates two frequencies signal, so the part of comparator circuit will added filter circuit for discriminating two signals before. Then, the two filtered signals are compared with fix value. Finally, the signals are converted into high or low signal and passed through parallel port of computer. However, all the processes produce delays time from electronic circuits which can be measured as the figure showed below.

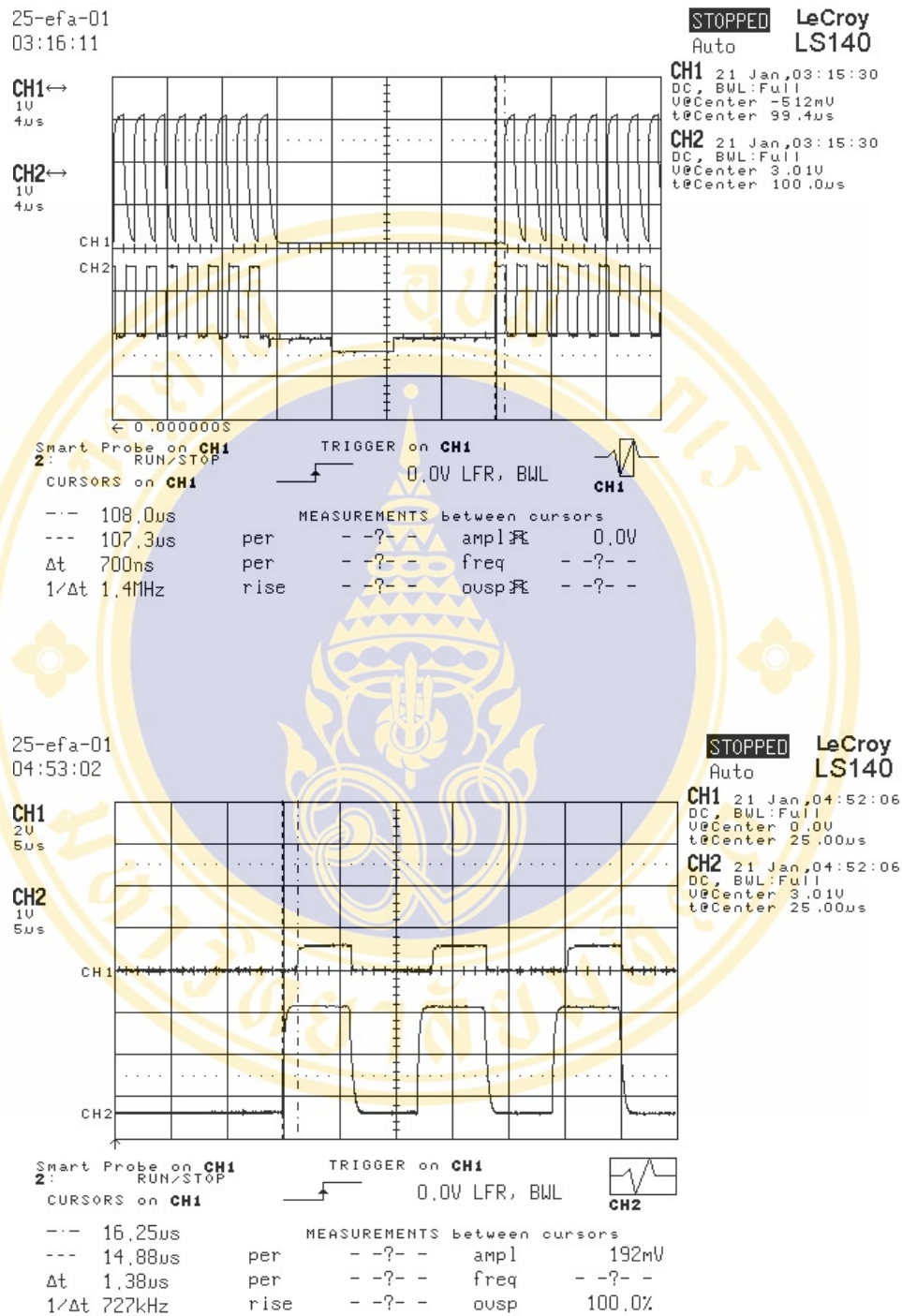


Figure 4.4 Target and Non-target delay times

From the above figure, the delay times from the Target signals about 700 ns and the delay times from the Non-target signals about 1.38 us. However, the delay times of two signals are less significant.

4.4 Result of P300 amplitude and latency

Subject's ERPs are recorded with Medelec Sapphire Premiere machine in the form of ASCII. However, It can be converted into text file for plot in matlab program (see appendix B).

The amplitude and latency of P300 is measured both from the standard method and from the new method in each subject. Finding P300 amplitude, this is a first peak which can notice in latency of 250-400 milliseconds. Next, Finding P300 latency, it's relating to P300 amplitude. The location of P300 amplitude indicates the P300 latency.

The table below compares the latency and amplitude at Fz and Cz of 30 subjects.

Subject	New Fz		New Cz		Standard Fz		Standard Cz	
	Lat	amp	Lat	amp	Lat	amp	Lat	amp
1	316	-5.971	315	-4.487	281	-4.103	280	-1.949
2	323	-12.585	322	-12.08	264	-10.965	264	-9.767
3	340	-8.385	340	-7.664	266	-7.32	267	-5.305
4	307	-13.755	309	-11.083	216	-9.012	216	-3.633
5	360	-5.551	359	-4.98	254	-2.867	254	-3.7
6	274	-10.623	274	-8.291	287	-4.817	286	-2.591
7	320	-4.162	321	-3.422	284	-0.071	283	0.417
8	368	-12.995	365	-9.997	290	-1.959	288	1.716
9	312	-14.323	312	-11.354	278	-14.337	279	-13.735
10	327	-9.54	328	-4.549	265	-3.08	265	-1.125
11	324	-7.076	324	-1.411	253	-1.57	254	2.025
12	332	-10.994	333	-8.884	257	-4.568	257	-2.221
13	305	-11.909	310	-6.78	291	-4.435	292	-1.78
14	338	-8.116	338	-5.822	338	-5.085	338	-5.398
15	312	-9.982	310	-5.941	291	-6.672	295	-3.065
16	359	-4.872	356	-4.533	308	0.123	308	-1.41
17	293	-2.472	293	-2.84	301	3.681	304	1.253
18	297	-11.192	298	-9.248	313	-2.268	313	-1.772
19	335	-10.286	334	-9.319	287	-1.657	288	-0.777
20	339	-6.399	338	-6.188	287	-4.283	289	-1.375
21	291	-11.424	291	-11.056	283	-2.121	280	-1.679

Subject	New Fz		New Cz		Standard Fz		Standard Cz	
	Lat	amp	Lat	amp	Lat	amp	Lat	amp
22	305	-9.32	304	-6.366	281	-2.062	277	-2.383
23	347	-6.601	346	-6.818	295	-2.916	293	-2.797
24	374	-11.267	374	-5.551	337	-0.936	340	2.057
25	284	-11.859	280	-6.916	302	-12.13	301	-6.009
26	302	-13.011	304	-11.358	292	-11.3	290	-6.236
27	331	-7.823	331	-5.629	254	-2.417	260	-0.94
28	308	-11.004	308	-7.631	251	-2.15	253	-0.136
29	297	-10.909	299	-7.806	283	-2.439	281	2.13
30	307	-8.274	310	-6.819	279	14.569	279	11.125

Table 4.1 P300 amplitude and latency from subjects are tested with new picture (portrait) and standard picture (checkerboard)

From the above table, the latency of P300 is measured in interval of 250-400 ms of the signal. The highest peak in this interval is p300 amplitude also. So we can observe that one subject has two data (from two methods), and the hypothesis of this experiment is that there is no difference between those two methods.

4.5 Paired-t test

First, to compare amplitude into two methods, we calculate amplitude of Fz = 0.8803, and amplitude of Cz = 0.8937, and then, select alpha (α) = 0.05. This value indicates an opportunity for an error in the test hypothesis.

From the table IV in Probability and Statistics in Engineering and Management Science in page 679 define that $t_{0.025, 29} = 2.045$.

Due to $|t_0| < t_{0.025, 29}$ ($0.8803 < 2.045$, and $0.8937 < 2.045$) so, we can conclude that there are no difference of amplitude between new pictures and standard pictures.

Next, compares latency of two methods, we calculate latency of Fz = 0.9756, and latency of Cz = 0.9729. Then, select alpha (α) = 0.05.

Due to $|t_0| < t_{0.025, 29}$ ($0.9756 < 2.045$, and $0.9729 < 2.045$) so, we can conclude that there are no difference of latency between new pictures and standard pictures.

This part we can conclude that the result of P300 and latency from the new method are not different from the standard method.

4.6 Energy density of signal

From the two methods, it can be seen that, the prominent peak of ERPs signals from new method last longer than standard method (see appendix B).

There are some authors recommend the object recognition constitutes between lower and higher levels of image analysis. The low-level shape features that are useful for identifying an object - edges, contours, surface curvature and the like - are typically ambiguous in natural scenes, so they cannot be computed directly based on a local analysis of the image. Rather, they must be inferred based on global context and higher-level knowledge (24). Consequently, if complicated images are presented to subjects, the time is used to identify an image a prolongation.

This prolonged signal can be implied that the brain uses high energy for analyzing picture in recognition process. The table below shows the brain energy during the tested subjects with new method and standard method.

Subject	New picture		Standard picture	
	Fz	Cz	Fz	Cz
1	15,045	9,383	5,477	5,220
2	34,360	34,064	30,595	22,231
3	40,050	21,752	22,801	10,114
4	<u>130,279</u>	70,867	<u>134,941</u>	64,075
5	14,579	12,748	2,980	4,010
6	336,740	107,973	318,937	98,290
7	17,158	35,310	14,404	15,601
8	98,884	49,261	86,336	47,387
9	230,076	110,703	135,613	98,819
10	209,636	100,872	147,156	87,477
11	19,848	13,690	18,635	11,381

Subject	New picture		Standard picture	
	Fz	Cz	Fz	Cz
12	181,140	50,455	101,296	39,055
13	53,363	21,346	14,019	12,561
14	30,040	17,349	14,016	15,277
15	110,002	24,961	99,447	24,396
16	20,312	4,121	3,938	3,472
17	145,430	35,840	93,504	31,425
18	58,657	41,291	12,888	12,859
19	35,327	27,134	34,298	25,168
20	10,649	9,504	11,048	9,244
21	40,246	25,493	16,425	21,524
22	127,294	44,296	56,216	34,696
23	22,244	29,258	21,888	21,764
24	38,309	14,740	25,346	7,852
25	32,001	21,609	25,528	12,164
26	89,262	46,140	44,067	20,974
27	12,913	19,469	11,984	12,162
28	42,253	22,467	12,233	10,242
29	214,059	108,844	194,217	88,019
30	187,951	66,071	199,380	80,911

Table 4.2 Energy density of signal from subjects is tested with new picture (portrait) and standard picture (checkerboard)

4.7 Correlation

The table below showed a correlation of energy density of signal between the new and standard Cz and Fz.

New Cz (x)	Standard Cz (y)	x ²	y ²	xy
9383	5220	88040689	27248400	48979260
34064	22231	1160356096	494217361	757276784
21752	10114	473149504	102292996	219999728
70867	64075	5022131689	4105605625	4540803025

New Cz (x)	Standard Cz (y)	x ²	y ²	xy
12748	4010	162511504	16080100	51119480
107973	98290	11658168729	9660924100	10612666170
35310	15601	1246796100	243391201	550871310
49261	47387	2426646121	2245527769	2334331007
110703	98819	12255154209	9765194761	10939559757
100872	87477	10175160384	7652225529	8823979944
13690	11381	187416100	129527161	155805890
50455	39055	2545707025	1525293025	1970520025
21346	12561	455651716	157778721	268127106
17349	15277	300987801	233386729	265040673
24961	24396	623051521	595164816	608948556
4121	3472	16982641	12054784	14308112
35840	31425	1284505600	987530625	1126272000
41291	12859	1704946681	165353881	530960969
27134	25168	736253956	633428224	682908512
9504	9244	90326016	85451536	87854976
25493	21524	649893049	463282576	548711332
44296	34696	1962135616	1203812416	1536894016
29258	21764	856030564	473671696	636771112
14740	7852	217267600	61653904	115738480
21609	12164	466948881	147962896	262851876
46140	20974	2128899600	439908676	967740360
19469	12162	379041961	147914244	236781978
22467	10242	504766089	104898564	230107014
108844	88019	11847016336	7747344361	9580340036
66071	80911	4365377041	6546589921	5345870681
1197011	948370	75991320819	56174716598	64052140169

Table 4.3 Correlation in energy density of signal at Cz location

New Fz (x)	Standard Fz (y)	x ²	y ²	xy
15045	5477	226352025	29997529	82401465
34360	30595	1180609600	936054025	1051244200
40050	22801	1604002500	519885601	913180050

New Fz (x)	Standard Fz (y)	x ²	y ²	xy
130279	134941	16972617841	18209073481	17579978539
14579	2980	212547241	8880400	43445420
336740	318937	1.13394E+11	1.01721E+11	1.07399E+11
17158	14404	294396964	207475216	247143832
98884	86336	9778045456	7453904896	8537249024
230076	135613	52934965776	18390885769	31201296588
209636	147156	43947252496	21654888336	30849195216
19848	18635	393943104	347263225	369867480
181140	101296	32811699600	10260879616	18348757440
53363	14019	2847609769	196532361	748095897
30040	14016	902401600	196448256	421040640
110002	99447	12100440004	9889705809	10939368894
20312	3938	412577344	15507844	79988656
145430	93504	21149884900	8742998016	13598286720
58657	12888	3440643649	166100544	755971416
35327	34298	1247996929	1176352804	1211645446
10649	11048	113401201	122058304	117650152
40246	16425	1619740516	269780625	661040550
127294	56216	16203762436	3160238656	7155959504
22244	21888	494795536	479084544	486876672
38309	25346	1467579481	642419716	970979914
32001	25528	1024064001	651678784	816921528
89262	44067	7967704644	1941900489	3933508554
12913	11984	166745569	143616256	154749392
42253	12233	1785316009	149646289	516880949
214059	194217	45821255481	37720243089	41573896803
187951	199380	35325578401	39752384400	37473670380
2598107	1909613	4.27842E+11	2.85157E+11	3.38239E+11

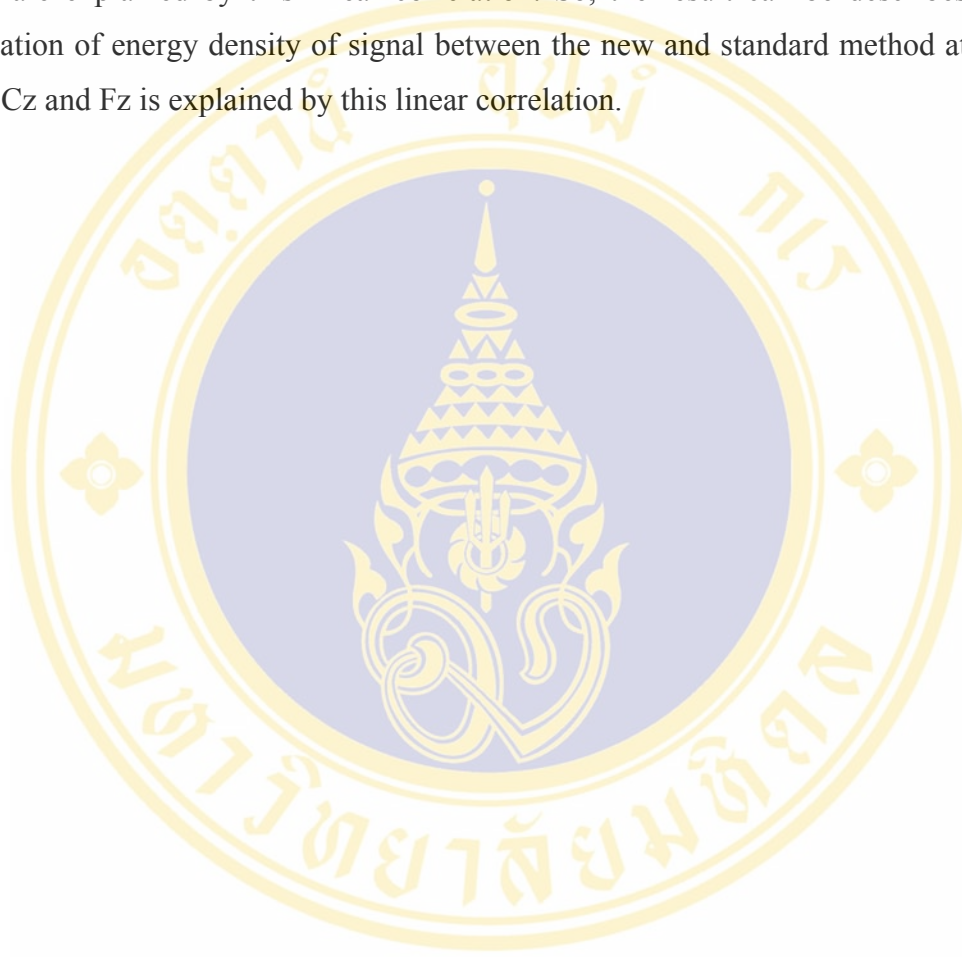
Table 4.4 Correlation in energy density of signal at Fz location

From the table 4.3 and 4.4, we can calculate r^2 from Cz and Fz

r^2 from Cz = 0.929

r^2 from Fz = 0.935

This means that 92.9 % of variable value in Cz and 93.5 % of variable value in Fz are explained by this linear correlation. So, the result can be describes that the relation of energy density of signal between the new and standard method at location of Cz and Fz is explained by this linear correlation.



CHAPTER V

DISCUSSION AND CONCLUSION

This chapter represents the conclusion and discussion of analyzing ERPs signal related to paired t test and energy density of signal.

Objective 1

To analyze the differences between amplitude and latency elicited by new visual stimuli and standard method in using paired t test. Are the two small sets of quantitative data related?

Discussion

According to the theory, the ERPs signal of each subject represents as his or her characteristic memory. Although, the visual stimuli are different, however, the ERPs signal should be the same. This means that the main prominent peaks of ERPs signal should appear equal. In this thesis, the new visual stimuli are presented in portrait pictures (human including man and woman for target picture) and scenic pictures (including tree, mountain, river, etc. for non target picture). In the experiment with 30 subjects, it can be found that the main peaks of ERPs signal have the same appearing as the standard picture. However, the new visual stimuli takes longer time to peak (P100 and P300 are seen clearly) when it is compared to the standard method.

The reason for the difference is that it may be the new visual stimuli presented with the color picture, but the standard picture presents with black & white color. However, the idea supporting this hypothesis is the next objective.

After the test, both of the results from 30 subjects are collected in pairs, paired t test was used to test the hypothesis. If there is no difference between the two sets of data, then this means that the differences should be zero. The result shows that there are no differences in amplitude and latency from new pictures and standard pictures.

Conclusion

According to the hypothesis, there were no significant differences between new visual stimuli and standard method. The results pointed out that the new visual stimulus is as a standardized process as is from the standard method.

Objective 2

To examine the total energy density of brain signal with Parseval's relation.
Are the new visual stimuli and standard method related?

Discussion

From the above objective, it is thought that the new visual stimuli make the brain consume more energy. The evidence was that graph from new visual stimuli shows longer time to peak. Although, taking longer time is proved to be due to electronic circuits, but it is only a little effect.

In addition, some authors recommended that the object recognition constitute between lower and higher levels of image analysis. The low-leveled analysis is an important factor to identify an object. If the object is more complicated in edges, contours, surface curvature etc., the low-leveled analysis must work harder. Consequently, the time used to identify an image is prolonged.

The result showed that the energy density of signal from the new visual stimuli is higher than that of the standard method, when they both are compared to each other. Although, results from 3 subjects 4, 20 and 30 do not support to this idea clearly, results from 2 subjects 4 and 20 nearly supported this, (Cz, but not Fz supported).

From the test, by means of the correlation in order to check for the direct line of energy density of brain signal. The calculated figures showed the relation between the old method and the new one in the position of Cz and Fz, which both have the direct line highly to 93% and 94%. So this means energy density of brain signal of Cz and Fz have the relations which are the same.

Conclusion

Considering those result from three subjects which represented some different details. The subjects sometimes felt bored during their tasks, or they themselves might

pay less concentration to the test. However, their result inclined to the theory of the color picture (high detail) which suggested that the brain's memory consumed high energy.



Future Works

- Test the new visual stimuli with brain's disease patients and compare result with standard visual stimuli in the same patients.
- Design picture for specific brain's disease patient. Because different of brain's disease need different visual stimuli for activate.



REFERENCES

1. Donchin E. Event-related potentials: a tool in the study of human in the study of human information processing. In: Evoked potentials and behavior, Vol2. Edited by Begleiter H. New York: Plenum Press, 1979: 13-88.
2. Hillyard SA. Cognitive functions and event-related brain potentials. In: Evoked Potential II The Second International Evoked Potentials Symposium. Edited by Nodar RH, Barber C. London: Butterworth, 1982: 52-59.
3. Event -Related Brain Potentials (ERPs).
<http://www.technion.ac.il/~mnirit/erp.html>.
4. Picton TW, Stuss DT, Champagne SC, Nelson RF. The effects of age on human event-related potentials. *Psychophysiology* 1984; 21: 312-25.
5. Sanquit T, Rohrbaugh J, Syndulko K, Lindsley D. Electrocortical signs of levels of processing: perceptual analysis and recognition memory. *Psychophysiology* 1980; 17: 568-576.
6. Johnson R, Pfefferbaum A, Kopell B. P300 and long-term memory: latency predicts recognition performance. *Psychophysiology* 1985; 22: 497-507.
7. Visual Pathway.
<http://thalamus.wustl.edu/course/basvis.html>.
8. Kandel E, Schwartz J, Jessell T. Principles of neural science. Fourth Edition 2000, Chapter 26: Visual Processing by the retina; 507-22.
9. Kandel E, Schwartz J, Jessell T. Principles of neural science. Fourth Edition 2000, Chapter 27: Central Visual Pathway; 524-47.

10. Basic Visual Pathway.

<http://www.hhmi.org/senses/b150.html>.

<http://thalamus.wustl.edu/course/basvis.html>.

11. Memory. Encyclopedia Americana.

<http://ea.grolier.com/cgi-bin/build-page?artbaseid=0266760>.

12. The Brain Mechanisms of Memory.

<http://www.epub.org.br/cm/n01/memo/mechanisms.htm>

13. Michael J. Zigmond, Floyd E. Bloom, Story C. Landis, James L. Roberts, Larry R. Squire. Chapter 56: Learning and Memory: System Analysis. *Fundamental Neuroscience*; 1999: 1455-83.

14. Psychophysiological measures.

<http://chem.larc.nasa.gov/HumanFactors/physiomeasures.htm>

15. Monica Fabiani, Gabriele Gratton, Michale G.H. Cloes. Event-related brain potentials; methods, theory, and applications. *Handbook of psychophysiology*, 2nd ed. 2000; 53-76.

16. Axel Mecklinger, D. Yves von Cramon and Gabi Matthes-von Cramon. Event-related potential evidence for a specific recognition memory deficit in adult survivors of cerebral hypoxia. *Brain* 1998; 121: 1919-35.

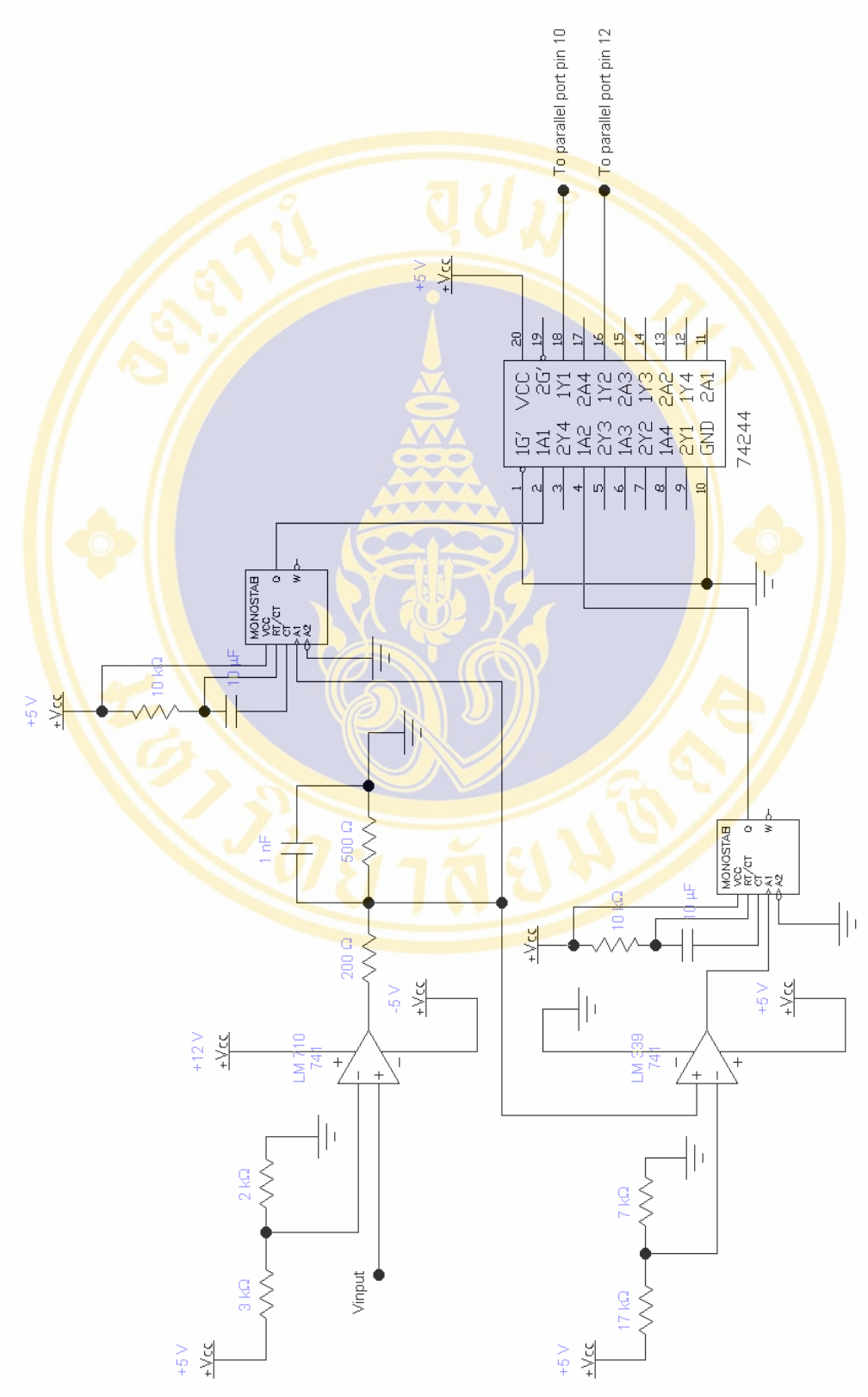
17. K. Hirata, A. Hozumi, H. Tanaka, J. Kubo, X.H. Zeng, K. Yamazaki, K. Asahi, T. Nakano. Abnormal information processing in dementia of Alzheimer type; A study using the Event-related potential's field. *Eur Arch Psychiatry Clin Neurosci* 2000; 250: 152-55.

18. P. A. Williams, G. H. Jones, M. briscor, R. Thomas and P. Cronin. P300 and Reaction-Time measures in Senile Dementia of the Alzheimer Type. *British Journal of Psychiatry* 1991; 159: 410-14.
19. Lihong Wang, Yoshiyuki Kuroiwa, Toshiaki Kamitani, Tatsuya Takahashi, Yume Suzuki, Osamu Hasegawa. Effect of interstimulus interval on visual P300 in Parkinson's disease. *J. Neurol Neurosurg Psychiatry* 1999; 67; 497-503.
20. Ryuji Neshige, Geoff Barrett, Hiroshi Shibasaki. Auditory long latency event-related potentials in Alzheimer's disease and multi-infarct dementia. *Journal of Neurology, Neurosurgery, and Psychiatry* 1988; 51: 1120-25.
21. John Polich. P300 clinical utility and control of variability. *Journal of Clinical Neurophysiology* 1998; 15 (1): 14-33.
22. William W. Hines, and Douglas C. Montgomery. Test of Hypotheses. *Probability and Statistics in Engineering and Management Science*; Third edition: 288-313
23. Tadanori Fukami, Tsuyoshi Shiina, Fumito Ishikawa, Yoichi Saito. Relation between auditory stimulus intensity and response of P300 with eyes-opened and eyes-close state. *Proceedings of the 20th annual international conference of the IEEE Engineering in Medicine and Biology Society*, 1998; Vol.20, No4: 1996-99.
24. Jeffrey S. Jonson, and Bruno A. Olshausen. Timecourse of neural signatures of objects recognition. *Journal of Vision*, 2003; vol 3; 499-512

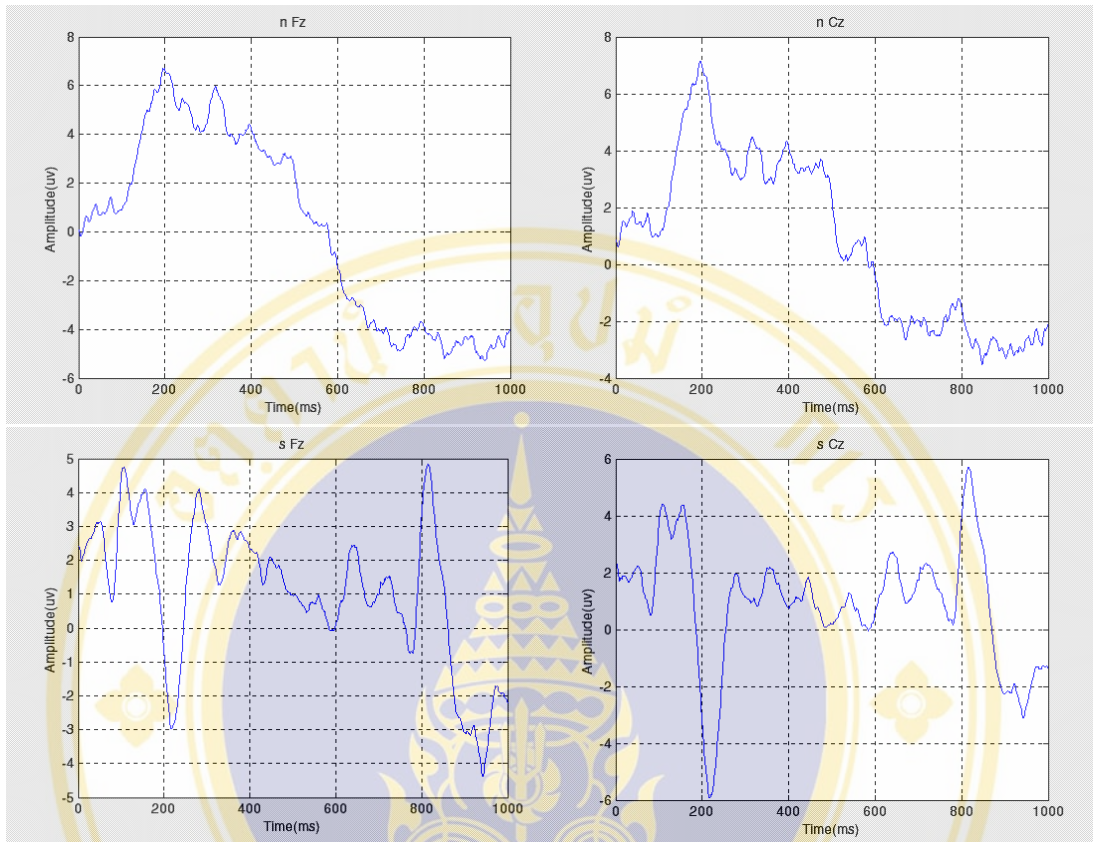
APPENDIX



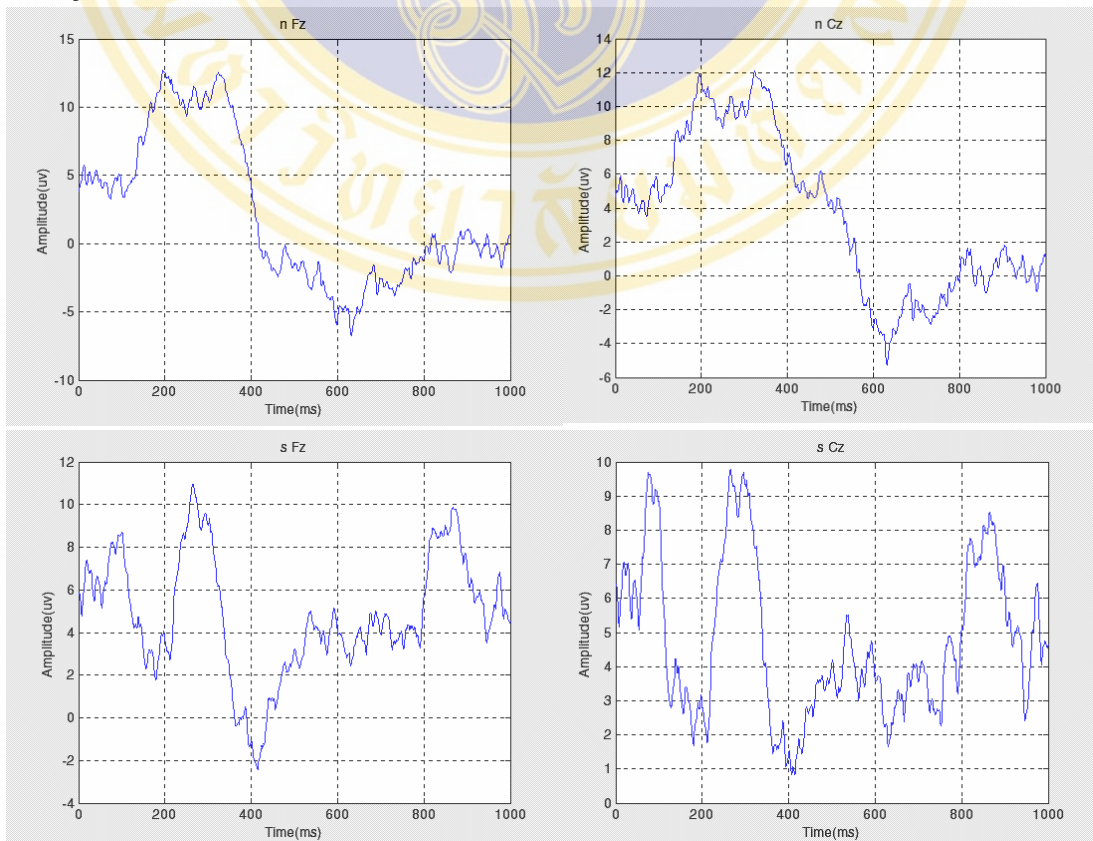
APPENDIX A: schematic diagram of circuit



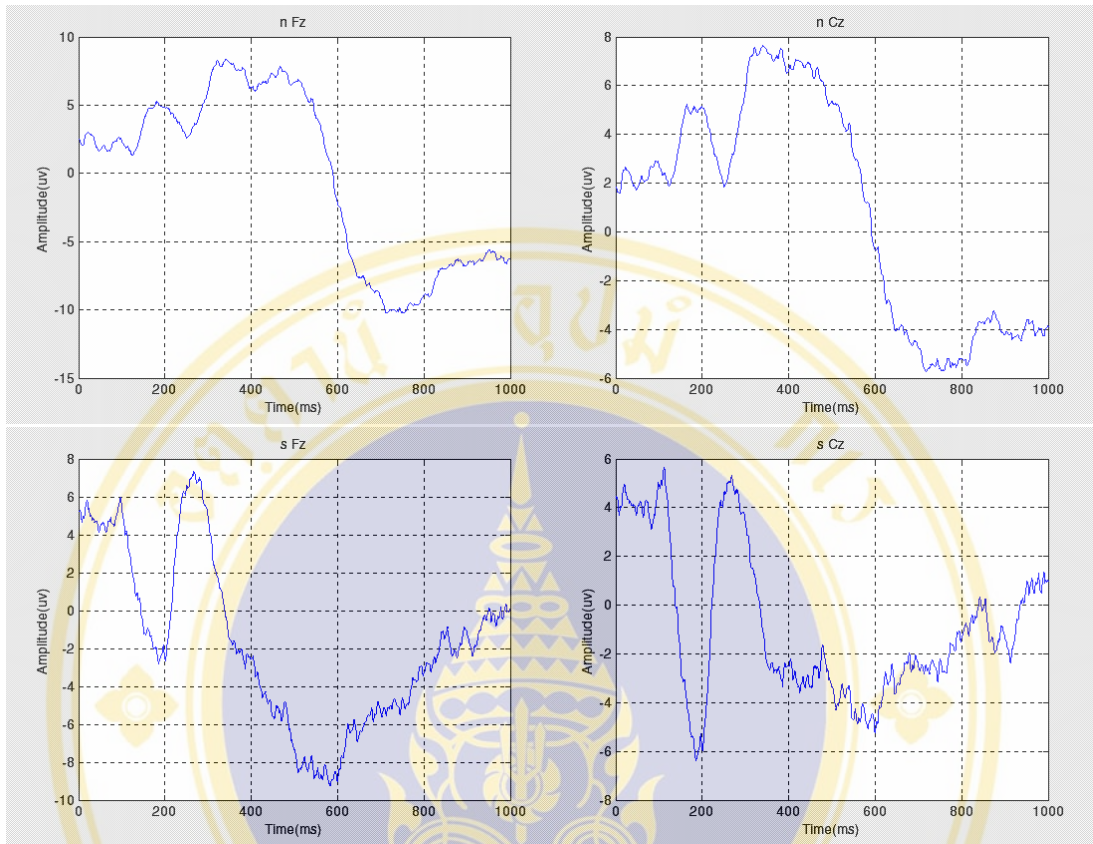
APPENDIX B: Subject's ERPs between new picture and standard picture
Subject 1st



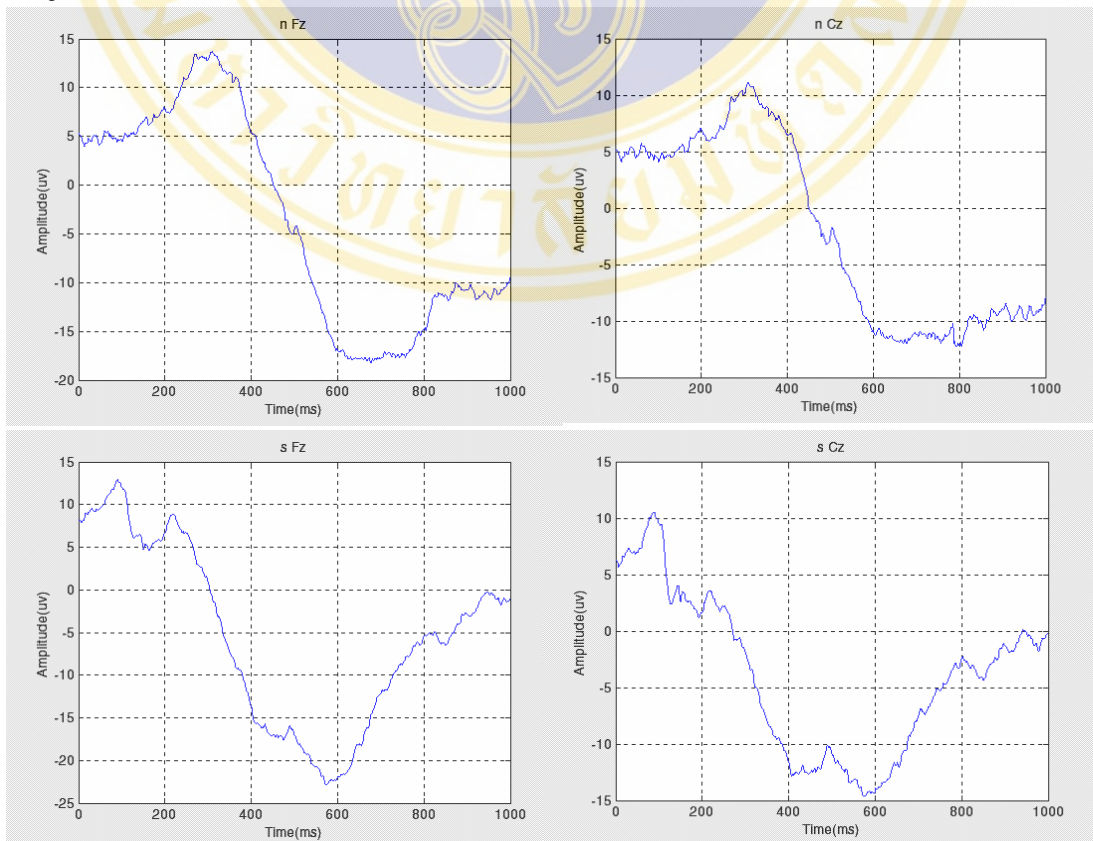
Subject 2nd



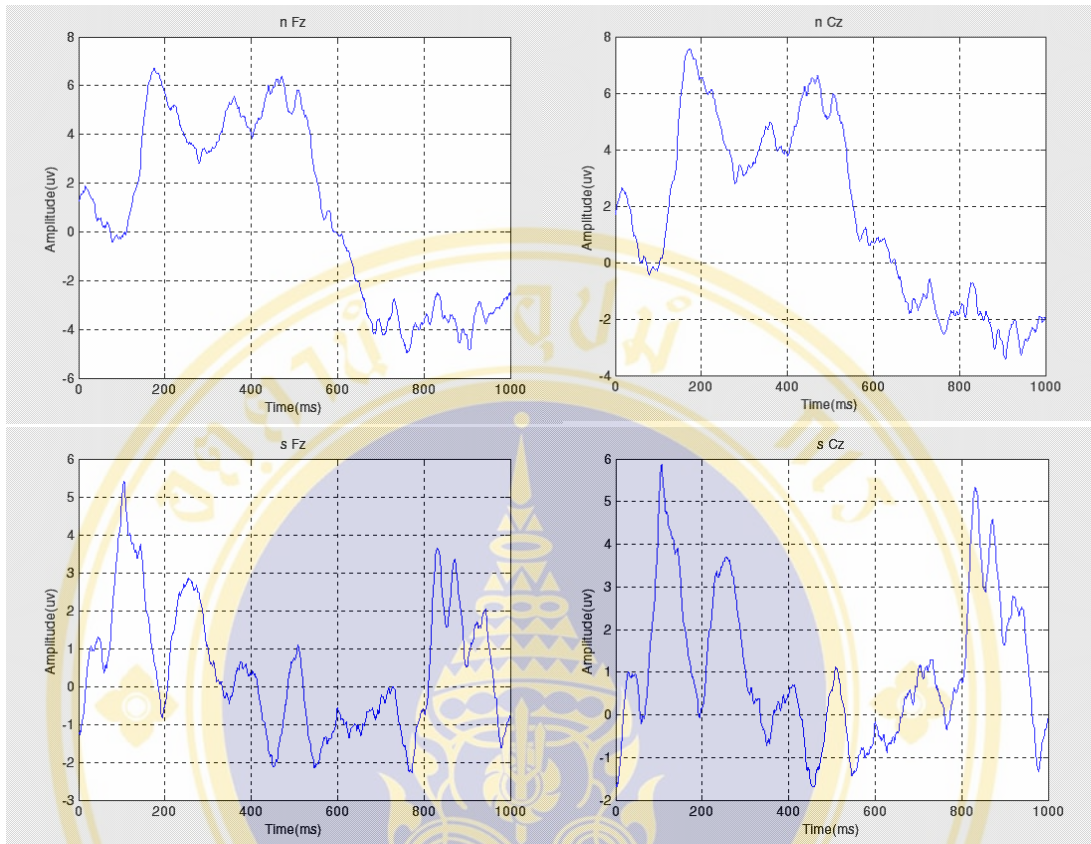
Subject 3rd



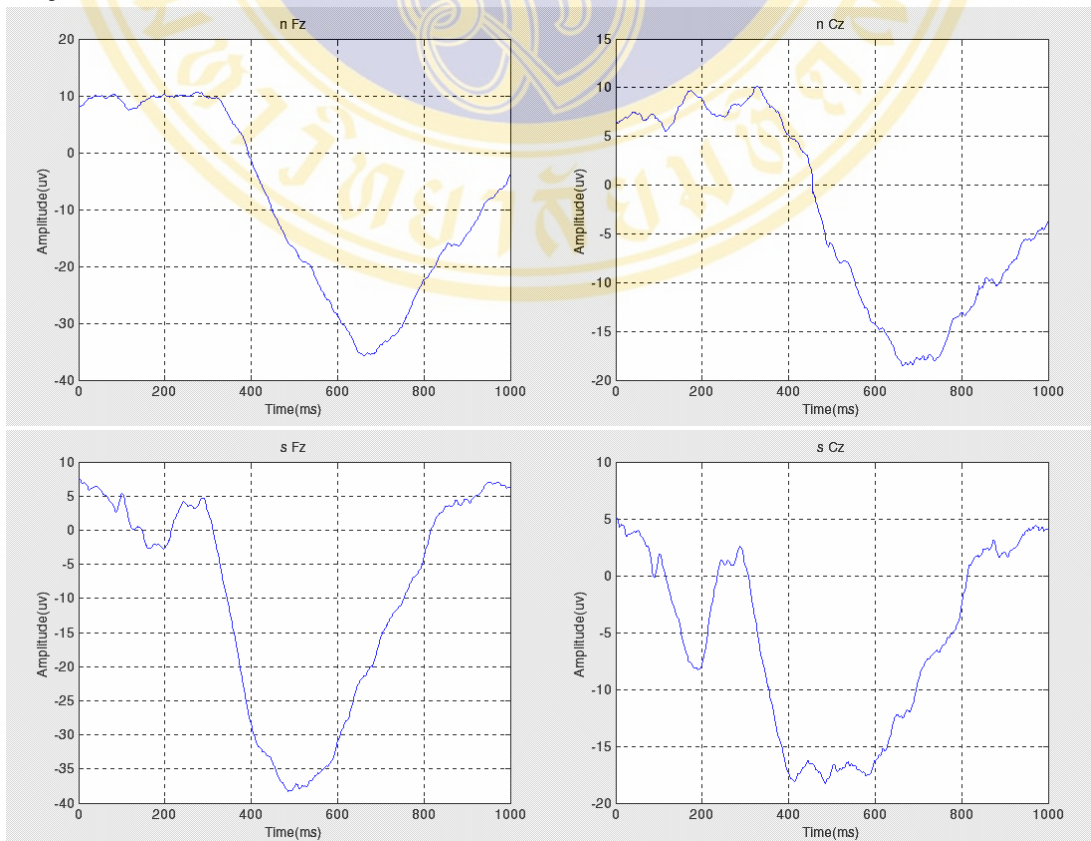
Subject 4th



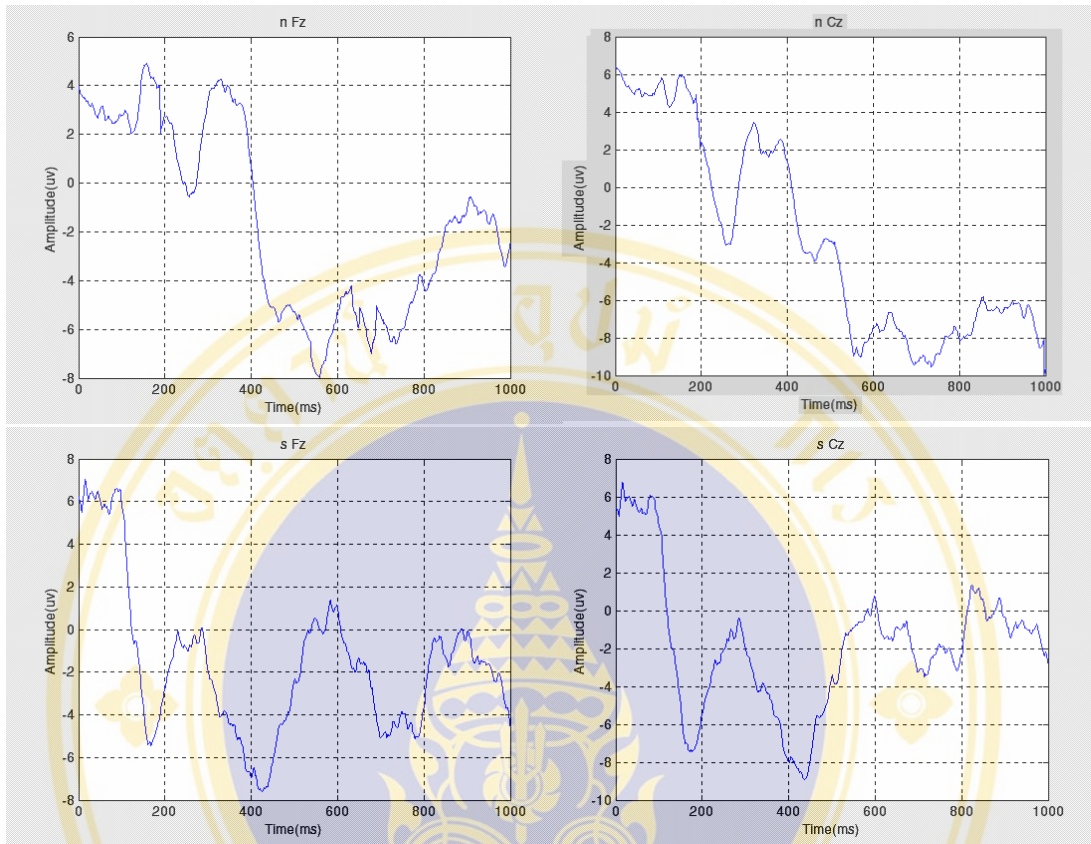
Subject 5th



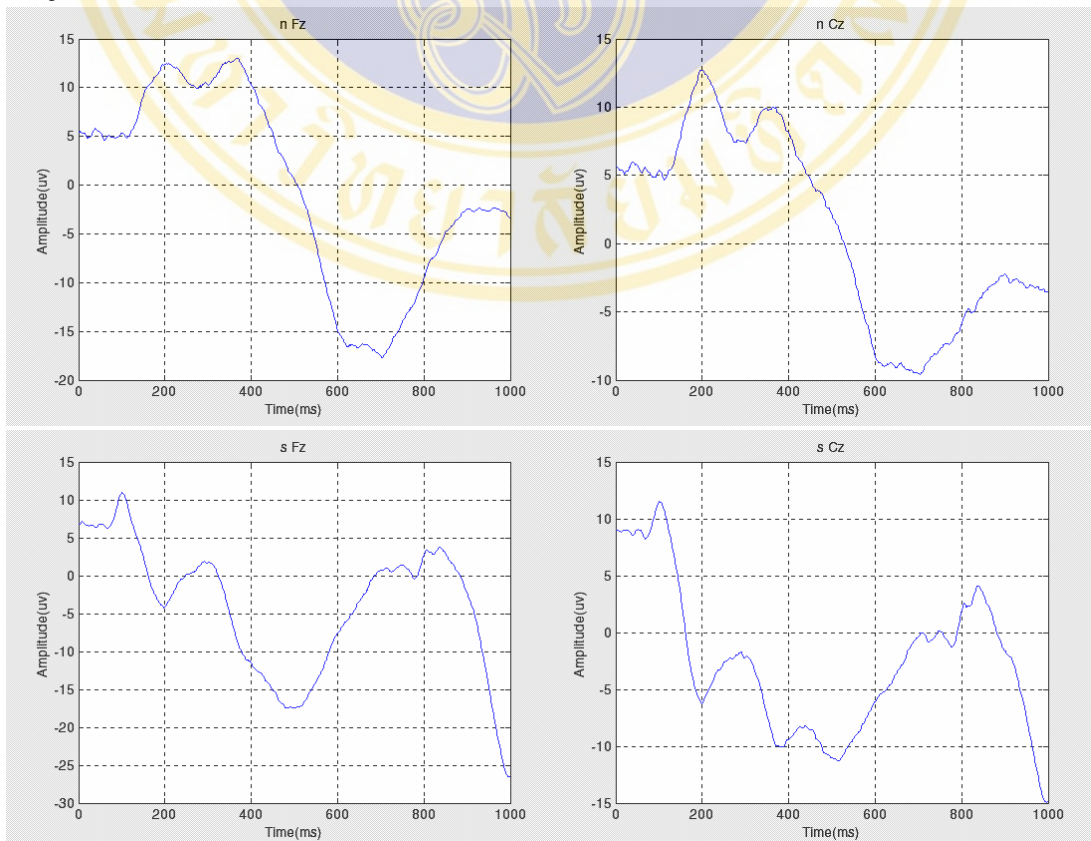
Subject 6th



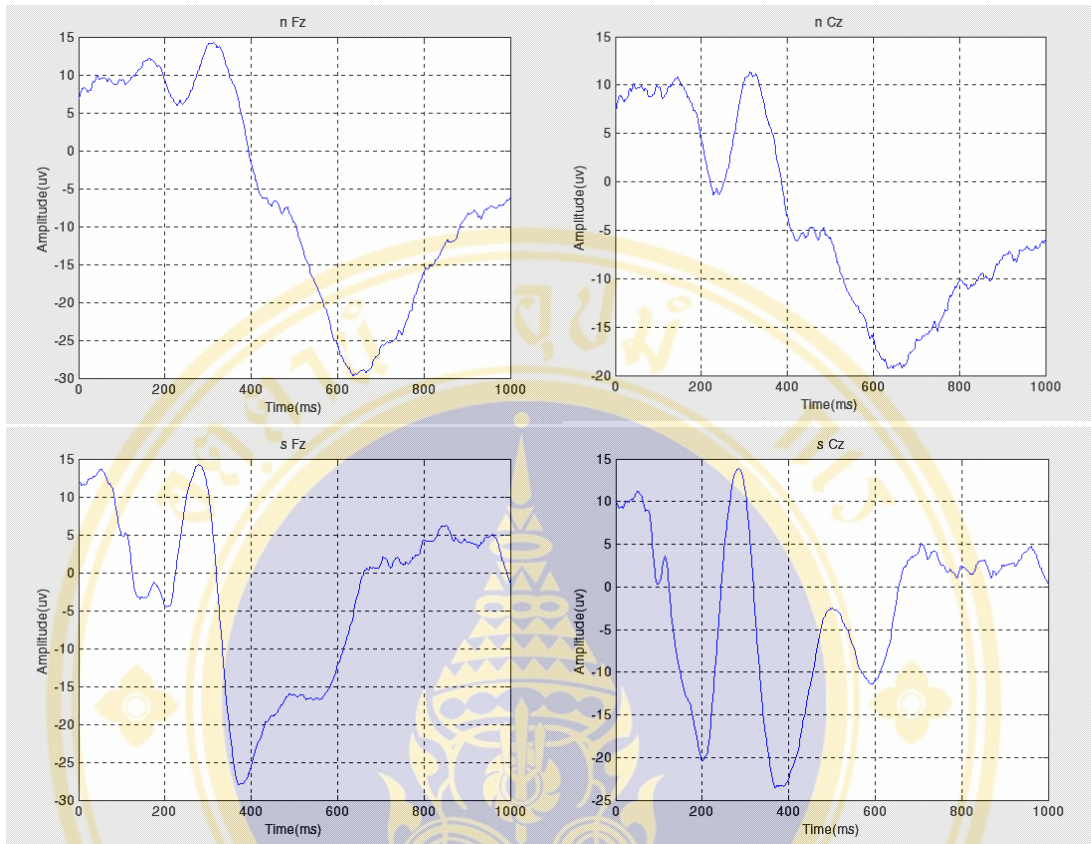
Subject 7th



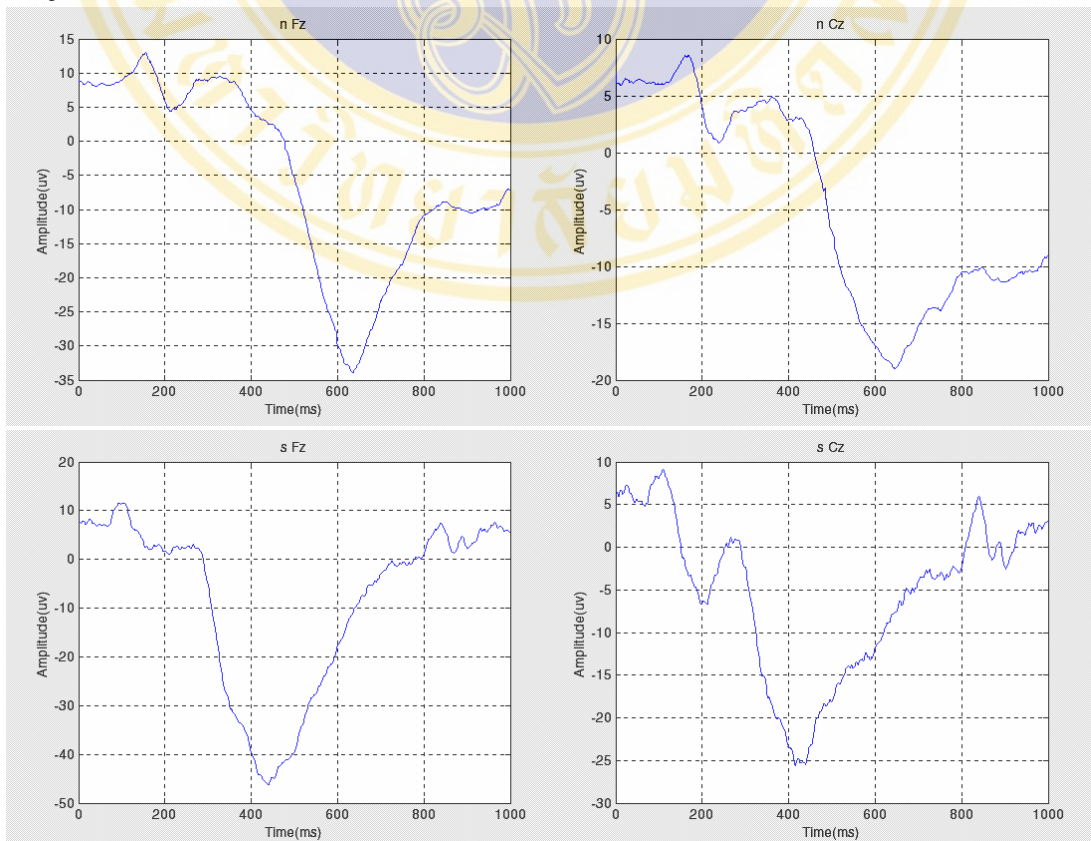
Subject 8th



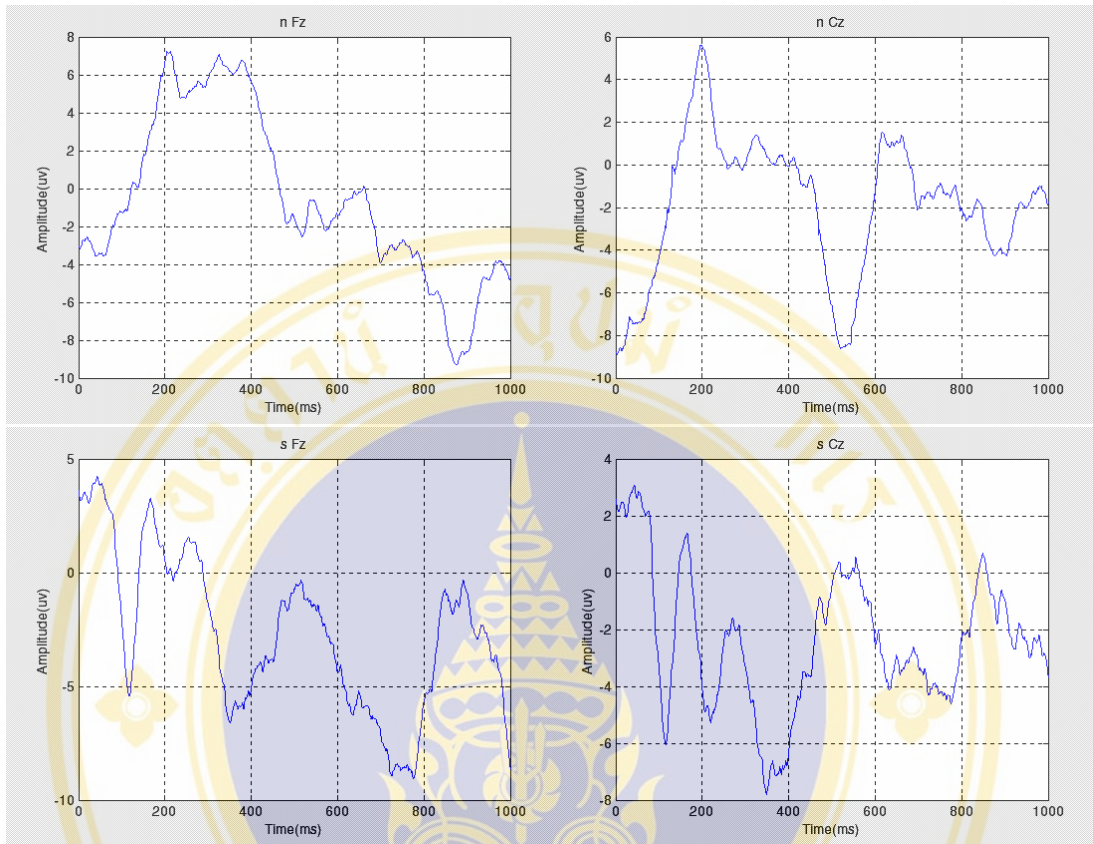
Subject 9th



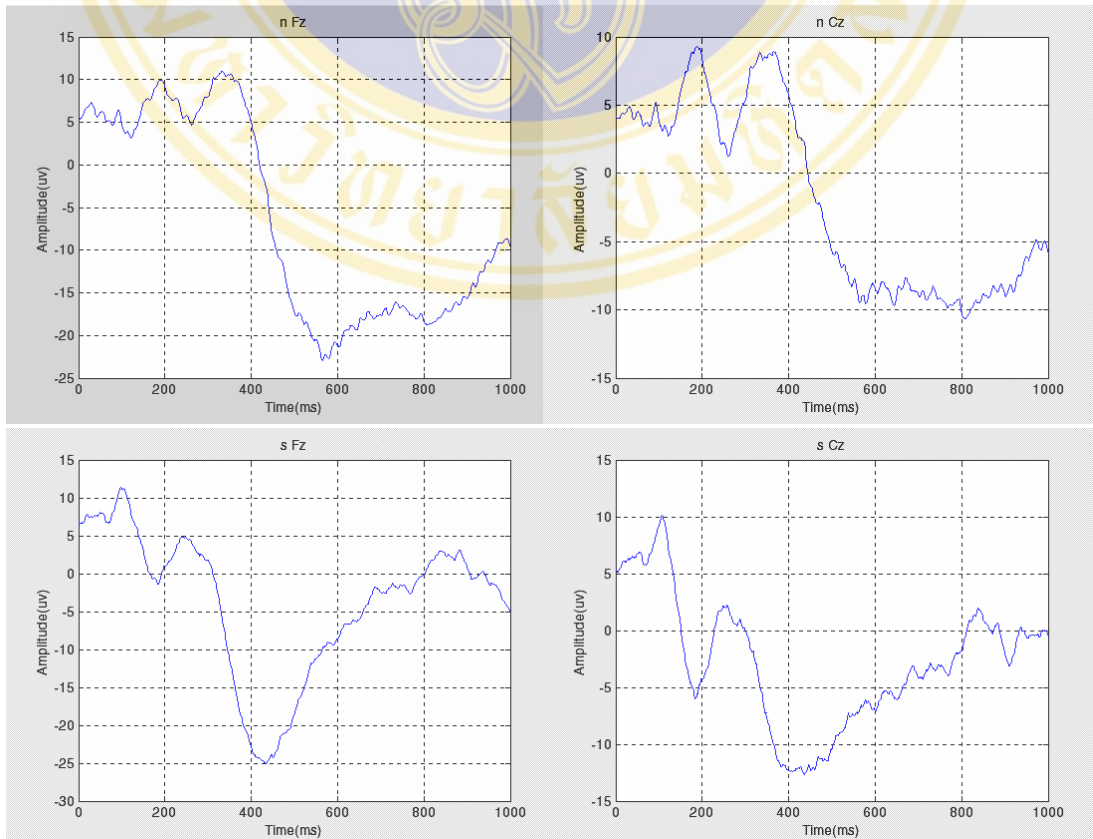
Subject 10th



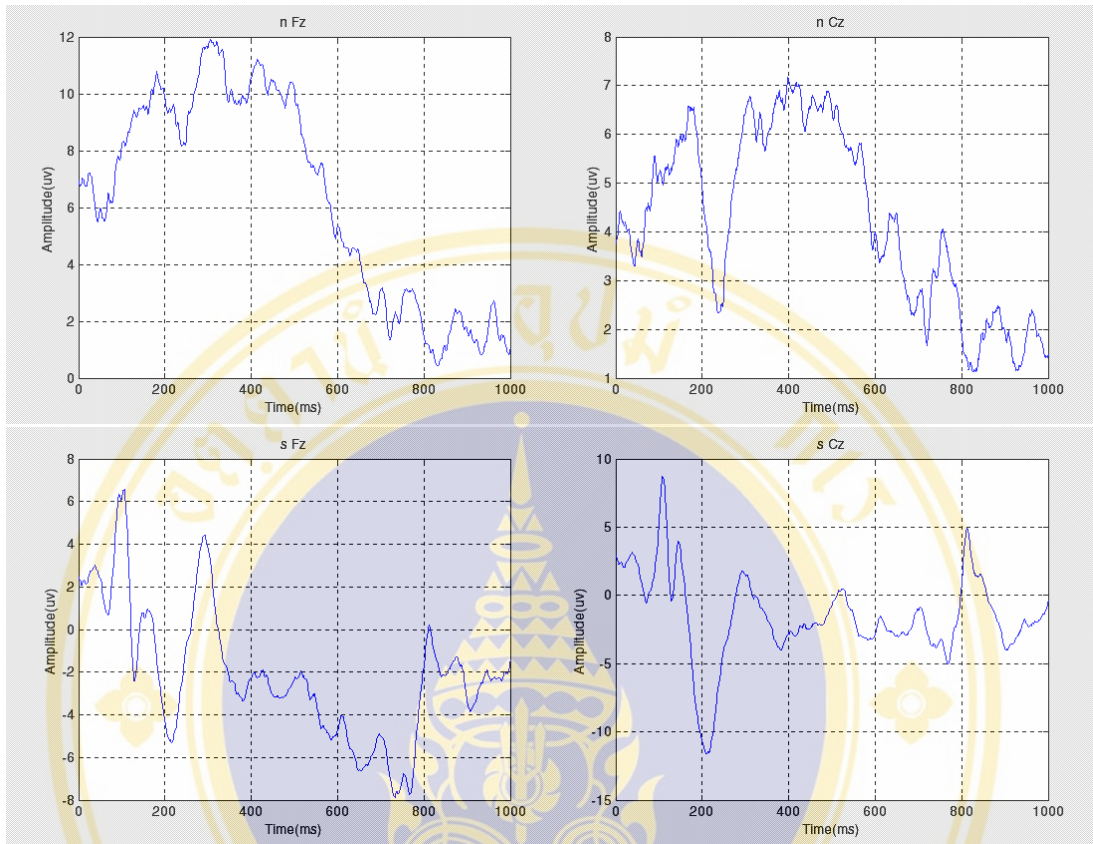
Subject 11th



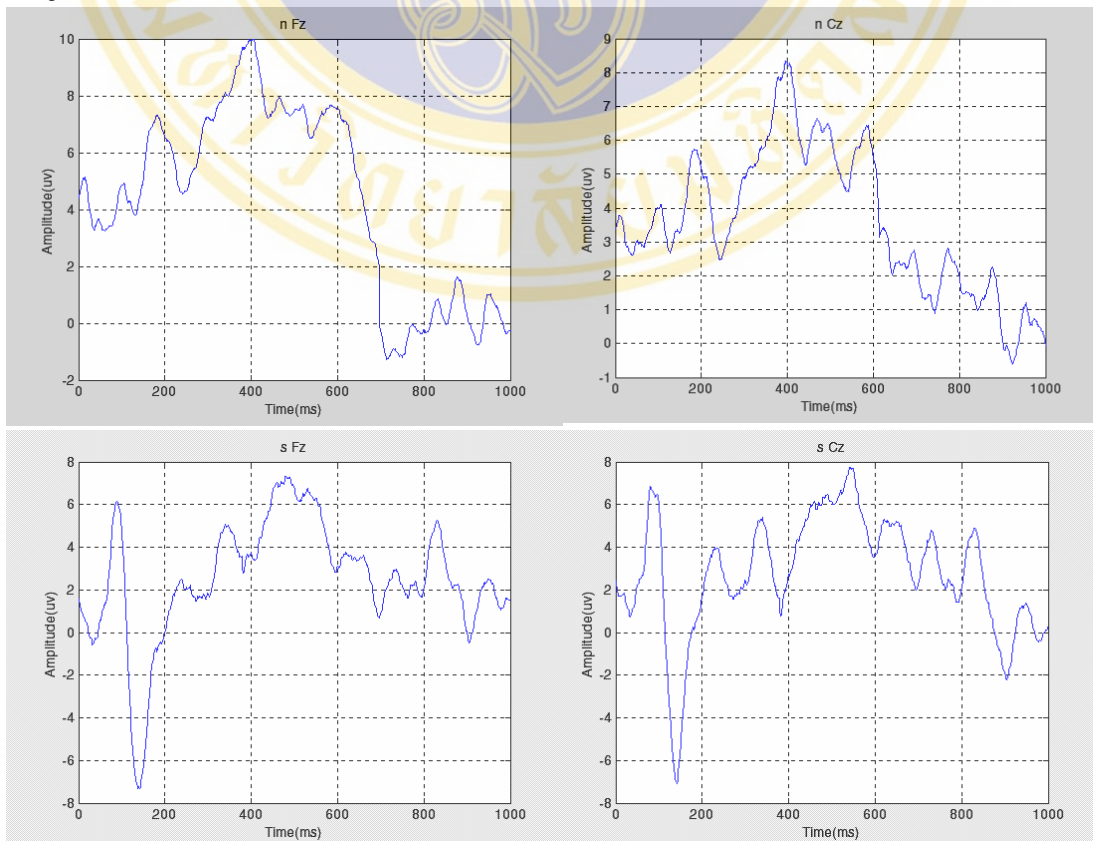
Subject 12th



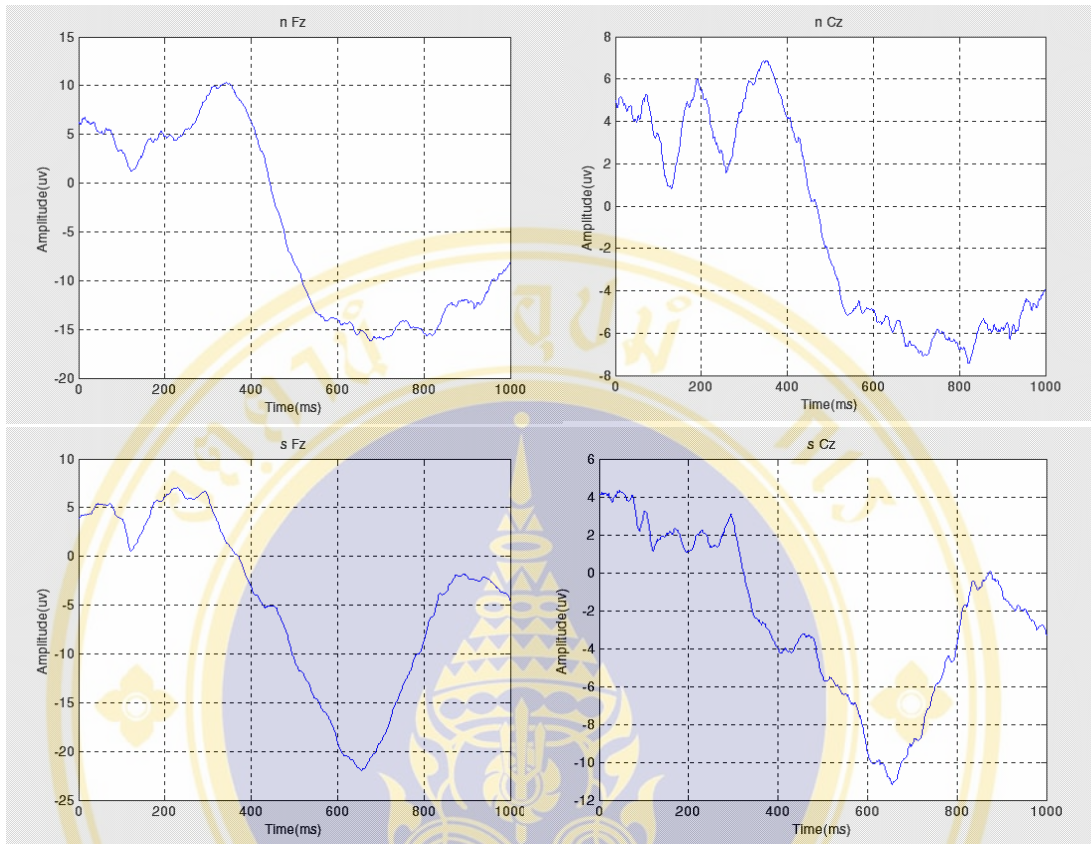
Subject 13th



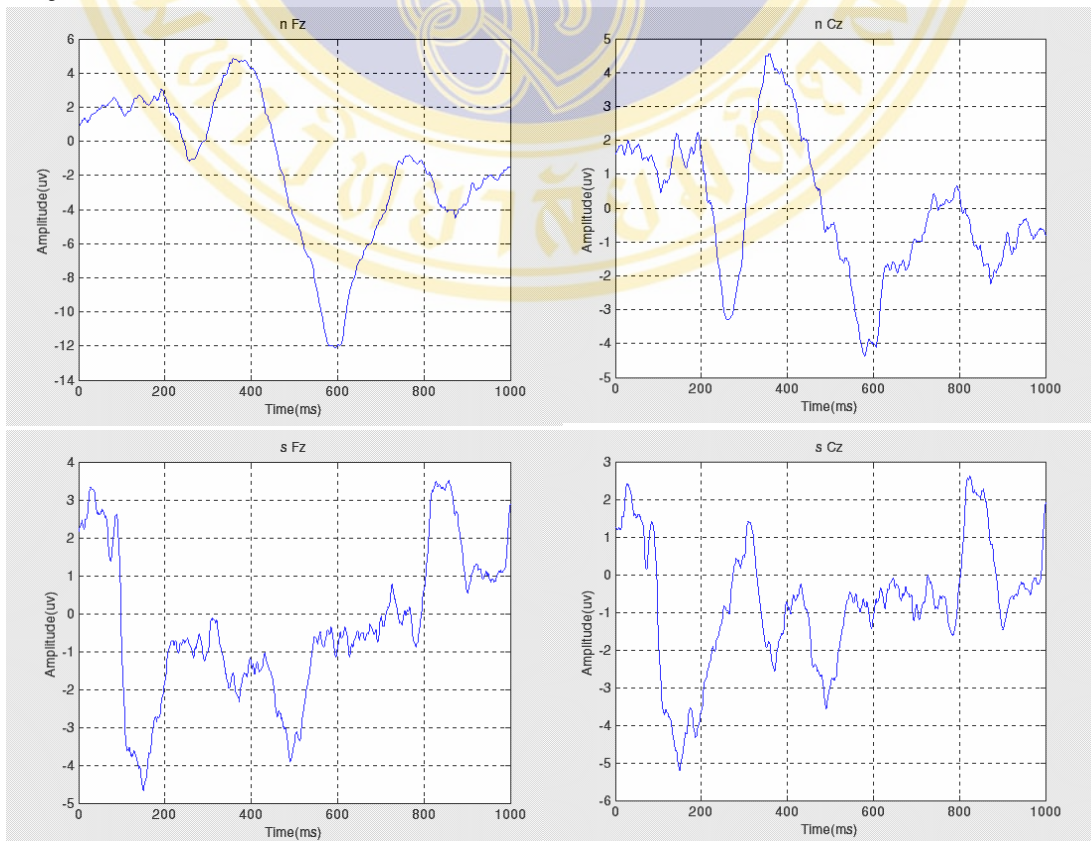
Subject 14th



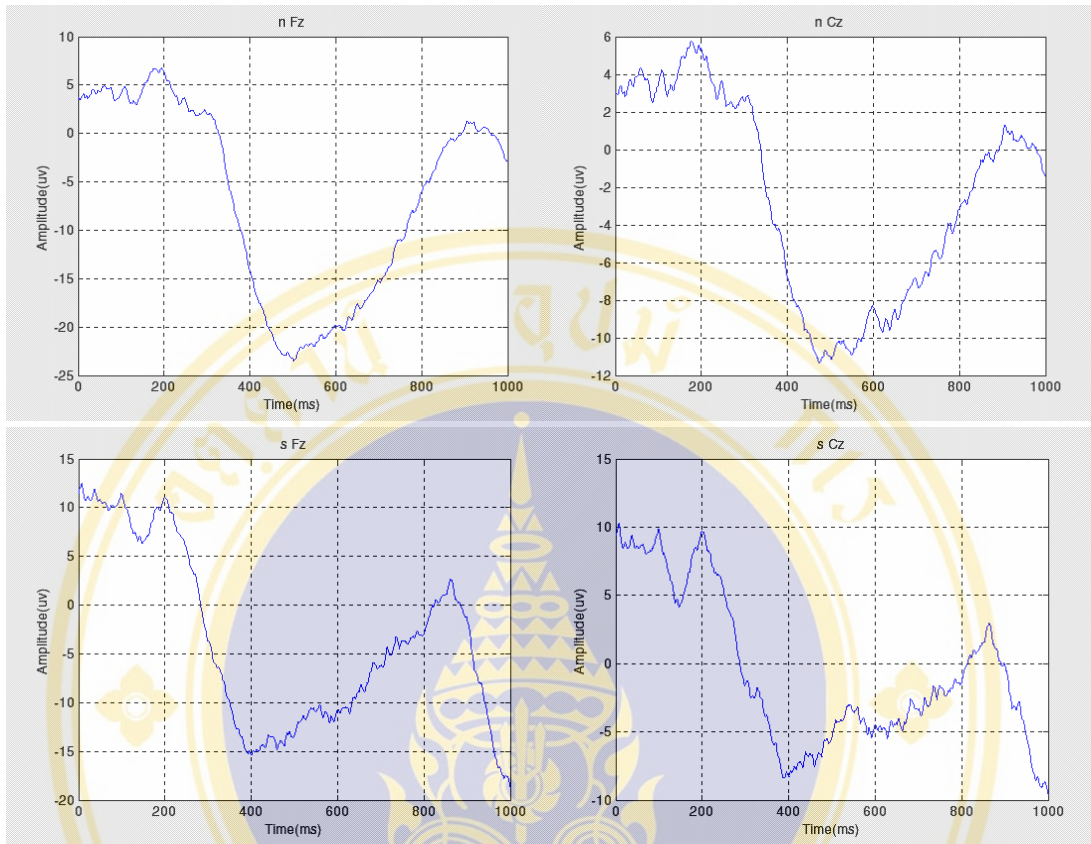
Subject 15th



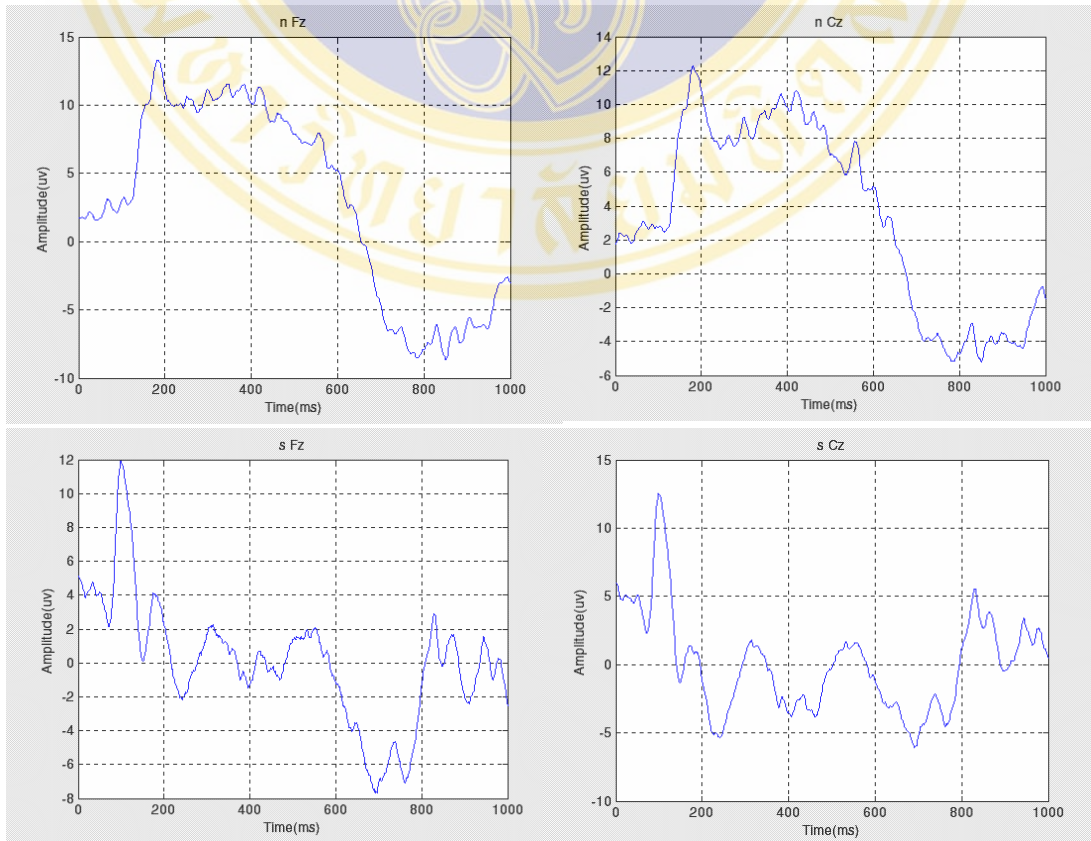
Subject 16th



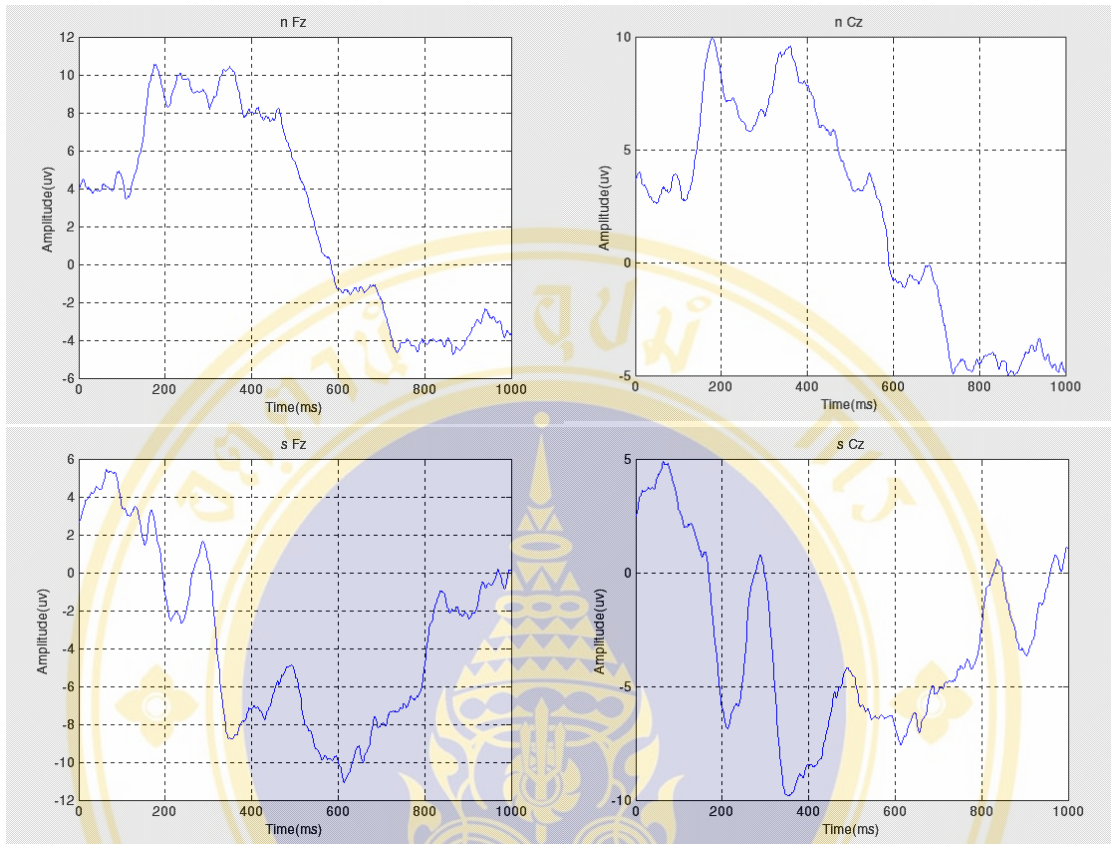
Subject 17th



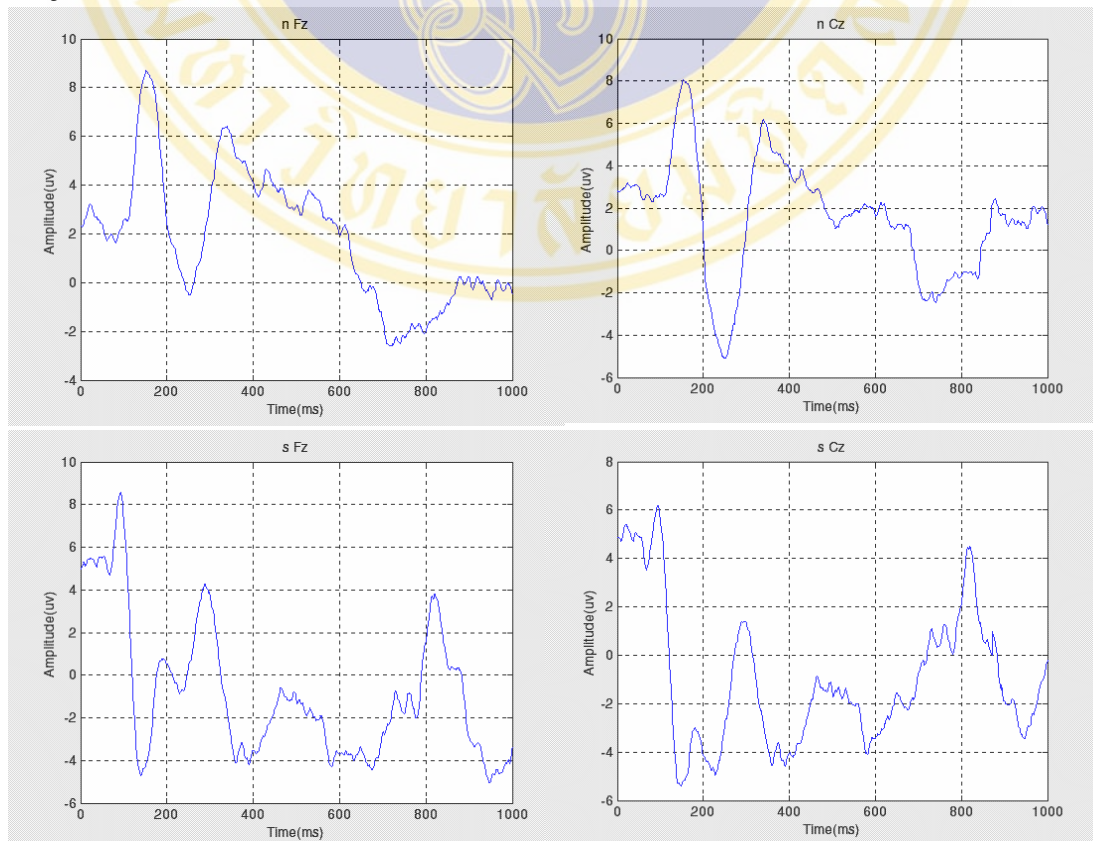
Subject 18th



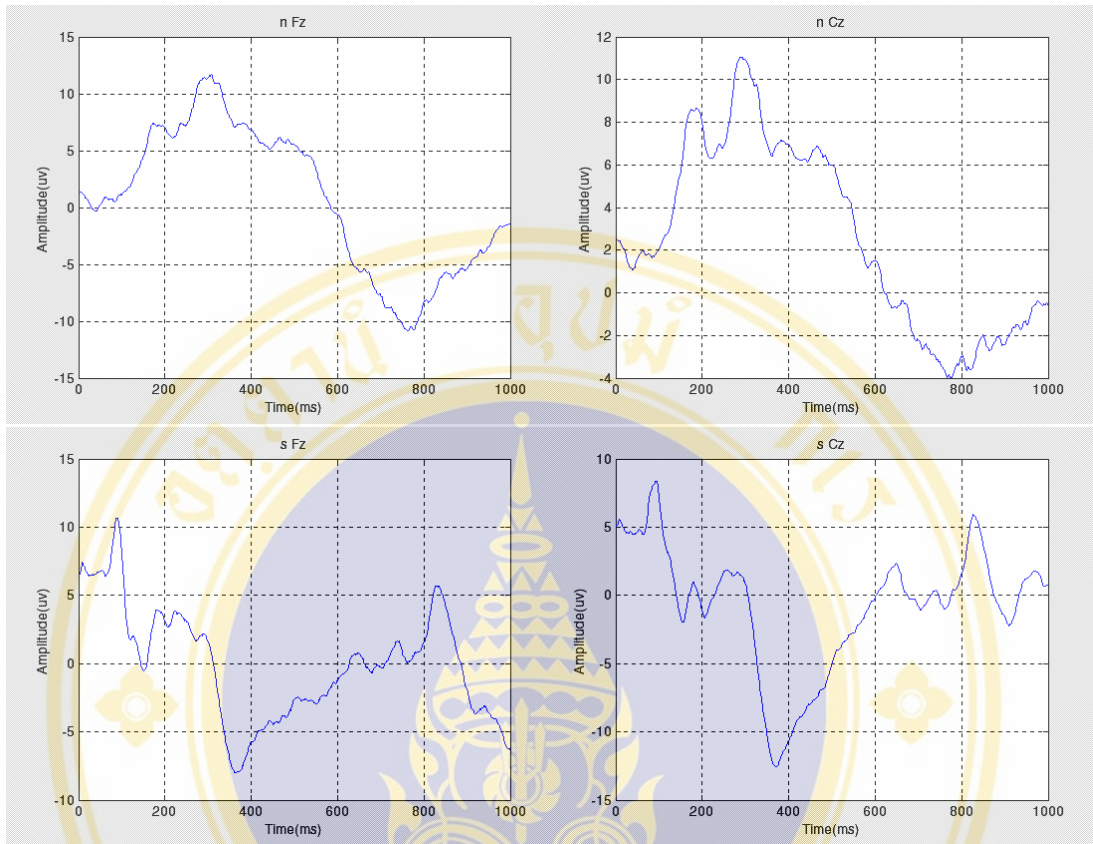
Subject 19th



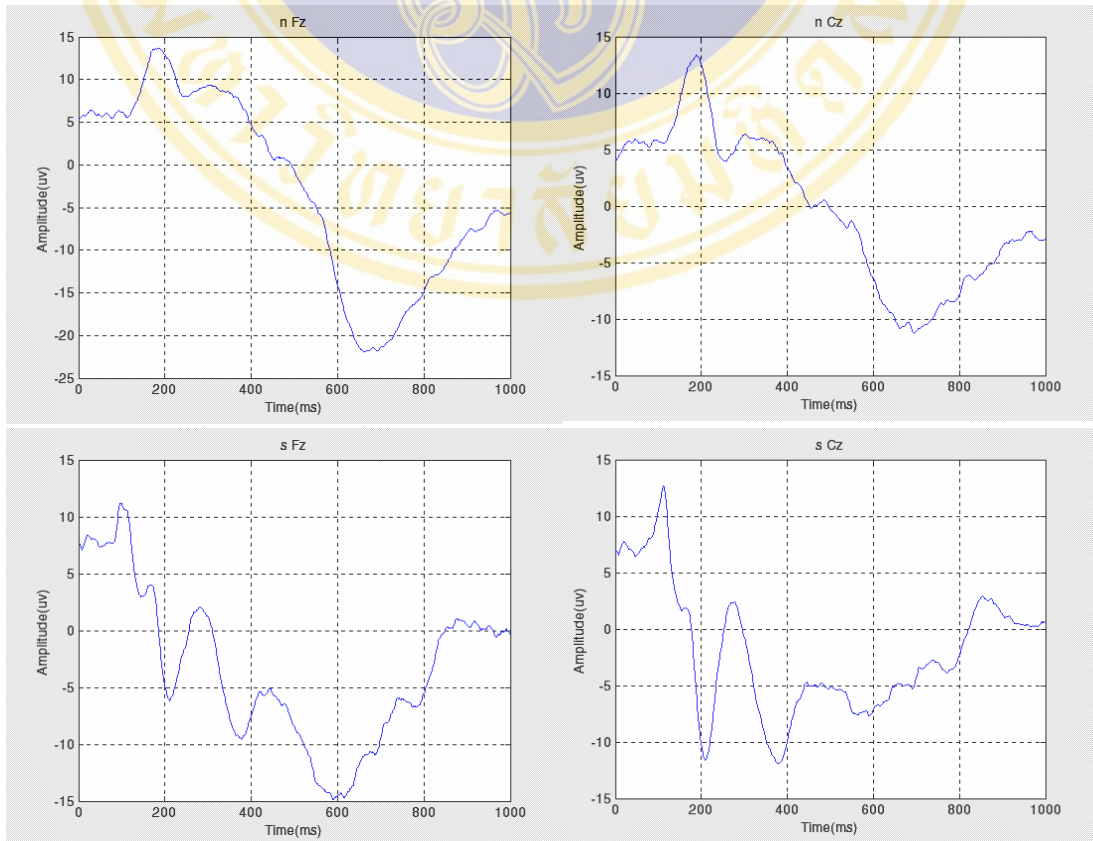
Subject 20th



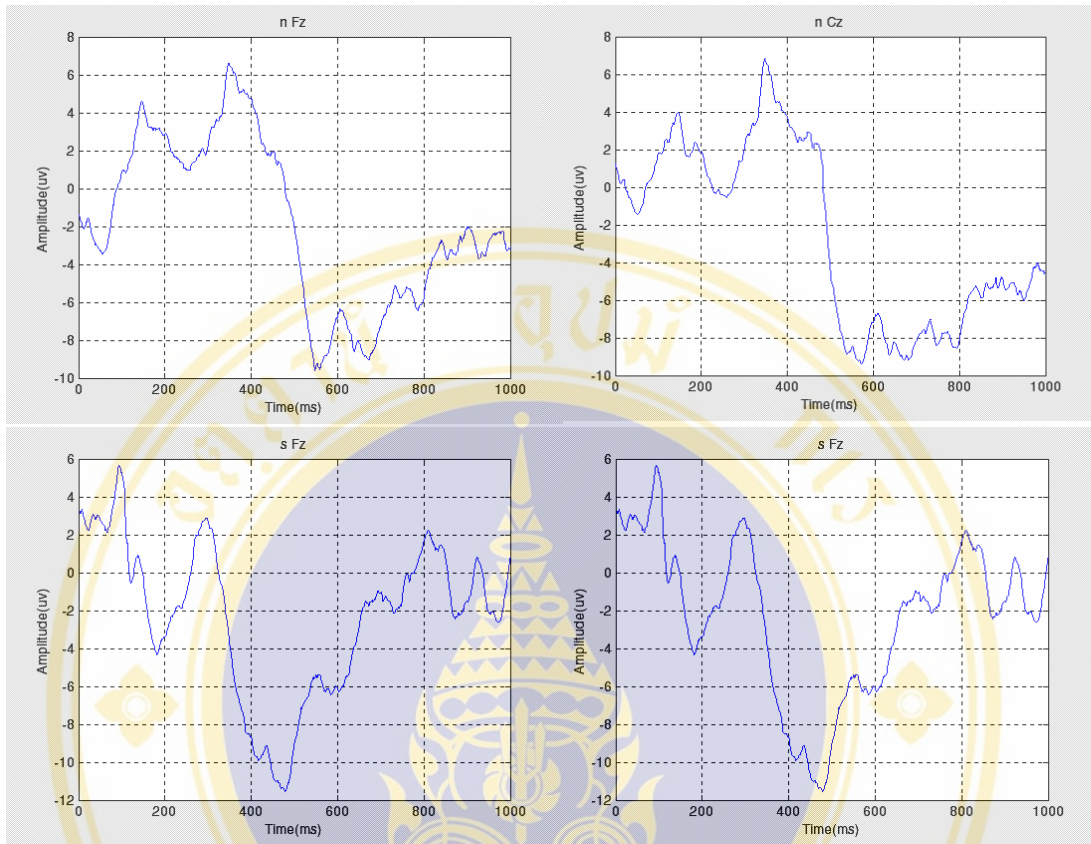
Subject 21st



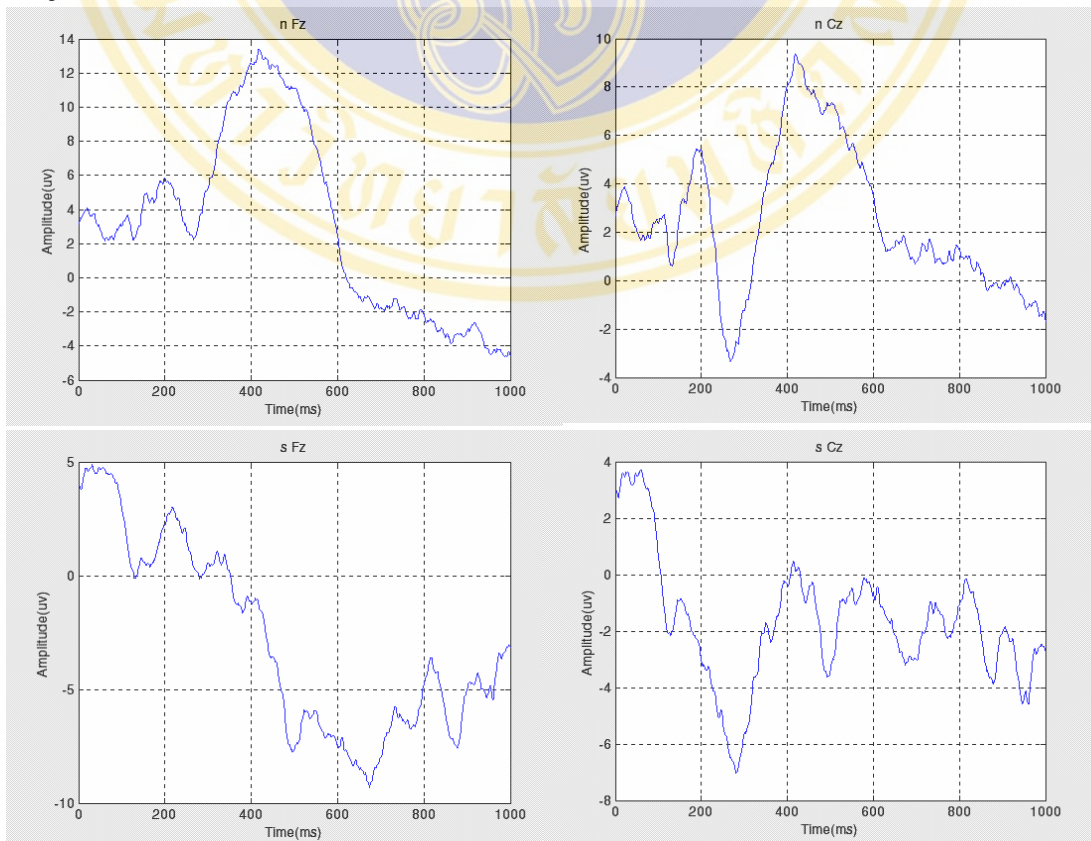
Subject 22nd



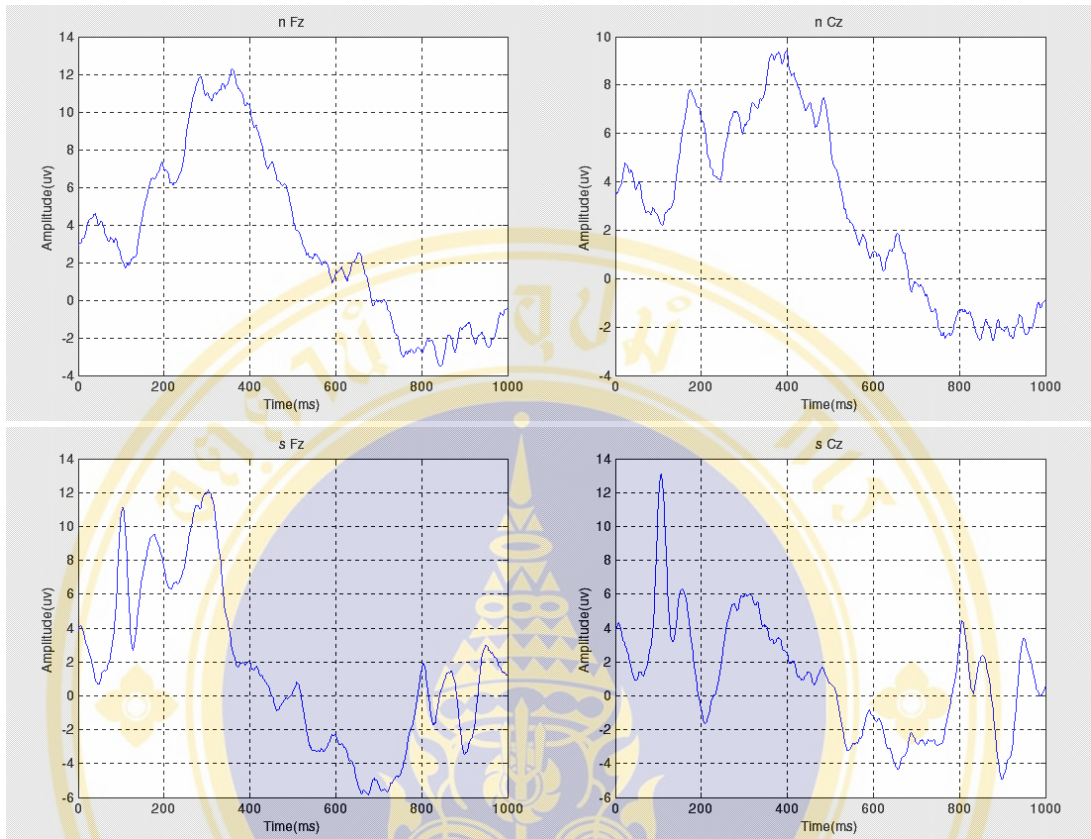
Subject 23rd



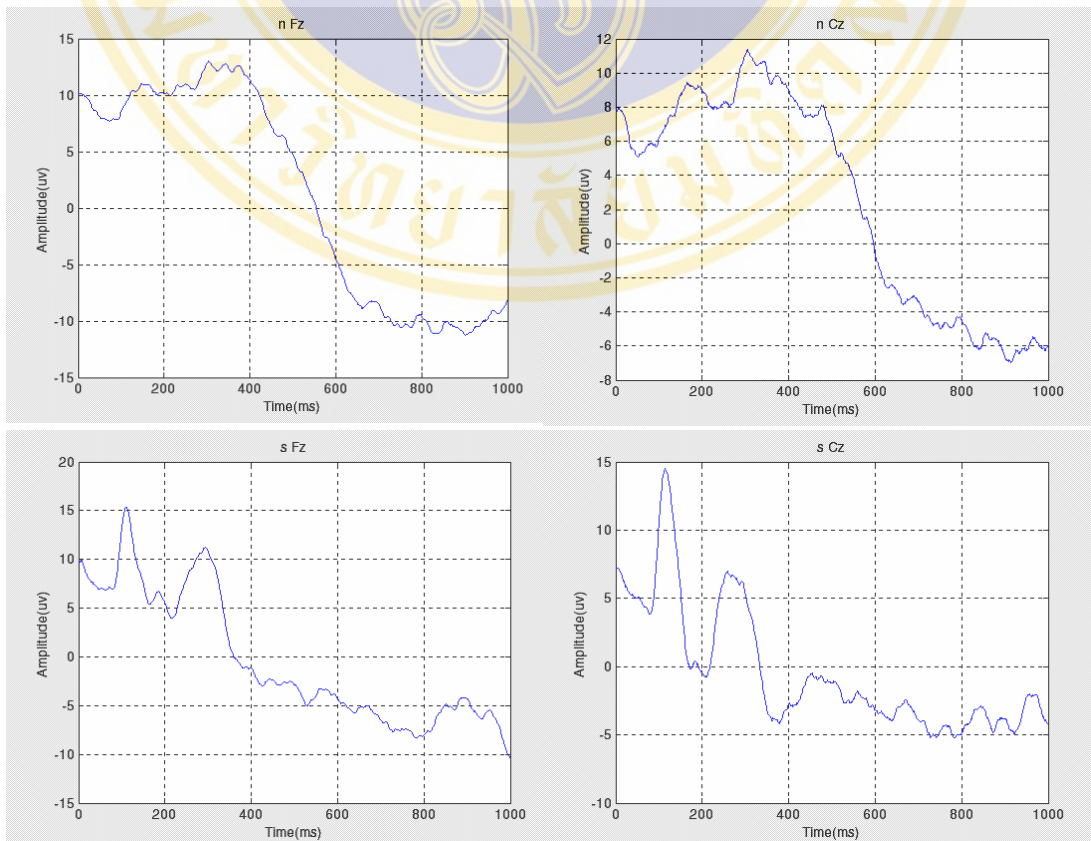
Subject 24th



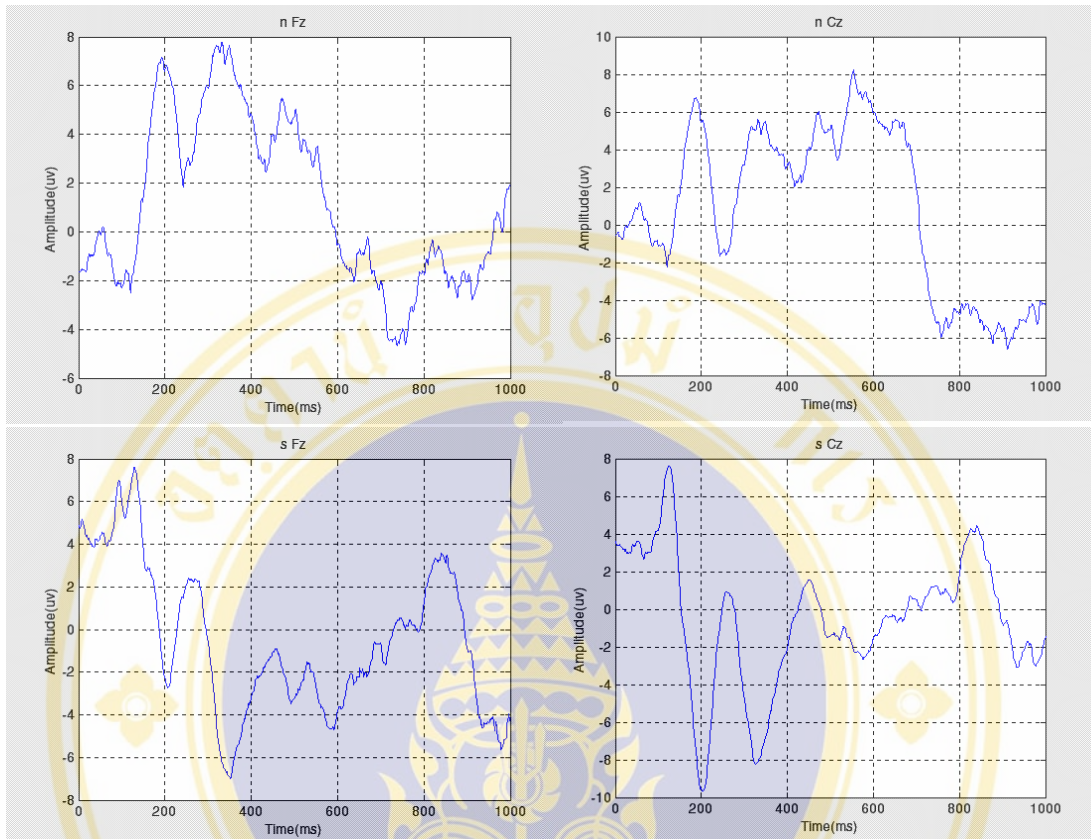
Subject 25th



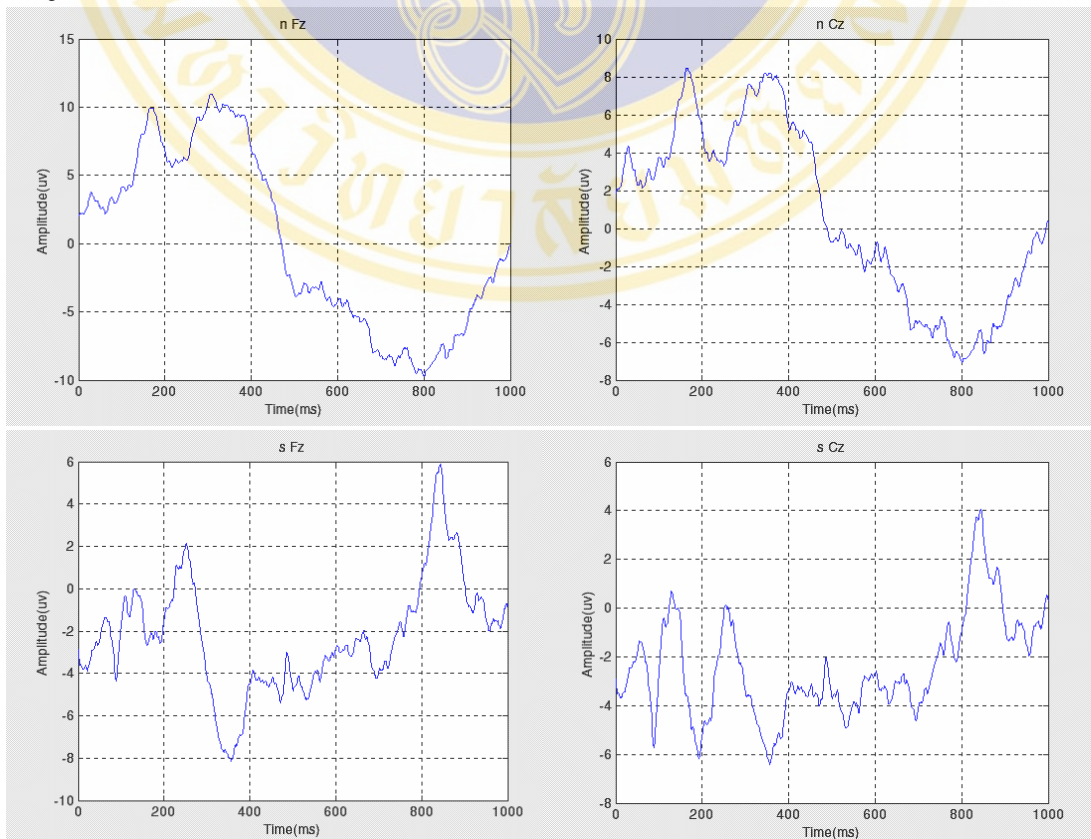
Subject 26th



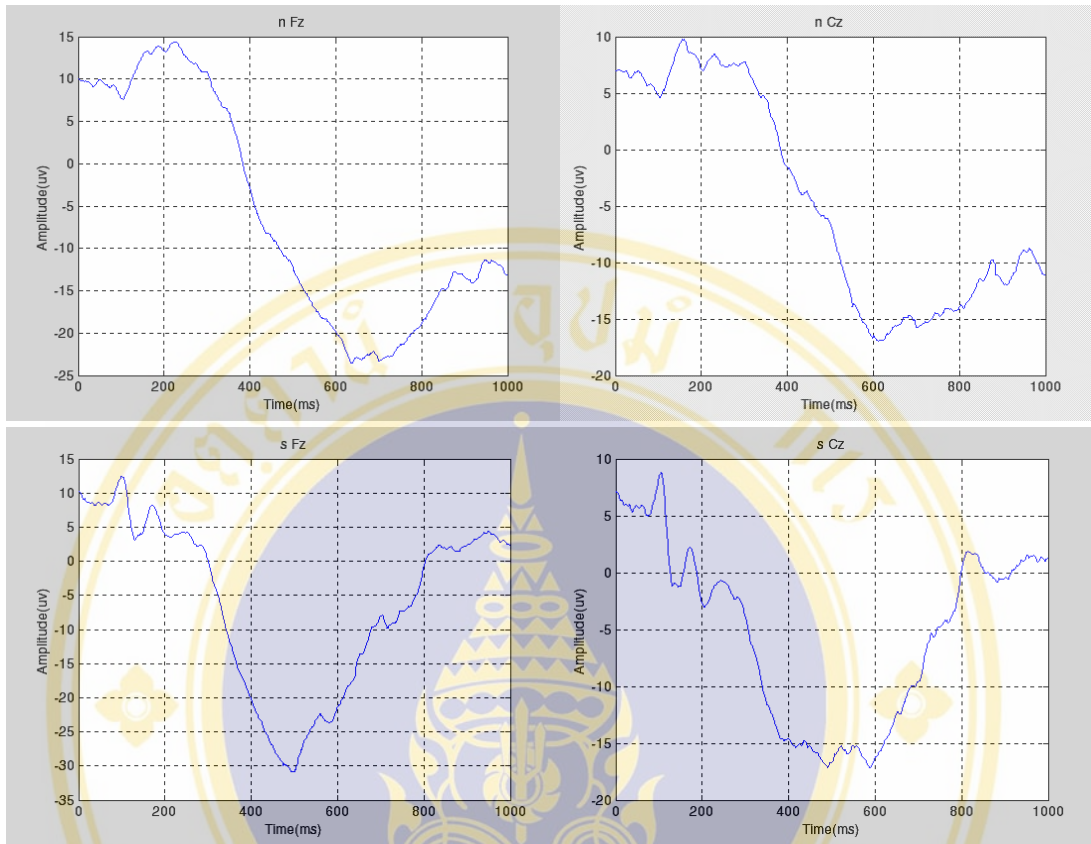
Subject 27th



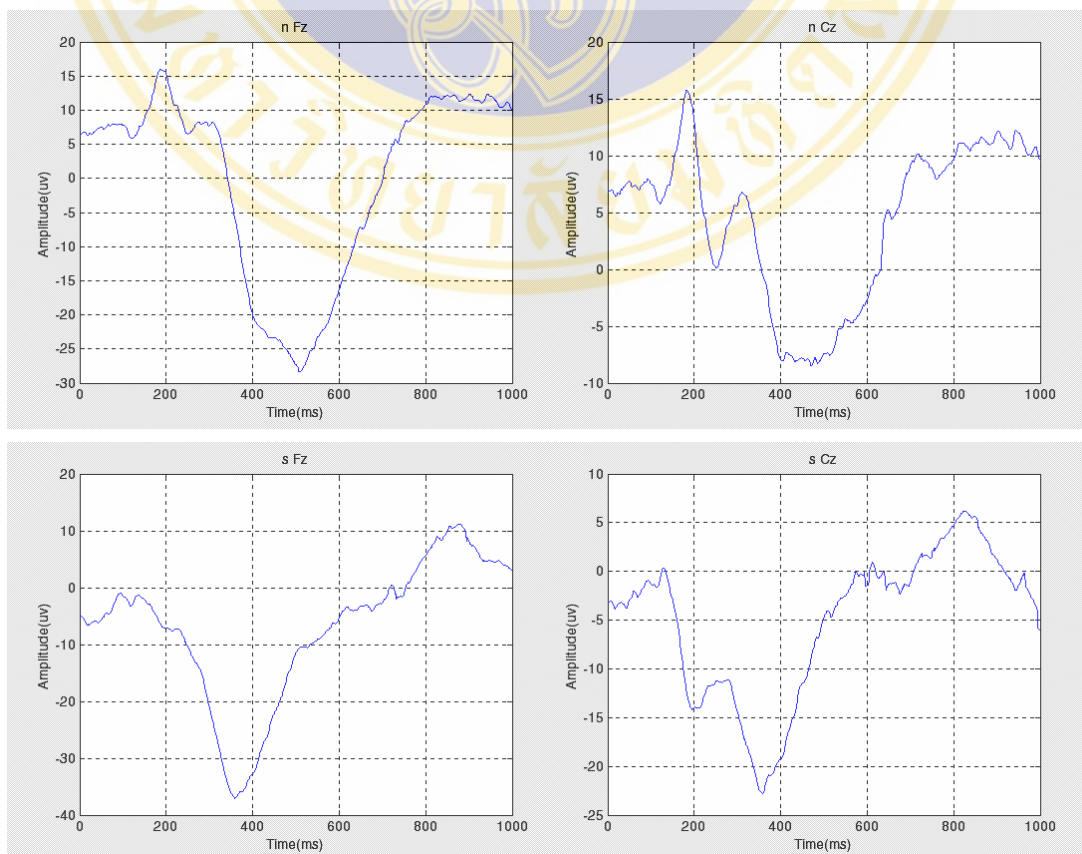
Subject 28th



Subject 29th



Subject 30th



APPENDIX C: C program

```

%%%%%%%%%%%%%%%%
% CMOUSE
%%%%%%%%%%%%%%%%
#if !defined(__mouse_h)
#define __mouse_h
void m_init(int *find,int *buttons);
void m_display_cursor(void);
void m_hide_cursor(void);
void m_get_status(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
void m_set_cursor_location(int *x,int *y);
int m_get_Lbutton_press(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
int m_get_Rbutton_press(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
int m_get_Cbutton_press(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
int m_get_Lbutton_release(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
int m_get_Rbutton_release(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
int m_get_Cbutton_release(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton);
void m_set_horizontal_cursor_limits(int x1,int x2);//07
void m_set_vertical_cursor_limits(int y1,int y2);//08
void m_set_area_limits(int x1,int x2,int y1,int y2);//07,08
void m_change_cursor(int mask[],int mask_x,int mask_y);
void m_speed(int Ratio_x,int Ratio_y);
void m_set_nocursor_area(int x1,int y1,int x2,int y2);
void m_init(int *find,int *buttons)
{
    union REGS regs;
    regs.x.ax = 0; /* init mouse */
    int86(0x33, &regs, &regs);
    *find = regs.x.ax; /* 0000h if mouse not available */
    *buttons = regs.x.bx; /* number of mouse buttons */
}
void m_display_cursor(void)
{
    union REGS regs;
    regs.x.ax = 1; /* set cursor position */
    int86(0x33, &regs, &regs);
}
void m_hide_cursor(void)
{
    union REGS regs;
    regs.x.ax = 2; /* conceal(hide) cursor */
    int86(0x33, &regs, &regs);
}
void m_get_status(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C;
    asm{

```

```

        mov ax,3
        int 0x33
        mov X,cx
        mov Y,dx
        mov ax,bx
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X;          /* CX = horizon position -> x */
    *y=Y;          /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
}
void m_set_cursor_location(int *x,int *y)
{
    union REGS regs;
    regs.x.ax = 4; /* set mouse pointer location */
    regs.x.cx = *x; /* set horizontal cursor position */
    regs.x.dx = *y; /* set vertical cursor position */
    int86(0x33, &regs, &regs);
}
int m_get_Lbutton_press(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C,count;
    asm{
        mov ax,5
        mov bx,0          /* for Left button _BX = 0 */
        int 0x33
        mov X,cx
        mov Y,dx
        mov count,bx
        mov bx,ax
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
        shr bx,1
    }
}

```

```

        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X;          /* CX = horizon position -> x */
    *y=Y;          /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
    return count;
}
int m_get_Rbutton_press(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C,count;
    asm{
        mov ax,5
        mov bx,1          /* for Right button _BX = 1 */
        int 0x33
        mov X,cx
        mov Y,dx
        mov count,bx
        mov bx,ax
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X;          /* CX = horizon position -> x */
    *y=Y;          /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
    return count;
}
int m_get_Cbutton_press(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C,count;
    asm{
        mov ax,5
        mov bx,2          /* for Center button _BX = 2 */
        int 0x33
        mov X,cx

```

```

        mov Y,dx
        mov count,bx
        mov bx,ax
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X;          /* CX = horizon position -> x */
    *y=Y;          /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
    return count;
}
int m_get_Lbutton_release(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C,count;
    asm{
        mov ax,6
        mov bx,0          /* for Left button _BX = 0 */
        int 0x33
        mov X,cx
        mov Y,dx
        mov count,bx
        mov bx,ax
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X;          /* CX = horizon position -> x */
    *y=Y;          /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
}

```

```

    return count;
}
int m_get_Rbutton_release(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C,count;
    asm{
        mov ax,6
        mov bx,1    /* for Right button _BX = 1 */
        int 0x33
        mov X,cx
        mov Y,dx
        mov count,bx
        mov bx,ax
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X;          /* CX = horizon position -> x */
    *y=Y;          /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
    return count;
}
int m_get_Cbutton_release(int *x,int *y,int *Lbutton,int *Rbutton,int *Cbutton)
{
    int X,Y,L,R,C,count;
    asm{
        mov ax,6
        mov bx,2    /* for Center button _BX = 2 */
        int 0x33
        mov X,cx
        mov Y,dx
        mov count,bx
        mov bx,ax
        and ax,1
        mov L,ax
        shr bx,1
        mov ax,bx
        and ax,1
        mov R,ax
    }
}

```

```

        shr bx,1
        mov ax,bx
        and ax,1
        mov C,ax
    }
    *x=X; /* CX = horizon position -> x */
    *y=Y; /* DX = vertical position -> y */
    *Lbutton=L;
    *Rbutton=R;
    *Cbutton=C;
    return count;
}
void m_set_horizontal_cursor_limits(int x1,int x2)
{
    union REGS regs;
    regs.x.ax = 7; /* set horizontal mouse pointer location */
    regs.x.cx = x1; /* set minimum horizontal cursor position */
    regs.x.dx = x2; /* set maximum horizontal cursor position */
    int86(0x33, &regs, &regs);
}
void m_set_vertical_cursor_limits(int y1,int y2)
{
    union REGS regs;
    regs.x.ax = 8; /* set vertical mouse pointer location */
    regs.x.cx = y1; /* set minimum vertical cursor position */
    regs.x.dx = y2; /* set maximum vertical cursor position */
    int86(0x33, &regs, &regs);
}
void m_set_area_limits(int x1,int x2,int y1,int y2)
{
    m_set_horizontal_cursor_limits(x1,x2);
    m_set_vertical_cursor_limits(y1,y2);
}
void m_change_cursor(int mask[],int mask_x,int mask_y)
{
    union REGS regs;
    struct SREGS sregs;
    regs.x.ax = 9; /* change cursor shape */
    regs.x.bx = mask_x; /* x cursor's point position */
    regs.x.cx = mask_y; /* y cursor's point position */
    regs.x.dx = (int)mask;
    segread(&sregs);
    sregs.es = sregs.ds;
    int86x(0x33, &regs, &sregs);
}
void m_speed(int Ratio_x,int Ratio_y)
{
    union REGS regs;

```

```

regs.x.ax = 0x0f; /* set pixel rate movement */
regs.x.cx = Ratio_x;
regs.x.dx = Ratio_y;
int86(0x33, &regs, &regs);
}
void m_set_nocursor_area(int x1,int y1,int x2,int y2)
{
union REGS regs;
regs.x.ax = 0x10; /* set mouse pointer exclusion area */
regs.x.cx = x1; /* set upper left horizontal cursor position */
regs.x.dx = y1; /* set upper left vertical cursor position */
regs.x.si = x2; /* set lower right horizontal cursor position */
regs.x.di = y2; /* set lower right vertical cursor position */
int86(0x33, &regs, &regs);
}
#endif

%%%%%%%%%%%%%%
%BODY
%%%%%%%%%%%%%%
#ifndef __PRJ_H
#define __PRJ_H
void StartGraphic(void);
void DrawWindw(int wx,int wy,int ww,int wh,int brd);
void DrawWindwBar(int wx,int wy,int ww,char *str);
void DrawButton(int x,int y,char *str);
void ButtonClick(int wx,int wy,int ww,int wh,char *str);
#define PALETTE_MASK 0x3c6
#define PALETTE_REGISTER_RD 0x3c7
#define PALETTE_REGISTER_WR 0x3c8
#define PALETTE_DATA 0x3c9
typedef struct RGB_color_typ
{
unsigned char red;
unsigned char green;
unsigned char blue;
} RGB_color, *RGB_color_ptr;
typedef struct pcx_picture_typ
{
char header[128];
RGB_color palette[256];
char far *buffer;
} pcx_picture, *pcx_picture_ptr;
unsigned char far *video = (char far *)0xA0000000L;
void setvideomode(int mode);
void vdo_out(int x,int y,unsigned char color);
void Set_Palette_Register(int index, RGB_color_ptr color);
void PCX_Load_targets(void);

```

```
void PCX_Load_Test(void);
void outtextxy(int x, int y, int index, char *stringin);
void outchar(int x, int y, int index, char character);
char far *FONTADDR;
void putpixel(int x, int y, int index);
void gbox(int x1, int y1, int x2, int y2, int index);
void gethline(int x1, int x2, int y, void far *image);
void puthline(int x1, int x2, int y, void far *image);
void Setfont(void);
void button(int x1, int y1, int x2, int y2, int mindex, int hindex, int lindex);
void abstr(void);
void intro(void);
void Ex_targets(void);
void Ex_nontargets(void);
char *nontarget[136] =
{
    nontarget[1] = "c:\\tc\\graphic\\ntarget\\001.pcx",
    nontarget[2] = "c:\\tc\\graphic\\ntarget\\002.pcx",
    nontarget[3] = "c:\\tc\\graphic\\ntarget\\003.pcx",
    nontarget[4] = "c:\\tc\\graphic\\ntarget\\004.pcx",
    nontarget[5] = "c:\\tc\\graphic\\ntarget\\005.pcx",
    nontarget[6] = "c:\\tc\\graphic\\ntarget\\006.pcx",
    nontarget[7] = "c:\\tc\\graphic\\ntarget\\007.pcx",
    nontarget[8] = "c:\\tc\\graphic\\ntarget\\008.pcx",
    nontarget[9] = "c:\\tc\\graphic\\ntarget\\009.pcx",
    nontarget[10] = "c:\\tc\\graphic\\ntarget\\010.pcx",
    nontarget[11] = "c:\\tc\\graphic\\ntarget\\011.pcx",
    nontarget[12] = "c:\\tc\\graphic\\ntarget\\012.pcx",
    nontarget[13] = "c:\\tc\\graphic\\ntarget\\013.pcx",
    nontarget[14] = "c:\\tc\\graphic\\ntarget\\014.pcx",
    nontarget[15] = "c:\\tc\\graphic\\ntarget\\015.pcx",
    nontarget[16] = "c:\\tc\\graphic\\ntarget\\016.pcx",
    nontarget[17] = "c:\\tc\\graphic\\ntarget\\017.pcx",
    nontarget[18] = "c:\\tc\\graphic\\ntarget\\018.pcx",
    nontarget[19] = "c:\\tc\\graphic\\ntarget\\019.pcx",
    nontarget[20] = "c:\\tc\\graphic\\ntarget\\020.pcx",
    nontarget[21] = "c:\\tc\\graphic\\ntarget\\021.pcx",
    nontarget[22] = "c:\\tc\\graphic\\ntarget\\022.pcx",
    nontarget[23] = "c:\\tc\\graphic\\ntarget\\023.pcx",
    nontarget[24] = "c:\\tc\\graphic\\ntarget\\024.pcx",
    nontarget[25] = "c:\\tc\\graphic\\ntarget\\025.pcx",
    nontarget[26] = "c:\\tc\\graphic\\ntarget\\026.pcx",
    nontarget[27] = "c:\\tc\\graphic\\ntarget\\027.pcx",
    nontarget[28] = "c:\\tc\\graphic\\ntarget\\028.pcx",
    nontarget[29] = "c:\\tc\\graphic\\ntarget\\029.pcx",
    nontarget[30] = "c:\\tc\\graphic\\ntarget\\030.pcx",
    nontarget[31] = "c:\\tc\\graphic\\ntarget\\031.pcx",
    nontarget[32] = "c:\\tc\\graphic\\ntarget\\032.pcx",
```

nontarget[33] = "c:\\tc\\graphic\\nontarget\\033.pcx",
nontarget[34] = "c:\\tc\\graphic\\nontarget\\034.pcx",
nontarget[35] = "c:\\tc\\graphic\\nontarget\\035.pcx",
nontarget[36] = "c:\\tc\\graphic\\nontarget\\036.pcx",
nontarget[37] = "c:\\tc\\graphic\\nontarget\\037.pcx",
nontarget[38] = "c:\\tc\\graphic\\nontarget\\038.pcx",
nontarget[39] = "c:\\tc\\graphic\\nontarget\\039.pcx",
nontarget[40] = "c:\\tc\\graphic\\nontarget\\040.pcx",
nontarget[41] = "c:\\tc\\graphic\\nontarget\\041.pcx",
nontarget[42] = "c:\\tc\\graphic\\nontarget\\042.pcx",
nontarget[43] = "c:\\tc\\graphic\\nontarget\\043.pcx",
nontarget[44] = "c:\\tc\\graphic\\nontarget\\044.pcx",
nontarget[45] = "c:\\tc\\graphic\\nontarget\\045.pcx",
nontarget[46] = "c:\\tc\\graphic\\nontarget\\046.pcx",
nontarget[47] = "c:\\tc\\graphic\\nontarget\\047.pcx",
nontarget[48] = "c:\\tc\\graphic\\nontarget\\048.pcx",
nontarget[49] = "c:\\tc\\graphic\\nontarget\\049.pcx",
nontarget[50] = "c:\\tc\\graphic\\nontarget\\050.pcx",
nontarget[51] = "c:\\tc\\graphic\\nontarget\\051.pcx",
nontarget[52] = "c:\\tc\\graphic\\nontarget\\052.pcx",
nontarget[53] = "c:\\tc\\graphic\\nontarget\\053.pcx",
nontarget[54] = "c:\\tc\\graphic\\nontarget\\054.pcx",
nontarget[55] = "c:\\tc\\graphic\\nontarget\\055.pcx",
nontarget[56] = "c:\\tc\\graphic\\nontarget\\056.pcx",
nontarget[57] = "c:\\tc\\graphic\\nontarget\\057.pcx",
nontarget[58] = "c:\\tc\\graphic\\nontarget\\058.pcx",
nontarget[59] = "c:\\tc\\graphic\\nontarget\\059.pcx",
nontarget[60] = "c:\\tc\\graphic\\nontarget\\060.pcx",
nontarget[61] = "c:\\tc\\graphic\\nontarget\\061.pcx",
nontarget[62] = "c:\\tc\\graphic\\nontarget\\062.pcx",
nontarget[63] = "c:\\tc\\graphic\\nontarget\\063.pcx",
nontarget[64] = "c:\\tc\\graphic\\nontarget\\064.pcx",
nontarget[65] = "c:\\tc\\graphic\\nontarget\\065.pcx",
nontarget[66] = "c:\\tc\\graphic\\nontarget\\066.pcx",
nontarget[67] = "c:\\tc\\graphic\\nontarget\\067.pcx",
nontarget[68] = "c:\\tc\\graphic\\nontarget\\068.pcx",
nontarget[69] = "c:\\tc\\graphic\\nontarget\\069.pcx",
nontarget[70] = "c:\\tc\\graphic\\nontarget\\070.pcx",
nontarget[71] = "c:\\tc\\graphic\\nontarget\\071.pcx",
nontarget[72] = "c:\\tc\\graphic\\nontarget\\072.pcx",
nontarget[73] = "c:\\tc\\graphic\\nontarget\\073.pcx",
nontarget[74] = "c:\\tc\\graphic\\nontarget\\074.pcx",
nontarget[75] = "c:\\tc\\graphic\\nontarget\\075.pcx",
nontarget[76] = "c:\\tc\\graphic\\nontarget\\076.pcx",
nontarget[77] = "c:\\tc\\graphic\\nontarget\\077.pcx",
nontarget[78] = "c:\\tc\\graphic\\nontarget\\078.pcx",
nontarget[79] = "c:\\tc\\graphic\\nontarget\\079.pcx",
nontarget[80] = "c:\\tc\\graphic\\nontarget\\080.pcx",

nontarget[81] = "c:\\tc\\graphic\\ntarget\\081.pcx",
nontarget[82] = "c:\\tc\\graphic\\ntarget\\082.pcx",
nontarget[83] = "c:\\tc\\graphic\\ntarget\\083.pcx",
nontarget[84] = "c:\\tc\\graphic\\ntarget\\084.pcx",
nontarget[85] = "c:\\tc\\graphic\\ntarget\\085.pcx",
nontarget[86] = "c:\\tc\\graphic\\ntarget\\086.pcx",
nontarget[87] = "c:\\tc\\graphic\\ntarget\\087.pcx",
nontarget[88] = "c:\\tc\\graphic\\ntarget\\088.pcx",
nontarget[89] = "c:\\tc\\graphic\\ntarget\\089.pcx",
nontarget[90] = "c:\\tc\\graphic\\ntarget\\090.pcx",
nontarget[91] = "c:\\tc\\graphic\\ntarget\\091.pcx",
nontarget[92] = "c:\\tc\\graphic\\ntarget\\092.pcx",
nontarget[93] = "c:\\tc\\graphic\\ntarget\\093.pcx",
nontarget[94] = "c:\\tc\\graphic\\ntarget\\094.pcx",
nontarget[95] = "c:\\tc\\graphic\\ntarget\\095.pcx",
nontarget[96] = "c:\\tc\\graphic\\ntarget\\096.pcx",
nontarget[97] = "c:\\tc\\graphic\\ntarget\\097.pcx",
nontarget[98] = "c:\\tc\\graphic\\ntarget\\098.pcx",
nontarget[99] = "c:\\tc\\graphic\\ntarget\\099.pcx",
nontarget[100] = "c:\\tc\\graphic\\ntarget\\100.pcx",
nontarget[101] = "c:\\tc\\graphic\\ntarget\\101.pcx",
nontarget[102] = "c:\\tc\\graphic\\ntarget\\102.pcx",
nontarget[103] = "c:\\tc\\graphic\\ntarget\\103.pcx",
nontarget[104] = "c:\\tc\\graphic\\ntarget\\104.pcx",
nontarget[105] = "c:\\tc\\graphic\\ntarget\\105.pcx",
nontarget[106] = "c:\\tc\\graphic\\ntarget\\106.pcx",
nontarget[107] = "c:\\tc\\graphic\\ntarget\\107.pcx",
nontarget[108] = "c:\\tc\\graphic\\ntarget\\108.pcx",
nontarget[109] = "c:\\tc\\graphic\\ntarget\\109.pcx",
nontarget[110] = "c:\\tc\\graphic\\ntarget\\110.pcx",
nontarget[111] = "c:\\tc\\graphic\\ntarget\\111.pcx",
nontarget[112] = "c:\\tc\\graphic\\ntarget\\112.pcx",
nontarget[113] = "c:\\tc\\graphic\\ntarget\\113.pcx",
nontarget[114] = "c:\\tc\\graphic\\ntarget\\114.pcx",
nontarget[115] = "c:\\tc\\graphic\\ntarget\\115.pcx",
nontarget[116] = "c:\\tc\\graphic\\ntarget\\116.pcx",
nontarget[117] = "c:\\tc\\graphic\\ntarget\\117.pcx",
nontarget[118] = "c:\\tc\\graphic\\ntarget\\118.pcx",
nontarget[119] = "c:\\tc\\graphic\\ntarget\\119.pcx",
nontarget[120] = "c:\\tc\\graphic\\ntarget\\120.pcx",
nontarget[121] = "c:\\tc\\graphic\\ntarget\\121.pcx",
nontarget[122] = "c:\\tc\\graphic\\ntarget\\122.pcx",
nontarget[123] = "c:\\tc\\graphic\\ntarget\\123.pcx",
nontarget[124] = "c:\\tc\\graphic\\ntarget\\124.pcx",
nontarget[125] = "c:\\tc\\graphic\\ntarget\\125.pcx",
nontarget[126] = "c:\\tc\\graphic\\ntarget\\126.pcx",
nontarget[127] = "c:\\tc\\graphic\\ntarget\\127.pcx",
nontarget[128] = "c:\\tc\\graphic\\ntarget\\128.pcx",

```
nontarget[129] = "c:\\tc\\graphic\\ntarget\\129.pcx",
nontarget[130] = "c:\\tc\\graphic\\ntarget\\130.pcx",
nontarget[131] = "c:\\tc\\graphic\\ntarget\\131.pcx",
nontarget[132] = "c:\\tc\\graphic\\ntarget\\132.pcx",
nontarget[133] = "c:\\tc\\graphic\\ntarget\\133.pcx",
nontarget[134] = "c:\\tc\\graphic\\ntarget\\134.pcx",
nontarget[135] = "c:\\tc\\graphic\\ntarget\\135.pcx",
};
char *target[64] =
{
    target[1] = "c:\\tc\\graphic\\target\\01.pcx",
    target[2] = "c:\\tc\\graphic\\target\\02.pcx",
    target[3] = "c:\\tc\\graphic\\target\\03.pcx",
    target[4] = "c:\\tc\\graphic\\target\\04.pcx",
    target[5] = "c:\\tc\\graphic\\target\\05.pcx",
    target[6] = "c:\\tc\\graphic\\target\\06.pcx",
    target[7] = "c:\\tc\\graphic\\target\\07.pcx",
    target[8] = "c:\\tc\\graphic\\target\\08.pcx",
    target[9] = "c:\\tc\\graphic\\target\\09.pcx",
    target[10] = "c:\\tc\\graphic\\target\\10.pcx",
    target[11] = "c:\\tc\\graphic\\target\\11.pcx",
    target[12] = "c:\\tc\\graphic\\target\\12.pcx",
    target[13] = "c:\\tc\\graphic\\target\\13.pcx",
    target[14] = "c:\\tc\\graphic\\target\\14.pcx",
    target[15] = "c:\\tc\\graphic\\target\\15.pcx",
    target[16] = "c:\\tc\\graphic\\target\\16.pcx",
    target[17] = "c:\\tc\\graphic\\target\\17.pcx",
    target[18] = "c:\\tc\\graphic\\target\\18.pcx",
    target[19] = "c:\\tc\\graphic\\target\\19.pcx",
    target[20] = "c:\\tc\\graphic\\target\\20.pcx",
    target[21] = "c:\\tc\\graphic\\target\\21.pcx",
    target[22] = "c:\\tc\\graphic\\target\\22.pcx",
    target[23] = "c:\\tc\\graphic\\target\\23.pcx",
    target[24] = "c:\\tc\\graphic\\target\\24.pcx",
    target[25] = "c:\\tc\\graphic\\target\\25.pcx",
    target[26] = "c:\\tc\\graphic\\target\\26.pcx",
    target[27] = "c:\\tc\\graphic\\target\\27.pcx",
    target[28] = "c:\\tc\\graphic\\target\\28.pcx",
    target[29] = "c:\\tc\\graphic\\target\\29.pcx",
    target[30] = "c:\\tc\\graphic\\target\\30.pcx",
    target[31] = "c:\\tc\\graphic\\target\\31.pcx",
    target[32] = "c:\\tc\\graphic\\target\\32.pcx",
    target[33] = "c:\\tc\\graphic\\target\\33.pcx",
    target[34] = "c:\\tc\\graphic\\target\\34.pcx",
    target[35] = "c:\\tc\\graphic\\target\\35.pcx",
    target[36] = "c:\\tc\\graphic\\target\\36.pcx",
    target[37] = "c:\\tc\\graphic\\target\\37.pcx",
    target[38] = "c:\\tc\\graphic\\target\\38.pcx",
```

```

target[39] = "c:\\tc\\graphic\\target\\39.pcx",
target[40] = "c:\\tc\\graphic\\target\\40.pcx",
target[41] = "c:\\tc\\graphic\\target\\41.pcx",
target[42] = "c:\\tc\\graphic\\target\\42.pcx",
target[43] = "c:\\tc\\graphic\\target\\43.pcx",
target[44] = "c:\\tc\\graphic\\target\\44.pcx",
target[45] = "c:\\tc\\graphic\\target\\45.pcx",
target[46] = "c:\\tc\\graphic\\target\\46.pcx",
target[47] = "c:\\tc\\graphic\\target\\47.pcx",
target[48] = "c:\\tc\\graphic\\target\\48.pcx",
target[49] = "c:\\tc\\graphic\\target\\49.pcx",
target[50] = "c:\\tc\\graphic\\target\\50.pcx",
target[51] = "c:\\tc\\graphic\\target\\51.pcx",
target[52] = "c:\\tc\\graphic\\target\\52.pcx",
target[53] = "c:\\tc\\graphic\\target\\53.pcx",
target[54] = "c:\\tc\\graphic\\target\\54.pcx",
target[55] = "c:\\tc\\graphic\\target\\55.pcx",
target[56] = "c:\\tc\\graphic\\target\\56.pcx",
target[57] = "c:\\tc\\graphic\\target\\57.pcx",
target[58] = "c:\\tc\\graphic\\target\\58.pcx",
target[59] = "c:\\tc\\graphic\\target\\59.pcx",
target[60] = "c:\\tc\\graphic\\target\\60.pcx",
target[61] = "c:\\tc\\graphic\\target\\61.pcx",
target[62] = "c:\\tc\\graphic\\target\\62.pcx",
target[63] = "c:\\tc\\graphic\\target\\63.pcx",
};
char *exnontarget[6] =
{
    exnontarget[0] = "c:\\tc\\graphic\\ntarget\\exnont1.pcx",
    exnontarget[1] = "c:\\tc\\graphic\\ntarget\\exnont2.pcx",
    exnontarget[2] = "c:\\tc\\graphic\\ntarget\\exnont3.pcx",
    exnontarget[3] = "c:\\tc\\graphic\\ntarget\\exnont4.pcx",
    exnontarget[4] = "c:\\tc\\graphic\\ntarget\\exnont5.pcx",
    exnontarget[5] = "c:\\tc\\graphic\\ntarget\\exnont6.pcx",
};
void StartGraphic(void)
{
    int gdriver = DETECT, gmode, errorcode;
    initgraph(&gdriver, &gmode, "c:\\tc\\bgi");
    errorcode = graphresult();
    if(errorcode != grOk)
    {
        printf("Graphics error: %s\n", grapherrormsg(errorcode));
        printf("Press any key to exit:");
        getch();
        exit(1);
    }
}

```

```

void DrawWindw(int wx,int wy,int ww,int wh,int brd)
{
    setcolor(WHITE);
    moveto(wx+ww, wy);
    lineto(wx, wy);
    lineto(wx, wy+wh);
    moveto(wx+ww-1, wy+1);
    lineto(wx+1, wy+1);
    lineto(wx+1, wy+wh-1);
    setcolor(DARKGRAY);
    moveto(wx+1, wy+wh);
    lineto(wx+ww, wy+wh);
    lineto(wx+ww, wy);
    moveto(wx+2, wy+wh-1);
    lineto(wx+ww-1, wy+wh-1);
    lineto(wx+ww-1, wy+1);
    setfillstyle(SOLID_FILL, LIGHTGRAY);
    bar(wx+2, wy+2, wx+ww-2, wy+wh-2);
    if(brd==1)
    {
        setcolor(DARKGRAY);
        moveto(wx+ww-10, wy+10);
        lineto(wx+10, wy+10);
        lineto(wx+10, wy+wh-10);
        setcolor(WHITE);
        lineto(wx+ww-10, wy+wh-10);
        lineto(wx+ww-10, wy+10);
    }
}

void DrawWindwBar(int wx,int wy,int ww,char *str)
{
    char label[50];
    strcpy(label, str);
    setcolor(WHITE);
    moveto(wx+20, wy+40);
    lineto(wx+20, wy+20);
    lineto(wx+ww-20, wy+20);
    setcolor(BLACK);
    lineto(wx+ww-20, wy+40);
    lineto(wx+20, wy+40);
    setfillstyle(SOLID_FILL, DARKGRAY);
    bar(wx+21, wy+21, wx+ww-21, wy+39);
    setcolor(LIGHTMAGENTA);
    int x = (wx+ww/2)-(strlen(label)*4);
    outtextxy(x, wy+27, label);
}

void DrawButton(int wx,int wy,char *str)
{

```

```

    char label[15];
    DrawWindw(wx,wy,90,32,0);
    strcpy(label, str);
    int x = (wx+90/2)-(strlen(label)*4);
    setcolor(BLACK);
    outtextxy(x, wy+12, label);
}
void ButtonClick(int wx,int wy,int ww,int wh,char *str)
{
    int *buff;
    char label[15];
    m_hide_cursor();
    int size = imagesize(wx+2,wy+2,wx+ww-2,wy+wh-2);
    buff = (int *)malloc(size);
    if(buff)
    {
        getimage(wx+2,wy+2,wx+ww-2,wy+wh-2,buff);
        putimage(wx+3,wy+3,buff,COPY_PUT);
        free(buff);
    }
    setcolor(DARKGRAY);
    moveto(wx+ww,wy);
    lineto(wx,wy);
    lineto(wx,wy+wh);
    moveto(wx+ww-1,wy+1);
    lineto(wx+1,wy+1);
    lineto(wx+1,wy+wh-1);
    setcolor(WHITE);
    moveto(wx+1,wy+wh);
    lineto(wx+ww,wy+wh);
    lineto(wx+ww,wy);
    moveto(wx+2,wy+wh-1);
    lineto(wx+ww-1,wy+wh-1);
    lineto(wx+ww-1,wy+1);
    strcpy(label, str);
    delay(100);
    DrawButton(wx,wy,label);
    m_display_cursor();
}
void setvideomode(int mode)
{
    _AH = 0x00;
    _AL = mode;
    geninterrupt (0x10);
}
void vdo_out(int x,int y,unsigned char color)
{
    video[x+(y*320)]=color;
}

```

```

}
void Set_Palette_Register (int index, RGB_color_ptr color)
{
    outp(PALETTE_MASK, 0xff);
    outp(PALETTE_REGISTER_WR, index);
    outp(PALETTE_DATA, color->red);
    outp(PALETTE_DATA, color->green);
    outp(PALETTE_DATA, color->blue);
}
void outtextxy(int x, int y, int index, char *stringin)
{
    char *strin;
    int a=x;
    strin = stringin;
    while((*strin)!=0&&(a<1024))
    {
        outchar(a,y,index,*strin);
        a+=8;
        strin++;
    }
}
void outchar(int x, int y, int index, char character)
{
    char far *af;
    int a;
    af = FONTADDR+(unsigned int)character*0x0008;
    for(a=0; a<8; a++)
    {
        if((*af)>>7)&0x01!=0) putpixel(x ,y+a,index);
        if((*af)>>6)&0x01!=0) putpixel(x+1,y+a,index);
        if((*af)>>5)&0x01!=0) putpixel(x+2,y+a,index);
        if((*af)>>4)&0x01!=0) putpixel(x+3,y+a,index);
        if((*af)>>3)&0x01!=0) putpixel(x+4,y+a,index);
        if((*af)>>2)&0x01!=0) putpixel(x+5,y+a,index);
        if((*af)>>1)&0x01!=0) putpixel(x+6,y+a,index);
        if((*af)&1)!=0) putpixel(x+7,y+a,index);
        af++;
    }
}
void putpixel(int x, int y, int index)
{
    pokeb(0xa000,(long)x+(long)y*320l,
        (unsigned char)index);
}
void gbox(int x1, int y1, int x2, int y2, int index)
{
    int x,y;
    char grest[1000];

```

```

        for(x=x1; x<=x2; x++)
            putpixel(x,y1,index);
        gethline(x1,x2,y1,(void huge *)grest);
        for(y=y1+1; y<=y2; y++)
            puthline(x1,x2,y,grest);
    }
void gethline(int x1, int x2, int y, void far *image)
{
    movedata(0xa000,(long)x1+(long)y*320l,
            FP_SEG(image),FP_OFF(image),x2-x1+1);
}
void puthline(int x1, int x2, int y, void far *image)
{
    movedata(FP_SEG(image),FP_OFF(image),0xa000,
            (long)x1+(long)y*320l,x2-x1+1);
}
void Setfont(void)
{
    _AX = 0x1130;
    _BH = 0x03;
    geninterrupt(0x10);
    FONTADDR = (char far *)MK_FP(_ES, _BP);
}
void button(int x1, int y1, int x2, int y2,
            int mindex, int hindex, int lindex)
{
    gbox(x1+1,y1+1,x2-1,y2-1,mindex);
    gbox(x1,y1,x2,y1,hindex);
    gbox(x1,y1,x1,y2,hindex);
    gbox(x1,y2,x2,y2,lindex);
    gbox(x2,y1,x2,y2,lindex);
}
void abstr(void)
{
    int gdriver = DETECT, gmode, errorcode;
    initgraph(&gdriver, &gmode, "c:\\tc\\bgi");
    errorcode = graphresult();
    if(errorcode != grOk)
    {
        printf("Graphics error: %s\n", grapherrormsg(errorcode));
        printf("Press any key to exit:");
        getch();
        exit(1);
    }
    setcolor(7);
    settxtstyle(1,0,1);
    outtextxy(50,120,"This is an example of target pictures (human) and non-");
}

```

```

    outtextxy(0,140,"target pictures in this program.");
    outtextxy(70,180,"Press 1 for non-target picture");
    outtextxy(70,200,"Press 2 for target pictures");
    setcolor(12);
    outtextxy(0,430,"          Press ENTER to continue");
}
void intro(void)
{
    FILE *fp;
    unsigned char data;
    int num_bytes,index,count;
    int i=0;
    char cha;
    clrscr();
    abstr();
    while((cha=getch()) != 13)
    {
        if(cha==49)
        {
            setvideomode(0x13);
            Ex_nontargets();
            getch();
            setvideomode(0x03);
            abstr();
        }
        if(cha==50)
        {
            setvideomode(0x13);
            Ex_targets();
            getch();
            setvideomode(0x03);
            abstr();
        }
    }
    cleardevice();
    setvideomode(0x13);
    for(int x=0;x<320;x++)
    for(int y=0;y<200;y++)
    {
        video[x+(y*320)]=0;
    }
    Setfont();
    outtextxy(30,2,7,"An example of picture for test");
    outtextxy(30,148,7,"Click left mouse button if you think");
    outtextxy(0,158,7,"it is a target picture. Do not do any-");
    outtextxy(0,168,7,"thing if you think it is a non-target");
    outtextxy(0,178,7,"picture.");
    outtextxy(65,190,30,"Press ENTER to continue");
}

```

```

int adx=100, ady=50;
if((fp = fopen(exnontarget[0],"rb"))== NULL)
{
    fprintf(stderr, "Cannot open input file.\n");
    exit(1);
}
RGB_color palette[256];
fseek(fp,-768L,SEEK_END);
for(index = 0; index < 256; index++)
{
    palette[index].red = (getc(fp) >> 2);
    palette[index].green = (getc(fp) >> 2);
    palette[index].blue = (getc(fp) >> 2);
}
for(index = 0; index < 256; index++)
{
    Set_Palette_Register (index, &palette[index]);
}
int LineX=0,LineY=0;
unsigned char *buf;
fseek(fp,128,SEEK_SET);
count = 0;
while (LineY<90)
{
    data = (unsigned char)getc(fp);
    if((int)data >= 192 && (int)data <= 255)
    {
        num_bytes = data-192;
        data = getc(fp);
        for(count=0; count<num_bytes; ++count)
        {
            if(LineX>89)
            {
                LineX=0;
                LineY++;
            }
            vdo_out(LineX+adx,LineY+ady,data);
            LineX++;
        }
    }
    else
    {
        if(LineX>89)
        {
            LineX=0;
            LineY++;
        }
        vdo_out(LineX+adx,LineY+ady,data);
    }
}

```

```

        LineX++;
    }
}
fclose(fp);
getch();
for(x=0;x<320;x++)
for(y=0;y<200;y++)
{
    video[x+(y*320)]=0;
}
SetFont();
for(i=0; i<5; i++)
{
    if(i==0)    fp = fopen(exnontarget[1],"rb");
    if(i==1)    fp = fopen(exnontarget[2],"rb");
    if(i==2)    fp = fopen(target[19],"rb");
    if(i==3)    fp = fopen(exnontarget[3],"rb");
    if(i==4)    fp = fopen(target[26],"rb");

    RGB_color palette[256];
    fseek(fp,-768L,SEEK_END);
    for(index = 0; index < 256; index++)
    {
        palette[index].red = (getc(fp) >> 2);
        palette[index].green = (getc(fp) >> 2);
        palette[index].blue = (getc(fp) >> 2);
    }
    for(index = 0; index < 256; index++)
    {
        Set_Palette_Register (index, &palette[index]);
    }
    int LineX=0,LineY=0;
    unsigned char *buf;
    fseek(fp,128,SEEK_SET);
    count = 0;
    while (LineY<90)
    {
        data = (unsigned char)getc(fp);
        if((int)data >= 192 && (int)data <= 255)
        {
            num_bytes = data-192;
            data = getc(fp);
            for(count=0; count<num_bytes; ++count)
            {
                if(LineX>89)
                {
                    LineX=0;
                    LineY++;
                }
            }
        }
    }
}

```

```

    }
    vdo_out(LineX+adx,LineY+ady,data);
    LineX++;
}
}
else
{
    if(LineX>89)
    {
        LineX=0;
        LineY++;
    }
    vdo_out(LineX+adx,LineY+ady,data);
    LineX++;
}
}
fclose(fp);
if(i==0||i==1||i==3)
{
    delay(2300);
    outtextxy(80,158,130,"DO NOT CLICK MOUSE");
    delay(1000);
    outtextxy(80,158,0,"DO NOT CLICK MOUSE");
}
if(i==2||i==4)
{
    delay(2300);
    outtextxy(80,158,9,"LEFT MOUSE CLICK");
    delay(1000);
    outtextxy(80,158,0,"LEFT MOUSE CLICK");
}
}
}
int numbentar;
void PicAdj(void)
{
    FILE *fptar;

    if((fptar=fopen("tardata.dat","w+"))==NULL)
    {
        printf("File could not be opened.\n");
        exit(0);
    }
    else
    {
        printf("\n\nEnter the number of Target picture.\n");
        printf("==> ");
        scanf("%d",&numbentar);
    }
}

```

```

        fprintf(fptar,"%d",numbertar);
        fclose(fptar);
    }
}
int tarpic[126]={0};
int nontarpic[405]={0};
void PCX_Load_targets(void)
{
    FILE *fp,*fptar;
    unsigned char data;
    int num_bytes,index,count;
// random target number into tarpic[0-62],[63-125]
    int copp=0,tar=0,nont=0;
    int *chk;

    randomize();
    taragain:    copp = rand()%64;
    if(copp==0)
        goto taragain;
    tarpic[tar] = copp;
    tarpic[tar+63] = copp;

    for(tar=1; tar<63; tar++)
    {
        tarstart: copp = rand()%64;
        if(copp==0)
            goto tarstart;
        for(int inc=0; inc<tar; inc++)
        {
            chk = &tarpic[inc];
            if(*chk != copp)
                chk++;
            if(*chk == copp)
                goto tarstart;
        }
        tarpic[tar] = copp;
        tarpic[tar+63] = copp;
    }
// random nontarget number into nontarpic[0-134],
// [134-269],[270-405]
    copp=0;
    randomize();
    nontagain:    copp = rand()%136;
    if(copp==0)
        goto nontagain;
    nontarpic[nont] = copp;
    nontarpic[nont+135] = copp;
    nontarpic[nont+270] = copp;
}

```

```

for(nont=1; nont<135; nont++)
{
    nontstart: copp = rand()%136;
    if(copp==0)
        goto nontstart;
    for(int inc=0; inc<nont; inc++)
    {
        chk = &nontarpic[inc];
        if(*chk != copp)
            chk++;
        if(*chk == copp)
            goto nontstart;
    }
    nontarpic[nont] = copp;
    nontarpic[nont+135] = copp;
    nontarpic[nont+270] = copp;
}
SetFont();
outtextxy(50,80,7,"Prepare yourself for testing");
outtextxy(30,185,9," Press ENTER to begin test ");
}
void Ex_targets(void)
{
    FILE *fp;
    unsigned char data;
    int num_bytes,index,count;
    int extar,adx,ady;
    for(int LineX=0; LineX<320; LineX++)
    for(int LineY=0; LineY<200; LineY++)
        vdo_out(LineX, LineY, 25);
    for(extar=20; extar<56; extar=extar+7)
    {
        if(extar==20)
        {
            adx=10; ady=3;
        }
        if(extar==27||extar==34||extar==48||extar==55)
        {
            adx+=100;
        }
        if(extar==41)
        {
            adx=10; ady+=93;
        }
        if((fp = fopen(target[extar-1],"rb")) == NULL)
        {
            fprintf(stderr, "Cannot open input file.\n");
            exit(1);
        }
    }
}

```

```

}
RGB_color palette[256];
fseek(fp,-768L,SEEK_END);
for(index = 0; index < 256; index++)
{
    palette[index].red = (getc(fp) >> 2);
    palette[index].green = (getc(fp) >> 2);
    palette[index].blue = (getc(fp) >> 2);
}
for(index = 0; index < 256; index++)
{
    Set_Palette_Register (index, &palette[index]);
}
LineX=0,LineY=0;
unsigned char *buf;
fseek(fp,128,SEEK_SET);
count = 0;
while(LineY<90)
{
    data = (unsigned char)getc(fp);
    if((int)data >= 192 && (int)data <= 255)
    {
        num_bytes = data-192;
        data = getc(fp);
        for(count=0; count<num_bytes; ++count)
        {
            if(LineX>89)
            {
                LineX=0;
                LineY++;
            }
            vdo_out(LineX+adx,LineY+ady,data);
            LineX++;
        }
    }
    else
    {
        if(LineX>89)
        {
            LineX=0;
            LineY++;
        }
        vdo_out(LineX+adx,LineY+ady,data);
        LineX++;
    }
}
fclose(fp);
}

```

```

    Setfont();
    outtextxy(60,189,28,"Press ENTER to continue.");
}
void Ex_nontargets(void)
{
    FILE *fp;
    unsigned char data;
    int num_bytes,index,count;
    int exnont=1,adx,ady;
    for(int LineX=0; LineX<320; LineX++)
    for(int LineY=0; LineY<200; LineY++)
        vdo_out(LineX, LineY, 0);
    for(exnont=1;exnont<7;exnont++)
    {
        if(exnont==1)
        {
            adx=10; ady=3;
        }
        if(exnont==2||exnont==3||exnont==5||exnont==6)
        {
            adx+=100;
        }
        if(exnont==4)
        {
            adx=10; ady+=93;
        }
    }
    if((fp = fopen(exnontarget[exnont-1],"rb")) == NULL)
    {
        fprintf(stderr, "Cannot open input file.\n");
        exit(1);
    }
    RGB_color palette[256];
    fseek(fp,-768L,SEEK_END);
    for(index = 0; index < 256; index++)
    {
        palette[index].red = (getc(fp) >> 2);
        palette[index].green = (getc(fp) >> 2);
        palette[index].blue = (getc(fp) >> 2);
    }
    for(index = 0; index < 256; index++)
    {
        Set_Palette_Register (index, &palette[index]);
    }
    LineX=0,LineY=0;
    unsigned char *buf;
    fseek(fp,128,SEEK_SET);
    count = 0;

```

```

while(LineY<90)
{
    data = (unsigned char)getc(fp);
    if((int)data >= 192 && (int)data <= 255)
    {
        num_bytes = data-192;
        data = getc(fp);
        for(count=0; count<num_bytes; ++count)
        {
            if(LineX>89)
            {
                LineX=0;
                LineY++;
            }
            vdo_out(LineX+adx,LineY+ady,data);
            LineX++;
        }
    }
    else
    {
        if(LineX>89)
        {
            LineX=0;
            LineY++;
        }
        vdo_out(LineX+adx,LineY+ady,data);
        LineX++;
    }
}
fclose(fp);
}
SetFont();
outtextxy(60,189,25,"Press ENTER to continue.");
}
#define DATA 0x378
#define STATUS 0x379
#define CONTROL 0x37a
void PCX_Load_Test(void)
{
    FILE *fp,*fptar;
    unsigned char data;
    int num_bytes,index,count;
    int a,b,x,y,L,R,C,Lpress;
    int tarcorrect=0,tarpress=0;
    char c;
    unsigned char in;
    outportb(CONTROL, inportb(CONTROL) & 0xF0 | 0x04);
    if((fptar=fopen("tardata.dat", "r+"))==NULL)

```

```

{
    printf("File could not be opened.\n");
    exit(0);
}
else
{
    fscanf(fp, "%d", &numbertar);
    fclose(fp);
}
for(x=0; x<320; x++)
for(y=0; y<200; y++)
{
    video[x+(y*320)]=0;
}
int showpic=0;
int tar=0, nont=0;
for(tar=0; tar<numbertar; )
{
    in = (inport(STATUS) & 0x20);
    if(kbhit()    c=getch());
    if(c==27)    break;
    if((in&&0x20)==1)
    {
        delay(2);
        in = (inport(STATUS) & 0x40);
        if((in&&0x40)==1)
        {
            showpic=1;
        }
        else if((in&&0x40)==0)
        {
            showpic=2;
        }
    }
}
if(showpic==1)
{
    int adx=100, ady=50;
    if((fp = fopen(nontarget[nontarpic[nont]-1], "rb"))== NULL)
    {
        fprintf(stderr, "Cannot open input file.\n");
        exit(1);
    }
    RGB_color palette[256];
    fseek(fp, -768L, SEEK_END);
    for(index = 0; index < 256; index++)
    {
        palette[index].red = (getc(fp) >> 2);
        palette[index].green = (getc(fp) >> 2);
    }
}

```

```

        palette[index].blue = (getc(fp) >> 2);
    }
    for(index = 0; index < 256; index++)
    {
        Set_Palette_Register (index, &palette[index]);
    }
    int LineX=0,LineY=0;
    unsigned char *buf;
    fseek(fp,128,SEEK_SET);
    count = 0;
    while (LineY<90)
    {
        data = (unsigned char)getc(fp);
        if((int)data >= 192 && (int)data <= 255)
        {
            num_bytes = data-192;
            data = getc(fp);
            for(count=0; count<num_bytes; ++count)
            {
                if(LineX>89)
                {
                    LineX=0;
                    LineY++;
                }
                vdo_out(LineX+adx,LineY+ady,data);
                LineX++;
            }
        }
        else
        {
            if(LineX>89)
            {
                LineX=0;
                LineY++;
            }
            vdo_out(LineX+adx,LineY+ady,data);
            LineX++;
        }
    }
}
fclose(fp);
showpic=0;
in = (inport(STATUS) & 0x40);
while((in&&0x40)==1)
{
    in = (inport(STATUS) & 0x40);
}
in = (inport(STATUS) & 0xF0);
while((in&0x20)==0x00)

```

```

    {
        in = (inport(STATUS) & 0xF0);
    }
    nont++;
}
if(showpic==2)
{
    int adx=100, ady=50;
    m_hide_cursor();
    if((fp = fopen(target[tarpic[tar]-1], "rb")) == NULL)
    {
        fprintf(stderr, "Cannot open input file.\n");
        exit(1);
    }
    RGB_color palette[256];
    fseek(fp, -768L, SEEK_END);
    for(index = 0; index < 256; index++)
    {
        palette[index].red = (getc(fp) >> 2);
        palette[index].green = (getc(fp) >> 2);
        palette[index].blue = (getc(fp) >> 2);
    }
    for(index = 0; index < 256; index++)
    {
        Set_Palette_Register (index, &palette[index]);
    }

    int LineX=0, LineY=0;
    unsigned char *buf;
    fseek(fp, 128, SEEK_SET);
    count = 0;
    while (LineY < 90)
    {
        data = (unsigned char)getc(fp);
        if((int)data >= 192 && (int)data <= 255)
        {
            num_bytes = data - 192;
            data = getc(fp);
            for(count=0; count < num_bytes; ++count)
            {
                if(LineX > 89)
                {
                    LineX=0;
                    LineY++;
                }
                vdo_out(LineX+adx, LineY+ady, data);
                LineX++;
            }
        }
    }
}

```

```

    }
    else
    {
        if(LineX>89)
        {
            LineX=0;
            LineY++;
        }
        vdo_out(LineX+adx,LineY+ady,data);
        LineX++;
    }
}
fclose(fp);
showpic=0;
in = (inport(STATUS) & 0x20);
while(in==0x20)
{
    Lpress=m_get_Lbutton_press(&x,&y,&L,&R,&C);
    if(Lpress==1)
    {
        tarpress=1;
    }
    in = (inport(STATUS) & 0x20);
}
tar++;
tarcorrect=tarcorrect+tarpress;
tarpress=0;
}
}
for(x=0;x<320;x++)
for(y=0;y<200;y++)
    video[x+(y*320)]=0;
SetFont();
outtextxy(60,90,3,"Thank you for testing");
delay(2000);
setvideomode(0x03);
textcolor(13);
cprintf("\r\n\n");
cprintf("    ALL %d Target picture was shown\r\n\r\n",numbertar);
cprintf("        Your score is...");
textcolor(10);
cprintf("\r\n\n\n");
cprintf("        Target picture press    = %d\r\n\r\n",tarcorrect);
cprintf("        Target picture not press = %d\r\n\r\n",numbertar-tarcorrect);
textcolor(12);
cprintf("\r\n\n");
cprintf("        Press ENTER to continue...");
}

```

```

#endif

%%%%%%%%%%%%%%
%P300
%%%%%%%%%%%%%%
#include <alloc.h>
#include <bios.h>
#include <conio.h>
#include <dos.h>
#include <graphics.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include "c:\tc\thesis\cmouse.h"
#include "c:\tc\thesis\body.h"
void main(void)
{
    int a,b,x,y,L,R,C;
    int quitt=1;
    clrscr();
    for(;quitt!=0;)
    {
        StartGraphic();
        DrawWindw(0,0,639,479,1);
        DrawWindwBar(0,0,639,"MEMORY TEST");
        DrawButton(80,150,"PIC.ADJ");
        DrawButton(80,210,"INTRODUCE");
        DrawButton(80,270,"START");
        DrawButton(80,330,"QUIT");
        m_set_area_limits(10,629,10,469);
        m_display_cursor();
        int Lpres,exitt=1;
        for(;exitt!=0;)
        {
            Lpres=m_get_Lbutton_press(&x,&y,&L,&R,&C);
            if(x>82 && x<168 && y>152 && y<180 && Lpres==1)
            {
                m_hide_cursor();
                ButtonClick(80,150,90,32,"PIC.ADJ");
                m_display_cursor();
                setvideomode(0x03);
                PicAdj();
                exitt=0;
            }
            else if(x>82 && x<168 && y>212 && y<240 && Lpres==1)
            {
                m_hide_cursor();
                ButtonClick(80,210,90,32,"INTRODUCE");
            }
        }
    }
}

```

```
        m_display_cursor();
        setvideomode(0x03);
        intro();
        exitt=0;
    }
    else if(x>82 && x<168 && y>272 && y<300 && Lpres==1)
    {
        Lpres=0,quitt=1;
        m_hide_cursor();
        ButtonClick(80,272,90,32,"START");
        m_display_cursor();
        setvideomode(0x13);
        PCX_Load_targets();
        getch();
        PCX_Load_Test();
        getch();
        exitt=0;
    }
    else if(x>82 && x<168 && y>332 && y<360 && Lpres==1)
    {
        m_hide_cursor();
        ButtonClick(80,330,90,32,"QUIT");
        m_display_cursor();
        exitt=0;
        quitt=0;
    }
}
closegraph();
}
```

APPENDIX D: Matlab programming

```

%%%%%%%%%%
%SUBJECT 1st
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n011.txt;
load c:\matlabr12\thesis\subject\n012.txt;
load c:\matlabr12\thesis\subject\s011.txt;
load c:\matlabr12\thesis\subject\s012.txt;

plot(n011(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n012(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s011(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s012(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n011(:,2)*(-1)).^2))
areanCz = trapz(l,((n012(:,2)*(-1)).^2))
areasFz = trapz(l,((s011(:,2)*(-1)).^2))
areasCz = trapz(l,((s012(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 2nd
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n021.txt;
load c:\matlabr12\thesis\subject\n022.txt;
load c:\matlabr12\thesis\subject\s021.txt;
load c:\matlabr12\thesis\subject\s022.txt;

plot(n021(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n022(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s021(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s022(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n021(:,2)*(-1)).^2))
areanCz = trapz(l,((n022(:,2)*(-1)).^2))
areasFz = trapz(l,((s021(:,2)*(-1)).^2))
areasCz = trapz(l,((s022(:,2)*(-1)).^2))

```

```

%%%%%%%%%
%SUBJECT 3rd
%%%%%%%%%
load c:\matlabr12\thesis\subject\n031.txt;
load c:\matlabr12\thesis\subject\n032.txt;
load c:\matlabr12\thesis\subject\s031.txt;
load c:\matlabr12\thesis\subject\s032.txt;

plot(n031(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n032(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s031(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s032(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n031(:,2)*(-1)).^2))
areanCz = trapz(l,((n032(:,2)*(-1)).^2))
areasFz = trapz(l,((s031(:,2)*(-1)).^2))
areasCz = trapz(l,((s032(:,2)*(-1)).^2))

%%%%%%%%%
%SUBJECT 4th
%%%%%%%%%
load c:\matlabr12\thesis\subject\n041.txt;
load c:\matlabr12\thesis\subject\n042.txt;
load c:\matlabr12\thesis\subject\s041.txt;
load c:\matlabr12\thesis\subject\s042.txt;

plot(n041(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n042(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s041(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s042(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n041(:,2)*(-1)).^2))
areanCz = trapz(l,((n042(:,2)*(-1)).^2))
areasFz = trapz(l,((s041(:,2)*(-1)).^2))
areasCz = trapz(l,((s042(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 5th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n051.txt;
load c:\matlabr12\thesis\subject\n052.txt;
load c:\matlabr12\thesis\subject\s051.txt;
load c:\matlabr12\thesis\subject\s052.txt;

plot(n051(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n052(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s051(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s052(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n051(:,2)*(-1)).^2))
areanCz = trapz(l,((n052(:,2)*(-1)).^2))
areasFz = trapz(l,((s051(:,2)*(-1)).^2))
areasCz = trapz(l,((s052(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 6th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n061.txt;
load c:\matlabr12\thesis\subject\n062.txt;
load c:\matlabr12\thesis\subject\s061.txt;
load c:\matlabr12\thesis\subject\s062.txt;

plot(n061(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n062(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s061(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;

figure(4); plot(s062(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n061(:,2)*(-1)).^2))
areanCz = trapz(l,((n062(:,2)*(-1)).^2))
areasFz = trapz(l,((s061(:,2)*(-1)).^2))
areasCz = trapz(l,((s062(:,2)*(-1)).^2))

```

```

%%%%%%%%%
%SUBJECT 7th
%%%%%%%%%
load c:\matlabr12\thesis\subject\n071.txt;
load c:\matlabr12\thesis\subject\n072.txt;
load c:\matlabr12\thesis\subject\s071.txt;
load c:\matlabr12\thesis\subject\s072.txt;

plot(n071(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n072(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s071(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s072(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n071(:,2)*(-1)).^2))
areanCz = trapz(l,((n072(:,2)*(-1)).^2))
areasFz = trapz(l,((s071(:,2)*(-1)).^2))
areasCz = trapz(l,((s072(:,2)*(-1)).^2))

%%%%%%%%%
%SUBJECT 8th
%%%%%%%%%
load c:\matlabr12\thesis\subject\n081.txt;
load c:\matlabr12\thesis\subject\n082.txt;
load c:\matlabr12\thesis\subject\s081.txt;
load c:\matlabr12\thesis\subject\s082.txt;

plot(n081(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n082(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s081(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s082(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n081(:,2)*(-1)).^2))
areanCz = trapz(l,((n082(:,2)*(-1)).^2))
areasFz = trapz(l,((s081(:,2)*(-1)).^2))
areasCz = trapz(l,((s082(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 9th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n091.txt;
load c:\matlabr12\thesis\subject\n092.txt;
load c:\matlabr12\thesis\subject\s091.txt;
load c:\matlabr12\thesis\subject\s092.txt;

plot(n091(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n092(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s091(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s092(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n091(:,2)*(-1)).^2))
areanCz = trapz(l,((n092(:,2)*(-1)).^2))
areasFz = trapz(l,((s091(:,2)*(-1)).^2))
areasCz = trapz(l,((s092(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 10th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n101.txt;
load c:\matlabr12\thesis\subject\n102.txt;
load c:\matlabr12\thesis\subject\s101.txt;
load c:\matlabr12\thesis\subject\s102.txt;

plot(n101(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n102(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s101(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s102(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n101(:,2)*(-1)).^2))
areanCz = trapz(l,((n102(:,2)*(-1)).^2))
areasFz = trapz(l,((s101(:,2)*(-1)).^2))
areasCz = trapz(l,((s102(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 11th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n111.txt;
load c:\matlabr12\thesis\subject\n112.txt;
load c:\matlabr12\thesis\subject\s111.txt;
load c:\matlabr12\thesis\subject\s112.txt;

plot(n111(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n112(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s111(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s112(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n111(:,2)*(-1)).^2))
areanCz = trapz(l,((n112(:,2)*(-1)).^2))
areasFz = trapz(l,((s111(:,2)*(-1)).^2))
areasCz = trapz(l,((s112(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 12th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n121.txt;
load c:\matlabr12\thesis\subject\n122.txt;
load c:\matlabr12\thesis\subject\s121.txt;
load c:\matlabr12\thesis\subject\s122.txt;

plot(n121(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n122(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s121(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s122(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n121(:,2)*(-1)).^2))
areanCz = trapz(l,((n122(:,2)*(-1)).^2))
areasFz = trapz(l,((s121(:,2)*(-1)).^2))
areasCz = trapz(l,((s122(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 13th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n131.txt;
load c:\matlabr12\thesis\subject\n132.txt;
load c:\matlabr12\thesis\subject\s131.txt;
load c:\matlabr12\thesis\subject\s132.txt;

plot(n131(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n132(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s131(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s132(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n131(:,2)*(-1)).^2))
areanCz = trapz(l,((n132(:,2)*(-1)).^2))
areasFz = trapz(l,((s131(:,2)*(-1)).^2))
areasCz = trapz(l,((s132(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 14th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n141.txt;
load c:\matlabr12\thesis\subject\n142.txt;
load c:\matlabr12\thesis\subject\s141.txt;
load c:\matlabr12\thesis\subject\s142.txt;

plot(n141(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n142(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s141(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s142(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n141(:,2)*(-1)).^2))
areanCz = trapz(l,((n142(:,2)*(-1)).^2))
areasFz = trapz(l,((s141(:,2)*(-1)).^2))
areasCz = trapz(l,((s142(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 15th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n151.txt;
load c:\matlabr12\thesis\subject\n152.txt;
load c:\matlabr12\thesis\subject\s151.txt;
load c:\matlabr12\thesis\subject\s152.txt;

plot(n151(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n152(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s151(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s152(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n151(:,2)*(-1)).^2))
areanCz = trapz(l,((n152(:,2)*(-1)).^2))
areasFz = trapz(l,((s151(:,2)*(-1)).^2))
areasCz = trapz(l,((s152(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 16th
%%%%%%%%%%

load c:\matlabr12\thesis\subject\n161.txt;
load c:\matlabr12\thesis\subject\n162.txt;
load c:\matlabr12\thesis\subject\s161.txt;
load c:\matlabr12\thesis\subject\s162.txt;

plot(n161(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n162(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s161(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s162(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n161(:,2)*(-1)).^2))
areanCz = trapz(l,((n162(:,2)*(-1)).^2))
areasFz = trapz(l,((s161(:,2)*(-1)).^2))
areasCz = trapz(l,((s162(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 17th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n171.txt;
load c:\matlabr12\thesis\subject\n172.txt;
load c:\matlabr12\thesis\subject\s171.txt;
load c:\matlabr12\thesis\subject\s172.txt;

plot(n171(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n172(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s171(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s172(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n171(:,2)*(-1)).^2))
areanCz = trapz(l,((n172(:,2)*(-1)).^2))
areasFz = trapz(l,((s171(:,2)*(-1)).^2))
areasCz = trapz(l,((s172(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 18th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n181.txt;
load c:\matlabr12\thesis\subject\n182.txt;
load c:\matlabr12\thesis\subject\s181.txt;
load c:\matlabr12\thesis\subject\s182.txt;

plot(n181(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n182(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s181(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s182(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n181(:,2)*(-1)).^2))
areanCz = trapz(l,((n182(:,2)*(-1)).^2))
areasFz = trapz(l,((s181(:,2)*(-1)).^2))
areasCz = trapz(l,((s182(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 19th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n191.txt;
load c:\matlabr12\thesis\subject\n192.txt;
load c:\matlabr12\thesis\subject\s191.txt;
load c:\matlabr12\thesis\subject\s192.txt;

plot(n191(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n192(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s191(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s192(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n191(:,2)*(-1)).^2))
areanCz = trapz(l,((n192(:,2)*(-1)).^2))
areasFz = trapz(l,((s191(:,2)*(-1)).^2))
areasCz = trapz(l,((s192(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 20th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n201.txt;
load c:\matlabr12\thesis\subject\n202.txt;
load c:\matlabr12\thesis\subject\s201.txt;
load c:\matlabr12\thesis\subject\s202.txt;

plot(n201(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n202(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s201(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s202(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n201(:,2)*(-1)).^2))
areanCz = trapz(l,((n202(:,2)*(-1)).^2))
areasFz = trapz(l,((s201(:,2)*(-1)).^2))
areasCz = trapz(l,((s202(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 21st
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n211.txt;
load c:\matlabr12\thesis\subject\n212.txt;
load c:\matlabr12\thesis\subject\s211.txt;
load c:\matlabr12\thesis\subject\s212.txt;

plot(n211(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n212(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s211(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s212(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n211(:,2)*(-1)).^2))
areanCz = trapz(l,((n212(:,2)*(-1)).^2))
areasFz = trapz(l,((s211(:,2)*(-1)).^2))
areasCz = trapz(l,((s212(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 22nd
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n221.txt;
load c:\matlabr12\thesis\subject\n222.txt;
load c:\matlabr12\thesis\subject\s221.txt;
load c:\matlabr12\thesis\subject\s222.txt;

plot(n221(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n222(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s221(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4);
plot(s222(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n221(:,2)*(-1)).^2))
areanCz = trapz(l,((n222(:,2)*(-1)).^2))
areasFz = trapz(l,((s221(:,2)*(-1)).^2))
areasCz = trapz(l,((s222(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 23rd
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n231.txt;
load c:\matlabr12\thesis\subject\n232.txt;
load c:\matlabr12\thesis\subject\s231.txt;
load c:\matlabr12\thesis\subject\s232.txt;

plot(n231(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n232(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s231(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s232(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n231(:,2)*(-1)).^2))
areanCz = trapz(l,((n232(:,2)*(-1)).^2))
areasFz = trapz(l,((s231(:,2)*(-1)).^2))
areasCz = trapz(l,((s232(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 24th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n241.txt;
load c:\matlabr12\thesis\subject\n242.txt;
load c:\matlabr12\thesis\subject\s241.txt;
load c:\matlabr12\thesis\subject\s242.txt;

plot(n241(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n242(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s241(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s242(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n241(:,2)*(-1)).^2))
areanCz = trapz(l,((n242(:,2)*(-1)).^2))
areasFz = trapz(l,((s241(:,2)*(-1)).^2))
areasCz = trapz(l,((s242(:,2)*(-1)).^2))

```

```

%%%%%%%%%
%SUBJECT 25th
%%%%%%%%%
load c:\matlabr12\thesis\subject\n251.txt;
load c:\matlabr12\thesis\subject\n252.txt;
load c:\matlabr12\thesis\subject\s251.txt;
load c:\matlabr12\thesis\subject\s252.txt;

plot(n251(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n252(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s251(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s252(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;
l=linspace(0,1000,1000);
areanFz = trapz(l,((n251(:,2)*(-1)).^2))
areanCz = trapz(l,((n252(:,2)*(-1)).^2))
areasFz = trapz(l,((s251(:,2)*(-1)).^2))
areasCz = trapz(l,((s252(:,2)*(-1)).^2))

%%%%%%%%%
%SUBJECT 26th
%%%%%%%%%
load c:\matlabr12\thesis\subject\n261.txt;
load c:\matlabr12\thesis\subject\n262.txt;
load c:\matlabr12\thesis\subject\s261.txt;
load c:\matlabr12\thesis\subject\s262.txt;

plot(n261(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n262(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s261(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s262(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n261(:,2)*(-1)).^2))
areanCz = trapz(l,((n262(:,2)*(-1)).^2))
areasFz = trapz(l,((s261(:,2)*(-1)).^2))
areasCz = trapz(l,((s262(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 27th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n271.txt;
load c:\matlabr12\thesis\subject\n272.txt;
load c:\matlabr12\thesis\subject\s271.txt;
load c:\matlabr12\thesis\subject\s272.txt;

plot(n271(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n272(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s271(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;

figure(4); plot(s272(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n271(:,2)*(-1)).^2))
areanCz = trapz(l,((n272(:,2)*(-1)).^2))
areasFz = trapz(l,((s271(:,2)*(-1)).^2))
areasCz = trapz(l,((s272(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 28th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n281.txt;
load c:\matlabr12\thesis\subject\n282.txt;
load c:\matlabr12\thesis\subject\s281.txt;
load c:\matlabr12\thesis\subject\s282.txt;

plot(n281(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n282(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s281(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s282(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n281(:,2)*(-1)).^2))
areanCz = trapz(l,((n282(:,2)*(-1)).^2))
areasFz = trapz(l,((s281(:,2)*(-1)).^2))
areasCz = trapz(l,((s282(:,2)*(-1)).^2))

```

```

%%%%%%%%%%
%SUBJECT 29th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n291.txt;
load c:\matlabr12\thesis\subject\n292.txt;
load c:\matlabr12\thesis\subject\s291.txt;
load c:\matlabr12\thesis\subject\s292.txt;

plot(n291(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;

figure(2); plot(n292(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s291(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s292(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n291(:,2)*(-1)).^2))
areanCz = trapz(l,((n292(:,2)*(-1)).^2))
areasFz = trapz(l,((s291(:,2)*(-1)).^2))
areasCz = trapz(l,((s292(:,2)*(-1)).^2))

%%%%%%%%%%
%SUBJECT 30th
%%%%%%%%%%
load c:\matlabr12\thesis\subject\n301.txt;
load c:\matlabr12\thesis\subject\n302.txt;
load c:\matlabr12\thesis\subject\s301.txt;
load c:\matlabr12\thesis\subject\s302.txt;

plot(n301(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Fz'); grid on;
figure(2); plot(n302(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('n Cz'); grid on;
figure(3); plot(s301(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Fz'); grid on;
figure(4); plot(s302(:,2)*(-1)); xlabel('Time(ms)'); ylabel('Amplitude(uv)');
title('s Cz'); grid on;

l=linspace(0,1000,1000);
areanFz = trapz(l,((n301(:,2)*(-1)).^2))
areanCz = trapz(l,((n302(:,2)*(-1)).^2))
areasFz = trapz(l,((s301(:,2)*(-1)).^2))
areasCz = trapz(l,((s302(:,2)*(-1)).^2))

```

Paired t test**%compare amplitude**

```
load c:\matlabr12\thesis\subject\newFz.txt;
load c:\matlabr12\thesis\subject\newCz.txt;
load c:\matlabr12\thesis\subject\standFz.txt;
load c:\matlabr12\thesis\subject\standCz.txt;
```

```
nfz=newFz(:,2); ncz=newCz(:,2);
sfz=standFz(:,2); scz=standCz(:,2);
diff=sfz-nfz;
tf=0;
```

```
for n=1:30;
    tf = tf+diff(n);
    tf2 = (tf+diff(n))^2;
```

```
end
dfbar = tf/30;
sf2=(tf2-((tf^2)/30))/29;
ptf=dfbar/((sqrt(sf2))/(sqrt(30)))
```

```
difc=scz-ncz;
tc=0;
for n=1:30;
    tc = tc+difc(n);
    tc2 = (tc+difc(n))^2;
```

```
end
dcbar = tc/30;
sc2=(tc2-((tc^2)/30))/29;
ptc=dcbar/((sqrt(sc2))/(sqrt(30)))
```

```
% alpha = 0.01 (error 1% convince 99%)
% t.005,29 = 2.756
% t amplitude Fz = 0.8803
% |t| < t.05,29
% ud = 0, not different
% t amplitude Cz = 0.8937
% ud = 0, not different
```

%compare latency

```
load c:\matlabr12\thesis\subject\newFz.txt;
load c:\matlabr12\thesis\subject\newCz.txt;
load c:\matlabr12\thesis\subject\standFz.txt;
load c:\matlabr12\thesis\subject\standCz.txt;
nfz=newFz(:,1); ncz=newCz(:,1);
sfz=standFz(:,1); scz=standCz(:,1);
```

```
diff=nfz-sfz;
tf=0;
for n=1:30;
    tf = tf+diff(n);
```

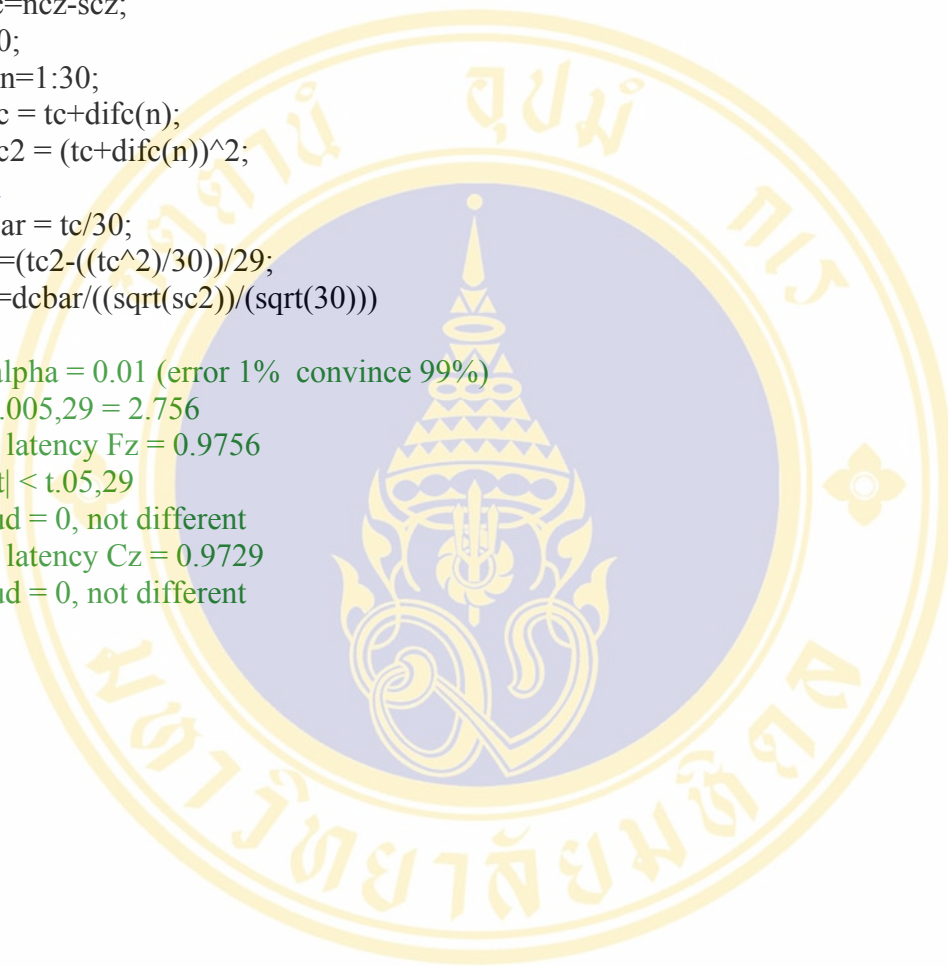
```

    tf2 = (tf+diff(n))^2;
end
dfbar = tf/30;
sf2=(tf2-((tf^2)/30))/29;
ptf=dfbar/((sqrt(sf2))/(sqrt(30)))

difc=ncz-scz;
tc=0;
for n=1:30;
    tc = tc+difc(n);
    tc2 = (tc+difc(n))^2;
end
dcbar = tc/30;
sc2=(tc2-((tc^2)/30))/29;
ptc=dcbar/((sqrt(sc2))/(sqrt(30)))

% alpha = 0.01 (error 1% convince 99%)
% t.005,29 = 2.756
% t latency Fz = 0.9756
% |t| < t.05,29
% ud = 0, not different
% t latency Cz = 0.9729
% ud = 0, not different

```



Correlation

%correlation Cz

$$zX = 1197011;$$

$$zY = 948370;$$

$$zXX = 75991320819;$$

$$zYY = 56174716598;$$

$$zXY = 64052140169;$$

$$n = 30;$$

$$b = ((n*(zXY))-(zX*zY))/((n*zXX)-(zX^2))$$

$$SStotal = zYY-(zY^2/n)$$

$$SSexplaned = b^2*(zXX-(zX^2/n))$$

$$SSunexplaned = SStotal-SSexplaned$$

$$r2 = SSexplaned/SStotal$$

$$\%b = 9.285050174714562e-001$$

$$\%SStotal = 2.619452803466667e+010$$

$$\%SSexplaned = 2.433781514595599e+010$$

$$\%SSunexplaned = 1.856712888710674e+009$$

$$\%r2 = 9.291182919480954e-001$$

%correlation Fz

$$zX = 2598107;$$

$$zY = 1909613;$$

$$zXX = 4.27842*(10^{12});$$

$$zYY = 2.85157*(10^{12});$$

$$zXY = 3.38239*(10^{12});$$

$$n = 30;$$

$$b = ((n*(zXY))-(zX*zY))/((n*zXX)-(zX^2))$$

$$SStotal = zYY-(zY^2/n)$$

$$SSexplaned = b^2*(zXX-(zX^2/n))$$

$$SSunexplaned = SStotal-SSexplaned$$

$$r2 = SSexplaned/SStotal$$

$$\%b = 7.936544783498853e-001$$

$$\%SStotal = 2.730015939674367e+012$$

$$\%SSexplaned = 2.553194951531469e+012$$

$$\%SSunexplaned = 1.768209881428980e+011$$

$$\%r2 = 9.352307854422312e-001$$

BIOGRAPHY

NAME	Mr. Manosh Suparang
DATE OF BIRTH	8 April 1977
PLACE OF BIRTH	Bangkok, Thailand
INSTITUTION ATTENDED	Mahidol University, 1994-1998: Bachelor of Engineering (Electrical Engineering) Mahidol University, 1999-2004: Master of Engineering (Biomedical Engineering)
ADDRESS	172/48, Sriayuthaya Road, Dusit, Bangkok, 10300
CONTACT	01-5561596 manosh_s@hotmail.com