Development of a general purpose computer-based platform to provide functional assistance to people with severe motor disabilities

Franco Senatore

A dissertation submitted to the Faculty of Engineering and the Built Environment at The University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering.

Johannesburg, March 2009

Declaration

I declare that this dissertation is my own, unaided work, except where otherwise acknowledged. It is being submitted for the degree of Master of Science in Engineering in the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination in any other university.

Signed this ___ day of ___________ 20___

____

Franco Senatore.

Abstract

Research and development into a generic assistive platform, which can accommodate a variety of patients suffering from a wide range of motor disabilities is described. Methodologies were established, whereby the design could be made sufficiently flexible, such that it could be programmed to suit these people in terms of their needs and level of motor disability. This needed to be achieved without redesigning the system for each person.

Suitable sensors were chosen to sense the residual motor function of the disabled individual, while being non-invasive and safe for use. These sensors included a dual-axis accelerometer (tilt switch), a 6-key touch sensor and a SCATIR switch (blink/wink sensor). The placement of the sensors, for the purpose of this study, were restricted to sensing arm (dual-axis accelerometer) or finger movements (touch sensors), head and neck movements (accelerometer) and blink/wink and/or eye-brow movements (SCATIR switch). These input devices were used to control a variety of different output functions, as required by the user, while being non-invasive and safe for use.

After ethics approval was obtained, volunteers with various motor disabilities were subsequently invited to test the system and thereafter requested to answer a series of questions regarding the performance and potential usefulness of the system. The input sensors were found to be comfortable and easy to use, while performing predictably and with very little to no fatigue experienced. The system performed as expected and accepted all of the input sensors attached to it, while repeating specific tasks multiple times. It was also established that the system was customisable in terms of providing a specific output for a specific and voluntary input. The system could be improved by further compacting and simplifying the design and operation, while using wireless sensors were necessary. It was thereafter concluded that the system, in general, was capable of satisfying the various users' diverse requirements, thereby achieving the required objectives.

Acknowledgements

The following research was performed under the auspices of the Biomedical Engineering Department at the University of the Witwatersrand, Johannesburg, South Africa.

I would like to thank my supervisors, Prof. David Rubin and Prof. George Gibbon, for their continued support, guidance and patience throughout the duration of this research. I would also like to thank all of the volunteers who came forward to assist with the performance and testing phase of the research and whose names are not mentioned here to maintain confidentiality, as well as Mr Vincent Gore who put me into contact with most of these volunteers. A special mention must be made to Mrs Celeste Mukheibir of Inclusive Solutions for kindly lending me one of the input sensors at no cost, as well as Brittan Healthcare for providing the safety testing at no cost, it is greatly appreciated.

I would also like to extend a special mention to my colleagues at the Department of Electrical and Information Engineering A-Lab, namely Megan Russell for kindly assisting me with the testing phase of the research as well as Rolan Christian, Ryan Van Den Bergh and David Vannucci for helping with various aspects of the project. A thanks goes out to Mr Tom Lattimer, Mr David Tshabalala, Mr John Manchidi and Mr Sakhi Xaba of the Electronics Workshop for their assistance with regards to technical help and support. Finally, I would like to thank the Human Research Ethics Committee (HREC) of the University of the Witwatersrand for reviewing the protocol and providing feedback on the study.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Physical disability interrupts normal development on many levels and can present substantial individual and family burdens, as well as major economic consequences. Some physically disabled individuals may experience pain; while others may have difficulty with motor control, with speech impairments etc. Examples of conditions that result in motor disability include cerebral palsy, Spina Bifida and most commonly spinal cord injury (SCI), resulting in the loss of central nervous system (CNS) function. This leads to a state of paralysis of the limbs [7]. The range of resulting clinical manifestations include, but are not limited to, quadriplegia, paraplegia, hemiplegia [8].

In many cases motor disorders are severely incapacitating, and the patient may not be able to perform the most basic of tasks. There is therefore a need to aid patients who suffer from these particular disabilities, in any way possible, to offer them a better quality of life and better social and professional integration.

1.1 Research Objectives

The essential objective is the research and development of a generic assistive platform, capable of accommodating patients suffering from a wide range of acute motor disabilities and with fairly diverse requirements. Methodologies must therefore be established, whereby the design can be made sufficiently flexible, such that it can be programmed to suit these people in terms of their needs and level of motor disability, without re-designing the system for each person.

The approach must be based on current theories of the action mechanism, motor control, and sensory-motor learning. Most importantly the design must be non-invasive and therefore safe for use by the end-user.

1.2 Approach

Conventional rehabilitation must be used to maximize the retained motor function of the disabled individual. Partial or complete transection of the spinal cord, resulting in various motor disabilities, comprises a substantial aspect of this study. The project results in the development and testing of methodologies to assess and characterize motor disabilities, through the design and development of the generic assistive system. Once ethics approval has been obtained, various volunteers are invited to test the system and thereafter requested to answer a series of questions regarding the performance and potential usefulness of the system, where the results obtained are analysed in detail.

1.3 Guide to the Report

This report consists of seven chapters which discuss the research into and design of biomedical technologies which may assist various people suffering from acute motor disabilities.

Chapter 2: Provides some background information into motor disabilities resulting from spinal cord injuries and their classification thereof. Motor control and the various existing technologies are discussed.

Chapter 3: Classifies the type of people for which the system is intended and the control methods which will be adopted.

Chapter 4: Provides some rationale on some of the design choices prior to the commencement of the final design stage.

Chapter 5: Details the design process regarding the "Generic Assistive System" to be used by people suffering from various acute motor disabilities.

Chapter 6: Includes the protocols followed prior to the testing stage, i.e. ethics approval and safety testing of the system. This chapter also details the testing procedure employed and the results obtained. These results are subsequently analysed.

Chapter 7: Reviews and summarizes the work covered in previous chapters, as well as confirming that the contribution has met the objectives outlined in the problem statement. Furthermore it examines the recommendations which shall be required for further research.

Chapter 2

Background Information

Reasonable background knowledge is required prior to the commencement of the design. This ensures that a reasonable knowledge of various motor deficiencies and their associated symptoms is attained. Other important factors include motor control and the use of existing technologies for use by people suffering from various motor disabilities.

2.1 Motor Disabilities

The design being performed is aimed at aiding people suffering from motor disabilities. It is therefore important to provide information on the causes of motor disabilities as well as the different categories of motor disabilities existing today.

2.1.1 Spinal Cord Injury

Frequently motor disabilities arise as a result of spinal cord injuries, which occurs when there is damage to the spinal cord, resulting in a loss of function, such as mobility or feeling [9]. Frequent causes of damage are trauma or disease (Polio, Spina Bifida, etc.). The spinal cord does not have to be severed in order for a loss of functioning to occur. In fact, in most people with SCI, the spinal cord is intact, but the damage to it results in loss of functioning [7]. When the spinal cord is injured, there is an interruption of the messages sent along the nerves and pathways going to and from various parts of the body and the brain. Spinal cord injuries can result in a state of paralysis of the limbs, where paralysis of the legs is called paraplegia and paralysis of the legs and arms is called quadriplegia [7].

The life expectancy of people living with SCIs has steadily increased over the last five decades, and with constantly improving methods of treatment, this trend should continue.

Patients with SCI are initially totally dependent on the people around them and need expert care if they are once again to become independent members of the community [7].

Spinal Cord and Vertebrae

As mentioned, spinal cord injury is the most common cause of motor disabilities. The spinal cord and vertebrae, therefore, need to be discussed briefly and understood before any further discussion is made on the various types of motor disabilities.

The spinal cord is roughly 46 centimetres long and extends from the base of the brain, down the middle of the back, to about the waist. The spinal cord is the major bundle of nerves that carries nerve impulses to and from the brain to the rest of the body. The brain and the spinal cord constitute the *Central Nervous System (CNS)*. Motor and sensory nerves outside the central nervous system constitute the Peripheral Nervous System (PNS), and another diffuse system of nerves that control involuntary functions such as blood pressure and temperature regulation are the Sympathetic (SPS) and Parasympathetic (PNS) Nervous Systems.

The spinal cord is surrounded by rings of bone called vertebrae. These bones constitute the spinal column (back bones). In general, the higher in the spinal column the injury occurs, the more dysfunction a person will experience. There are 30 segments (vertebrae) in the spinal cord: 8 cervical, 12 thoracic, 5 lumbar and 5 sacral. It is evident from the names given, that each of these groups of vertebrae is named according to their location. The eight vertebrae in the neck are called the Cervical Vertebrae. The top vertebra is called C-1; the next is C-2, etc. The twelve vertebrae in the chest are called the Thoracic Vertebrae. The first thoracic vertebra, T-1, is the vertebra where the top rib attaches. These are illustrated in Figure 2.1 below.

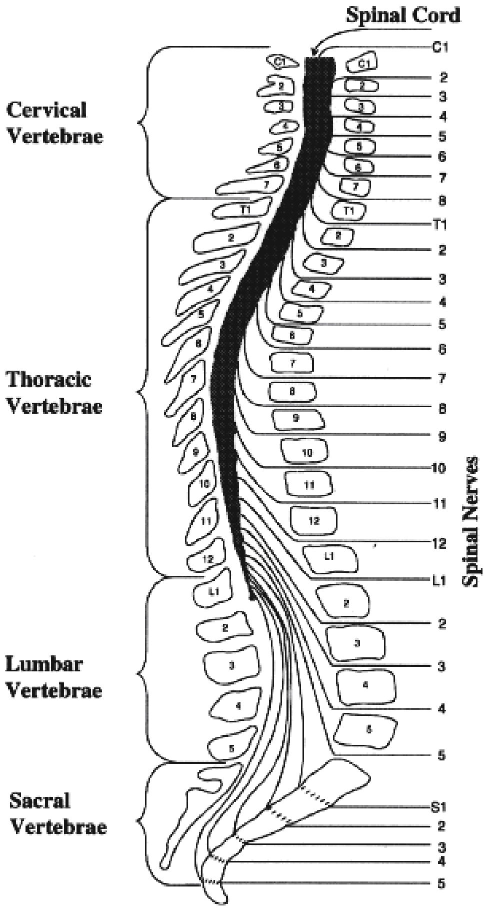


Figure 2.1: The Spinal Cord - The vertebrae are numbered and named according to their location in the spinal column. The spinal nerves are numbered and indicate their corresponding vertebrae [1]

Effects of Spinal Cord Injuries

The level of injury is helpful in predicting what parts of the body might be affected by paralysis and loss of function. The effects of SCI depend on the type and level of the injury. SCI can be divided into two types of injury - *complete* and *incomplete* [9].

Complete Injury A person who has a complete injury has no function below the level of the injury i.e. no sensation and no voluntary movement. Both sides of the body are equally affected.

Incomplete Injury An incomplete injury means that there is some functioning below the primary level of the injury. The disabled individual may be able to move one limb more than another and may experience feeling in parts of the body that cannot be moved, or may have more functioning on one side of the body than the other. With the advances in acute treatment of SCI, incomplete injuries are becoming more common. In incomplete injuries there will be some variation in these prognoses.

Injuries in the **thoracic region** usually affect the chest and the legs and result in **paraplegia**. Injuries to the five Lumbar vertebra (L-1 thru L-5) and similarly to the five Sacral Vertebra (S-1 thru S-5) generally result in some loss of functioning in the hips and legs. Further information regarding the classification of spinal cord injuries, is discussed in Section 2.1.2 [9, 10].

Cervical (neck) injuries usually result in **quadriplegia**. Injuries above the C-4 level may require a ventilator for the person to breathe. C-5 injuries often result in shoulder and biceps control, but no control at the wrist or hand. C-6 injuries generally yield wrist control, but no hand function. Individuals with C-7 and T-1 injuries can straighten their arms but still may have dexterity problems with the hand and fingers. Injuries at the thoracic level and below result in paraplegia, with the hands not affected. At T-1 to T-8 there is most often control of the hands, but poor trunk control as the result of lack of abdominal muscle control. Lower T-injuries (T-9 to T-12) allow good trunk control and good abdominal muscle control. Sitting balance is very good. Lumbar and Sacral injuries yield decreasing control of the hip flexors and legs [7, 9, 10].

Besides a loss of sensation or motor functioning, individuals with SCI also experience other changes. For example, they may experience dysfunction of the bowel and bladder. Very high injuries (C-1, C-2) can result in a loss of many involuntary functions including the ability to breathe, necessitating breathing aids such as mechanical ventilators. Other effects of SCI may include low blood pressure, inability to regulate blood pressure effectively, reduced control of body temperature, inability to sweat below the level of injury, and chronic pain [7, 10].

2.1.2 Classification of the Level of Lesion

Several methods of classification of the level of lesion are in use throughout the world. The system used in this paper, is to give the most *distal* uninvolved segment of the cord together with the *skeletal level*, e.g. for paraplegia (complete or incomplete), below T11, due to fracture/dislocation of vertebrae T9-10. A lesion may not be the same on both sides, e.g. C5L/C7R. To give some idea of the neurological involvement in incomplete lesions, the most distal uninvolved segment is given together with the last segment transmitting any normal function, e.g. incomplete below C5, complete below C7. In this case, some motor power or sensation, supplied by C6 and C7, is present [3].

The degree of paralysis, loss of sensation and the inability to perform activities of daily living demonstrate the severity of an injury, and in order to identify that level of disability, measurements are required in all those areas. The neurological level and the motor, sensory and functional score, for a patient, are calculated by utilising "The international standard for neurological and functional classification of spinal cord injury" (refer to Figure A.2 of Appendix A) [3]. The neurological levels are determined by examination of the following:

- A key sensory point within 28 dermatomes on each side of the body.
- A key muscle within each of 10 myotomes on each side of the body. The sensation and motor power present are quantified, giving a final numerical score.

This is achieved by using the *American Spinal Cord Injuries Impairment (ASIA) scale* [5] (Table 2.1) to grade impairment of sensation and motor power, and the *Functional Independence Measure (FIM)* (Figure A.1 in Appendix A) to measure disability and grade function. The ASIA scale (Table 2.1) is based on the Frankel scale (1969). The letters A-E are used to denote **degrees of impairment**. The FIM, as its name states, is devised to measure function for each disability. Each area of function is evaluated in terms of independence using a seven point scale. A total score from all these measures is calculated each time an assessment is carried out and progress can be readily seen [2].

In addition to these measures, the *Ashworth Scale* is used to measure muscle spasticity as shown in Table 2.2 [6].

Table 2.1: The ASIA Scale [5]

Grade	Description
A	Complete: No motor or sensory function is preserved in the sacral segments S4-S5.
B	Incomplete: Sensory but not motor function is preserved below the neurological level, and includes the sacral segments S4-S5.
C	Incomplete: Motor function is preserved below the neurological level, and the majority of key muscles below the neurological level have a muscle grade less than 3 on the *Ashworth scale*.
D	Incomplete: Motor function is preserved below the neurological level and at least half of key muscles below the neurological level have a muscle grade of 3 or more on the *Ashworth scale*.
E	Normal: Motor and sensory function is normal.

Table 2.2: The Ashworth Scale [6]

Grade	Description
0	Normal muscle tone.
1	Slight increase in the muscle tone, ‘catch’ when limb is moved.
2	More marked increase in muscle tone, but limb easily flexed.
3	Considerable increase in muscle tone.
4	Limb rigid in flexion or extension.

Looking at these two tables collectively, one can deduce that a fully able-bodied person would fit into category E on the ASIA scale, with a muscle grade greater than 3 on the Ashworth scale, in accordance with research performed by Damiano et al. [6]. A severely disabled person would possibly fit into category A on the ASIA scale, with a muscle grade of 0 spasticity on the Ashworth scale.

2.2 Motor Control

The various types of movement must be understood in order to grasp the concepts being discussed, relating to motor control. In motor behaviour and control various terms are used to describe the tasks and movements. These terms must be firstly understood to communicate effectively about the field. The second reason is that relationship between certain independent and dependent variables is often different for one kind of task or behaviour as compared to another. Classification is therefore done in order to ensure that the laws of motor control are not more complicated than they should be.

2.2.1 Open versus Closed Movements

One way to classify skills is in terms of the extent to which the environment is predictable during the performance [11].

Open Movements

Open skills are those for which the environment is constantly changing, so that the performer can not effectively plan the response. Examples of these include boxing and wrestling, which are both activities for which the opponent's behaviour is unpredictable. Success in open skills is determined by the extent to which the performer is successful in adapting the behaviour to the changing environment [11].

Closed Movements

Closed skills are those for which the environment is predictable and thus stable. Examples of these include archery and playing snooker. A predictable environment can arise when the environment is variable, but the changes are either predictable and/or have been learned as a result of practice; examples are juggling or working on a production line [11].

2.2.2 Controlling Externally Powered Devices

According to research conducted by Kumar V et al., there are primarily two main approaches to controlling externally powered devices [12]:

1. The activation of prosthetic joints with the aid of myoelectric signals (EMG) from intact musculature (muscles).
2. Control by displacement signals obtained from body movements.

Electromyographic Control (EMG)

Electromyographic (EMG), or myoelectric, control uses *the electric signal due to depolarization of the cell membrane of muscle fibers during contraction* [13]. The signal is sensed through electrodes, which thereafter amplifies, processes the signal, which can subsequently be used as an input to the actuators (sensors).

The main drawback of EMG control [14] is its open loop character (open movement) due to the absence of position proprioception[1] i.e. the environment is unpredictable because its constantly changing. Subsequently, the user would be required to adapt to this change (refer to Section 2.2.1).

Because synergistic muscle groups are responsible for activating the joints of the natural limb, any "natural" control scheme must take as input various EMG signals from the shoulder and chest. The key technical challenge is to interpret the patterns of EMG and to match these patterns to specific movements. However, myoelectric signals are often inconsistent and the reliability and the benefits of such an approach is questionable [12].

Body Movement Control

Visual and auditory feedback are slower, less automated, and less programmed than the normal proprioceptive feedback [12]. With EMG control, the amputee relies on vision and exteroceptive feedback to determine how well his intentions have been executed by the prosthesis. In contrast, if the movement of the device is physically linked to body movement, not only does it **reduce demands on system response**, but it also **provides proprioceptive feedback**, and hence, a closed-loop system i.e. those requiring closed skills (as described in Section 2.2.1).

In order for the design to be successful, a basic understanding motor control must be understood. The design will therefore take most of its ideas from the body's natural motor control mechanism. The term motor control refers to *the study of postures and movements and also to the functions of the mind and body that govern posture and movement*. In this context *posture* means the static position of any part of the body, rather than the description of a sitting or standing position of the whole body, as its common usage suggests [15]. Postures of the limbs, trunk and the whole body are maintained by muscular effort. *Movements* are *the transitions from one posture to another* [15]. Postures and movements can be assumed consciously or as automatic adjustments. This distinguishes *reflex actions* from *voluntary actions*. Reflexes are best described as *responses evoked with great probability by particular stimuli* [15]. Humans are born with only very rudimentary movement, whereby all other movement must be learnt by active practice during the early stages of a child's development to adulthood.

Body movement control can be divided into **discrete**, **continuous** and **serial** inputs/movements.

[1]The ability to sense the position and location and orientation and movement of the body and its parts [14]

Discrete Movements: Discrete skills are those with a recognizable beginning and end. Kicking, throwing, striking a match and shifting the gears of a motor vehicle are some typical examples. The finishing point is defined by the skill in question, not arbitrarily by when an observer ceased examining it, as would be the case with swimming and jogging. Another characteristic of discrete skills is that they are usually rapid, seldom requiring more than a second to complete [11].

Discrete signals/movements that are effected by body movements include switches operated by digits (eg. touch switches), shoulder displacement switches, feet and shoulder switches, and various body switches. The disadvantage of discrete control is that it usually relates the duration of switch closure to the distance moved. This relationship does not conform to natural modes of control and coordination of multi-joint movement becomes difficult [12].

Continuous Movements: This is defined as having no recognizable beginning and end, with behaviour continuing until the response is arbitrarily stopped. Examples are swimming, running and steering a motor vehicle. Continuous tasks tend to have longer movement times than do the discrete tasks [11].

A continuous signal/movement, as opposed to a discrete signal/movement, offers **superior control**. For example, a person who uses the bi-scapular movement of the shoulders to flex and extend the elbow of a conventional cable-operated arm has a sense of being linked to the arm. The user exerts a force on the cable which moves the artificial joint. The movement at the joint is linked to the amount of movement permitted at the shoulder. Any sensor which utilises continuous movement acts as an extension of the user and provides force and position information to the user [12].

Serial Movements: There are a number of skills which are neither discrete nor continuous, as they seem to be made of a series of discrete actions acting over a certain time period. Examples of this are starting a motor vehicle, filling and lighting a pipe, as well as various other tasks performed on production lines in industry. Such tasks may take several seconds to complete, and may thus appear to be continuous, although these tasks might also have discrete beginnings and ends. Serial tasks can be loosely though of as a number of ordered discrete tasks which act together over a particular period of time [11].

2.3 Functional Electrical Stimulation (FES)

There are two available approaches for the restoration of movements in humans with paralysis. These include [16]:

1. **The functional activation of paralysed muscles:** This is called Functional Electrical Stimulation (FES).
2. **The combined usage of an FES and a mechanical orthosis:** This is called Hybrid Assistive System (HAS). This deals with prosthesis related to externally controlled and powered artificial limbs, with specific emphasis on myoelectric controlled devices.

Major limitations in daily application relate to insufficiently adaptive and robust control methods. The complexity of central nervous system (CNS) control and the interface between voluntary control and external artificial control are still challenging unanswered questions. Hierarchical control methods, combining symbolic models at the highest level and analytic models at lower levels are needed for further progress.

This approach was considered for the design, but one of the design criteria is to ensure that the final design is as non-invasive as possible. FES is generally quite an **invasive** approach and is also out of the scope of the required design. Resultantly this approach was not considered for the purposes of the research [16].

2.4 Existing Technologies

While conventional rehabilitation maximizes the retained function of the disabled individual, few interventions increase control of the paralyzed or paretic functions.

2.4.1 Single-switch Access

People who have very limited mobility use this type of device. If a person can move only the head, for example, a switch could be placed to the side of the head that would allow the person to click it with head movements. This clicking action is usually interpreted by special software on the computer, allowing the user to navigate through the operating system, Web pages, and other environments. Some software facilitates the typing of words by using an auto-complete feature that tries to guess what the person is typing, and allowing the person to choose between the words that it guesses [17].

2.4.2 Sip-and-puff switch

Similar in functionality to the single switch described above, sip and puff switches are able to interpret the user's breath actions as on/off signals, and can be used for a variety of purposes, from controlling a wheelchair to navigating a computer. The hardware can be combined with software that extends the functionality of this simple device for more sophisticated applications [17].

2.4.3 Oversized trackball mouse

A trackball mouse is not necessarily an assistive technology–some people without disabilities simply prefer it to the standard mouse–but it is often easier for a person with a motor disability to operate than a standard mouse. Someone may, for example, use a trackball mouse in conjunction with a head wand or mouth stick. It is relatively easy to manipulate a trackball with these devices and much harder to manipulate a standard mouse. Someone with tremors in the hands may also find this kind of mouse more useful because once the person moves the mouse cursor to the right location, there is less danger of accidentally moving the cursor while trying to click on the mouse button. A person with tremors in the hands could also manipulate the trackball mouse with a foot, if there is enough motor control in the feet [17].

2.4.4 Adaptive Keyboard

In cases where a person does not have reliable muscle control in the hands for precision movements, an adaptive keyboard can be useful. Some adaptive keyboards have raised areas in between the keys, rather than lowered areas, to allow the person to first place the hand down on the keyboard, then slide the finger into the correct key. A person with tremors, or spastic movements could benefit from this type of keyboard. Keyboard overlays are also available as an adaptation to standard keyboards, which achieve the same results. In some cases, adaptive keyboards come with specialized software with word-completion technology, allowing the person to type with fewer keystrokes, since typing can be rather laborious and slow otherwise [17].

2.4.5 Eye Tracking

Eye tracking devices can be a powerful alternative for individuals with no control, or only limited control, over their hand movements. The device follows the movement of the eyes and allows the person to navigate through the Web with only eye movements. Special software allows the person to type, and may include word-completion technology to speed up the process. These systems can be expensive usually in the thousands of South African Rands, so they are less common than the less sophisticated devices, such as mouth sticks and head wands [17].

2.4.6 Voice Recognition

Another alternative is to install software that allows a person to control the computer by speaking. This assumes that the person has a voice that is easy to understand. Some people with motor disabilities–those with cerebral palsy in particular–may have a difficult time speaking in a way that the software can understand them, since the muscles that control the voice are slow to respond, and speech is often slurred, despite the fact that these people do not have any slowness in their mental capacity [17].

Chapter 3

Subject Classification

It is important to characterise the motor disabilities of the potential end-users of the generic assistive device before designing the system.

3.1 Types of Motor Disabilities

The system has being designed for people suffering from various *localised and/or severe motor deficiencies* (especially those resulting from spinal cord injuries (SCIs)), who have experiences a *loss of central nervous system function*. These disabled people have undergone full paralysis of their legs, but may have limited use of their hands and arms (as is the case with quadriplegics).

Other types of motor deficiencies include: motor neuron disorder and tetraplegia. Motor neuron disorder is, conversely, different to SCI related motor loss as it is a degenerative disease; whereby the level of motor deficiency and hence paralysis is dependent on the how far the disease has spread throughout the body.

3.1.1 Level of Lesion Classification

These subjects typically have a lesion level of C6/7 (complete or incomplete) and above[1]. C6/7 injuries generally yields some wrist control, where the subject may be capable of straighten their arms but may still have dexterity problems with their hands and fingers (little to no movement of the fingers and hands). Higher level C-5 injuries often result in shoulder and bicep control, but no wrist or hand control; where injuries above the C-4 level

[1]The higher the level of lesion the more acute the motor deficiency becomes, as is the case with C/5 injuries and above

may require a ventilator for the person to breathe (refer to section 2.1.1). These subjects would typically have a *degree of impairment* of A on the ASIA scale (Table 2.1), and a *muscle grade* of 0 on the Ashworth scale (Table 2.2) (refer to section 2.1.2).

3.2 Body Movement Control

Of the types of control of externally powered devices, the design would involve the **control by displacement signals obtained from body movements** [12], since the system is being designed for people who have motor disabilities.

As mentioned in Section 2.2.2, these users are best suited to using a closed-looped system, i.e. the end-user would require closed skills, whereby the environment is predictable and thus stable.

3.3 Motor Response Classification

The motor response classification is dependant on the type and degree of motor deficiency, as the acuteness and locality of the various subjects' disabilities differs.

As mentioned in Section 2.2.1, open looped skills would not be favourable, as the environment is constantly changing, and the only way the user can compensate is by adapting quickly to it which, in the case of people suffering from more acute motor disabilities, is very unlikely and therefore impractical.

Consequently, the motor response classifications, being catered for in this design, are **closed voluntary movements** (Section 2.2.1), which may be **discrete**, **continuous** or **serial** in nature (Section 2.2.2). Closed voluntary movements are intentional movements performed in a predictable and stable environment e.g. tapping a touch pad with one's finger. Sharp movements of the fingers or wrists would constitute a discrete action (using a touch sensor), whereas breathing in and out would form a continuous action (using a sip-and puff switch). Serial actions would be utilised when a subject wishes to execute a series of discrete actions, such as moving his/her head intermittently from side to side [11].

Chapter 4

Design Rationale

Before the design process could commence, various principles need to be discussed, relating to the research and design objectives. These principles form the central basis for the design stages to follow.

4.1 Design Objectives

In order to satisfy the research objectives, the system must capable of detecting and interpreting any **residual movement** that the subject may have, thereby allowing the user to perform routine tasks at their **own will**. For example, a subject who has a limited use of their hand and finger movements may use this system to detect his/her finger movements via a touch sensor, to instruct the system to switch on/off lights, control a wheelchair or call for assistance.

Consequently, the platform must be programmable, which would ensure that it suits the various subjects' unique range of abilities, thereby satisfying their diverse requirements. This would certify that the platform is truly generic in nature. This forms the central basis for the design stages to follow.

4.2 Basic Considerations in Motor Behaviour and Control Systems

In the field of motor behaviour and control, the problem of measurement can be approached in two different ways. First there is the requirement to quantifying the nature of the actual movements that the person/patient has made. Secondly, there is the requirement to how well the movement was achieved for some environmental goal that was inherent in the task e.g. whether sufficient pressure was applied to the input of a transducer. A fundamental issue is how motor behaviour is measured. Important features of an effective assistive system are [11]:

4.2.1 Reliability

This refers to the extent to which the measurement is repeatable under similar conditions. A lack of reliability can result in certain technological error.

4.2.2 Objectivity

A measurement system is *objective* if the two observers evaluating the same performance arrive at the same (or similar) measurements. Measurement of the distance that a javelin is thrown is objective as it yields similar results for various measurements taken. By comparison, evaluation of the performance of a Figure skater is more subjective as no physical measurement is taken.

4.2.3 Validity

Validity is the extent to which the test measures what the researcher intends it to measure.

4.2.4 Minimising Energy Required

This involves a minimum increase of the energy rate with respect to the able-bodied subjects performing the same task.

4.2.5 Minimising Disruption

This involves the minimum disruption of normal activities when utilising the system.

4.2.6 Safety

Safety forms one of the most important characteristics, which all measurement systems must employ.

4.2.7 Cosmetics

It is important that the user finds the device aesthetically pleasing. This can be thought of as "form following function".

4.2.8 Practicality

Practicality is concerned with actual use rather than theoretical possibilities of the device.

These features form the basis of the testing phase of the research, which will be discussed later in the document.

4.3 General Concerns

When detecting and recording the input signal, there are certain issues of concern that influence the fidelity of the signal. These are described below.

4.3.1 The Signal to Noise Ratio (SNR)

The first issue, present in all measurement systems is **noise**. In general, noise is defined as *electrical signals that are not part of the wanted input signal* [18]. It is important to reduce the noise, while increasing the fidelity of the wanted signal. This is determined by the signal-to-noise ratio (SNR), which is defined as the*the ratio of the signal power to the power of the combined noise sources*. This is specified in decibel (dB) units [18]. The SNR[1] can be improved by digital filtering , which forms part of the design process and is discussed in detail in Section B.3.2 of Appendix B.

[1]It is important to note that the combined sources of the noise components will be amplified along with the signal as it passes through the amplifier. Consequently, it is important to analyse the signal to noise ratio (SNR) rather than the level of the combined noise sources.

4.3.2 Ambient (Transmission Line) Noise

Power line radiation (50 Hz) is a dominant source of electrical noise. It is possible to design devices that have a notch-filter at this frequency. Theoretically, this type of filter would only remove the unwanted power line frequency; however, practical implementations also remove portions of the adjacent frequency components. Because the dominant energy of e.g. EMG (biopotential) signals are located in the 50-100 Hz range, the use of notch filters is not advisable when there are alternative methods of dealing with the power line radiation [4, 16]. The risk of distorted input signals also arises when a notch filter is used and this may affect the eventual readings taken. A simpler and far more accurate method would be to include some form of **digital filtering** after the A/D stage, which would effectively remove the unwanted transmission noise using a DSP (digital signal processor).

4.3.3 Signal Distortion

Signal distortion is prevalent in measurement systems [18, 16]. One factor which may cause this is **stimulus/motion artifact**, which is a very common source of possible error occurring in biomedically related systems, such as the one being designed [2]. There are two main sources of motion artifact [19]:

1. From the interference between the detection surface of the electrodes of biopotential sensors and the skin.
2. From movement of the cable connecting the electrode to the amplifier.

Both of these sources can be essentially reduced by proper design of the electronics circuitry, i.e. filtering the signal. Such filtering, typically high-pass, however, can severely distort the biopotential signal being measured [16]. Digital-signal-processing (DSP) would, therefore, be necessary to identify and delete the artifact from the final display and processing. The electrical signals of both noise sources have most of their energy in the frequency range from 0 to 20 Hz, which incidentally coincides with the fact that essentially all **measured body movements** are contained within the frequency components **below 20 Hz** (99% of the energy in gait is contained below 15 Hz) [20, 21].

For the particular design, no biopotential sensors have been used for the input stage, but the generic platform must allow for the use of these types of sensors. It must be noted that

[2]This system needs to be designed for use with bioelectric sensors, e.g. EMG (electromyography), even though this design does not include the use of these particular sensors

stimulus artifact may vaguely be present when utilising input sensors such as the dual-axis accelerometer and touch sensor, which may come into contact with the skin (discussed in Section 4.5.

4.3.4 Inherent Instability of the Signal

The amplitude of signals is quasi-random in nature. The frequency components between 0 and 20 Hz are particularly unstable because they are affected by the quasi-random nature of the firing rate of the motor units which, in most conditions, fire in this frequency region (if biopotential sensors were utilised). Because of the unstable nature of these components of the signal, it is advisable to consider them as unwanted noise and remove them from the signal [18].

As mentioned already, normal human motion does not exceed 20 Hz, and it is for this reason that it can become difficult to detect voluntary motor action unit potentials over unwanted noise.

4.4 Real-time Systems

The system being employed must be capable of continuous, real-time measurement, to process the various input signals as well as to perform the required output functions. The characteristics required of all real-time systems are described as follows [22]:

- **The system must be extremely reliable and safe:** Embedded systems typically control the environment in which they operate. Failure to operate can result in loss of life, damage to the environment or financial loss.
- **Guaranteed response times:** The worst case response times must be confidently predicted. Efficiency is important but predictability is essential.

4.5 Input Sensor Selection

Suitable sensors have been chosen for the system on the basis that they are **capable of detecting the retained motor functions of diverse disabled people**. The sensor choice has also been based on the fact that they will be used on the upper extremities of the disabled subjects' bodies, as motor function may only be possible in this region, for those people suffering with severe motor disabilities (as is the case in quadriplegics).

For this study, the input sensors have been restricted to the following:

1. **Movement sensors:** The Analog Device™ ADXL210 Dual-Axis Accelerometer [23].
2. **Touch sensors:** The QProx™ QT160D 6-Key Charge-Transfer QTouch™ Sensor IC [24].
3. **Blink/Wink sensor:** The Tash™ Self-Calibrating Auditory Tone Infrared (SCATIR) Switch [25].

These sensors must be positioned in such as way as to **efficiently detect residual movements** that the subject may have. The placement of the sensors, for the purpose of this study, will be restricted to sensing arm (dual-axis accelerometer) or **finger movements** (touch sensors), **head and neck movements** (accelerometer) and **blink/wink** and/or **eyebrow movements** (SCATIR switch).

The selected input devices are all examples of **passive sensors**, as they do not add energy as part of the measurement process; i.e. all measurements are converted from physical to electrical quantities. These devices were chosen primarily on the basis that they are **non-invasive** and therefore **safe** to use, which consequently meets the required design objectives. Further detailed information regarding these sensors is discussed in Section B.2 of Appendix B. The design and application of these sensors is discussed in Chapter 5, which follows.

4.6 Instrumentation Requirements

Assuming that suitable input devices have been chosen for the application, the primary hardware concerns for reading the input signal become the **sampling rate** and **signal conditioning**. These are discussed in further detail in Section 5.3.2 in the following chapter.

Chapter 5

Design and Implementation

The following chapter deals with the procedures followed in the design and implementation of the final system.

The design stages include:1. **the power supply and conditioning stage**, 2. **the input (sensor) stage**, 3. **the signal selection and signal conditioning stage**, 4. **data acquisition (DAQ) and control stage**, 5. **the software application stage**, which includes the digital signal processing (DSP), 6. **the output stage** and finally 7. **the design implementation and functionality testing stage**.

The input stage, coupled with the signal selection and conditioning stage are illustrated in Figure 5.1 [18], [26], and [4].

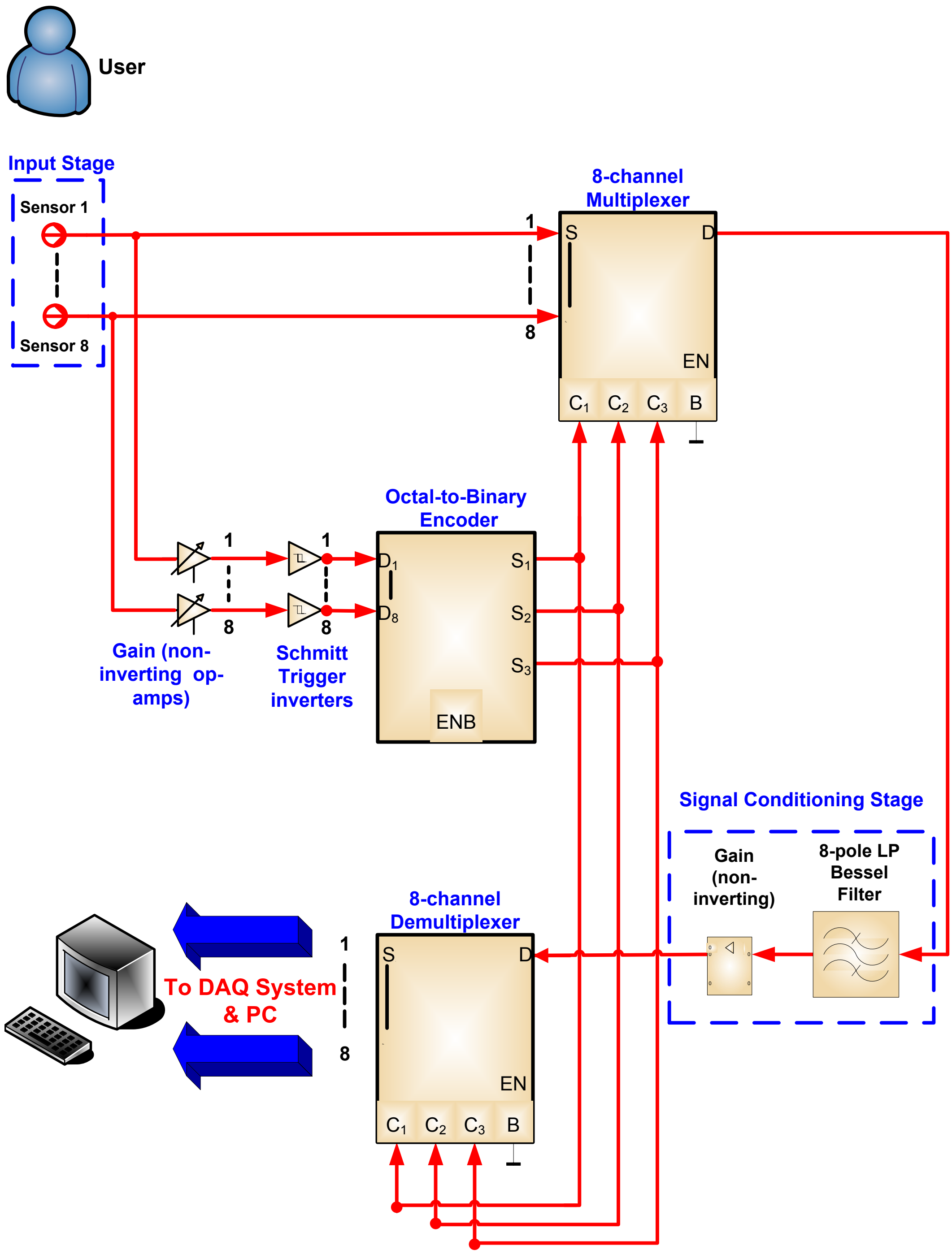


Figure 5.1: Input, signal selection and signal conditioning stages

5.1 Stage 1 - Power Supply and Conditioning

5.1.1 Input Hardware Power Supply

Power has been supplied to the hardware portion of the system (i.e. all input sensors and hardware occur.

Voltage Regulation

Voltage regulation is performed on the $\pm 12V$ input battery DC voltage and is regulated to $+5V$ and $-5V$ using the LM7805 and LM7905 [27, 28] voltage regulators respectively. Protection is provided via $2.2\mu F$ and $1.0\mu F$ capacitors on the input and outputs of the voltage regulators respectively. The 1N4148 zener diodes are used for absolute stability of the output signal and prevent spikes in the output voltage.

The full CAM schematic and board diagrams of the power supply rectifier circuit are illustrated in detail in Figures F.1 and F.2 respectively in Appendix F.

5.1.2 Data Acquisition Card and PC Power Supply

Power has been provided to the PC and consequently the NI™ PCI-6221 DAQ card via the 220 V (AC) mains voltage supply. To ensure absolute safety a 220 - 220 V isolation voltage transformer has been attached between the mains supply and the PC. This isolates the secondary side for improved electrical safety.

5.2 Stage 2 - Input Stage (Input Sensor Design)

The process involved in the design and implementation of the selected input transducers is discussed in this stage. The transducers that have been chosen for the design are an ADXL250 dual-axis accelerometer, a QT160 6-channel touch sensor and a SCATIR blink/wink switch.

5.2.1 The ADXL250 Dual-axis Accelerometer

Design Choice and Operation

A dual-axis ADXL250 accelerometer is used as the primary input detection transducer. This particular transducer was chosen is it detects acceleration and tilt and is therefore capable of detecting residual movement provided by a subject's active muscle regions e.g. wrist, head, neck and shoulder regions (pg. 9 of [23]). The ADXL250 offers lower noise, a wider dynamic range (i.e. more values can be taken), reduced power consumption and improved zero g bias drift when compared to previous models and most importantly it is relatively cheap, only costing in the region of R50.

The signal-to-noise ratio (SNR) is $80dB$, which is very similar to the SNR of the 16-bit analogue-to-digital converter (ADC), which has an $SNR = 98dB$ (discussed later in Section 5.3.2). The ADXL250 allows for a resolution of signals as low as 10mg, yet still providing a $\pm 50g$ full-scale range. The two sensitive axes (X and Y) are orthogonal (90^0) to each other [23].

This transducer may be used for all types of motor response i.e. continuous, discrete and serial actions (refer to Section 2.2.2 of Appendix 2). The most probable motor response would be discrete in nature, as the user may wish to tilt his/her wrist or head or provide a sudden "jilt" of acceleration in order to perform a specific action. Further information is discussed in Section B.2.1 of Appendix B.

Signal Pre-conditioning

Pre-conditioning was required for this particular sensor. Pre-conditioning was performed by utilising 1) unity gain buffers and by introducing a 2) DC offset.

Unity Gain Buffer: The X and Y analogue outputs of the accelerometer have a high output impedances of 32 $k\Omega$ per channel and are therefore not designed to drive a load directly ([23] pg. 10). Op-amp followers were therefore required to buffer each of these analogue outputs, therefore ensuring for better impedance matching.

DC Offset: The X and Y analogue outputs of the accelerometer have an offset of $V_{DD}/2 = 2.5V$. The problem with this is that the full range of voltages may not be achieved once amplification has been performed, as saturation may occur at the output of the amplifier, just below V_{DD} ($\approx 4.3V$). To ensure that this does not occur, a -2.5 V DC signal is added to the offset of 2.5 V using an inverted summer. This ensures that the offset is $\approx 0V$. A voltage divider was used to provide the -2.5 V and is finely tuned using a variable resistor. This variable resistor is required for the adjustment of the output in order to cater for zero g drift.

The full CAM schematic and board diagrams are illustrated in detail in Figures F.3 and F.4 respectively in Appendix F.

5.2.2 Six-Channel Touch Sensor

Design Choice and Operation

The *QT160* 6-channel touch sensor was chosen as it is a *self-contained digital controller, which is capable of detecting close-proximity or touch from up to 6 electrodes* (pg. 1 [24]). This provides very useful in the detection of a subject's slight touch, especially considering that many quadriplegics may have slight movement of their fingers.

The touch sensor was chosen over the use of a cantilever strain-gauge as it is far cheaper and provides for more accurate readings, by detecting touch (which is capacitive in nature) rather than strain (which is resistive in nature). The QT160 touch sensor IC can also easily be set for a variety of different sensitivity levels and can therefore be customised to the level of touch required. Another advantage of this sensor is that the user has the option of selectable toggle mode or DC outputs . This sensor may be used for both *discrete* and *serial movements* (refer to Section 2.2.2 of Appendix 2).

As mentioned in Section B.2.1 of Appendix B, the sensitivity may be altered to suit various application and situations on a channel-by-channel basis. The easiest way to change the sensitivity is to decrease or increase C_s (between $10\mu F$ and $47\mu F$), thereby decreasing or increasing the sensitivity respectively.

To ensure that the sensitivity is customisable to a particular subject's touch level, plug-in sockets have been incorporated into the design, thereby allowing the user to plug different values of C_s into the circuit.

Signal Pre-conditioning

No further signal pre-conditioning needed to be performed as the *QT160* IC is a self-contained digital controller.

The 6-channel touch sensor circuit was designed and built with the aid of Figure 1-1 of *QT160 Touch Sensor Datasheet* [24]. The full CAM schematic and board diagrams are illustrated in detail in Figures F.5 and F.6 respectively in Appendix F.

5.2.3 The Self-Calibrating Auditory Tone Infrared (SCATIR) Switch

Design Choice and Operation

The SCATIR blink/wink switch is the third sensor chosen for the design. This device is a stand-alone device, which has been designed and tested already and was borrowed for the research project as a device to be used on people with specifically limited levels of movement i.e. those who can only move there facial muscles and eyes. This sensor could also be used in conjunction with the other sensors, where movement is less limited, to provide more functionality to the user. Detailed information regarding this sensor is described in Section B.2.1 of Appendix B.

5.3 Stage 3 - Signal Selection and Signal Conditioning

Signals from sensors do not usually have suitable characteristics for display, recording, transmission, or further processing. The signal voltages output from each sensor are usually in the range of a few mV and therefore lack the amplitude, power level, or bandwidth required for the analogue-to-digital conversion (ADC) stage. These signals generally carry superimposed interference that masks the desired information [26].

The signal conditioning design involves the detection of the residual motor function (user input). For this stage the muscular activity is detected from the user's e.g. weak hand and forearm muscles, as in the case of many quadriplegics and tetraplegics. Before this input may be monitored it must be passed through the signal conditioning circuit. The signal conditioning circuit, therefore, essentially "prepares" the signal for the A/D stage while also improving the fidelity of the signal.

The design rationale for the signal conditioning unit was to design a circuit capable of accepting up to 8 input sensors and pass them through a single anti-aliasing filter and gain stage. The signal selection and conditioning stage is illustrated in Figure 5.1.

5.3.1 Signal Selection

Signal selection is a unique method incorporated into the design, as it utilises a single signal conditioning unit, whilst still allowing for multiple input channels (8 channels in this case) connected to various input sensors. Multiple signal conditioning units could have been chosen for the design, each being associated with a single input channel. The problem associated with this is that is tends to increase the size, cost and complexity of the circuit dramatically. It was therefore decided that a "generic" signal conditioning unit, which could cater for all of the various input sensors, would best be suited to the design. The rationale of the selection is based on passing only the "active" or operational channel's signal though the signal conditioning unit at any one moment in time.

Signal selection was accomplished by firstly passing each of the sensor inputs into 8 separate gain and Schmitt trigger inverter sections. The sensor inputs, passed through the gain sections, are also passed into the inputs ports of the CD4051 8-channel multiplexer [29]. The Schmitt triggers have positive and negative going thresholds between $0.8V$ and $1.6V$ respectively. The gain stages ensure that the sensor input voltages operate above and below these threshold values, thereby ensuring that the input signal can be triggered. This ensures that the output voltage is in the 0-5 V range or active low-high logic levels. It must be noted that the output of the Schmitt trigger goes "low" when the input level exceeds $1.6V$ and "high" when the signal drops below $0.8V$ [30].

The Schmitt trigger outputs thereafter proceed to go through an octal-to-binary priority encoder. This encoder encodes the 8 signals into 3 binary outputs, with an active low level (≈ 0) resulting in a unique binary output. A priority encoder was chosen as it ensures that if two signals are selected simultaneously, the signal with highest priority is chosen. These three binary outputs are "tied" to the control lines of the 8-channel multiplexer, thereby controlling which input passes through the anti-aliasing and gain section (signal conditioning stage). The signal conditioning process is discussed in further detail in Section 5.3.2 which follows. The input signal is thereafter de-multiplexed, resulting in 8 separate outputs, which thereafter leads to the data acquisition stage. This ensures that each input results in a specific output. To achieve this, the control lines of the multiplexer and the demultiplexer are tied together, ensuring that the states of the three control lines are the same for each input and output [30].

The only drawback of this design is that the input signals are output as positive signals as the Schmitt rigger thresholds are between 0.8 - 1.6 V and therefore don't trigger below the $0.8V$ threshold, i.e. any negative voltage would therefore be output as an active high at the output of the Schmitt trigger inverter. Zener diodes have been placed on the inputs of the

Schmitt triggers, which effectively “clamp” the input signals to voltages at and above 0 V, thereby ensuring that only the positive voltage levels of the various signals pass through the Schmitt triggers. This, however, does fit the specifications as most sensors used to detect the retained motor function are positive going signals. The signal selection process is illustrated in Figure 5.1.

5.3.2 Signal Conditioning

The signal conditioning circuit is illustrated in Figure 5.2 below [26].

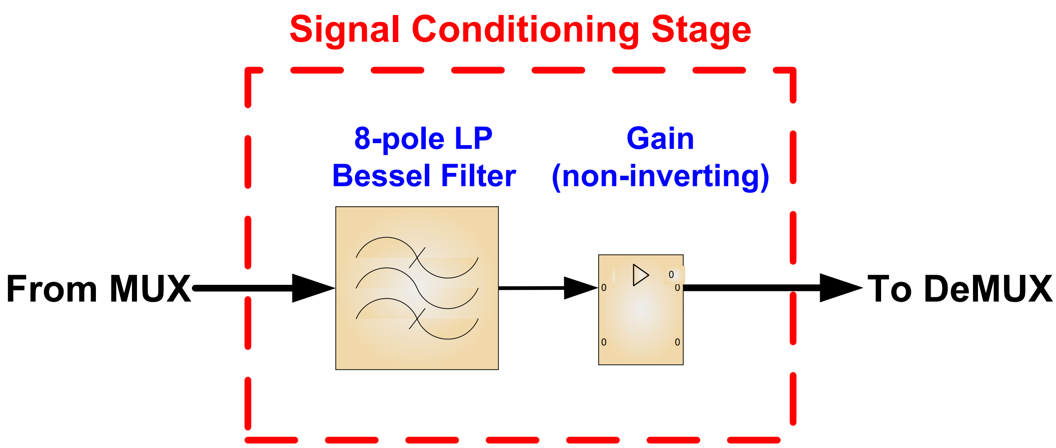


Figure 5.2: Signal Conditioning Circuit

Antialiasing Filter Design (Averaging)

It is desirable to obtain a signal that contains the maximum amount of information from the input device signal and the minimum amount of contamination from electrical noise. Thus, the maximization of the signal-to-noise ratio (SNR) should be done with minimal distortion to the input device signal. Therefore, it is important that any detecting and recording device process the signal linearly. It is very important to avoid saturation of the signal, and therefore distortion of the signal peaks. Also no unnecessary filtering should be performed [26].

Anti-aliasing filter choice: A comparison was done between the various types of filters available to perform the anti-aliasing function. Butterworth filters have fairly good amplitude and transient characteristics. The Chebyshev family of filters offer increased selectivity but poor transient behaviour. Neither approximation to an ideal filter is directed toward obtaining a constant delay in the passband [31], [4]. The Bessel Filter, however, has a transfer which has been optimised to obtain a linear phase, i.e. it has a maximally flat delay. The step response has essentially no overshoot or ringing and the impulse response lacks oscillatory behaviour i.e. there is no ripple in the pass or stopbands. The Bessel filter is used where transient properties are a major consideration as the steepness of the roll-off is gradual.

The Bessel filter was therefore chosen, as its distinguishing characteristic is the near constant group delay throughout the passband of the low pass filter [31], [4]. This is a very important characteristic of the filter as any linear phase shift in the input signal at each frequency would otherwise result in a phase shift in the output signal at each frequency. This would consequently cause a distortion in the output waveform of the filter. The only draw-back of this filter type is that its frequency response is far less selective than the other filter types i.e. there is a gradual roll-off, but it is sufficient for this application as it is only used for ant-aliasing and not for improving the S/N ratio (noise rejection) or discriminating between noise and the input signal resulting from retained motor action. Also the roll-off has been improved by increasing the order of the filter to an 8-pole filter. This is discussed in further detail below and in Appendix C.

Anti-aliasing design rationale: The sampling rate of the measured analogue signal, taken from the patient's movements using the input device e.g. strain gauge accelerometer, must be sufficient to capture the desired phenomenon, and also to satisfy the Nyquist rule i.e. the sampling rate must be at least twice that of the highest frequency component of the signal to be measured. Conversely, it is also important to use anti-aliasing filters to remove high frequency components that are more than one-half the sampling rate in order to eliminate false characteristics in the reconstructed waveform.

This first important point to consider is that normal human movement does not exceed $20Hz$ [17]. To satisfy the Nyquist Theorem, $f_{sb} = 2 \times f_m$, where f_m is the highest frequency component of interest of the input signal (-3 dB corner frequency) i.e. $20Hz$ and f_{sb} is the stopband frequency, where $f_{sb} = 40Hz$. To ensure that no aliasing results, the cut-off frequency has been oversampled and therefore increased to $f_c = 1kHz$. Oversampling is an important technique employed when performing low-pass filtering as is used as it a) *improves the ADC resolution* and b) *increases the SNR*. Further information is provided in Section C.2.4 of Appendix C.

Appendix C discusses the design of the filter further in detail. It is important to note that if the sample rate were too high, a faster and therefore more expensive ADC would be required to process the higher number of samples. The filter chosen is as an 8^{th} order ($n = 8$) Bessel filter, which satisfies the various criteria. The active filter has been designed, with four 2^{nd} order ($n = 2$) sections and is illustrated in Figure C.1 of Appendix C.

Frequency Response Testing: The filter response was tested, by introducing a 100 mV (peak) sinusoidal signal into the system, via a signal generator and subsequently recording the output signal peak voltages for the $10Hz - 3.3kHz$ frequency range. The gain was determined using the following equation:

$$Gain(dB) = log\frac{V_{out}}{V_{in}} \quad (5.1)$$

The gain was thereafter normalised thereafter by:

$$Gain(dB)_{norm} = log\frac{Gain(dB)}{Gain(dB)_{f=30}} \quad (5.2)$$

where the numerator is the presently calculated gain, occurring at the various frequencies. $Gain(dB)_{f=40}$ is the gain $\approx 18dB$, measured at a $frequency = 40Hz$, which is well within the expected frequency response or operating range of the filter.

The frequency response curve is shown in Figure 5.3. The cutoff-frequency $f_c \approx 1,027kHz$ (at gain level = -3 dB) and the stop-band frequency $f_s \approx 6,4kHz$ (at gain level = -80 dB), which is similar to the theoretical values $f_c = 1kHz$ (at gain level = -3 dB) and the stop-band frequency $f_s = 6kHz$ (at gain level = -80 dB).

Amplification (Gain)

As in the design of the anti-aliasing filter, it was again important to consider the ADC when performing the amplification. The A/D converter, included on the National Instruments™ PCI-6221 DAQ board, has a maximum dynamic range of $\pm 10V$, so it is important to ensure that the overall gain of all the signals does not exceed this range. A non-inverting amplifier was chosen to provide this particular gain. The gain is set via the feedback resistor $R_{feedback} = 100k\Omega$, with the input resistance $R_{in} = 1k\Omega$. This provides a maximum gain $K = 1 + (R_{feedback/Rin}) = 101$.

The full CAM schematic and board diagrams for the signal selection and conditioning circuit are illustrated in detail in Figures F.7 and F.8 respectively in Appendix F.

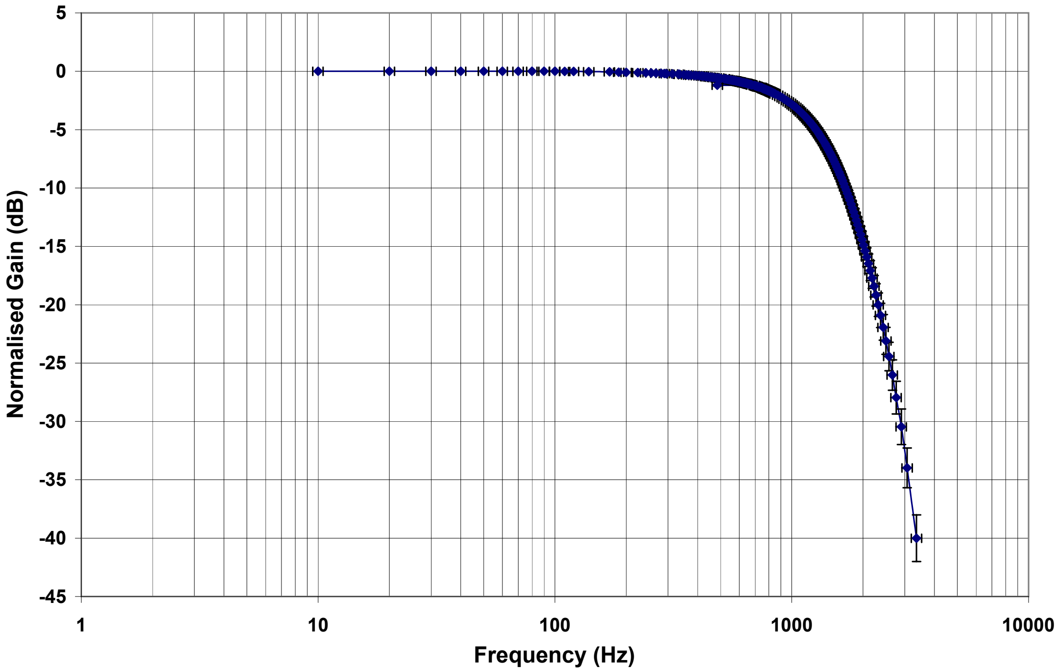


Figure 5.3: Frequency Response of the 8-pole anti-aliasing Bessel filter

5.4 Stage 4 - Data Acquisition (DAQ) and Control

This stage deals with the data acquisition and control of the various input signals, which in turn determines the accuracy and predictability of the input signal required to produce the correct output action at a specific time. The portion of the system is illustrated in Figure 5.4 below.

5.4.1 Prior data acquisition research

The Motorola™ MC68HC908KX8 microcontroller was initially considered to perform the A/D conversion and interfacing between the signal selection and conditioning stage and the PC. It was thought to be sufficient for the design as this microcontroller could perform a variety of different function when integrated onto a data acquisition board. The microcontroller has a 4-channel, 8-bit analogue-to-digital-converter (ADC). An 8-bit ADC is sufficient for signals resulting from human-motion, which occur at low frequencies $(max.20Hz)$, and generally would not require a high-speed ADC with a very large sampling rate. Other functionality included is an on board serial communication interface (SCI) module for transmitting and receiving data asynchronously (connected to the horizontal mount female 9-pin RS232 connector), as well a $MAX232$ serial driver. The board also includes an external 9.8304 MHz crystal oscillator, which would allow for ADC conversions of between $153.6S/s$ and $9.6kS/s$.

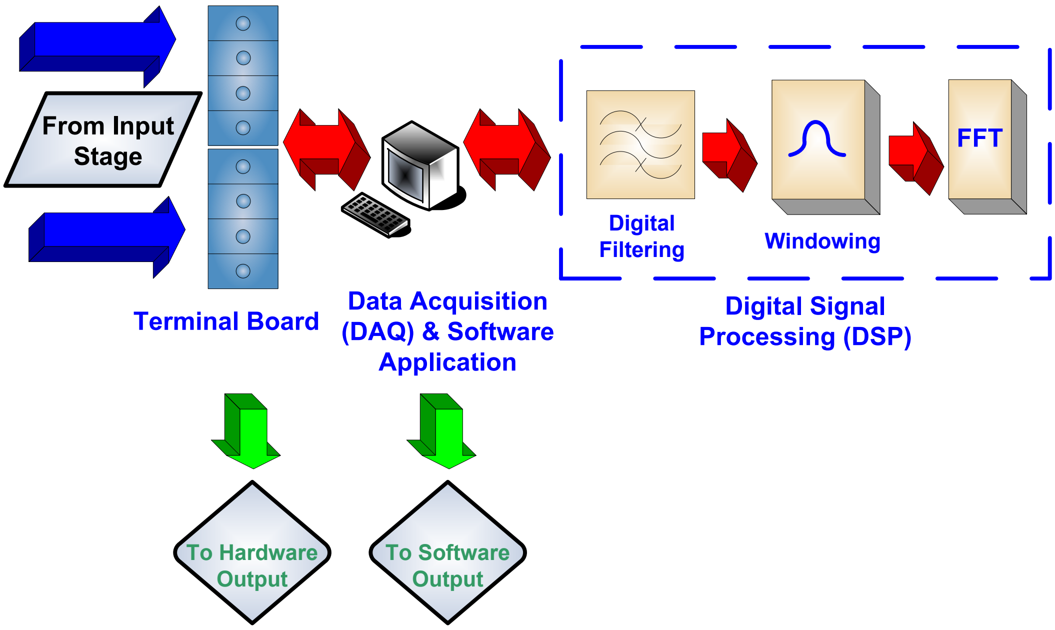


Figure 5.4: The data acquisition (DAQ), DSP and Software Application Stages

Extensive design was done using the $MC68HC908KX8$ microcontroller. This included low-level (assembly code) programming of the microcontroller for optimal ADC operation with a sampling rate $f_s = 9.6kS/s$. The use of this development board was eventually decided against as it proved far too simplistic and difficult in terms of interfacing with the software application being developed.

Further research was performed into other available technologies, which could be used for the data acquisition. A far more suitable system was available at no extra cost to the research. This is the *National Instruments® NI™ PCI-6221 data acquisition (DAQ) on board PC card*. This was chosen as it would be far more useful when utilising a variety of different input sensors for the data acquisition as opposed to the Motorola™ MC68HC908KX8 microcontroller board, which is only capable of asynchronous serial communication with only one channel at a time to a PC interface.

5.4.2 NI™ PCI-6221 DAQ Card

As the system requires the use of multiple input sensors it is important to employ a DAQ device which can make use of multiple inputs. As noted in [32], the NI™ PCI-6221 DAQ card has up to 16 single-ended and 8 differential analogue input channels. The most important reason for this choice of data acquisition is that the DAQ card slots into the PC's PCI slot and interfaces with the computer, thereby allowing for the development of a software application to interface with the DAQ device. All digital functionality is determined using higher level programming and is a lot simpler than programming at the low assembly code level (as used in the Motorola™ microcontroller). This therefore means that any software application that interfaces with the DAQ card can be customised to a particular user's requirements and, as a result, follows the design criterion more closely. A *NI™ CB-68LP* terminal block connection board serves as the interface between the "signal selection and conditioning board" and the NI™ PCI-6221 DAQ card.

The data acquisition (DAQ), as well as the digital signal processing and software application stages of the design, are illustrated in Figure 5.4.

5.4.3 Analogue-to-Digital Conversion

The analogue-digital-converter (ADC) converts the signal output from the signal conditioning circuit, into digital numbers so that the computer and digital processor (DSP) can (1) *acquire signals automatically*, (2) *store and retrieve information about the signals*, (3) *process and analyse the information*, and (4) *display measurement results* [33].

The two main functions of the ADC are **sampling** and **quantisation**. The NI™ PCI-6221 DAQ's on-board ADC has a 16 bit resolution; twice that of the Motorola™ MC68HC908KX8 microcontroller's ADC.

As mentioned previously, an 8-bit resolution for the ADC would be sufficient for the design, therefore a 16-bit ADC will serve only to increase the resolution of the digital output. The analogue-to-digital conversion forms the first part of digital signal processing. The sampling rate $S_r = 12kS/s$ for the design is implemented using the software application discussed in Section 5.5, which follows. According to [32], the maximum sampling rate of the ADC is $250kS/s$ and is this maximum decreases for each extra channel added, i.e.

$$f_s = \frac{250kS/s}{channels} \tag{5.3}$$

This would mean that even if 16 single-ended analogue inputs were chosen, the *maximum sampling rate/channel* $S_{max} = 15.625kS/s$, which is still higher than the required *sampling rate per channel* $S_r = 12kS/s$. In this case, however, 8 channels are utilised.

5.5 Stage 5 - Software Design and Implementation

5.5.1 Matlab®

An important tool used for design and analysis of data-acquisition systems is Matlab® . This was initially considered for use to perform the "offline" DSP functionality. As the design requires a generic platform, it is important to consider the use of a stand-alone software application to achieve the goals set out for the design. Matlab® , however, is not suited for this kind of application and would also prove too difficult to develop an application using it. The end-user would also require a copy of the Matlab® software application and a valid software license to run the designed application, which proves too costly for a general application, which would only requiring a small fraction of Matlab's® functionality and toolboxes.

5.5.2 Object-Oriented Design using MS™ Visual Studio C#.NET

The best way of employing a stand-alone software application is to use an object-oriented program such as C#. This has been incorporate in Microsoft's® Visual Studio.NET environment. The advantages of this form of programming over Matlab® , is that it is far less costly for the end-user, as he/she will not require a copy of the MS™ Visual Studio.NET program to run the designed application as once the executable has been compiled, it will automatically import and therefore incorporate the .NET framework into the executable. Another advantage of programming in $C\#$ over a matrix based language, such as Matlab®, or other object oriented design programs, like $C++$ and $Java$, is that it is much simpler and has been optimised for use in designing applications (forms) for the Windows™ and Linux (open-source) based operating systems. This would also prove far easier to use for the end-user, as most people, who are computer literate, have a reasonable knowledge in using Windows™ type programs, which make use of Windows™ type forms. A software application, designed in MS™ Visual Studio.NET would therefore have the benefit of portability.

Research was thereafter employed into the available technologies which could perform the required real-time digital signal processing and, therefore, be incorporated into the C# based software application. The National Instruments Measurement Studio™ was found to suit the required type of continuous based acquisition system as it is an add-on-tool for the MS Visual Studio.NET™ framework. This is discussed in further detail in Section 5.5.3 which follows.

5.5.3 National Instruments™ Measurement Studio and DAQmx 8.0

"National Instruments™ Measurement Studio is a suite of tools, which is used to compliment common development paradigms such as object-oriented programming, ActiveX, and Microsoft .NET technology in order to complete test, measurement, and control applications in less time" [34]. Various instruments can be controlled with as little or as much complexity as needed. Measurement Studio includes libraries, controls, and classes for sending commands to instruments over GPIB or serial ports. This simplified the various tasks immensely and provided far more functionality then if the code had been written in its entirety. It also has graphical tools, which aided in demonstrating these functions visually.

The main advantage of using this add-on tool is that it has been designed to incorporate already available software drivers , for use with National Instruments hardware products, such as the NI PCI-6221™ DAQ card. It therefore provides increased performance, which includes faster single-point analogue I/O and multithreading. The particular driver being used for the DAQ card is the NI-DAQmx™ 8.0 driver. The advantages of using this driver is, first of all it simplifies the task of having to write one's own software drivers for the NI™ DAQ card, which would be quite a difficult and laborious task (refer to [32]).

5.5.4 Requirements Engineering

The software application could only be designed once the **requirements engineering** process had been completed. This is a document which outlines the design rationale and procedures followed prior to the final software design and implementation stages. The complete *Requirements Engineering Document* is provided in Appendix D.

5.5.5 Functionality Diagram

The best way to define the requirements is via a flow diagram which visually describes the basic functionality of the proposed software application at language level (high level). This diagram includes only the primary functions performed by the application, where error messages (as tabulated in Table D.1 of the *Requirements Engineering Document* in Appendix D) and other detailed information has not been included. The basic functionality diagram for the "Generic Assistive Software Application", as it has been named, is illustrated in Figure 5.5 below.

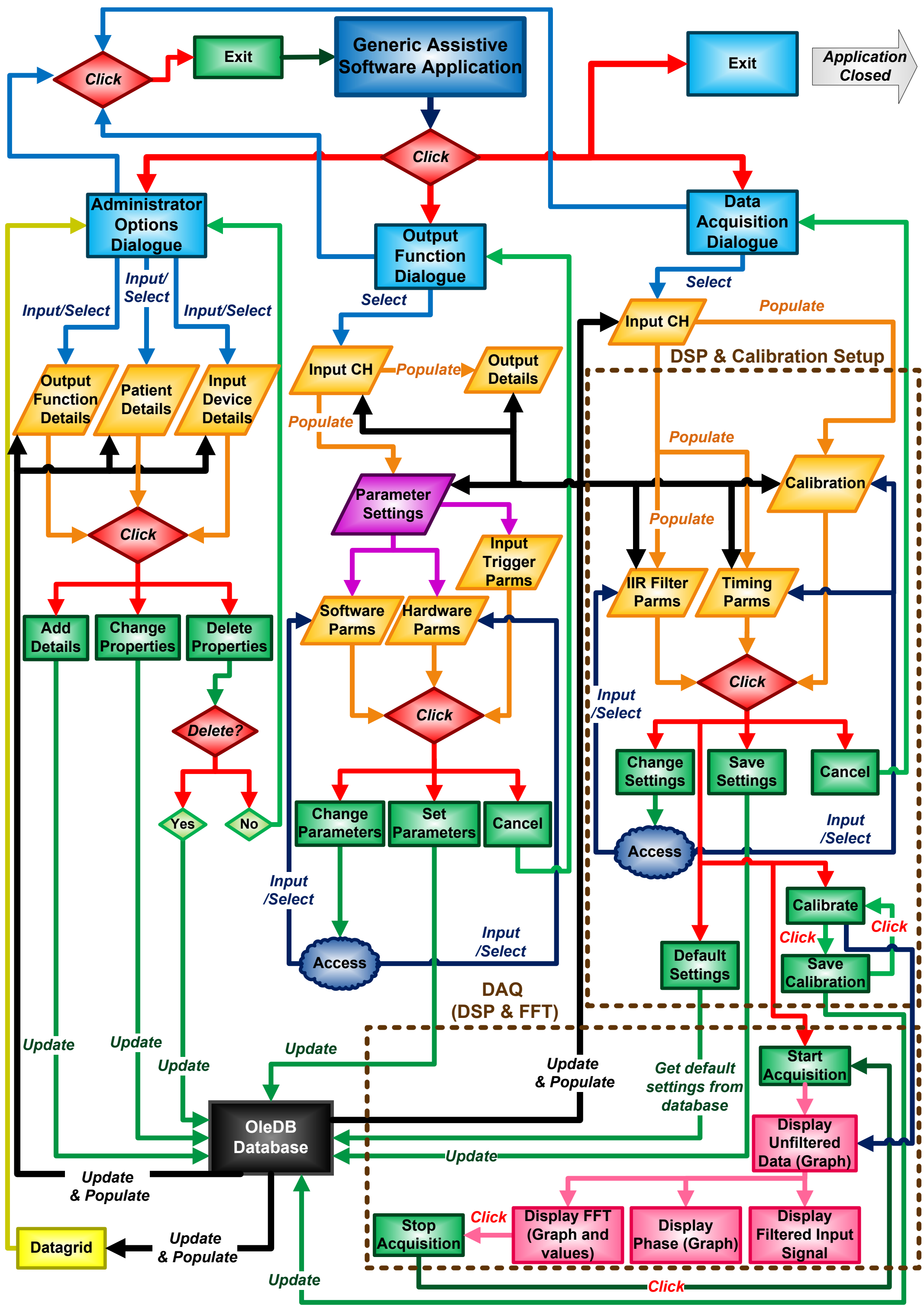


Figure 5.5: Basic Functionality Diagram of the Generic Assistive Software Application

5.5.6 Software Implementation

The software application was implemented by following the procedures described by the *Requirements Engineering Document* (Appendix D) and using the above illustrated *functionality diagram* (Figure 5.5). C#.NET, incorporated in Microsoft's® Visual Studio.NET environment (as described in section 5.5.2 above), was used to write the application. The software application acquires the information from the NI PCI-6221 DAQ card using the NI-DAQmx™ 8.0 device driver as described in section 5.5.3.

Software Setup and Operation

The software application setup and operation is described in full detail in section H.2.4 of the *User Manual*.

Digital Signal Processing

Digital signal processing forms a very important part of all measurement and control systems. The digital signal processing stages includes *digital filtering* and the *windowing* process as well as *spectral analysis*, which in turn involves performing a *Fast Fourier Transform* (FFT) on the various input signals.

The digital signal processing and analysis routines were implemented in the software application using National Instruments™ Measurement Studio.NET (as mentioned in section 5.5.3), which was achieved by utilising various classes, toolboxes and libraries already available in the software suite. All digital signal processing is setup in the "Data Acquisition Dialogue", as shown in Figure 5.5 and is described in full detail in section H.2.4 of the *User Manual* (Appendix H).

Digital Filtering: Digital filtering is important in all data acquisition based systems and forms the first part of the digital signal processing. As noted earlier, the analogue Bessel filter, designed in Appendix C, is used for anti-aliasing (averaging) and does not perform significant noise reduction, as the highest frequency component (stopband frequency) chosen was $6kHz$. This means that many unwanted/noise signals may be prevalent at frequencies lower than $6kHz$. Noise reduction is therefore performed by digital filtering, which approximates the desired response. This also increases the signal-to-noise ratio significantly and ensures that the signal acquired occurs as a result of the user input and not as a result of environmental, sensor or transmission noise [4].

Digital Filter Design Choices: The procedures outlined in Section B.3.2 of Appendix B were employed in determining the digital filter characteristics.

Two filter classes needed to be considered, namely the *FIR (Finite Impulse Response)* and the *IIR (Infinite Impulse Response)* filters.

- **FIR filters** are generally preferred over their IIR counterparts, because they offer a number of advantages compared with their IIR equivalents. These include the fact that FIR filters are generally more stable, offer exactly linear phase (constant group delay) vs. frequency behaviour and the fact that there exist a number of efficient algorithms for designing optimum FIR filters with arbitrary specifications.
- **IIR filters** have an advantage over FIR filters in that FIR filter designs generally require more computation to implement. Also IIR filters can generally approximate a filter design using a lower-order filter than that required by an FIR filter design to perform a similar filtering operation [4].

As mentioned in Section 4.4 of Appendix 4, real-time systems must be **efficient** and **predictable**, and it is for this reason that an IIR filter has been chosen for the digital filtering design as it executes much faster and does not require extra memory, as it executes in place and is therefore optimal for the real-time system being used. The biggest disadvantage of IIR filters, however, is that they have a **non-linear phase response**. It is therefore important during the testing phase to ensure that the non-linear phase response does not affect the end design adversely [4].

Digital Filtering Implementation: All digital filtering has been performed in Microsoft® Visual $C\#.NET$ using National Instruments™ Measurement Studio. The code has been written such that the user has the option of selecting the various filter types and designs . The filter types, included, are the lowpass, highpass, bandpass and bandstop filters (i.e. all of the various filter types), though the lowpass filter is generally the most common selection required for the signals. The most general filter designs have been incorporated into the design, and include the Elliptic, Bessel, Butterworth, Chebyshev and Inverted Chebyshev filters. The program has also been written such that the user may enter the upper and lower cutoff frequencies, the filter order, attenuation and ripple of the digital filter. For information on the digital filtering setup and implementation, refer to section H.2.4 of the *User Manual* (Appendix H). The sample code below is an example of digital filtering with a lowpass Bessel filter:

$NationalInstruments.Analysis.Dsp.Filters.BesselLowpassFilter;$

$besselLowpass = newBesselLowpassFilter((int)order.Value, sampleRate, lowerCutoff.Value);$

$filteredwave = besselLowpass.FilterData(waveform);$

As can be seen from the code block above, the user needs to specify the order, lower cutoff frequency and sample rate of the filter. The value "waveform" is the input signal to be filtered.

Spectral Analysis Spectral analysis provides an estimate of the distribution of signal power at different frequencies. All signal-processing is performed by the assisitive software application which interfaces with the certain dedicated instruments. This consequently allows the user to perform spectral analysis at the "touch of a button", thereby providing a visualisation of the results on a graphical plot. For further background information on spectral analysis refer to section B.4 of Appendix B.

The spectral analysis being performed by the software application, includes the fast Fourier transform with windowing functions, and is discussed as follows.

Fast-Fourier Transform: The discrete Fourier transform (DFT) converts the digital/discrete signal from the time-domain to the frequency domain. The DFT allows one to analyse the frequency spectrum of the incoming discrete signal graphically, by expressing the relative contributions of the various sinusoidal signals to the composition of the discrete input signal. The physical interpretation of the spectrum for the discrete Fourier transform is obtained with the amplitude and phase spectra, with respect to frequency. These are defined by equations B.9 and B.10 in Appendix B.

The FFT is not a single algorithm, but rather a large family of algorithms which can increase the **computational efficiency** of the DFT [35]. As demonstrated by Panerai et al., where the given number of samples $N = 1024$ (which in turn is the sampling rate for the various input signals), using an FFT algorithm is **100 times faster** than the direct implementation of Equations B.7 and B.8 (Appendix B), which is especially important when using a real-time system such as the one being designed.

A FFT is performed on all the various input signals as a means to ensure that only the voluntary actions are detected, and not those which are involuntary. The software application performs the FFT using the FFT function defined in the NI™ Measurement Studio toolbox. The application has been coded to decompose the input signal into its magnitude and phase plots. From this the peak magnitude, and its associated frequency is determined by the software application, from the amplitude versus frequency plot.

The application allows the user to calibrate the trigger level at which the input, from a particular sensor, produces the required output response or function. The trigger level is equal to the peak magnitude level (dB) of the input signal, occurring within a small calculated frequency range. An output function is only produced when this trigger level is equaled or exceeded, while occurring at or within the calibrated frequency range. This therefore ensures that only voluntary actions result in an output response being triggered. For information on calibration setup and operation of the FFT refer to section H.2.4 of the *User Manual* (Appendix H). The principles behind fast Fourier transforms and FFT spectral analysis are discussed in detail in sections B.4.2 and B.4.3 respectively (Appendix B). The FFT software application code is provided below:

```
// Apply window function to filtered waveform ("FFTwaveform" below) before performing FFT.
scaleWindow.Apply(FFTwaveform, out equivalentNoiseBandwidth, out coherentGain);

//Plots previous FFT data for the current channel selected
if(currentchannel == number) windowedPlot.PlotY(FFTwaveform);

// Calculate the FFT of waveform array.
FFTValue = NationalInstruments.Analysis.Dsp.Transforms.RealFft(FFTwaveform);

// Get the magnitudes and phases of FFT array.
NationalInstruments.ComplexDouble.DecomposeArrayPolar(FFTValue, out magnitudes, out phases);
```

Windowing: Windowing suppresses "glitches" or leakage effects which occur when performing a FFT and so avoids the broadening of the frequency spectrum caused by these glitches and is discussed further in section B.4.4 of Appendix B. Windowing can narrow the spectrum and make it closer to what was expected. The windowing functions catered for here include: the Hanning, Hamming, Blackman-Harris, Exact Blackman, Blackman, FlatTop, 4-Term and 7-term Blackman-Harris as well as rectangular windows. For information on the windowing setup and implementation, refer to section H.2.4 of the *User Manual* (Appendix H). A sample code is provided below:

```
stringWindowtype = WindowRow["Window"].ToString().Trim();

winType = ScaledWindowType.Hamming;
```

This code provides the e.g. Hamming windowing function.

5.5.7 Software Testing and Validation

Once the software application had been designed and implemented, testing and validation needed to be performed to ensure that the application met its desired requirements (see requirements documentation). Testing was performed using the procedures described in the testing and validation document in Appendix E. The testing and validation performed at this stage only included tests and validation performed on the software application alone and not of the whole system. The full assistive system was later tested as described in section 5.7.2.

Coverage based testing

Coverage based testing involved verifying that the overall program operated as required. This was achieved by following the procedures outlined in section E.2.1 of the *Testing and Validation document* in Appendix E.

One of the more important functions tested for was that of the data acquisition process (performed by clicking the “Start” button). To test this a **known signal** was inputted via a signal generator into the NI CB-68LP terminal connection board, connecting to the data acquisition card. Various signals of known amplitude and frequency was inputted into one of the channels, in order to test the *Spectral analysis* i.e. the *FFT* and *windowing* ability of the program. The results were fairly accurate and could be improved by increasing the number of samples/channel towards the sample rate of 12 kS/s, i.e. when 6 kS/channel was chosen, the FFT resulted in a singular peak at a frequency within 1 Hz of the signal provided from the signal generator, whereas samples chosen between 2 - 4 kS/channel resulted in a 2 - 3 Hz error from the peak magnitude frequency for the signal provided. The trade-off, however, was that of accuracy and resolution versus the data acquisition performance, where higher samples resulted in a more accurate FFT, but reduced the program’s performance and visa-versa. The typical value was therefore chosen at 4 kS/channel, which was found to be accurate enough to fit the requirements of the design.

The coverage based testing procedures were performed numerous times until all of the required program functions operated correctly and without complication.

Error-based testing

This focuses on testing all the possible erroneous inputs and events which may occur while using the software application. The error-based testing was performed by following the procedures, as outlined in section E.2.2 of the *Testing and Validation* document, to ensure that the various erroneous inputs would generate the required error messages.

This form of testing was performed at various stages throughout the software design and implementation and was tested extensively in order to ensure that the user would be notified of all the possible illegal inputs or events which could result. The error-based testing was subsequently tested for successfully.

User interface and functionality testing

These set of tests were performed by following the procedures outlined in section E.2.3 of the *Testing and Validation* document.

The overall result from the user interface and functionality testing resulted in all five criteria being highly rated. The tasks were weighted evenly and the scores were added and divided by the sample size of 5 people. The following were the results of the tests:

Usability = 4.4
Performance = 3.6
Enjoyment level = 4.6
Visual Appeal = 5.0
Customer feedback and helpfulness = 4.2

These tests yielded good results on the whole, especially in terms of the software program's usability, enjoyment level and visual appeal. The performance, as predicted, was above average, however this can be improved by redesigning the program code more efficiently with a less graphics intensive GUI (at the expense of visual appeal).

Miscellaneous testing

The miscellaneous testing was performed by following the procedures outlined in section E.2.4 of the *Testing and Validation* document. These tests were completed successfully, whereby the accessibility and colour testing passed each of their requirements, however the software application was found to be a little graphics intensive and resultantly slowed the program performance slightly when compared to using simpler graphics. I was determined, however, that the program had sufficient performance to execute the required tasks successfully, even with the higher resolution graphics.

Fault based testing

This focused on detecting the faults which may arise when using the program, and was performed by following the procedures established in section E.2.5 of the *Testing and Validation* document. The one fault which occurred periodically was that the program would "freeze" 10% of the time, while performing the data acquisition process. This was a performance related problem, whereby the program would acquired a new set of samples before the previous samples had been processed, as a result of this sluggish performance, which is comparable to the "buffer under run error" which occurs when copying a CD. This was corrected by decreasing the number of samples/channel, meaning that the program was less resource intensive. Another program fault which occured was that of reading from and writing to the database during the data acquistion proccess. This was subsequently corrected by changing the incorrect code, which related to the way the database updated.

All of the major faults had been corrected, however not all of the faults may have been detected, as this would require a lot of time to do properly. This is not really a problem, since these would probably be minor faults, which shouldn't affect the way the program operates as a whole. It must be noted that the data acquisition process is the most important task performed by the program and must not contain any errors or faults.

5.6 Stage 6 - Output Stage

The output stage forms the final stage in the design process. As mentioned throughout the report, the device must be capable of adapting to the user's varying needs, i.e. the user should be able to customise the device to control a variety of different environments. These environments may be hardware or software in nature. The device has been designed to control both types.

Both physical output signals and software functions may be customised using the *Generic Assistive System Software Application* mentioned in Stage 5. The output functions may be customised by selecting the "Output Function Setup" button, which consequently opens the "Output Function Setup" dialog, as mentioned in Section H.2.4 of the *The User Manual* in Appendix H.

5.6.1 Hardware/Physical Outputs

The hardware output in this case, is based on any physical output signal which would control an externally powered hardware device or system. This output signal must be customisable and would link to a control unit which would e.g. operate an electric wheelchair or bed.

As mentioned in the *NI™ 622X Specification Sheet*, the DAQ card can provide digital and analogue voltage and current outputs, but for purposes of this research only analogue voltage outputs have been catered for [32]. These output pins are labeled "aio" and "ai1" and occur on pins 22 and 21 of the NI PCI-6221 DAQ card respectively.

As mentioned in Section 1b of the *The User Manual* in Appendix H, the user may control all outputs by specifying, in the "Output Function Setup" dialog, the output channel to be used as well as the voltage range and actual minimum and maximum voltage levels. The closer the min/max voltage range values are to the actual minimum and maximum input values, the better the resolution of the D/A conversion for the output signal.

5.6.2 Software Outputs

To ensure that the system is truly customisable, the design must allow for the execution and control of software applications and operating systems. To ensure this the required software functions or applications must be written such that it they may interface with the "Generic Assistive Software Application".

The requirement of the plug-in software is that they must be written such that the software output functions relate to the different input channels "ai0" to "ai7". The example plug-in application provided for this project is the execution and control of Windows Media Player via the various input sensors and is called "WindowsMPlayer.exe".

As mentioned in Section 1b of the *The User Manual* in Appendix H, for a software type output the user must select the "..." browse button in the "Output Function Setup" dialog, in order to search for a particular plug-in application executable file.

5.7 Stage 7 - Design Implementation and Functionality Testing

5.7.1 Implementation

The system components described in stages 1 to 3 (excluding the SCATIR switch) were implemented using Eagle-CAD® to design the PCB (printed circuit board). A milling machine was used to mill and cut the required circuit boards, onto which the various circuit components would be soldered. The Eagle-CAD schematics and boards are illustrated in Appendix F.

Once the various circuit boards had been designed and built they were placed into various enclosures for safety and aesthetic reasons. The enclosures also protect the PCBs from damage.

5.7.2 Operation and Functionality Testing

Individual Component Testing

Once the various system components were built, testing was performed on each component to ensure their individual operation. This involved testing each of the individual input sensors by varying the levels of movement (in the case of the accelerometer and blink/wink switch) or sensitivity (in the case of the touch sensor) and thereafter checking that the correct output voltage resulted from the different sensors. The signal selection and conditioning unit was tested by inputing various test voltages, using a signal generator, into the system and thereafter checking that the required signal outputs resulted.

Hardware System Testing

The system was subsequently assembled by connecting the input sensors to the signal selection and conditioning unit, utilising the guidelines followed in Sections H.2.1 and H.2.2 of the *Operation Manual* in Appendix H.

Functionality testing: This involves the testing of the entire hardware portion of the system in terms of its functionality. This was executed by activating each input sensor individually and checking whether each sensor input resulted in the correct output signal voltage on the correct output pin terminals of the signal selection and conditioning circuit. Priority of the input sensors was performed by activating more than one sensor at any given time, and thereafter checking that only the higher priority input sensor signal resulted in an output and not those with a lower priority.

Full System Testing and Validation

The hardware system was thereafter connected to the PC via the DAQ terminal block, as described in Section H.2.3 of the *User Manual* (Appendix H). From this the **software testing and validation**[1] was performed, as outlined in Appendix E. Detailed information on the software application and its operation is provided in Section H.2.4 of the *Operation Manual* (Appendix H).

The final system, with all its various components, is provided in Figure 5.6.

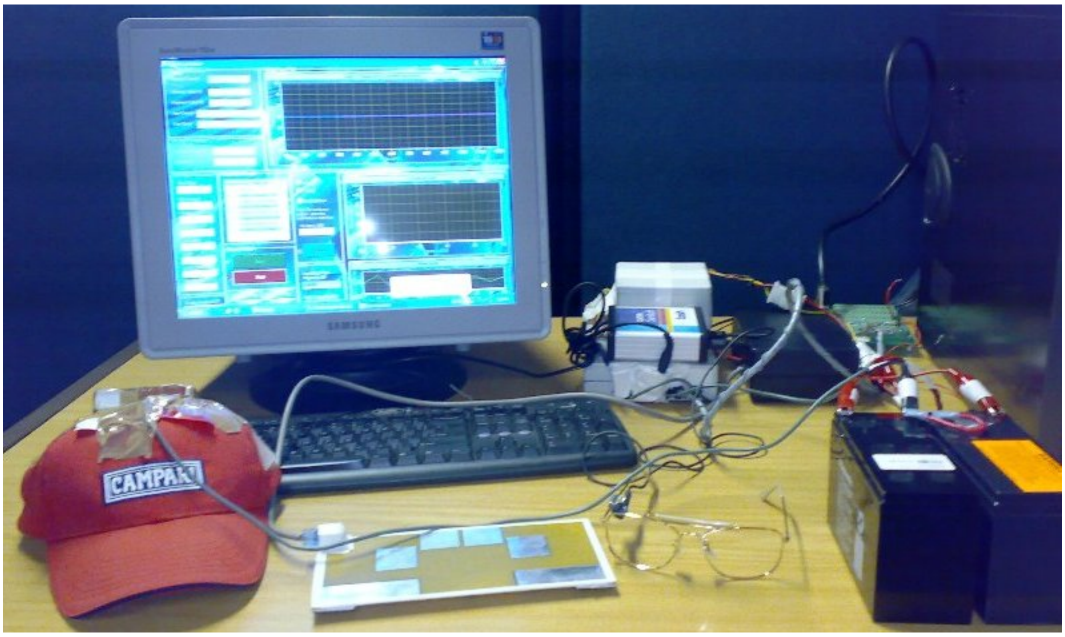


Figure 5.6: The Generic Assistive System

[1]Software testing cannot be performed properly without interfacing the hardware devices to the software application.

Chapter 6

Testing and Results

Results based testing forms the most important phase of the research. Certain tests have been performed by a selection of volunteer subjects, each with a different level of motor deficiency. The results must relate back to the research objectives in terms of establishing whether the assistive system could be customised to the individual subjects, each with their own specific needs, in terms of controlling their environment.

6.1 Ethics Approval

Whenever human subjects or volunteers are required to perform certain tests, ethics approval must be obtained. In this case the testing was going to be performed using volunteer subjects for medical research. Subsequently an application form was filled out and sent to the *Human Research Ethics Committee (HREC)* for clearance of medical research. The approved ethics document is provided in section G.1 of Appendix G.

One of the stipulations of the HREC ethics approval was that a *Participant Information Sheet*, *Informed Consent Form* and *Questionnaire* (if applicable) needed to be completed and attached to the document. These documents were given to each of the volunteers prior to the commencement of testing.

The Participant Information Sheet: This introduces the test subject to the researcher as well as providing him/her with basic information regarding the research and the various tests to be performed. The full participant information sheet is provided in Section G.2 of Appendix G.

The Information Leaflet and Informed Consent Form: This introduces the purpose of the study and the testing procedures to be followed. Written informed consent was to be provided once the document had been read thoroughly. This is provided in Section G.3 of Appendix G.

The Questionnaire: This provides all essential questions which the volunteers were asked to answer. More information is provided in Section 6.4.5 to follow. The actual questionnaire used is provided in section G.4 of Appendix G.

It must be noted that all of these above mentioned documents and forms stipulate that no risks would be encountered throughout the testing procedure and that the volunteers could leave at any time throughout the testing procedure, if they chose to do so, without any adverse consequences to themselves or the system.

6.2 Safety Precautions and Safety Testing

Even though the system makes use of low voltages and currents, it is still important to ensure absolute safety is followed when testing is performed with human subjects.

6.2.1 $\pm 12V$ Battery

The hardware portion of the system has been connected to a $\pm 12VDC$ battery supply as noted in Section 5.1.1. As already noted, this battery allows for better safety than using a dual-power desktop power supply as it is not in any way connected to the 220 V mains supply. It also provides better portability of the device.

6.2.2 The NI™ PCI-6221 DAQ card

The DAQ card, which interfaces between the hardware and software systems, is a key issue in terms of the safety of the system as it is physically connected to the PC, which in turn is connected to the 220 V mains supply.

As mentioned in the PCI-6221 specifications manual in [32], the DAQ card is designed to meet the requirements of the following *Standards of Safety for Electrical Equipment for Measurement, Control, and Laboratory Use* [36]:

- IEC 61010-1, EN 61010-1
- UL 61010-1
- CAN/CSA-C22.2 No. 61010-1

The NI™ PCI-6221 DAQ card also meets *The Essential Requirements of Applicable European Directives, as amended for CE Marking (CE compliance)*, as follows:

- **Low-Voltage Directive (safety) - 73/23/EEC:** which in Section 2.1 states that "The Electrical Equipment means any equipment designed for use with a voltage rating of between 50 and 1000 V for alternating current (A.C.) and between 75 and 1500 V for direct current (D.C.). Therefore, this directive is often called the Low Voltage Directive which applies to the vast majority of electrical equipment in everyday use. The Electrical Equipment may be placed on the market only if it does not endanger the safety of persons, domestic animals or property." [37].
- **Electromagnetic Compatibility [38]**
  - Electromagnetic Compatibility Directive (EMC) - 89/336/EEC
  - Emissions: EN 55011 Class A at 10 m; FCC Part 15A above $1GHz$.
  - Immunity: EN 61326:1997 + A2:2001, Table 1.

Further information on these safety standards for the NI™ PCI-6221 DAQ card are provided in the PCI-6221 specifications manual [32].

6.2.3 The Isolation Transformer

As mentioned in 5.1.2, the PC is plugged into a 220-220 V isolation transformer, which in turn is plugged into the 220 V (AC) mains supply. This isolates the secondary side for improved electrical safety.

6.2.4 Overall Safety Testing

The overall system was tested for safety at *Brittan Healthcare*. Leakage tests were performed on the DAQ card and the PC, where both passed successfully.

6.3 Risks

No risks are associated with the design and there is consequently no risk associated with its use thereof. No danger is posed to the users of the system as all of the input sensors are non-invasive i.e. they do not puncture the skin. This has been specified in the HREC approval form and attachments.

6.4 The Testing Procedure

In order to provide as little inconvenience to the subjects as possible, the tests were performed at the various subjects' places of residence.

6.4.1 The Test Subjects

Five test volunteers suffering with diverse levels of motor deficiency were chosen, who fit the subject classification described in Chapter 3. Each volunteer was chosen with a slightly different level of disability from the rest of the test subjects. The motor disability and classification levels, of the volunteers chosen, are shown in Table 6.1.

Table 6.1: Subject Classification and Level of Lesion

Input Channel	**Output Function**
"Subject A"	Quadraplegic - C4/C5
"Subject B"	Musclular Atrophy
"Subject C"	Quadraplegic - C6/C7
"Subject D"	Cerebral Palsy
"Subject E"	Quadraplegic - C5/C6

6.4.2 The Input Sensors

The subjects performed a set of tests, based on the operation of the system, by utilising all or some of the input sensors described in previous chapters. The sensors were selected based on their suitability for the subjects' retained functions[1]. Placement of the sensors, for the purpose of this study, were restricted to sensing arm or finger movements, head and neck movements and blink/wink eye movements.

[1]These functions may be based on residual movements.

6.4.3 System Setup

The system was initially setup[2] by following the steps outlined in the *User Manual* described in Appendix H. Once this was completed, the volunteer was asked to try use as many of the different types of sensors as possible, which would subsequently provide certain output functionality as described in the following section.

6.4.4 The Output Functions

The Physical Output

The physical output signal was provided by an output voltage, which was varied throughout testing to prove the customisability of the device (refer to Section 5.6.1 of the *Design Stages* chapter). This physical voltage was, however, not connected to any physical system, but measured rather with a multimeter to prove the concept that the system could provide physical output signals.

The Software Output

As mentioned in Section 5.6.2 of the *Design Stages* chapter, the software output consists of the execution and operation of *WindowsMPlayer.exe*, which was written such that it can could simply "plug" into the *Generic Assistive System*, with certain channels, connected to various input sensors, performing certain tasks with regard to using the *Windows Media Player* application.

Table 6.2 below describes the input channel setup which was used for the testing as well as the associated output functions performed:

N.B. This channel setup was maintained for all the tests performed on the various volunteers to ensure consistency in the testing procedure, which consequently should ensure that the results are consistent as well.

[2]To save time and maintain simplicity in the setup and testing procedure, various steps in the setup had to be completed prior to the assessment. These include the connections made between the input sensors and the signal selection and conditioning board and the associated connections to the NI LP-68LP terminal block connection board.

Table 6.2: Input Channels vs. Output Functions

Input Channel	Output Function
“ai0”	Provides a physical voltage output on “ao0”
“ai1”	Provides a physical voltage output on “ao1”
“ai2”	Not connected
“ai3”	Executes or opens Windows Media Player
“ai4”	Skips to the next media file in the playlist
“ai5”	Stops play
“ai6”	Closes Windows Media Player
“ai7”	Pauses/plays media content

6.4.5 The Questionnaire

The volunteers were subsequently asked to answer a questionnaire, concerning their assessment of the performance and potential usefulness of the system. These questions have been formulated using the*basic considerations in motor behaviour and control systems*, mentioned in Section 4.2 as a guide, e.g. minimising energy required, minimising disruption, practicality etc. The full questionnaire is provided in Section G.4 of Appendix G. It must be noted that the only subject information recorded was the extent of the volunteers’ motor disabilities.

The testing procedure took no longer than an hour to complete per subject, once the initial system setup had been completed. Full information on the system setup and operation is provided in the *User Manual* (Appendix H).

6.5 Results

On completion of the questionnaire, the following results were obtained, followed by a complete analysis of these results.

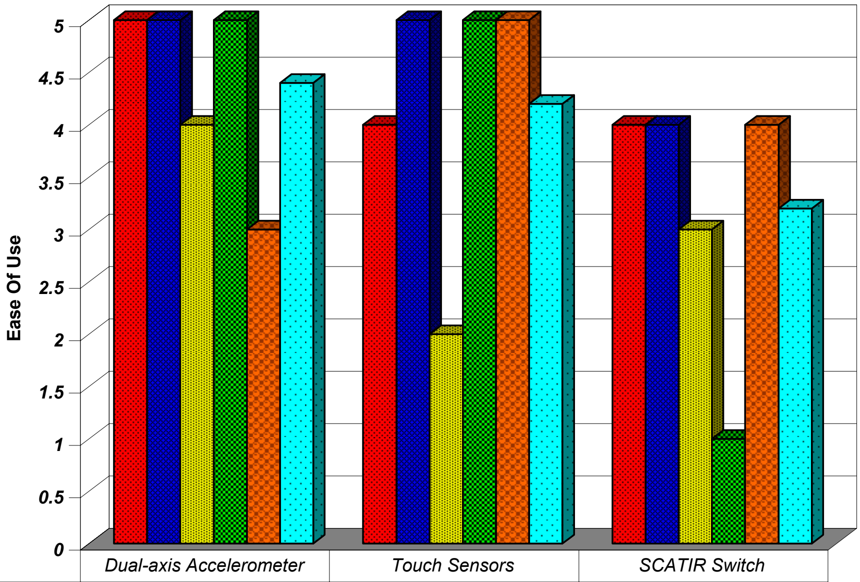
6.5.1 The Input Sensors

These results deal with various aspects of the input sensors[3]. All scores are rated out of 5.

[3]Please note that both the X and Y axes of the dual-axis accelerometer resulted in the same scores as well as the 4 touch sensor channels. The scores are therefore illustrated by the sensors used and not by each individual channel

Graphical Results

Figure 6.1 is a bar graph of the ease of use of the various input sensors, where a higher score determines a better ease of use (5 being the highest score).



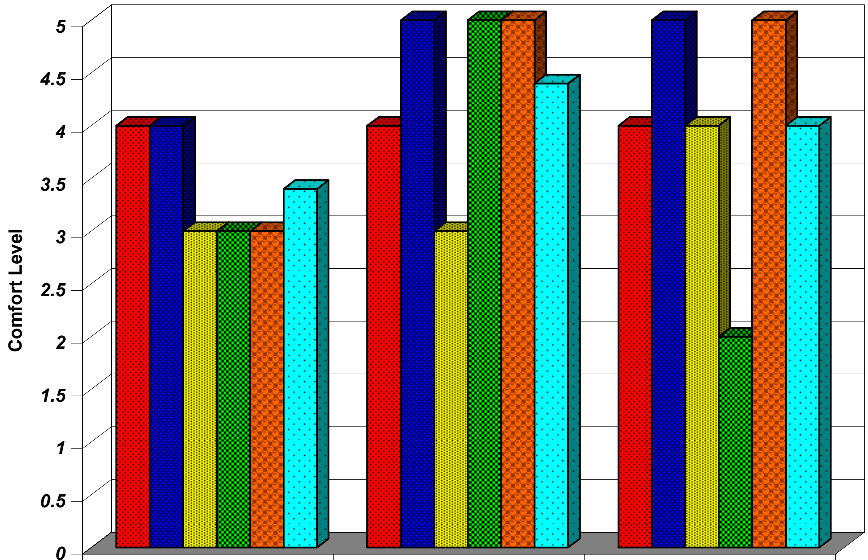
	Dual-axis Accelerometer	Touch Sensors	SCATIR Switch
A	5	4	4
B	5	5	4
C	4	2	3
D	5	5	1
E	3	5	4
(Average)	4.4	4.2	3.2

Figure 6.1: Ease of use of the various input sensors

Figure 6.2 is a bar graph of the comfort level experienced when utilising the various input sensors. A higher score determines a better comfort level (5 being the highest score).

Figure 6.3 is a bar graph of the performance and predictability of the various input sensors. A higher score determines a better performance and predictability (5 being the highest score).

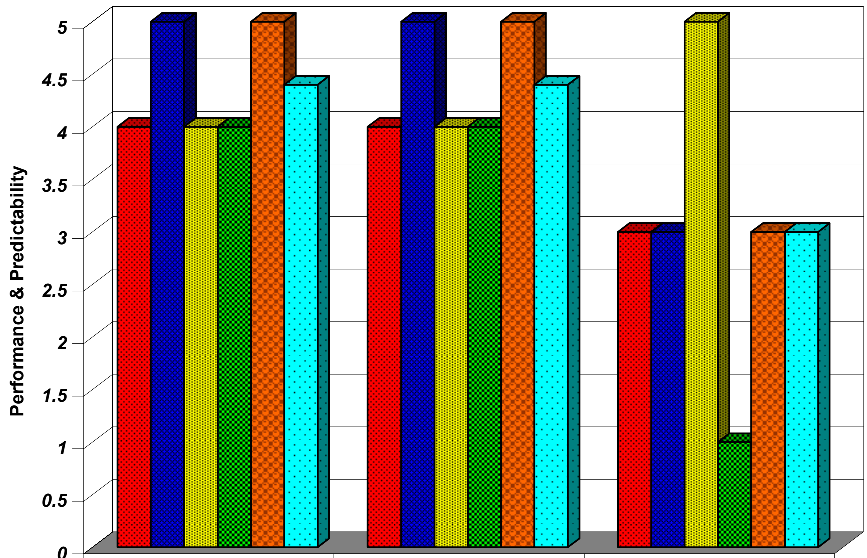
Figure 6.4 is a bar graph of the fatigue experienced by the subject when utilising the various input sensors. A lower score determines a lower level of fatigue (1 being the lowest).



	Dual-axis Accelerometer	Touch Sensors	SCATIR Switch
A	4	4	4
B	4	5	5
C	3	3	4
D	3	5	2
E	3	5	5
(Average)	3.4	4.4	4

Input Sensors

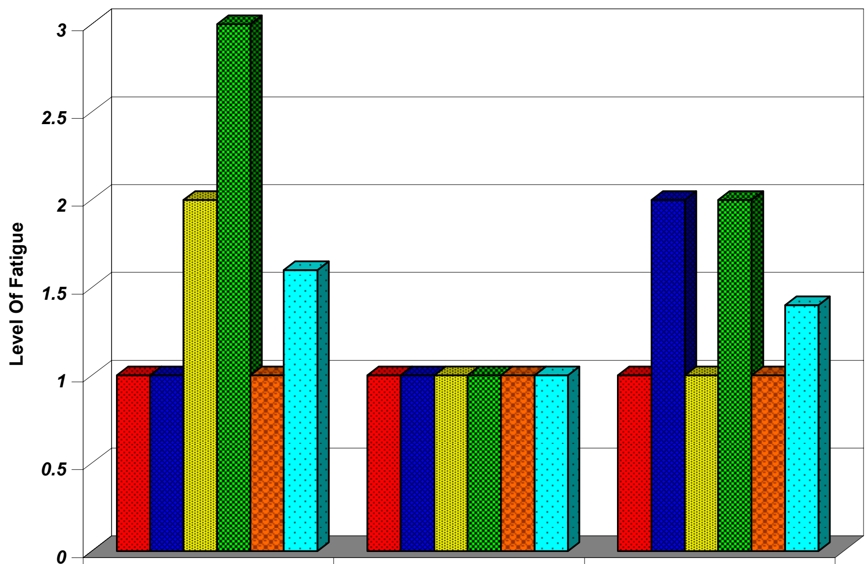
Figure 6.2: The comfort level experienced when utilising the various input sensors



	Dual-axis Accelerometer	Touch Sensors	SCATIR Switch
A	4	4	3
B	5	5	3
C	4	4	5
D	4	4	1
E	5	5	3
(Average)	4.4	4.4	3

Input Sensors

Figure 6.3: The performance and predictability of the input sensors



	Dual-axis Accelerometer	Touch Sensors	SCATIR Switch
A	1	1	1
B	1	1	2
C	2	1	1
D	3	1	2
E	1	1	1
(Average)	1.6	1	1.4

Input Sensors

Figure 6.4: Level of fatigue experienced when utilising the various input sensors

User Feedback on the input sensors

This section provides information on some possible improvements to the system as well as the various concerns which may have been experienced by the volunteers when utilising the system.

Subject A: Adjusting the input sensors to specific requirements was difficult, especially in the case of the blink/wink sensor, but once one is used to it, it should provide better results. The tilt-sensor needs to be reduced in size and could for example be attached to the eye-glasses.

Subject B: The blink/wink switch is too sensitive.

Subject C: The tilt sensor (dual-axis accelerometer) should be smaller and unobtrusive. The touch-sensor keys needs to be larger and customised to the specific user's hands. This sensor may not be as useful for quadriplegics at level C4/5 and below, whereas the other sensors would be fine for most acute disabilities.

Subject D: The keys of the touch pad should be closer together to suit this particular user, i.e. one should be able to customise the spacing of the keys to suit the different users' hands. The blink/wink sensor did not suit this particular user as he had poor muscle control of his eyes and could blink, but not wink. The input sensors should be designed such that they are easier to attach to the user, which was difficult when using the blink/wink and tilt sensors.

Subject E: The blink/wink sensor wasn't as predictable as perhaps required as it needed to be set for the particular user using it. The choice of input sensors doesn't really fit the requirements of this volunteer. The tilt sensor (dual-axis accelerometer) needs to be more compact. A tracker-ball mouse or eye-gazing sensor would possibly be more useful to this subject.

6.5.2 The Overall System

These results deal with certain aspects of the system as whole.

Graphical Results

Figure 6.5 is a bar graph depicting various system parameters, which have been used to determine the viability of the overall assistive system. These parameters are explained below and are each scored out of 5.

Customisability (a): Determines how easily the system accepts various input sensors, where a score of 5 signifies that all inputs are accepted by the system.

Customisability (b): Determines how customisability of the system in terms of providing a specific output for a specific and voluntary input. A higher score signifies that the system is fully customisable.

Repeatability: Determines how repeatable the system is i.e. how easily the system can repeat a specific task multiple times, where a higher score signifies that the system repeats all the various tasks more accurately.

Performance: Determines the performance level of the system i.e. if the system provides the required output for a specific input accurately, without complication and within an acceptable time frame. A higher score signifies that system performs excellently.

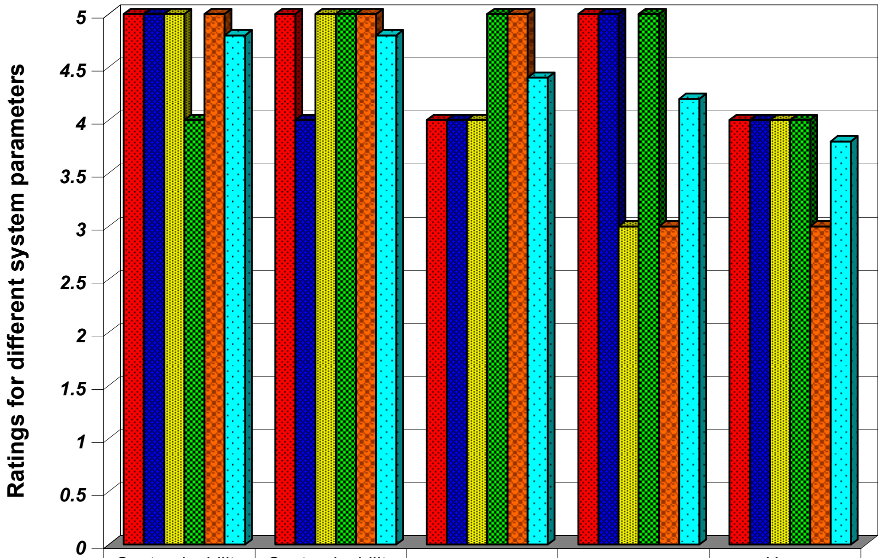
Requirements: Determines if the system is capable of satisfying the user's requirements, i.e. if the system is specific enough for the subject's needs. A higher score signifies that the system could meet any or all of the user's requirements.

User feedback on the system testing

Subject A: The system performs what was required, but took some getting used to.

Subject B: (*No Information provided*)

Subject C: This system might be best suited to more severely disabled people.



	Customisability (a)	Customisability (b)	Repeatability	Performance	User requirements
A	5	5	4	5	4
B	5	4	4	5	4
C	5	5	4	3	4
D	4	5	5	5	4
E	5	5	5	3	3
(Average)	4.8	4.8	4.4	4.2	3.8

Overall system parameters

Figure 6.5: Various system parameters used to determine the viability of the overall assistive system

Subject D: (*No Information provided*)

Subject E: (*No Information provided*)

6.5.3 User Feedback: General Comments and Possible Improvements to the System

Subject A:

- The system should be wireless.
- Voice activation sensors would be best suited for the design.
- The system is too bulky, i.e. it needs to be more compact.
- The system needs a little too much assistance to set up.

Subject B:

- The system would be much improved if it were wireless, or the wires would at the very least need to be longer.
- The system should be more simple in design.

Subject C:

- More simplicity is required with the software application, i.e. the user would need to be proficient in software to use the application.
- The system should rather be more a plug-and-play type system, which would require less knowledge to use and therefore easier to setup.
- The system would be better if it were wireless.

Subject D:

- The system should be wireless.
- On the positive side the touch pad is probably the most applicable to this particular user.

Subject E:

- The system should be wireless.
- The design need to be more compact.
- The system could be more aesthetically pleasing.
- On the positive side the system is very simple to use.

6.6 Analysis of Results

6.6.1 Input Sensor Characteristics

Ease of Use: One can determine from Figure 6.1, that all the sensors were relatively easy to use, with the dual-axis accelerometer scoring the highest average score of 4.4. The touch sensors and SCATIR switch scored well with average scores of 4.2 and 3.2 respectively. The average score for the SCATIR switch sensor was heavily reduced by a score of 1 given by *Subject D*, due to the fact that this particular user had poor facial muscle control. More specifically the subject's blinking action was not distinguishable over a winking action. The touch sensor keys were spaced too closely together or too far apart for *Subjects C* and *D* respectively. This could be rectified by ensuring that the touch sensor keys could be customised to different users' hands.

Comfort: The touch sensors and SCATIR switch were the most comfortable sensors to use, with scores of 4.4 and 4 respectively. The dual-axis accelerometer was the least comfortable to use of the sensors provided, averaging 3.4. This modest score could be attributed to the fact that it was slightly too large or bulky (as noted by *Subjects A*, *C* and *E*) and had to be attached to a peaked cap. As discussed with *Subject A*, the accelerometer would be less cumbersome and obtrusive if it were much smaller and attached to e.g. a set of eyeglasses, such that it would be less noticeable. As expected, *Subject C* gave the lowest score for the touch sensors, with a comfort level score of 3. This is expected, as this user also found this sensor the most difficult to use. This is also true for *Subject D*, who scored the SCATIR switch the lowest with a score of 2. This result again lowered the overall average score for this sensor, as the user had to perform the blink/wink actions multiple times to try and provide an accurate voluntary output. This was, however, achieved with little success, which in turn made this sensor more uncomfortable for the volunteer to use. From this, one can deduce that the ease of use and comfort levels are generally related. This becomes more evident when comparing these levels on Figure 6.2, where the highest scores for ease of use also resulted in high comfort level scores.

Performance and Predictability: Good to excellent results were produced, pertaining to the performance and predictability of the various input sensors. The dual-axis accelerometer and touch sensors scored the highest, with both averaging 4.4 and the SCATIR switch scoring the lowest, with an average performance level score of 3. The overall score was again lowered by a user rating of 1 given by *Subject D*, which means that this sensor was totally unpredictable, and consequently detected involuntary movement. The SCATIR switch also scored modest performance values of 3 for three of the users, with only one user scoring 5. The reason for this is that this particular sensor took some getting used to, where the sensor needed to be set to the particular user's facial movements (blinking/winking) in order to provide more predictable results. This feedback was given by *Subjects A*, *B* and *E*.

Fatigue: Excellent results were scored for the fatigue levels experienced by the subjects, when utilising the various input sensors, where a lower score meant that less fatigue was experienced by the user. The touch sensors scored the best with a perfect average of 1, meaning that no fatigue was experienced by the users, throughout the duration of testing. The SCATIR switch and dual-axis accelerometer also scored well, with overall average scores of 1.4 and 1.6 respectively. This meant that no to little fatigue was experienced throughout testing. *Subject D* scored the worst fatigue levels, with scores of 3 and 2 for the dual-axis accelerometer and SCATIR switch respectively, meaning that little to mild fatigue was experienced, when utilising these sensors. The reason for this, may be because the dual-axis accelerometer required slightly more movement to trigger a response, when compared to the rest of the sensors. This can be improved by setting the sensor to the particular user's head-movements, i.e. by tweaking the potentiometer screws located on the accelerometer sensor.

By comparing the fatigue with the performance & predictability levels, one can ascertain that they are directly related. This is especially the case with the SCATIR switch, which was the least predictable of all the sensors used. This was because the subjects were required to perform the winking actions multiple times to activate an output response, which increased the level of fatigue experienced by the facial muscles. By comparing the above mentioned input sensor parameters, one can determine that the **touch sensors** rated the best, on average, of all the sensors used.

6.6.2 The Overall System Characteristics

Figure 6.5 plots various information regarding the overall system operation.

"Customisability a": The system accepts all or most of the input sensors as required, with an overall average score of 4.8. A solitary, non perfect score of 4, was given by *Subject D* because the volunteer couldn't really use the SCATIR switch properly (as mentioned above), and therefore had little proof that the system could utilise this sensor as required.

"Customisability b": The system is also very customisable in terms of providing a specific output for a specific voluntary input, averaging 4.8. The lowest score of 4 was given by *Subject B*, which could have resulted, as only a few output function examples were provided for testing. More examples of software applications (e.g. using a word processor or email application) and hardware device control, may have resulted in a perfect average.

System Repeatability: A good average score of 4.4 was given for the system repeatability, meaning that the system is capable of repeating specific tasks multiple times and with a fair amount of accuracy.

Performance: This also scored fairly well with an average score of 4.2. The two lowest scores of 3, were given by *Subjects C* and *E*, because the program halted during testing. This lowered the average scores, given for the repeatability of the system, because certain tasks could not be repeated when required, as a result of the application halting. This occurred because too many samples were being written to memory, with the buffer size not large enough to handle this large influx of data. The program had to subsequently be restarted, which fixed the problem. This situation could be avoided by increasing the performance of the computer (memory and processing capabilities) or the samples/channel could be reduced, thereby reducing the number of samples required. The disadvantage of reducing the number of samples/CH, however, is that the FFT data would less accurate as less data would be acquired and processed. This therefore leads to a trade-off between the number of samples/channel and the accuracy required. The best solution is to design the code for efficiency, while utilising a less graphics intensive interface. This would increase the memory and processing requirements of the computer using the application.

The User Requirements: This scored the lowest of those concerning the overall system parameters, with an overall average score of 3.8, meaning that the system is can potentially satisfy the various users' diverse requirements. *Subject E* scored this the lowest with an average score of 3, meaning that only some of the volunteer's requirements could be met by utilising the system. According to this volunteer, the score would have been improved if different input devices were used, such as an eye-gazing sensor or a tracker-ball mouse. This would better demonstrate that the system was capable of utilising further input sensors, which would better meet the specific requirements of the volunteer.

6.7 Suggested System and Design Improvements

The majority of the volunteers suggested that the system would be greatly improved if it were wireless. The system would be greatly improved if wireless input sensors were used, which could transmit to a wireless receiver mounted on the signal selection and conditioning board. This would allow the user far more freedom as he/she would not be restricted by the length of cord of the input sensors attached to the system. Another recommendation suggested by the volunteers is that the system is far too bulky and should be more compact in design. This may be achieved by using a laptop instead of a PC, with the rest of the system components reduced in size by using e.g. dual-sided PCBs and compacting all of the system components, with exception of the input sensors, into to a single unit connected to the laptop. This would also reduce the setup time of the system and allow for easier mobility as the system would consist of fewer and more compact components. As suggested by *Subject C*, the "Generic Assistive Software Application" should also be further simplified and easier to use, i.e. it should be more user friendly. This would ensure that users with very basic computer literacy skills could setup and use the software effectively.

Most of the volunteers who performed the tests, also stated that one of the more useful technologies, which would better suite their requirements, is the use of voice activation software. The advantage of this form of input is that it can be used by different people suffering with varying degrees of motor deficiency, as even those suffering with very severe motor disabilities should be capable of speaking. Also, speech based commands would be limitless in their capabilities to provide the required output actions accurately and predictably, with no fatigue resulting. This contrasts with sensors, which are used to detect the disabled person's retained muscle movement, which may result in some fatigue being experienced by the user over an extended period of use.

Another limitations of the design is that it cannot output more than two voltages from the DAQ card, meaning that a maximum of two physical devices can be controlled by the assistive system at any one time. A slightly more expensive DAQ card, which consists of more output channels, would need to be used instead if additional devices were to be controlled by the system.

Chapter 7

Conclusion and Recommendations

7.1 Recommendations for Further Research

Reliability and validity are some of the more important characteristics required of assistive systems (Section 4.2). These characteristics were tested and scored in this pilot study. These need to be further improved, in order to produce a system which could reliably be utilised by severely disabled people. Better knowledge and research into motor disabilities would be required, to accurately model the retained muscle function of disabled people. To achieve this, the system would need to be customisable to each individual's retained and voluntary muscle movements, while utilising the correct input sensors for detection. The data obtained from these input sensors would thereafter be stored in a database and subsequently sent through a neural network, which could be trained to recognise the voluntary movements of the particular individual, over those which are accidental or involuntary.

Cellular and other telecommunications technologies could be used, to allow more freedom of movement for the end users. This means that the disabled person could leave his/her premises, without the restriction of having to be in the immediate vicinity of the system to utilise it. This could be achieved by using blue-tooth enabled wireless sensors, which would transmit a signal to and from a cellular phone and subsequently link to the rest of the blue-tooth enabled system using 3G/GPRS protocols. This is a possibility, especially with the increased reliability of GSM networks and computers.

Future work should also be directed at investigating further applications of this system, for use at different stages of the user's life, i.e. for educational purposes, employment and independent living etc. As discussed in section 6.7, different applications may require more suitable input devices than those demonstrated during testing. A personal computer is probably one of the most common technologies used in the workplace, at home and at schools. Eye-gazing technology could be used to control the mouse pointer on a PC, where voice recognition software could be used to write a document or email using speech-to-text inputs. Other examples of computer input devices include ergonomic keyboards, such as the raised keyboard, and the tracker-ball mouse (section 2.4).

Another future development of the device could include its use in Telecare, to offer remote care for disabled people. This would enable disabled individuals to remain independent in their own homes by providing person-centred reactive technologies to support the individual. Various sensors, such as a sip-puff switch, touch pads, single-switch access, joysticks (using the chin for control) or glove controllers (if residual movement of the hand is still present) (section 2.4) could be used by the disabled individual to control or send out a warning or emergency alert to the Telecare system through the use of the generic assistive system.

7.2 Conclusion

The essential research objective was the development of a generic assistive platform, which could accommodate various patients suffering from a wide range of motor disabilities and with fairly diverse requirements.

This was achieved by firstly developing and testing methodologies to assess and characterize motor disabilities and thereafter developing the system, such that it could maximize the retained motor function of the disabled individual. To ensure this, three input sensors were chosen which would make used of the limited or retained motor function available to the user; namely a dual-axis accelerometer (tilt-sensor), a 6-key touch sensor and a SCATIR (blink/wink) switch.

The design stages formed the largest portion of the research and proved challenging, as the final design needed to behave as a stand-alone generic system, which could be adaptable to the end-user's requirements and was therefore named the "Generic Assistive System". To ensure success, all stages needed to be designed with the proceeding stages in mind. This was illustrated vividly when designing the signal conditioning unit; as the A/D stage, to follow, needed to be taken into account when determining the filter order and type, as well as the sampling rate of the anti-aliasing filter. The input stage involved the selection criteria employed in determining the above mentioned input devices for the research, as well as their design.

The stage following thereafter concerned the design of the signal selection and conditioning circuit. This particular design is very unique, in that it utilises a solitary signal conditioning unit, which provides analogue filtering and anti-aliasing, as well as the amplification of the various input signals, which have been connected to multiple input channels. The data acquisition and control design stage centered on the A/D conversion, which would behave as an interface between the software application and the analogue signals, output from the signal selection and conditioning unit.

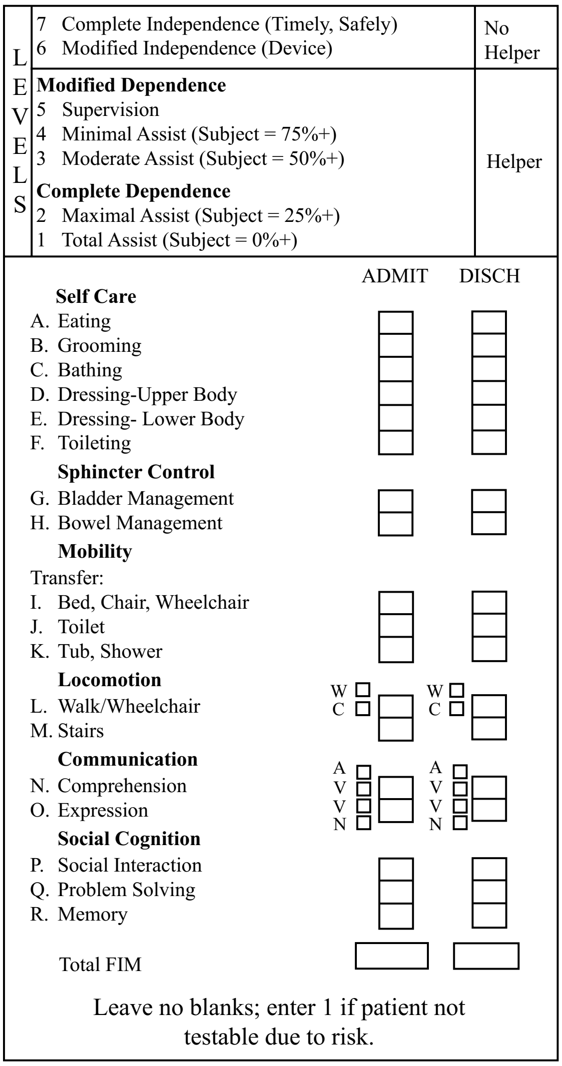
The software design stage involved the development of a stand-alone software application, which would provide the major portion of customisability to the system. The software application allows a user to customise the digital signal processing (DSP) of the various input signals, by allowing him/her to alter the digital filtering, windowing and FFT behaviour. The user would also be given the ability to alter the output functionality of the system, which formed the final design stage. This was achieved by allowing the user to tailor the application to perform various software-based tasks or even allowing the user to control a range of hardware devices, such as a wheel-chair or light switch. In order to control these physical devices, the user would need to customise the type of signal and amplitude required to operate these devices. These signals could be output in analogue and/or digital form from the DAQ card. The system was most importantly designed, such that it is non-invasive and safe for use by all prospective subjects. Methodologies were established on whether the design could be made sufficiently flexible, such that it could be programmed to suit these people in terms of their needs and level of motor disability. This needed to be achieved without re-designing the system for each person. Various subjects were subsequently invited to test the system and thereafter requested to answer a series of questions regarding the performance and potential usefulness of the system.

The input sensors were found to be comfortable and easy to use, with very little to no fatigue experienced when utilising them. Excellent results were obtained for the performance and predictability of the tilt switch and touch sensors, with the blink/wink switch scoring a modest rating. The system performed as expected and accepted all of the input sensors attached to it. The system repeated specific tasks multiple times and it was also established that the system was customisable in terms of providing a specific output for a specific and voluntary input.

The system could be improved by further compacting and simplifying the design and operation. Wireless sensors would consequently allow the user far more freedom as he/she would not be restricted by the length of cord of the input sensors attached to the system. It was thereafter concluded that the system was capable of satisfying the various users' diverse requirements, thereby achieving the required objectives.

Appendix A

Classification of Spinal Cord Injuries



LEVELS		
	7 Complete Independence (Timely, Safely) 6 Modified Independence (Device)	No Helper
	Modified Dependence 5 Supervision 4 Minimal Assist (Subject = 75%+) 3 Moderate Assist (Subject = 50%+) **Complete Dependence** 2 Maximal Assist (Subject = 25%+) 1 Total Assist (Subject = 0%+)	Helper

	ADMIT	DISCH
Self Care		
A. Eating		
B. Grooming		
C. Bathing		
D. Dressing-Upper Body		
E. Dressing- Lower Body		
F. Toileting		
Sphincter Control		
G. Bladder Management		
H. Bowel Management		
Mobility		
Transfer:		
I. Bed, Chair, Wheelchair		
J. Toilet		
K. Tub, Shower		
Locomotion		
L. Walk/Wheelchair	W C	W C
M. Stairs		
Communication		
N. Comprehension	A V V N	A V V N
O. Expression		
Social Cognition		
P. Social Interaction		
Q. Problem Solving		
R. Memory		
Total FIM		

Leave no blanks; enter 1 if patient not testable due to risk.

Figure A.1: The Functional Independence Measure (FIM) [2]

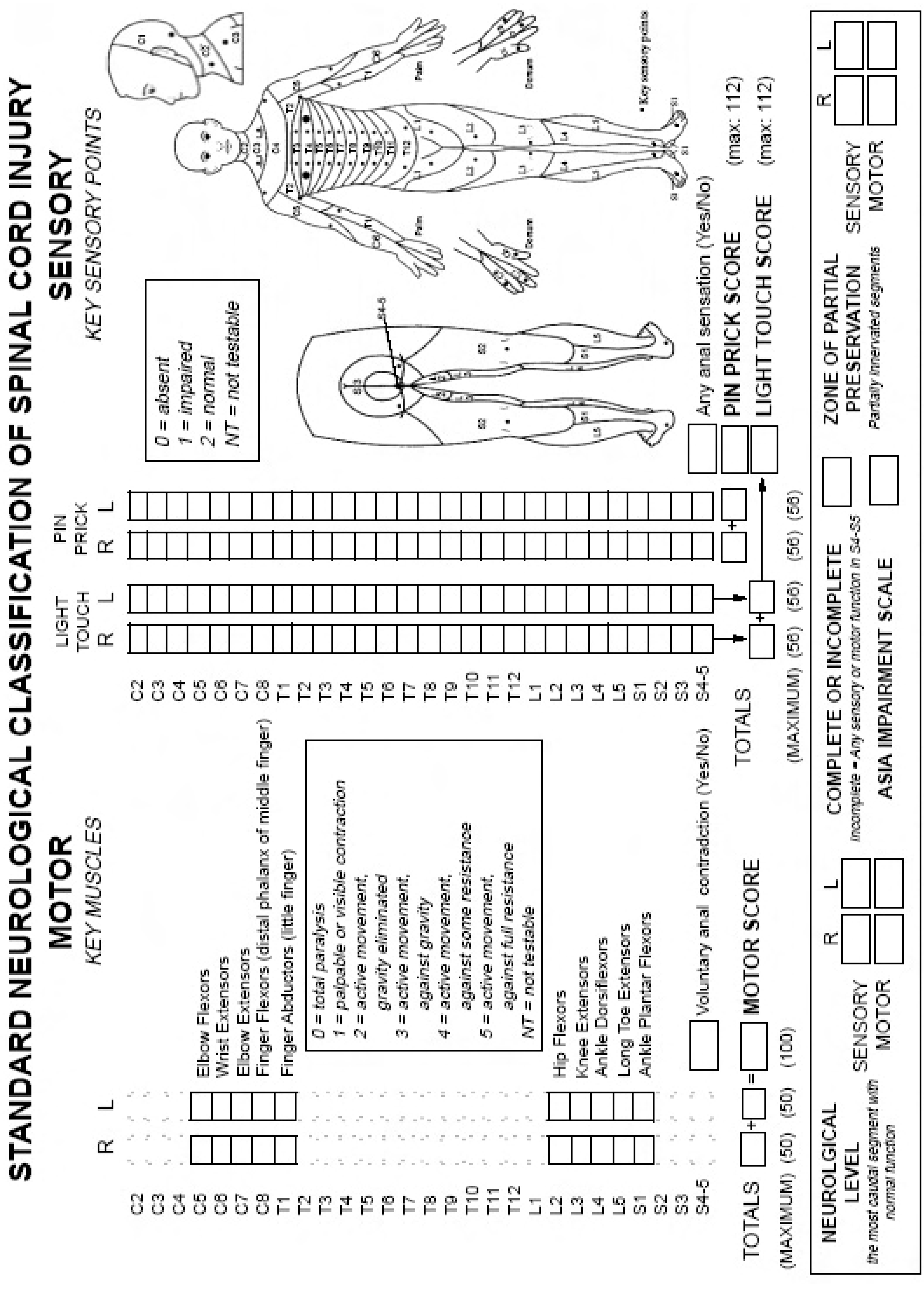


Figure A.2: Standard Neurological Classification of Spinal Cord Injury [3, 2]

Appendix B

Supplementary Prior Design Information

B.1 Introduction

Prior design information must be researched and understood before the final system can be designed and subsequently implemented. The following chapter discusses, in detail, the theory regarding some of the components which make up the system to be designed, as well as various design considerations that must be taken into account prior to commencing the final design and implementation stage of the project.

B.2 Generic Assistive System Principles and Components

This section provides some background information on the components and principles forming the generic assistive system being designed.

B.2.1 Input Sensors

A variety of input sensors have been chosen for the design of the assistive system. These include a dual-axis accelerometer, a 6-channel touch sensor and a self calibrating auditory tone infrared (SCATIR) switch.

The Dual-Axis Accelerometer

Accelerometer Principles: An accelerometer measures acceleration by detecting a change in the speed and/or direction of a body. The operating principle of the accelerometer couples a mass to a sensor that can generate an electrical signal. The sensor produces an output signal that is proportional to the acceleration. Acceleration is measured as the rate of change of the velocity that a mass undergoes when it is subjected to a force (Newton's 2nd law of motion) i.e.

$$Force = mass \times acceleration \tag{B.1}$$

Where, acceleration $[m/s^2]$ and mass $[Kg]$.

Note that the velocity is a vector quantity and consists of components for speed and direction. An object, therefore, undergoes acceleration when its speed, direction or both change. Furthermore, the direction of the accelerometer can be multi-dimensional, requiring a multi-axial accelerometer for complete measurement. Accelerometer mounting can be critical for achieving satisfactory results; a misalignment of only a few degrees can have a profound effect on the final result [39].

The ADXL210 Dual-Axis Accelerometer: The ADXL210 accelerometer is a Micro Electro Mechanical System (MEMS) capacitive dual-axis accelerometer. It is a low cost, low power sensor which incorporates 2-axis accelerometers with a low-g or measurement range of either $\pm 2g / \pm 10g$. The ADXL210 can measure both dynamic acceleration (e.g., vibration) and static acceleration (e.g.,gravity). Voltage outputs, proportional to acceleration, is available from the XFILT and YFILT pins. The bandwidth of the ADXL210 may be set from 0.01 Hz to 5 kHz via capacitors CX and CY (as on the data-sheet referenced by [23]).

As the accelerometer is to be used to measure vibration and distinct movements, it is important to consider the frequency of the signal. The sampling rate and signal conditioning chosen must capture the desired information while avoiding aliasing. As the waveform needs to be reconstructed in great detail, then the sampling rate must be chosen so that it is many times the highest frequency component of the vibration/movement [23].

The Quantum™ QT160 touch sensor

Theory of Operation: The basic theory of operation of the touch-sensor is that the *QT160* IC employs bursts of charge-transfer cycles to acquire signals. It detects the capacitance from a charge sampler capacitor C_s as well as the user's capacitance C_x and thereafter provides a DC output voltage of V_{DD} = 5 V when detection has occurred. The power supply voltage V_{DD} is used as a reference to provide the output DC voltage. The main purpose of Cs is to determine the available resolution required by the user. Larger values of $C_s \approx 47\mu F$ result in a higher resolution, whereas when C_x is high the resolution decreases (Section 1.1 of [24]). The rule $C_s >> C_x$ must be observed for proper operation. C_s must therefore occur between $10\mu F$ and $47\mu F$ to ensure that C_s is sufficiently larger than C_x.

Changing the Sensitivity: The sensitivity may be altered to suit various application and situations on a channel-by-channel basis. The easiest way to change the sensitivity is to decrease or increase C_s (between $10\mu F$ and $47\mu F$), thereby decreasing or increasing the sensitivity respectively. Other methods of changing the sensitivity of the sensor are discussed in Section 1.3.6 of [24].

Detailed information, regarding the QT160 touch sensor, is provided in [24].

The Self-Calibrating Auditory Feedback (SCATIR) Switch

Photoelectric Sensors: Photoelectric sensors utilise photoelectric emitters and receivers to detect presence, absence, or distance of target objects. They can be categorized into three main categories, *through beam*, *retroreflective*, and *proximity* [40].

- **Through beam photoelectric sensors** are designed with the emitter and detector opposite the target or intended object's path and sense presence when the beam is broken.
- **Retroreflective photoelectric sensors** are designed with the emitter and detector in the same housing and rely on a reflector to reflect the beam back across the path of the target. This sensor is usually polarized to minimize false reflections.
- **Proximity photoelectric sensors** have the emitter and detector in the same housing and rely upon reflection from the surface of the target. This mode can include presence sensing and distance measurement via analogue output. The proximity category can be further broken down into five sub-modes: *diffuse*, *divergent*, *convergent*, *fixed-field* and *adjustable field* [40].
  - With a diffuse sensor presence is detected when any portion of the diffuse reflected signal bounces back from the detected object.
  - Divergent beam sensors are short-range diffuse-type sensors without any collimating lenses.

- Convergent, fixed focus, or fixed distance optics (such as lenses) are used to focus the emitter beam at a fixed distance from the sensor.
- Fixed-field sensors are designed to have a distance limit beyond which objects are no longer detectable, no matter how reflective these objects are.
- Adjustable field sensors, as with fixed-field sensors, utilize a cutoff distance beyond which a target will not be detected, even if it is more reflective than the target. This sensor differs, however, in that the cut-off distance can be set by the user. Some photoelectric sensors can be set for multiple different optical sensing modes.

Reflective properties of the target and environment are important considerations in the choice and use of photoelectric sensors.

Photoelectric Sensor Parameters Important photoelectric sensor parameters include *sensing mode*, *detecting range*, *position measurement window*, *minimum detectable object*, and *response time*.

- Sensing modes is the presence or absence and position measurement. With a presence or absence sensor, the sensor detects presence or absence in an on/off mode. In a position measurement sensing mode the sensor can detect position in a linear region by the intensity of reflected light. Analogue output is linear with position in the measurement range.
- The detecting range is the range of sensor detection. For presence sensors, this goes up to the maximum distance for which the signal is stable. For position measurement sensors, this is the distance range over which the position vs. output response is linear and stable.
- The position measurement window is the width of linear region for the sensor. For example, if the sensor could measure between 5 and 10 cm, this window would be 5 cm.
- The minimum detectable object is the smallest sized object detectable by the sensor.
- The response time is the time from target object entering detection zone to the production of the detection signal.

Common configuration features for photoelectric sensors include beam visibility, light-on or dark-on modes, light and dark programmability, adjustable sensitivity, self-teaching, laser source, fiber optic glass, and fiber optic plastic. The body style of the sensor can be threaded barrel, cylindrical, limit switch, rectangular, slot, ring, and window or frame. The sensor may be self-contained and may have a remote head [40].

Theory of Operation The Self-Calibrating Auditory Tone Infrared (SCATIR) Switch is an experimental multipurpose momentary-contact optical switch with auditory feedback designed for use by persons who experience difficulty in activating mechanical switches. It works by detecting a beam of reflected pulsed infrared light, more specifically an infra-red-sensitive phototransistor. These two electro-optical devices are positioned so that they share virtually the same optical axis. This becomes the optical axis of the sensor. The switch is activated whenever an infrared-reflective surface (such as skin or light clothing) is placed on this optical axis at a distance within the range of the switch [25].

It is suitable for use with a variety of control gestures, including eye-blink, eyebrow movement, finger movement, head movement, and facial muscle movement. Because it works on an optical principal, it can be activated at a distance. The controlling body part therefore need not be in physical contact with the switch sensor.

B.3 Filtering

A filter can be defined as a signal processing system which has an output signal (or response) which differs from the output signal (or excitation), such that the output signal has certain properties [4].

Filtering is important for two functions. These include (1) anti-aliasing and (2) improving the signal-to-noise ratio (SNR). Analogue filters are always used for the design of anti-aliasing filters and digital (discrete) filters are used for noise reduction, which provides the desired response. This is specified in the frequency domain in terms of the desired magnitude response and/or the desired phase response. The *squared magnitude function* of a low pass analogue filter is illustrated in Figure B.1 below [4].

The *passband* of a low-pass filter is the region in the interval $[0, \Omega_p]$, where the desired characteristics of a given signal are preserved. In contrast, the *stopband* of a low-pass filter (the region $[\Omega_s, \infty]$) rejects signal components. The *transition band* is the region between $(\Omega_x - \Omega p)$, which would be 0 for an ideal filter [4].

B.3.1 Analogue Filtering

As noted above, analogue filtering is only used when performing antialiasing and not really for significant noise reduction. The antialiasing design choices and implementation are discussed in further detail in section 5.3.2 of the *Design and Implementation* document.

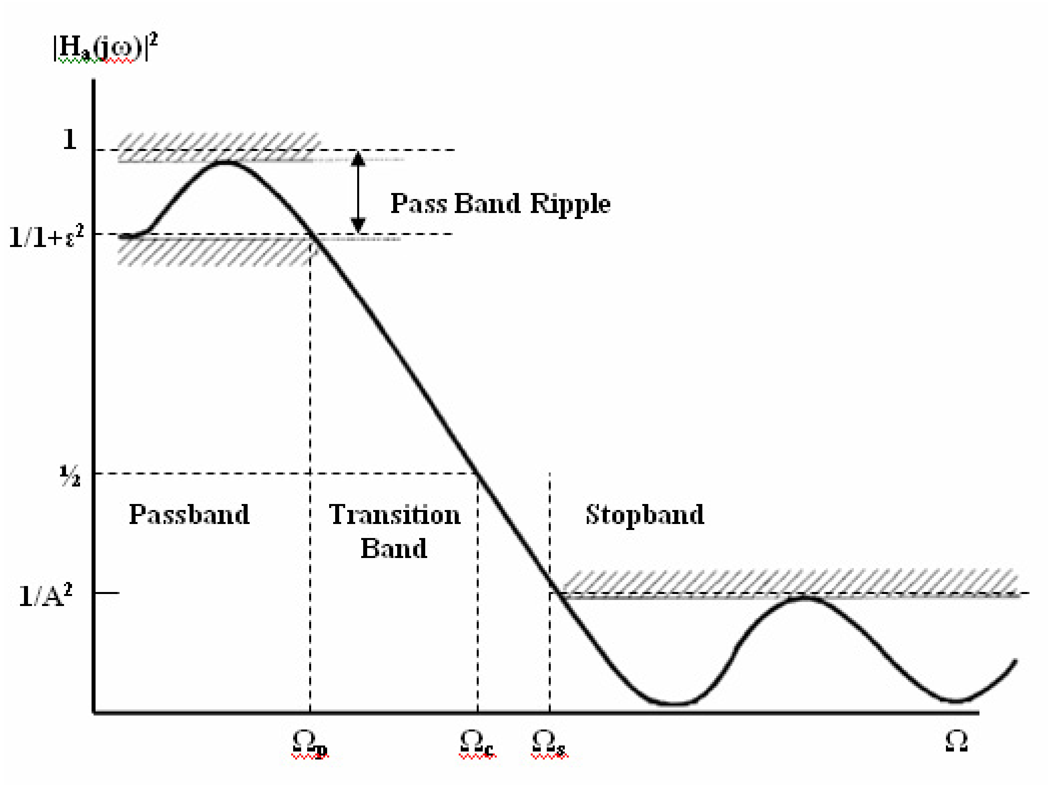


Figure B.1: The squared magnitude function of an analogue filter with ripple in the passband and stopband [4].

- Ω_p : Passband cutoff frequency (rad/s)
- Ω_s : Stopband cutoff frequency (rad/s)
- $\Omega_(c)$: -3 dB cutoff frequency (rad/s)
- ϵ : Permissible error in passband
- *1/A* : Permissible maximum magnitude in the stopband

B.3.2 Digital Filtering

The following procedure should always be followed in choosing a digital filter [4]:

1. Determine the desired response, which in this case is specified in the frequency domain and is specified i.t.o. the desired magnitude response and/or phase response.
2. Select the class of filters. In this (e.g. linear-phase FIR filters or IIR filters) to approximate the desired response.
3. Select the best member of the filter class.
4. Implement the best filter using a DSP.
5. Analyse the filter performance to determine whether the filter satisfies all the given criteria.

B.4 Spectral Analysis

Spectral analysis estimates the distribution of signal power at different frequencies. Spectral analysis and correlation techniques are an aid to the interpretation of the signals and to the systems that generate them.

As will be seen later, the signal-processing will be performed by a computer software package interfacing with dedicated instruments. This will, consequently allow the user to perform spectral analysis by utilising a mouse, thereby providing a visualisation of the results on a graphical plot. This saves time and money, however, the limitations of the fundamental techniques must be taken into account. This is needed in order to prevent misleading results [35].

B.4.1 Background Information

Once a continuous signal has been band-limited by the analogue antialiasing filter it is passed through an ADC, which transforms into a digital signal, thereby allowing it to be analysed by digital signal processors (DSPs). This signal is called x_n and represents a discrete-time signal with samples at $n = 0, 1, 2, ..., N1$. The Fourier theorem states that it is possible to decompose x_n as a sum of cosine and sine waveforms of different frequencies using an appropriate combination of amplitude coefficients [35]. Therefore,

$$x_n = a_0 + \sum_{k=1}^{N-1} a_k \cos\left(\frac{2\pi kn}{N}\right) + \sum_{k=1}^{N-1} b_k \sin\left(\frac{2\pi kn}{N}\right) \tag{B.2}$$

where $k = 1, 2, ..., N-1$ determines the frequency of each cosine and sine waveforms as $f_k = k/N\Delta t$. The corresponding coefficients are calculated from

$$a_0 = \frac{1}{N} \sum_{n=0}^{N-1} x_n \tag{B.3}$$

$$a_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n \cos\left(\frac{2\pi kn}{N}\right) \tag{B.4}$$

$$b_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n \sin\left(\frac{2\pi kn}{N}\right) \tag{B.5}$$

Note that a_0 represents the mean value of x_n. From Eulers formula, it is possible to combine the cosine and sine terms to express the DFT in exponential form:

$$e^{j\theta} = cos\,\theta + j\ \sin\ \theta \tag{B.6}$$

leading to

$$x_n = \sum_{k=0}^{N-1} c_k\, e^{j(2\pi kn/N)} \tag{B.7}$$

where

$$c_k = \frac{1}{N} \sum_{k=0}^{N-1} x_n\, e^{-j(2\pi kn/N)} \tag{B.8}$$

A graphic representation of the a_k, b_k and c_k coefficients for each value of k (or f_k) constitutes the *frequency spectrum* of x_n, expressing the relative contribution of different sinusoidal frequencies to the composition of x_n (Equation B.7). Since c_k is complex, a more meaningful physical interpretation of the spectrum is obtained with the *amplitude* and *phase spectra*, defined as

$$A_k = (a_k^2 + b_k^2)^{1/2} = |c_k| \tag{B.9}$$

$$\theta_k = \frac{b_k}{a_k} \tag{B.10}$$

As shall be discussed in section B.4.4 below, it is also important to apply a *window* to the data to **minimize the phenomenon of leakage**. For $k > N/2$ both spectra present symmetrical values. This can be easily demonstrated from the fact that cosine functions have even symmetry while sine has odd symmetry. From Equations B.9 and B.10 it follows that A_k and θ_k have even and odd symmetry, respectively [41]. Only **half the spectral components** ($k \leq N/2$) are therefore required to give a complete description of x_n in the frequency domain. The *amplitude spectra* indicates the combined amplitude of the cosine and sine terms to reconstruct x_n and also reflects the signal power at different frequencies. The *phase spectra* reflects the relative phase differences (or time delays) between the sinusoidal waveforms to generate the temporal pattern of x_n [35].

B.4.2 Fast Fourier Transform (FFT)

The concepts of spectral analysis of various input signals is based on the *fast Fourier transform (FFT)* approach. The FFT is not a single algorithm, but rather a large family of algorithms which can increase the **computational efficiency** of the DFT [35]. As demonstrated by Panerai et al., if the given number of samples $N = 1024$, using an FFT algorithm is 100 times faster than the direct implementation of Equations B.7 and B.8, which is especially important when using a real-time system such as the one being designed.

Most FFT algorithms are based on the principle of *decimation-in-time*, involving the decomposition of the original time (or frequency) sequence into smaller subsequences. To understand how this decomposition can reduce the number of operations, assume that N is even. In this case it is possible to show that Equation B.8 can be written as [42, 43].

$$c_k = \frac{1}{N}\sum_{r=0}^{(N/2)-1} x_r^e \cdot W_{N/2}^{kr} + \frac{1}{N}\sum_{r=0}^{(N/2)-1} x_r^o \cdot W_{N/2}^{kr} \tag{B.11}$$

with

$$W_N = e^{-j(2\pi/N)} \tag{B.12}$$

where x_r^e and x_r^o represent the even- and odd-order samples of x_n, respectively. Comparing Equations B.11 and B.8, it is clear that the latter represents two DFTs with dimension $N/2$, involving $2(N/2)^2$ operations rather than the N^2 operations required by Equation B.8. This process of decimation-in-time can be carried out further to improve computational performance and using the procedures followed by Ronney Panerai et al. it is possible to show

$$\text{efficiency gain} = \frac{N}{log_2 N} \tag{B.13}$$

Where the number of samples is N = 1024, the efficiency gain is approximately 100 as discussed above [35].

B.4.3 FFT Spectral Analysis

The previously mentioned deterministic signals, x_n can be expressed by a mathematical function and the amplitude and phase spectra can be calculated as an exact solution of Equations B.8, B.9 and B.10. In most practical applications, there is a need to perform spectral analysis of experimental measurements, corresponding to signals which, in general, cannot be described by simple mathematical functions. In this case the spectra has to be estimated by a numerical solution of Equations B.9 and B.10, which can be efficiently implemented on a digital computer with an FFT algorithm. Considerable distortions can result from applications of the FFT unless attention is paid to the following characteristics and properties of the measured signal and the DFT/FFT [35].

Limited Observation of signal in time

The limited observation of a *periodic* signal x_n in time is the multiplication of the original signal x_n^∞ by a rectangular window of duration $T = N\Delta t$ as exemplified by a single sinusoid, where the DFT is assumed to be periodic with period T. For example if the discrete input signal is a sinusoid of period $T = 20ms$, instead of a single harmonic at the centre frequency $f_c = 1/T = 50Hz$ of the original sinusoid, the power spectrum estimated with the FFT will have power at *other harmonics*. The spectral power, which should have been concentrated on a *single harmonic* $f_c = 50Hz$, has "leaked" to neighboring harmonics and for this reason this phenomenon is usually called leakage. This adds spurious power to neighboring harmonics and restricts the frequency resolution of the main spectral peaks [35].

The effects of spectral leakage can be reduced by

1. Increasing the period of observation and
2. Multiplying the original signal x_n by a window function with a smooth transition, as discussed in section B.4.4 below.

B.4.4 Windowing

The Fourier transform of a window function with tapered ends has smaller side lobes, thus reducing the unwanted leakage effects [35]. There are a large number of tapering windows which may be used, such as the rectangular, Hanning, Hamming and 4-term BlackmanHarris windows.

An example of the 4-term Blackman-Harris is provided as follows

$$\mathrm{w}_n = a_0 - a_1\, cos\left(\frac{2\pi n}{N}\right) + a_2\, cos\left(\frac{4\pi n}{N}\right) - a_3\, cos\left(\frac{6\pi n}{N}\right) \qquad n = 0, 1, 2, ...N - 1 \tag{B.14}$$

Windowing *attenuates* the contribution of signal samples at the beginning and end of the signal and, therefore, reduces its effective signal duration. This effect is reflected by the equivalent noise bandwidth (ENBW) defined as

$$\text{ENBW} = \frac{\sum_{n=0}^{N-1} \mathrm{w}_n^2}{\left[\sum_{n=0}^{N-1} \mathrm{w}_n\right]^2} \tag{B.15}$$

For a rectangular window the ENBW = 1.0 and for the 4-term BlackmanHarris window the corresponding value is 2.0. The majority of other window shapes have intermediate values of ENBW [44].

B.5 Conclusion and Summary

This document discussed various principles which needed to be researched and understood prior to commencing the final design stages. The theory of operation, principles and functionality was provided for the three input sensors chosen for use with the system; namely the dual-axis accelerometer, 6-channel touch sensors and the blink/wink SCATIR switch device. Filtering is one of the most important techniques utilised in control systems such as the one being designed. It can further be separated into analogue and digital filtering. Analogue filtering is performed on the analogue signal prior to the A/D stage in order to bandlimit the signal, so that aliasing does not occur when converting to a digital signal. It does not, however, perform significant noise reduction. This is performed by the digital filter, which forms part of the digital signal processor (DSP). In choosing a digital filter it is firstly important to determine the desired response and thereafter select the class of filter (IIR or FIR) and the best member of this filter class. This is followed by its implementation using the DSP and thereafter analysing the filter performance to determine whether the chosen filter satisfies all the given criteria.

Spectral analysis is used to estimate the distribution of signal power at different frequencies, using a DSP to perform a fast Fourier transform (FFT). The FFT is used to derive the plots for the magnitude and phase of a discrete periodic signal in the frequency domain and increases the computational efficiency of the discrete Fourier transform (DFT) 100 fold, when compared to a normal DFT computation. One problem, which may occur when utilising a FFT is "leakage", which adds power to neighboring harmonics of a DFT and restricts the frequency resolution of the main spectral peaks. To reduce this another important spectral analysis technique called windowing is used, which also ensures a smooth FFT transition.

Appendix C

Bessel Filter Design

C.1 Introduction

The following describes the stages followed in the design and implementation of the 8-pole analogue Bessel filter, for use in the Generic Assistive Device.

C.2 Filter Design Rationale

C.2.1 ADC Design Choice

It is important to note that the filter must be designed around the characteristics of the ADC being used, therefore, before the anti-aliasing filter can be designed, a choice of type of A/D converter needs to be decided.

As has been noted in Appendix 5 the *NI™ PCI-6221 DAQ* card has been chosen to perform the A/D conversion as well as other functions. This is discussed in further detail in Section 5.4.2 of the main document. The DAQ card's onboard ADC has a *16-bit resolution* and according to figure a-3 has an ideal SNR (or dynamic range) of 98 dB. The disadvantage, however, of using this ADC is that it is more sensitive to higher frequencies as a result of the high dynamic range. The 16-bit ADC has a sensitivity of up to 97.6μV at a $\pm$10V input range, with an average dynamic range DR = 98dB, which is 24dB higher than the average 12-bit ADC's dynamic range of DR = 74 dB [32] and equation C.3).

C.2.2 Filter Characteristics

Based on the ADC choice provided in Section C.2.1 above, the filter should be designed so that its gain at f_{sb} (stopband frequency) is at 98 dB below the passband gain [41]. Using figure 2.56 of [31], this particular stopband attenuation level is realised by utilising an *8-pole Bessel filter*, which has a stopband attenuation of $f_{sb} = 98dB \text{ @ } 8kHz$.

An oversampling rate of $f_{os} = 16kHz$ would, therefore be required for this filter to satisfy the Nyquist criterion. This sampling rate, however, is far *too large* as it *increases the performance requirements of the ADC*. Since the onboard *NI™ PCI-6221 DAQ* [32] card performs the A/D conversion, I/O and DSP functionality, the performance requirements of the PC's microprocessor would also be increased, as a result of this high sampling rate. The sampling has, hence been chosen at a more suitable level of $f_{sb} = 6kHz$, which has a stopband attenuation of -80 dB for an 8-pole Bessel filter design [45, 41] and [31]. The oversampling frequency is, therefore $f_{os} = 12kHz$, to satisfy Nyquist. This would mean that aliasing could possibly occur in the frequency range between 6 kHz (@ 80dB stopband attenuation) and 8 kHz (@ 98dB stopband attenuation), but since the attenuation levels between these frequencies are so low, the possible attenuation caused will not affect the overall signal performance (Figure 2.56 of [31]).

In practice, noise-signal amplitudes rarely match the amplitudes of signal components of interest, so this attenuation calculation represents the worst case scenario [45]. Also, there is little chance of aliasing as the Nyquist oversampling frequency is at $f_{os} = 12kHz$, which, indecently, occurs at the 140 dB (approx.) stopband attenuation level, which is far greater than the 98 dB level, occurring at 8 kHz [45, 41], [4].

C.2.3 Retained Motor Function

To satisfy the *Nyquist* criterion and starting with the fact that **human motion does not exceed** $f_c = 20Hz$ [45, 41], which in this case is the disabled individual's retained motor function. Initially:

$$f_{sb} = 2 \times f_c = 40Hz \tag{C.1}$$

, where f_c is the highest frequency component of interest of the input signal (-3 dB corner frequency) i.e. 20 Hz and f_{sb} is the stopband frequency where $f_{sb} = 40Hz$. To ensure that no aliasing results, the cut-off frequency has been *oversampled* and therefore increased to $f_c = 1kHz$.

C.2.4 Oversampling

Oversampling is an important technique employed when performing low-pass filtering as is used as it *improves the ADC resolution* and *increases the SNR*; as discussed below.

Improvement of the ADC Resolution

Oversampling ensures that no high frequency components are sampled, thereby preventing any possible aliasing. As discussed in Section C.2.2 the stopband frequency must be chosen with a $f_{sb} = 6kHz$ and with an associated oversampling frequency $f_{os} = 12kHz$, which has been chosen to satisfy Nyquist. According to C.2, this high oversampling frequency increases the *effective resolution* of the ADC by 4-bits from 16 to 20 bits. This is a far higher resolution than is required of low-frequency signals, such as those occurring as a result of human motion, but nevertheless improves the resolution of the original analogue signal.

To improve the resolution of the ADC from 16 to 20 bits, oversampling is performed:

$$f_{os} = 4^{\omega} f_{sb} \tag{C.2}$$

, where ω is the number of additional bits of resolution desired f_{sb} is the original sampling frequency requirement and f_{os} is the oversampling frequency

Therefore, $f_os = 4^4 f_{sb} = 256(40) = 10.24kS/s$, with an additional 4 bits of resolution. Adding 4 additional bits will, therefore result in $f_{os} \approx 12kS/s$, to satisfy the chosen oversampling frequency chosen above.

Increasing the SNR

Oversampling and averaging also increases the SNR (or dynamic range of the ADC) and is the most important reason for choosing such a high oversampling frequency, especially after comparing it with the original sampling frequency of 40 Hz [45, 41].

This oversampling would also improve the *SNR* or *dynamic range (DR)* of the ADC and is determined by [45, 41]:

$$DR = (6.02 \times ENOB) + 1.76 \tag{C.3}$$

, where ENOB is the effective number of bits, i.e. $SNR = (6.02 \times 20) + 1.76 = 122.17dB$, which is an excellent SNR, i.e. an increase of 24 dB from 98.08 dB (for the 16-bit ADC) [45, 41]. Oversampling from $f_{sb} = 40Hz$ to $f_{os} = 12kHz$ improves the SNR by 24 dB, which is quite significant and will help ensure that the digital signal is as error and noise free as possible before entering the A/D and DAQ (data acquisition) stages.

C.2.5 Quantisation Noise

The theoretical limit of the SNR of an ADC measurement is based on the *quantisation noise*, due to the quantisation error inherent in the A/D process, when there is no oversampling and averaging of the input analogue signal [45]. It is also important to note that the effectiveness of oversampling and averaging depends on the characteristics of the dominant noise signal. The key requirement is that the noise can be modeled as white noise, i.e. noise which has a uniform power spectral density over the frequency band of interest [41]. It won't be problematic, however, if oversampling and averaging is not effective in improving the SNR and resolution of the signal, as the SNR (dynamic range) and resolution of the 16-bit ADC is more than sufficient for this application.

C.2.6 Active vs. Passive Filter Design

The filter being designed is an *active low-pass filter*. **An active filter** is chosen over a passive LC filter as they are more practical, considering the fact that they can be designed at higher impedance levels so that the capacitor magnitudes can be reduced. Active filters are also generally smaller than LC filters since inductors are not required. The only limitation of active filters over passive LC filters is that they can only operate at frequencies of up to 50 kHz (newer models of up to 500 kHz are available at an increased cost). The active filter is suitable for this application, as the anti-aliasing filter is designed for sufficient noise rejection of signals (noise) above 6 kHz. As mentioned in Section C.2.3 the Bessel filter has been designed for a -3-dB cut-off frequency of $f_c = 1kHz$. This frequency forms the basis of the filter design.

C.3 Stages followed in the Bessel filter design

The stages involved in the design of the filter are discussed as follows:

C.3.1 Stage 1

This involves the computation of the steepness factor As i.e. the ratio of the 3-dB (corner frequency) and stop-band frequencies (refer to equation C.1). In this case the 3-dB frequency f_c is chosen as 1 kHz and the stop-band frequency as 6 kHz (at the highest possible frequency component occurring at a stopband attenuation of 80 dB) [41]. This relationship is described in the following equation C.4.

$$A_s = f_{sb}/f_c \tag{C.4}$$

The steepness factor is therefore $\omega = 6$.

C.3.2 Stage 2

The response curve of Fig 2.56 in [31] displays the attenuation characteristics for maximally flat delay Bessel filters, where the steepness factor is on the x-axis and the attenuation is on the y-axis (in dB). According to this response curve, the steepness factor of 6 occurs at - 80 dBs and the attenuation in the stop-band is achieved by an $n = 8$ (8^{th} order) Bessel filter, which is sufficient for the design.

To realise an 8^{th} order filter, four 2^{nd} order ($n = 2$) or 2-pole filter sections will be required. The following pole locations of a normalised 8^{th} order Bessel low-pass filter are obtained from Table 11-41 of [31]:

$$\begin{array}{rll} \text{Complex poles:} & \alpha = 1.7627 & \beta = 0.2737 \\ \text{and} & \alpha = 0.8955 & \beta = 2.0044 \\ \text{and} & \alpha = 1.3780 & \beta = 1.3926 \\ \text{and} & \alpha = 1.6419 & \beta = 0.8253 \end{array}$$

The normalised capacitor values for the four 2^{nd} order sections are calculated as follows:

$$C_1 = \frac{1}{\alpha} \tag{C.5}$$

$$\text{and} \qquad C_2 = \frac{\alpha^2}{\alpha^2 + \beta^2} \tag{C.6}$$

These values are already available from Table 11-43 of [31].

The normalised capacitor values for the 4 $n = 2$ sections are:

$$C_1 = 0.5673\,F \qquad C_2 = 0.5540\,F$$

$$C_1 = 0.6090\,F \qquad C_2 = 0.4861\,F$$

$$C_1 = 0.7257\,F \qquad C_2 = 0.3590\,F$$

$$C_1 = 1.1160\,F \qquad C_2 = 0.1857\,F$$

The normalised resistor values for all of the 2^{nd} order sections are normalised to 1Ω.

C.3.3 Stage 3

The normalised filter is denormalised by dividing all capacitor values by $FSF \times Z$, i.e.

$$C' = \frac{C}{FSF \times Z} \tag{C.7}$$

,where FSF is the frequency-scaling factor $2\pi f_c$ and Z is the impedance-scaling factor. The resistors are multiplied by Z, which results in equal resistors throughout of Z .

In this case, f_c is the 3-dB cut-off frequency of 1 kHz. The impedance scaling factor Z is chosen to be $1k\Omega$, as this resistive values is relatively small, and doesn't allow for high power dissipation, especially when dealing with small signals, such as those found in data acquisition and control systems. The 8-pole Bessel filter along with the finalised values for the elements are shown in Figure C.1 below.

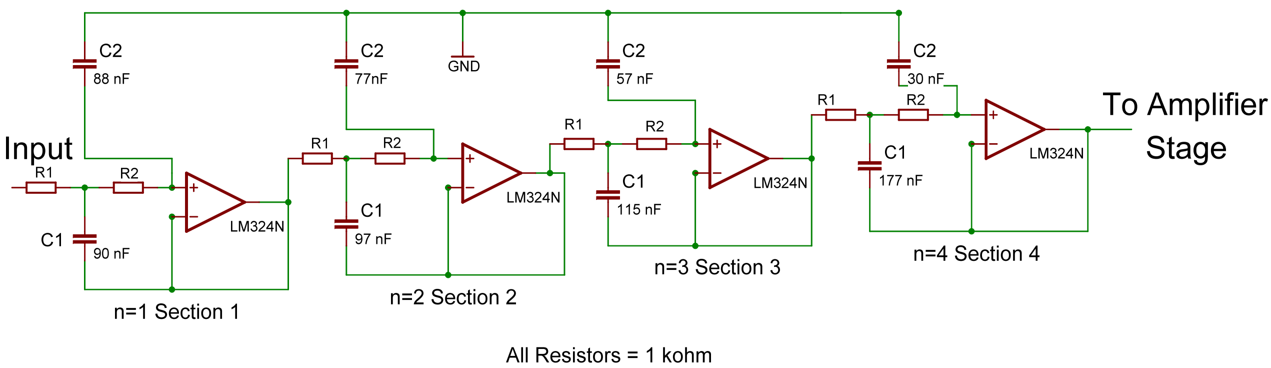


Figure C.1: 8-pole Bessel Filter

The problem with these values is that no capacitors are available with these values. A combination of capacitors had to be chosen to ensure that their values are as close as possible to those shown in the figure above. The actual capacitor values used in the design are shown in figure F.7 in Appendix F.

C.4 Conclusion and Summary

Analogue filtering forms as one of the most important design stages as it band-limits the the input signal thereby preventing aliasing of the signal during the A/D stage of the design. The filter chosen was a high order 8-pole active Bessel filter. The Bessel filter has the advantage that it contains no ripple in the pass or stopbands and has a constant group delay throughout the passband. A high order filter was chosen to increase the roll-off of the Bessel filter as well as achieve a stopband attenuation at $-80db$, in order to filter out signal frequencies which may occur at this low attenuation level when using the A/D, which incidentally has a high SNR (noise "floor") of $-98dB$. An active filter was chosen as its more practical and smaller than a passive filter. This filter also serves to increase the fidelity of the signal by increasing the signal-to-noise (SNR) as well as filtering out all high frequency components.

Appendix D

Requirements Engineering

D.1 Introduction

The following requirements engineering document describes the performance and operation of the proposed software program.

To ensure that the final assistive system is truly generic, a software program must be developed, which can be customised by the end-user, to perform a variety of basic everyday tasks. This software must therefore form as the interface between the input signals and the required output functions. Basic modeling of the software system as well as testing and validation shall be adopted to produce a prototype of the proposed system.

D.2 Program Requirements

The software program is to satisfy the following requirements:

- The system must allow the administrator to:
  - View a text file, stating the minimum system requirements as well as the installation procedure for the program.
  - Add, change or delete basic user information.
  - Add, change or delete the types of input devices being used, along with the associated channel.
  - Add change, delete various output function information.

  - Set or change a variety of input versus output function parameters (hardware and software outputs).
  - Customise the digital signal processing and timing parameters of the input signals.
  - Calibrate the input trigger variables, which will result in an output function.
  - Activate/discontinue the real-time data acquisition of the various input signals.
  - Select a help file, which will allow the administrator to search for help regarding the operation of the program as well as for troubleshooting problems incurred.

- It must operate in real-time
- The real-time digital plots of the various input signals, along with the FFT magnitude and phase plots must be displayed.
- The input signal FFT peak magnitudes and associated frequencies must be displayed numerically in real-time.
- The software program must be capable of interfacing with various input and output hardware devices as well as software programs.
- The software program must be compatible with popular operating systems such as MS Windows® .
- Most importantly; the proposed software program must be easy to use.

D.3 Overview Of The Program As A Result Of The Requirements

The software program must be capable of achieving the above design requirements, while providing an efficient and user-friendly interface. The software program is marketed at people suffering from severe motor deficiencies, who wish to simplify everyday tasks. These people are the "users" of the system. The program must (at least initially) be customised for the end-user by another able-bodied person, who will need to learn the program by utilising a simple and easy to follow user manual (refer to Appendix H). This person is the "administrator".

The administrator must firstly select the administrator options dialog available, which is to be used to add, change or delete: a) basic user information, b) the types of input devices being used, along with the associated channel and c) the various output function information. The administrator must thereafter select the output function dialog. This is required in order to set or change various output function parameters, which may be a physical output signal or software in nature. The final stage involves the selection of a general data acquisition dialog. This selection will allow the administrator to change various fields such as the digital filtering, windowing, timing and input voltage range parameters , of the input signal. A calibration routine must be selected in order to calibrate the trigger level/magnitude , frequency range and hysteresis of the input signal, which is required to generate the output function. These results must be saved to a central database. A start/stop button must be used to activate the real-time data acquisition of the various input signals, which are subsequently displayed digitally, along with the associated FFT magnitude and phase plots. The input signal FFT peak magnitudes and associated frequencies must be displayed numerically in real-time. A help file must be available which will allow the administrator to search for topics regarding the operation of the program as well as troubleshooting.

The program must be capable of allowing the end-user, who has a particular motor disability, to customise the system, without the assistance of an able-bodied person. The software program must also be informative and specific when an error occurs as to aid the user to fix any problems quickly. As the program is to operate as part of a real-time system, there must be guaranteed response times i.e. the worst case response times must be confidently predicted. Efficiency is important but predictability is essential. Real-time systems are also characterised by reliability and safety (Section 4.4 of Chapter 4. The software program's design must take into account user characteristics and objectives to satisfy a wide range of potential customers with varying degrees of expertise while being user-friendly and simple to use.

D.4 Functional Requirements

D.4.1 Accuracy

Certain applications need to perform to the highest possible degree of accuracy. To ensure the highest degree of accuracy, the algorithms used need to be mathematically sound for all the possible range of inputs. This accuracy is required, especially when performing the digital signal processing, namely the digital filtering and FFT. The accuracy of the input signal magnitude, frequency range, and hysteresis levels, which trigger certain output functions, is of extreme importance. Also the validity of the input channel corresponding to an input signal, as well as various other input/output information, need to be considered.

Rationale: Digital signal processing performed inaccurately will result in an incorrect and erroneous input signal reading. Also incorrect input versus output trigger parameters will result in an output function being triggered at the wrong moment i.e. when a voluntary action has not been performed by the end-user. This could also result in multiple output actions being performed when only one is requested.

D.4.2 Applicability and Efficiency

The program must be applicable to various operating systems and it must be capable of working with a wide variety of external applications. It must also operate at a high degree of efficiency.

Rationale: It is important to note that an inefficient program decreases the cost effectiveness of the application, as it may be taxing on the system resources, thereby requiring the end-user to purchase a far more powerful computer to run the application. An inefficient program may result in incorrect input and output information or at worst it may cause the application to crash, thereby rendering the program useless, especially when it is needed by the end-user to perform a basic task, such as driving a wheel-chair.

D.4.3 User-friendliness

The programs menus, error messages, user inputs and outputs must be in an intuitive and easily understood form, i.e. simplicity and user friendliness are very important. Any unneeded complexity in the program must, therefore be hidden from the end user.

Rationale: A user-friendly program increases user confidence and makes it more applicable to a larger consumer group i.e. not only trained professionals, resulting in a more widely used and economically viable product.

D.5 Interface Requirements

D.5.1 The Graphical User Interface (GUI)

The Graphics User Interface needs to be simple to use (i.e. user friendly). This is important as the target user would rather use program which is simple and quick to learn and operate than a more cumbersome user interface. The GUI portrays all of the program requirements graphically, thereby making the program much easier to understand and therefore use. The interface will also be responsible for reporting any errors which may occur during regular program operation. Table 1 lists all the possible error messages that can occur, and their corresponding meanings. The administrator and users may perform various tasks at numerous occasions in any one of the windows dialog boxes; including inputting and displaying various details accordingly. The administrator and users may also return to the previously displayed menu at any point in the GUI as well as exit form the program when the initial menu is displayed.

Rationale: An informative output, displaying both user inputs and outputs, increases the efficiency of the application as the user can double check that the correct inputs were entered, without having to rerun the program.

The importance of informative and specific error messages and notifications aids in the rapid solution of in-program problems. By making use of this particular GUI, the user and administrator may, in one session, input and display various results, without having to run the program repeatedly.

The following table Table D.1 describes all the possible error message and notification messages which may arise, as well as a brief description of why the error messages are initialized.

Table D.1: Error and notification messages which may arise in the software application

User	Button	Type of Message	Message	Error
Administrator	**Administrator Options**	Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source. This results when the program cannot access the database, as a result of the connection path to the database being incorrect.

User	Button	Type of Message	Message	Error
		Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source i.e. when there is a missing data source. This specific error would occur after attempting to add the pin assignments to the database from the DAQ system, with no channels yet available in the database.
	Add Details	Administrator Error	Please assign the detail to the 'User Name' field	This error results when the administrator does not input the required user's details, i.e. the field is null.
		Administrator Error	Please assign the detail to the 'Input Device Name' field	This error results when the administrator does not input the required input device detail corresponding to the end-user's requirements i.e. the field is null.
		Administrator Error	Please assign the detail to the 'Input Device Name' field	This error results when the administrator does not input the required input device detail corresponding to the end-user's requirements i.e. the field is null.

User	Button	Type of Message	Message	Error
		Administrator Error	Please assign the detail to the ‘Output Function for the device’ field	This error results when the administrator does not input the details for the output function required, which may be software or hardware in nature, i.e. the field is null.
		Administrator Error	Please assign details to the ‘User Name’ and ‘Input Device Name’ fields	This error results when the administrator does not input the required user and input device details corresponding to the end-user’s requirements, i.e. the fields are null.
		Administrator Error	Please assign details to the ‘Input Device Name’ and ‘Output Function’ fields	This error results when the administrator does not input the required input device and output function details, corresponding to the end-user’s requirements, i.e. the fields are null.
		Administrator Error	Please assign details to the ‘User Name’ and ‘Output Function for the device’ fields	This error results when the administrator does not input the user’s details and the required output function name, corresponding to the end-user’s requirements, i.e. the fields are null.

User	Button	Type of Message	Message	Error
		Administrator Error	Please assign details to the 'User Name', 'Input Device Name' and 'Output Function for the device' fields	This error results when the administrator does not input the user information and the required input device and output function details, corresponding to the end-user's requirements, i.e. the fields are null.
		Notification	Updated	The database has been updated successfully with the user's information, the pin assignment as well as the input device and output function details.
		Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source. This specific error occurs when certain information cannot be added to the database. These include the user details, pin assignment information as well as input device and output function details.
	Delete Properties	Administrator Error	Please select the 'Input Device Name' with the associated t/p connection	This error occurs when no specific row has been selected, on the datagrid for deletion from the database and consequently from the datagrid.

User	Button	Type of Message	Message	Error
		Notification	Are you sure you would like to delete the specified item?	This question is asked when information is being deleted form the database.
		Notification	[*Input device name*] DELETED.	
		Database (OleDb) Exception	An unhandled exception of type ‘OleDbException’ occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source. This specific exception results when the selected row of the datagrid does not result in the deletion of that row of data from the database and subsequently from the datagrid.
	Change Properties	Administrator Error	Please select the ’Input Device Name’ with the associates terminal pin connection from the list above	This error occurs when no specific row has been selected, on the datagrid, in order to change certain properties in that row in the database and consequently in the datagrid.

User	**Button**	**Type of Message**	**Message**	**Error**
		Error	Error in [*input channel row number*]	Gets a value indicating whether there are errors in any of the rows in any of the tables of the DataSet to which the table belongs.
		Notification	Updated [*device name*]	The database has been updated successfully with the changed details.
		Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system.data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source. This specific exception results when the selected row of the datagrid does not result in the deletion of that row of data from the database and subsequently from the datagrid.
	click datagrid	Notification	Please add details to the selected row for editing	At least one row of data must be present before the datagrid may be selected i.e. the administrator must add various details to the datagrid before it may be selected in order to change or delete information from the selected row.
		Notification	No row present	This notification is provided when the number of visible rows in the datagrid is less than those in the database.

User	Button	Type of Message	Message	Error
		Index out of range exception	There is no row at position [*x*]. Please select a valid row	This exception is thrown when an attempt is made to access an element of an array with an index that is outside the bounds of the array.
	Output Function Setup	Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system.data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source i.e. when the program cannot access the database. This may occur when the connection path to the database or the information in the database is incorrect.
	Set Parameters	Administrator Error	The 'Voltage Min Output' value cannot be less than the 'Minimum Range' Value	The minimum range value is the lower boundary for the minimum output signal voltage, and cannot be exceeded.
		Administrator Error	'Voltage Max Output' value cannot be more than the 'Maximum Range' Value	The maximum range value is the upper boundary for the maximum output signal voltage, and cannot be exceeded.

User	Button	Type of Message	Message	Error
		Error	Error in [*input channel row number*]	Gets a value indicating whether there are errors in any of the rows in any of the tables of the DataSet to which the table belongs.
		Notification	Updated [*input device name*]	The database has been updated successfully with various input trigger level, software and hardware parameters.
		Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source i.e. when the program cannot access the database. This occurs specifically when the database cannot be updated with certain information concerning various output parameters.
	Browse for plug-in program	Error	The requested file is not of type '.exe'	This error occurs when the administrator selects a file which is not of type *.exe (executable) to run as a plug-in software module for the application.

User	Button	Type of Message	Message	Error
	Data Acquisition	Database (OleDb) Exception	An unhandled exception of type 'OleD-bException' occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source i.e. when the program cannot access the database. This may occur when the connection path to the database or the information in the database is incorrect.
		Database (OleDb) Exception	An unhandled exception of type 'OleD-bException' occurred in sys-tem.data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source i.e. when there is a missing data source. In this particular case it occurs when adding the pin assignment to database.
	Calibrate	Notification	"Please perform an input trigger action e.g. select the touch pad or tilt the sensor, and thereafter select the 'Save Calibration' button to save the input trigger parameters."	

User	Button	Type of Message	Message	Error
	Save Calibration	Error	Error in [*input channel row number*]	Gets a value indicating whether there are errors in any of the rows in any of the tables of the DataSet to which the table belongs.
		Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system.data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source. In this case data concerning the software output cannot be written/changed to/from the database.
		Notification	Calibration performed successfully for [*input channel row number*].	
	Save Settings	Error	Error in [*input channel row number*]	Gets a value indicating whether there are errors in any of the rows in any of the tables of the DataSet to which the table belongs.
		Notification	Updated [*input device name*] for [*user name*]	The database has been updated successfully with various DSP, timing and voltage range parameters.

User	Button	Type of Message	Message	Error
	Start	DAQ Error	Error in [*input channel row number*]	Gets a value indicating whether there are errors in any of the rows in any of the tables of the DataSet to which the table belongs.
		DAQ Error	Device not available for routing. It is possible the device needs to be reset or that the device is being reset.	This occurs when there is no data acquisition being performed due to the fact that the DAQ card is either not ready or not connected and therefore non-operational.
		DAQ Error	*the current samples are no longer available*	This is a major error which causes the program to crash/freeze, which halts the DAQ process; occurring when too many samples are being read whilst the PC is low on resources (memory and processor power) i.e. whereby the PC is not capable of handling the large influx of data. This error could also occur when reading/writing from/to the database whilst performing real-time measurements/functions. To prevent this one should close any applications which are resource intensive or upgrade the computer processing speed and/or memory.

User	Button	Type of Message	Message	Error
		Exception	*DAQ Exception*	Represent the exception that is thrown when a NI-DAQmx driver error occurs. This occurs when the input samples are being initialised and read.
		Exception	*DAQ Exception*	Represent the exception that is thrown when a NI-DAQmx driver error occurs. This may occur when the program is trying to initialise and write samples for a hardware/-physical voltage output.
		Exception	Error in [*input channel row number*]	Gets a value indicating whether there are errors in any of the rows in any of the tables of the DataSet to which the table belongs.
		Database (OleDb) Exception	An unhandled exception of type 'OleDbException' occurred in system. data.dll	This is the exception that is thrown when the underlying provider returns a warning or error for an OLE DB data source. In this case data concerning the software output cannot be written/changed to/from the database.
		Exception	*General Exception*	This exception is thrown when the plug-in software application is not initialised and is therefore not run successfully.

D.5.2 The Database

Certain information is stored in the database, including subject details, input device and output parameter settings as well as information relating to the digital signal processing. The database should only be accessed by the administrator. These outputs shall be of the same format of that displayed on the screen. The database to be used is a simple OleDb database i.e. MS® Access 2003.

Rationale: All the information must be stored in a database, as each user, utilising specific input devices, has specific output requirements. The database information is allocated in order of input channel number. The most important fields in the database include the input trigger levels and output function information, which are assigned values in the 'Output Function' dialog box. This information is then retrieved from the database for use when performing the data acquisition.

D.6 Non-Functional Requirements

The main non-functional requirements applicable to this type of software program are that of reliability, maintainability, and compatibility. Table D.2 illustrates, clearly, the need for each of the above listed requirements and outlines ways in which they can be achieved within a software program.

Table D.2: The non-functional requirements of the Assistive System's software application

Non-Functional Requirement	**Description of Requirement**	**Why it is needed**	**How it will be achieved**
Reliability	The program will be required to perform faultlessly and accurately each and every time it is run.	A reliable program will achieve a loyal following and establish a consumer base for future products.	Sound programming techniques and extensive testing procedures will establish a reliable product.

Non-Functional Requirement	Description of Requirement	Why it is needed	How it will be achieved
Maintainability	Maintainability is the ability to be able to repair any in program errors to ensure the functionality of the program is maintained.	After market service if it is conducted effectively will establish a reliable consumer base for future products.	By adopting modular design techniques with different methods being responsible for different things, errors and sources of inaccuracy can be easily found and eliminated.
Compatibility	Compatibility is the ability of the program to run on most to all personnel and laptop computers, with no prerequisites on operating systems and system resources.	A highly compatible product will ensure a much large consumer group and increase the products economic viability.	By designing the program such that it does not require large amounts of system resources and by ensuring that it is not OS specific will increase its compatibility.

D.7 Operational Scenario

The software application operation is illustrated by Figure 5.5 in section 5.5.5 of the main document.

D.8 Conclusion and Summary

The requirements engineering document describes the performance and operation of the software application component of the proposed assistive system. Certain requirements, which needed to be satisfied prior to the design of the software application, were described and this was followed by an overview of the program as a result of the requirements. Other important factors are the functional requirements namely, the accuracy, applicability and efficiency and user-friendliness of the software program. One of the most important interface requirements is that the graphical user interface (GUI) needs to be simple or easy to use. The GUI also needs to provide all error messages and notifications in order to aid in the rapid solution of in-program problems. The database is used to store certain information pertaining to the assistive system, such as the subject details, input device names and output parameter settings, as well as information relating to the digital signal processing. The non-functional requirements, as the name suggests, deals with the overall behaviour of the the software system. The non-functional requirements, which must be satisfied by this software program are the reliability, maintainability, and compatibility of the application.

Appendix E

Testing and Validation

E.1 Introduction

Testing and validation are essential in ensuring that the application meets its desired requirements (see requirements documentation) and play an important role in product quality assurance. Testing usually occurs through out the software process life cycle and is generally associated with verifying the architectural design and program coding, whereas validation is usually done on the completed application to ensure that it accomplishes what it was designed to do. Testing involves executing a program to see whether it produces the correct output for a given input. It usually a very laborious and tedious task and can be the most time consuming stage of a software design. Testing was traditionally performed after the program requirements specifications, specifications and the program itself were completed as an interim to the final working program. The following document shall describe a test plan for the generic assistive system, and shall be implemented during various phases of the software design [46].

E.2 Test Plan

A suitable test plan is going to be developed and this shall be performed at all the various stages of the software design process. During the subsequent phases, testing is continued and refined. The earlier the errors are detected the cheaper it is to correct them. This would save time and money as well as making the software process more efficient to design and implement. There are two very important concepts which must be satisfied for the test plan. These are: a) **Verification** and b) **Validation**.

Verification is a test which verifies that the transition between subsequent phases of the software design process is correct. If one phase of the software process is completed with valid testing and verification it will result in a better confidence level when implementing the subsequent phase/s and hence the process will run more efficiently.

Validation is a test which checks that the software design process is still conforms to the **user requirements**. [46]

The system shall be designed for **error** (human action which produces an incorrect result), **faults** (manifestation of an error) and **failure** (fault which may result in a system/program failure). In the following subsections the various verification and validation activities shall be discussed which are to be performed during the various phases of the software design lifecycle [46]. Various testing techniques shall be utilized for the software design process. These include coverage-based, fault-base and error-based testing methods.

E.2.1 Coverage-based Testing

Program Execution

The program is executed by double left-clicking on the "GenAssDevice.exe" executable file.

Administrator Options

The administrator single left clicks on the "Administrator Options" button in the main dialog window. The "Administrator Options" dialog window is opened. The administrator must thereafter add/select various details to/from the "Add New Details" group box, which includes the "Patient Details", "Input Device Details" and "Output Function Details" group boxes. The administrator must, more specifically, add/select the subject user name and motor disability to/from the "Select/Input a User Name" and "Please Select/Input a Motor Disability" combo box lists respectively in the "Patient Details" group box. The "Input Device Details" group box contains the "Please Select/Input an Input Device Name" and "Select an Available Terminal Block Connection" combo box lists, to which the administrator may add/select the name of the input device and the channel to which it is associated respectively. The last essential details concern the "Output Function Details" group box, which requires the administrator to add/select the output function name to the "Please Select/Input an Output function for the Device" combo box list and thereafter select the output type, i.e. hardware or software, from the radio buttons. Once all the details have been added, the administrator must single left-click the “Add Details” button, which populates a row comprising the specified details to the “List of Input vs. Output Information for specified users” data grid.

Should the administrator wish to change the information in a particular row, he/she must select that row from the data grid and thereafter specify all information again in the “Add New Details” group box. Left-clicking the “Change Properties” button must result in the selected row of information being updated, along with all the changed information, except for the new pin assignment. The only way that the administrator may choose a new terminal pin connection, for a particular row, is to delete the row and subsequently add the required details again.

The administrator may delete a particular row of information, by selecting the row and thereafter left-clicking the “Delete Properties” button. It must be noted that selecting a particular row on the data grid must result in the “Input Device” and “Pin Assignment” combo boxes being populated with the input device name and terminal block connection name respectively. This ensures that the correct row was selected before it is changed or deleted. Also the information on the data grid must be listed in ascending order of the terminal pin connection names, from “Dev1/ai0” to “Dev1/ai7” (if chosen) respectively. The administrator may then return to the main menu by single left-clicking the “Exit” button, which closes the “Administrator Options” window.

Output Function Setup

The administrator single left clicks on the “Output Function Setup” button in the main dialog window. The “Output Function Setup” dialog window is opened. The first check that needs to be done is the administrator options validity check, which consists of the administrator checking whether the correct information is populated in the “Input Details” and “Output Details” group boxes.

The administrator thereafter may select an input channel from the “Physical Input Channel” combo box list, which is contained in the “Input Details” group box, in order to set certain parameters to follow. This selection must result in the “Output Function for the Device” combo box and the concording “software” or “hardware” radio buttons being populated, which are contained in the “Output Details” group box. Also all information relating to the selected input must be populated in the “Parameter Settings” group box.

The administrator may single left-click the “Change Properties” button. This will allow the administrator access to add/select parameter information to/from the “Parameter Settings” group box, which consists of the “Parameter Settings”, “Software Parameters” and “Hardware Output Channel Parameters” group boxes and the “Set Parameters” and “Cancel” buttons. The input parameters, which are found in the “Input Parameter Trigger Information” group box, must be inaccessible, with only information regarding the various input trigger parameters being populated.

The administrator will only be allowed access to the “Software Parameters” and “Hardware Output Channel Parameters” group boxes, depending on whether the required output function is a software or hardware type of output respectively. If the required output function is software in nature, the administrator may select the “...” browse button, which opens the “Open Required Software Plug-in Application” open file dialog. The administrator must select a plug-in application, which has been written for the assistive software application, by searching for and selecting the application executable file. This follows with a left-click of the “OK” button, which must close the open file dialog and consequently save this information to a database. If the open file dialog has been selected erroneously, with no selection required, the administrator may left-click the “Cancel” button. This returns the administrator to the “Output Function Setup” dialog. The “File/exe Path” combo box must be populated with the path of the plug-in executable file.

If the required output function is hardware in nature, the administrator will, as mentioned previously, have access to the “Hardware Output Channel Parameters” group box. The administrator will firstly have to select the required output channel by selecting the “Physical Output Channel” combo box list. The minimum and maximum range of the output signal must be entered into the “Minimum Range Value (V)” and “Maximum Range Value (V)” numeric up down boxes respectively. This subsequently follows with the actual minimum and maximum voltage levels, which are scrolled for in the “Voltage Min Output (V)” and “Voltage Max Output (V)” numeric up down boxes respectively. The user thereafter may left-click the “Save Parameters” button in order to save the entered and selected information to the database. If the “Change Parameters” button was selected erroneously, the administrator has the option of canceling this, as well as any information which may have been entered into the various fields, by left-clicking the “Cancel” button. The administrator may then return to the main menu by single left-clicking the “Exit” button, which closes the “Output Function Setup” window.

Data Acquisition Setup

The administrator single left clicks on the “Data Acquisition Setup” button in the main dialog window. The “Data Acquisition Setup” dialog window is opened. The first check that needs to be done is the administrator options validity check, which consists of checking whether the correct information is populated by the “Channel Parameters” group box, concerning the various physical input channels and their associated input devices and end-user names.

The administrator may thereafter left-click the “Change Settings” button, which allow him/her access to the “Timing Parameters”, “IIR Filter Parameters”, “FFT” and “Calibration” group boxes as well as the “Default Settings”, “Save Settings”, and “Cancel” buttons and “Minimum/Maximum Value (V)” numeric up down boxes (these are restricted by default along with the “Stop” button). This will also result in the “Start” and “Change Settings” buttons being restricted. The administrator will thereafter need to select the required input channel by selecting the “Physical Channel” combo box list, which can be found in the “Channel Parameters” group box. This must result in the “Input Device” and “User Name” combo boxes being populated with the associated input device name and the user’s name. This follows with the administrator scrolling for the appropriate input voltage range by selecting the “Minimum Value (V)” and “Maximum Value (V)” numeric up down boxes. The administrator may subsequently select/scroll through the “Samples/Channel” and “Rate (Hz)” numeric up down boxes in order set the timing parameters for the input signal. The “IIR Filter Parameters” group box consists of the “Filter Type”, “Filter Design”, and “Windowing” combo box lists, as well as the “Ripple”, “Attenuation”, “Order”, “Lower Cutoff” and “Upper Cutoff“” numeric up down boxes.

Selecting the “Calibrate” button must result in the button’s text changing to “Save Calibration”. This will also allow the user access to the various calibration parameters, which follows with the administrator having to select whether the input triggers above or below this calibrated value by selecting the “<mag” or “>mag” radio buttons respectively. The “hold/release” radio button may be selected in order to hold or release the state of the output function, with each subsequent output trigger action resulting in the output’s state being held or released. This follows with a value being entered into the “Hysteresis (dB)” numeric up down box. Hysteresis ensures that the output function is not triggered continuously while the input is at the required input trigger level. Selecting the “Calibration” button also starts the data acquisition process, including the display of the input signal as well as the FFT parameters. The administrator must perform a trigger input action, which would be used later to perform the specified output function for the selected input channel. Once satisfied with the results, the input trigger information may be stored to the database by selecting the “Save Calibration” button. This should result in the various input trigger parameters being inaccessible to the administrator. This also results in the button’s text changing back to “Calibrate”. It must be noted that if the calibration is performed again (once the calibration has been performed for the specified channel), it must not result in an output function being triggered for the same input trigger action. This should only result once the “Start” button has been selected.

Left-clicking the “Default Settings” button must result in all of the input fields being populated with default information and parameters. Left-clicking the “Save Settings” button must result in all of the information, which has been provided in the various fields, being stored in the database. The administrator must therefore be able to scroll through the “Physical Channel” combo box list, which must result in all of the parameters and settings provided being populated in the required fields, for the specified input channel. This information must be stored in the database.

If the “Change Settings” button was selected erroneously, the administrator has the option of canceling this, as well as any information/parameters which may have been entered into the various fields, by left-clicking the “Cancel” button. Once the “Cancel” and “Change Settings” buttons have been selected, the administrator must no longer have access to these and the “Default Settings” buttons. This should also restrict access to the “Timing Parameters”, “IIR Filter Parameters“ and “FFT” group boxes and “Minimum/Maximum Value (V)” numeric up down boxes, with access only granted by again left-clicking the “Change Settings” button.

Left-clicking the “Start button” must restrict access to the “Change Settings” and “Start” buttons and allow access to the “Stop” button. This must result in the data acquisition process. The unfiltered and filtered input signal as well as the windowing function must be displayed on the “Graph of Voltage for specified input device” plot. A hidden legend for this plot must come into view once the administrator passes the mouse pointer over the plot. The “Magnitude Graph” and “Phase Graph”, which are contained within the “Fourier Transform (FFT) Plots” group box, must contain the simultaneous plots (of all the added channels) for the FFT magnitude and phase respectively. Each magnitude plot must have a certain colour associated with it, with a legend being visible only once the administrator passes the mouse pointer over the graph. Also cursors must also be visible on the “Magnitude Graph” plot and must be singularly associated with a specific magnitude plot, whereby the cursor determines the position of the frequency for the associated peak magnitude per channel. The peak magnitude of the signal and the associated frequency, at which this peak occurs, must be displayed in the “Logarithmic Peak Magnitude (dB) and“Peak Frequency” combo boxes respectively. It must be noted that all of the plots and calculated data must be displayed in real-time, without any lag or interruptions. Lastly, by left-clicking the “Start” Button, the numerical values of all of the added raw or unfiltered (digitally) input signals must be displayed on the “Acquired Data (Unfiltered)” data grid.

Another check that needs to be performed is the output function setup validity check, which consists of checking whether the parameters that were set in the “Output Channel Setup” results in the required actions. When the input signal is within the range of the specified input parameter trigger settings, it must result in the performance of the specified output action. If an output function is not triggered, then the administrator will have to go back to the "Output Function Setup" and specify the various input parameter trigger details again. These results must therefore be predictable and efficient in nature, as expected of all real-time systems (Section 4.4 of Appendix 4).

Left-clicking the "Stop" button must firstly stop and dispose of the data acquisition process (task) as well as restrict access to the "Stop" button. Access must also be granted to the "Stop" and "Change Settings" buttons. The administrator may then return to the main menu by single left-clicking the "Exit" button, which closes the "Data Acquisition" window.

Help

The user may request help by selecting the "Help" button in the main dialog window. This will open a new dialog window, consisting of a basic help text file covering all possible information that may be required as well as troubleshooting possible problems which may occur.

Exiting the Application

The program is exited by selecting either the "Exit" button or the red crossed button in the top right-hand corner of the main dialog window.

E.2.2 Error-based Testing

This focuses on testing various error-prone points, such as 0, 1, or the upper bound of an array [46]. The possible errors and notifications, which may arise in the use of the program, have been provided in Table D.1 of the requirements engineering document (Appendix D). The administrator must thus follow the error-based testing in accordance with Table D.1. All of these tests must be performed in the same order as the table, with all of the error and exception messages as well as the notification messages being tested for.

E.2.3 User Interface and Functionality Testing

This testing method shall be performed once a prototype for the program is completed. To gain a more objective view of the application and its functionality, various other people will be chosen to evaluate the Generic Assistive Device software application. The volunteers will only have to evaluate the general setup information for the application; however, to give these people a better idea of the application, he/she may wish to test the general data acquisition, by utilising the various input sensors and subsequently evaluating certain output functions.

A reasonable sample size of 5 people, with reasonable computer literacy skills, will be chosen to perform the required tasks. It must be noted, that the actual testing of the system as a whole will be performed later by the intended disabled volunteers and would obviously exclude user interface testing, but rather involves functionality testing. This will only be performed once the application has been setup in order to meet the end-user's varying requirements.

The volunteers will perform the user interface and functionality testing by following the guidelines outlined in Appendix H, which provides detailed information on using the system as a whole.

Each of these tasks are rated based on 1) **Usability** 2) **Performance** 3) **Enjoyment level** 4) **Visual Appeal**. Each of these requirements will be rated out of 5 with 5 being excellent and 0 being very poor (requires redesign and development).

E.2.4 Miscellaneous Testing

Every aspect counts when trying to launch a product into the market place. All possible tests must be accounted for, under varying conditions. The following tests are very unusual, but form a very important part of the testing process [46].

Accessibility

By keeping the application clear of any extravagant graphics it ensures a more accessible application for people running older versions of software. The program is also required to be as simple as possible, as the potential users are generally accepted as having a very basic computer literacy level. It is for this reason that the application must be as compatible as possible for all user agents who would like to use the application.

Colour Testing

To ensure the application is not overly dependent on colour and that the colours won't cause problems, the following tests can be performed:

- Setting the monitor to monochrome to see how it renders;
- Print the pages in black and white, with backgrounds and images using grayscale, to ensure that there is enough contrast for the page to be easily read; and
- Reset Windows to use the High Contrast monitor setting.

E.2.5 Fault-based Testing

This focus is on detecting faults, for example, the percentage of seeded faults detected [46]. This is only performed once the error, coverage, GUI and functionality based tests have been completed. The only way this can be performed successfully, is by performing the coverage based testing numerous times until a fault occurs. This fault must thereafter be fixed to ensure that it cannot occur again.

E.3 Conclusion and Summary

Testing serves as a very important phase in the software development process and is important for in order to ensure the success of a program. The testing can also serve to illustrate the user interface and overall program functionality as designed by the software programmers, in the hope that the customer will be fully aware of the standard of application they will receive, before the final implementation stage of the design has been started. The testing is performed at all the different stages of the design. This is in part to safe guard the interests of both the programmers and customer and to decrease the amount of time spent in developing the application. The two important concepts that need to be satisfied are validation and verification. These two concepts need to be considered when implementing the various testing methods. Theses include the coverage, error and fault based testing methods. Other testing techniques performed are the GUI and functionality requirements test as well as other miscellaneous tests.

Appendix F

Eagle-CAD® Diagrams

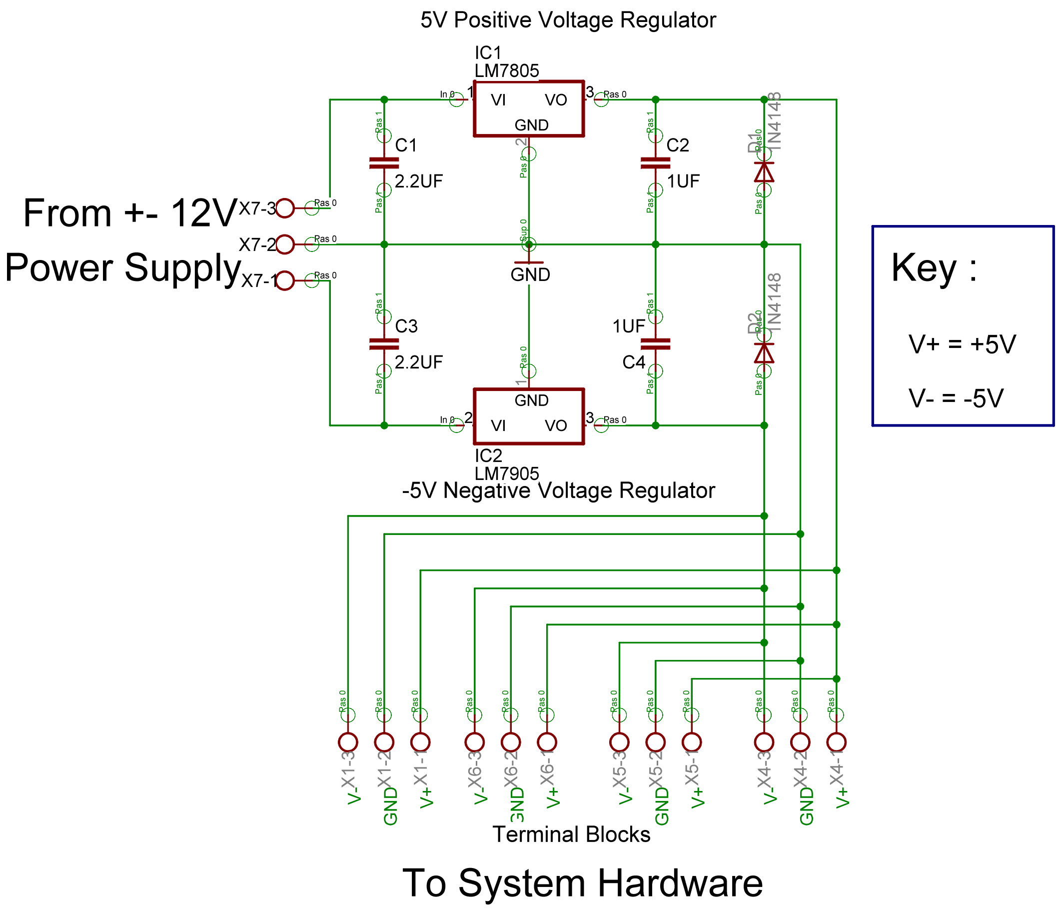


Figure F.1: The Power Supply Conditioning Circuit Schematic (Eagle CAD)

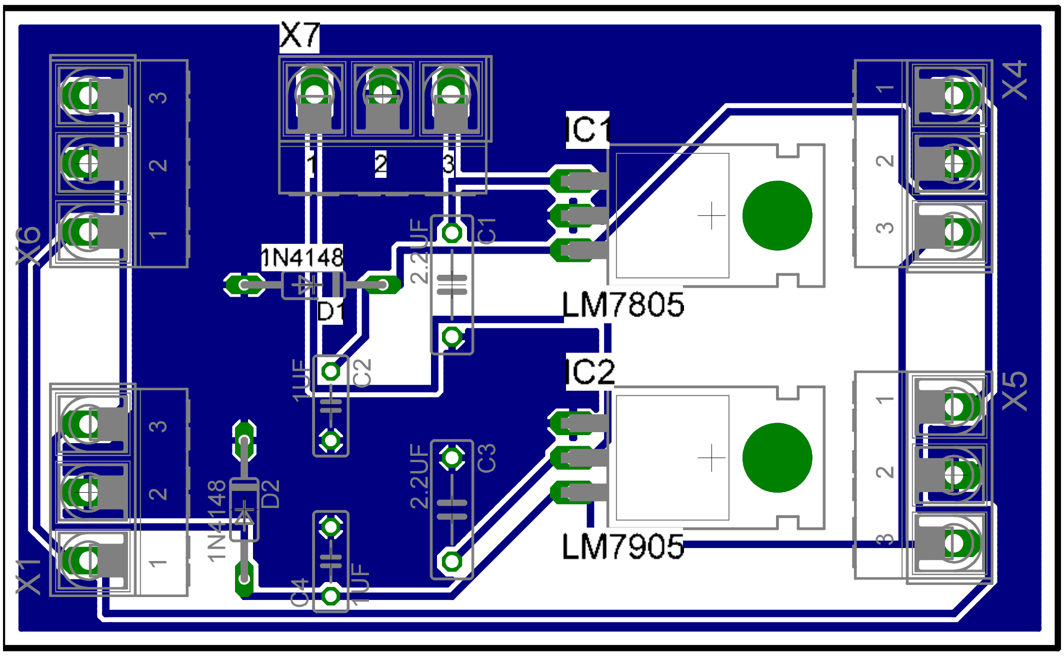


Figure F.2: The Power Supply Conditioning Circuit Board (Eagle CAD)

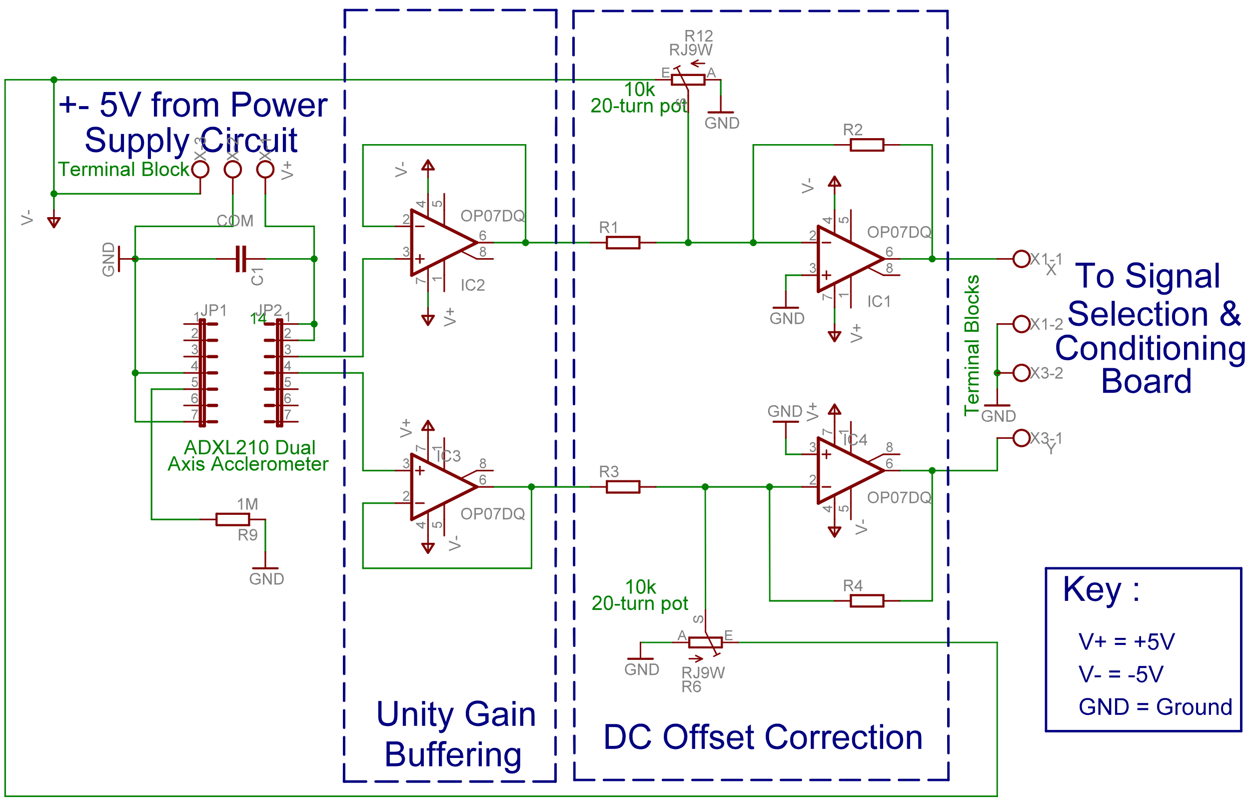


Figure F.3: The ADXL 210 Dual-Axis Accelerometer Sensor Circuit Schematic (Eagle CAD)

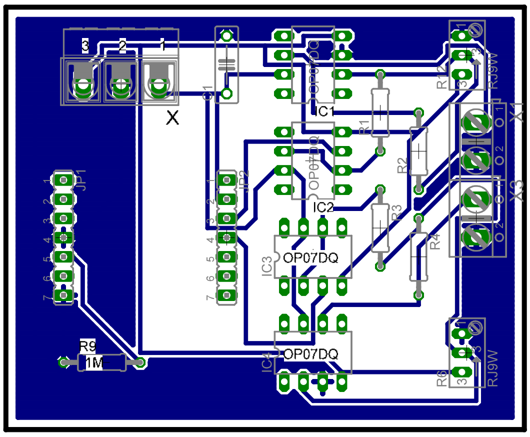


Figure F.4: The ADXL 210 Dual-Axis Accelerometer Sensor Circuit Board (Eagle CAD)

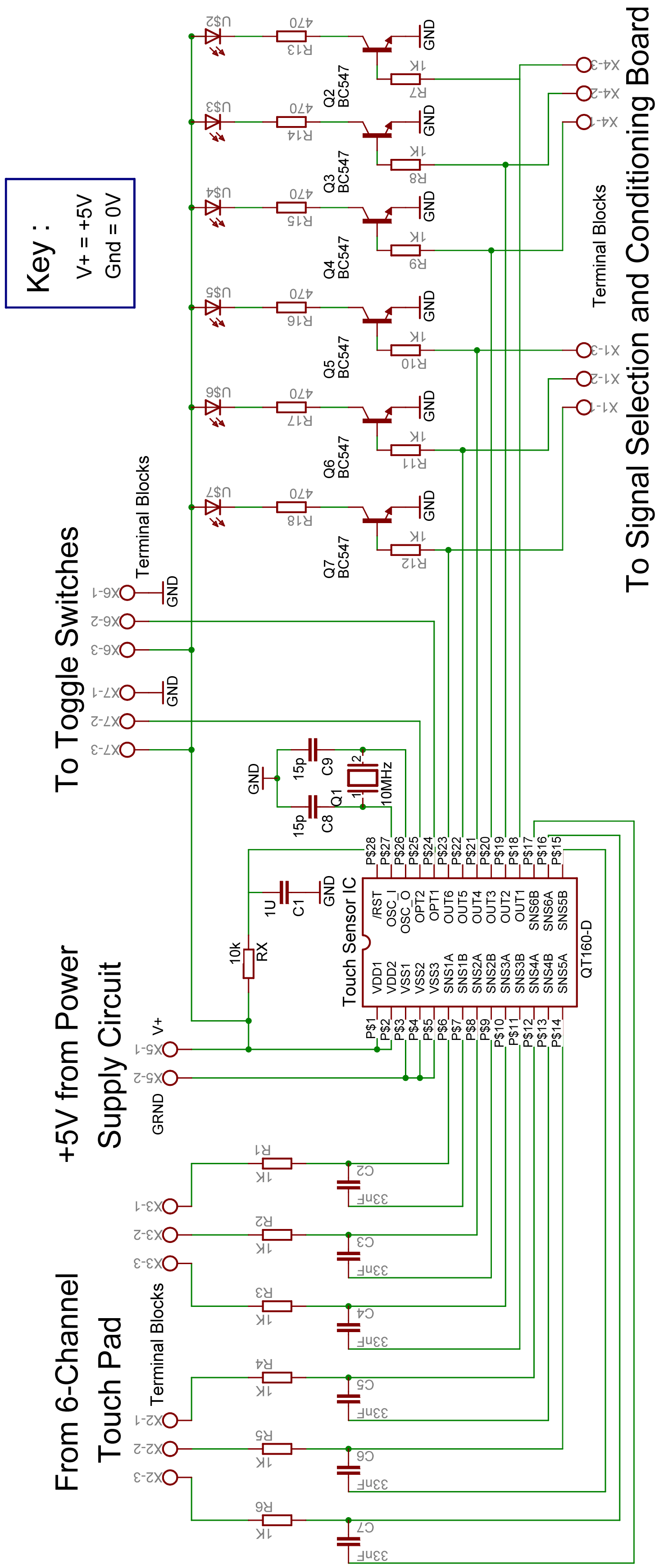


Figure F.5: The QT160D 6-Channel Touch Sensor Sensor Circuit Schematic (Eagle CAD)

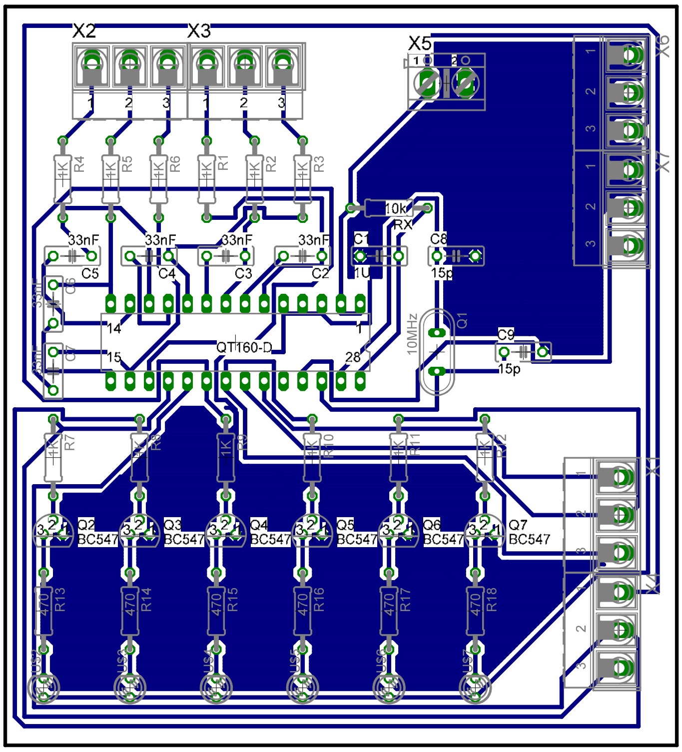


Figure F.6: The QT160D 6-Channel Touch Sensor Sensor Circuit Board (Eagle CAD)

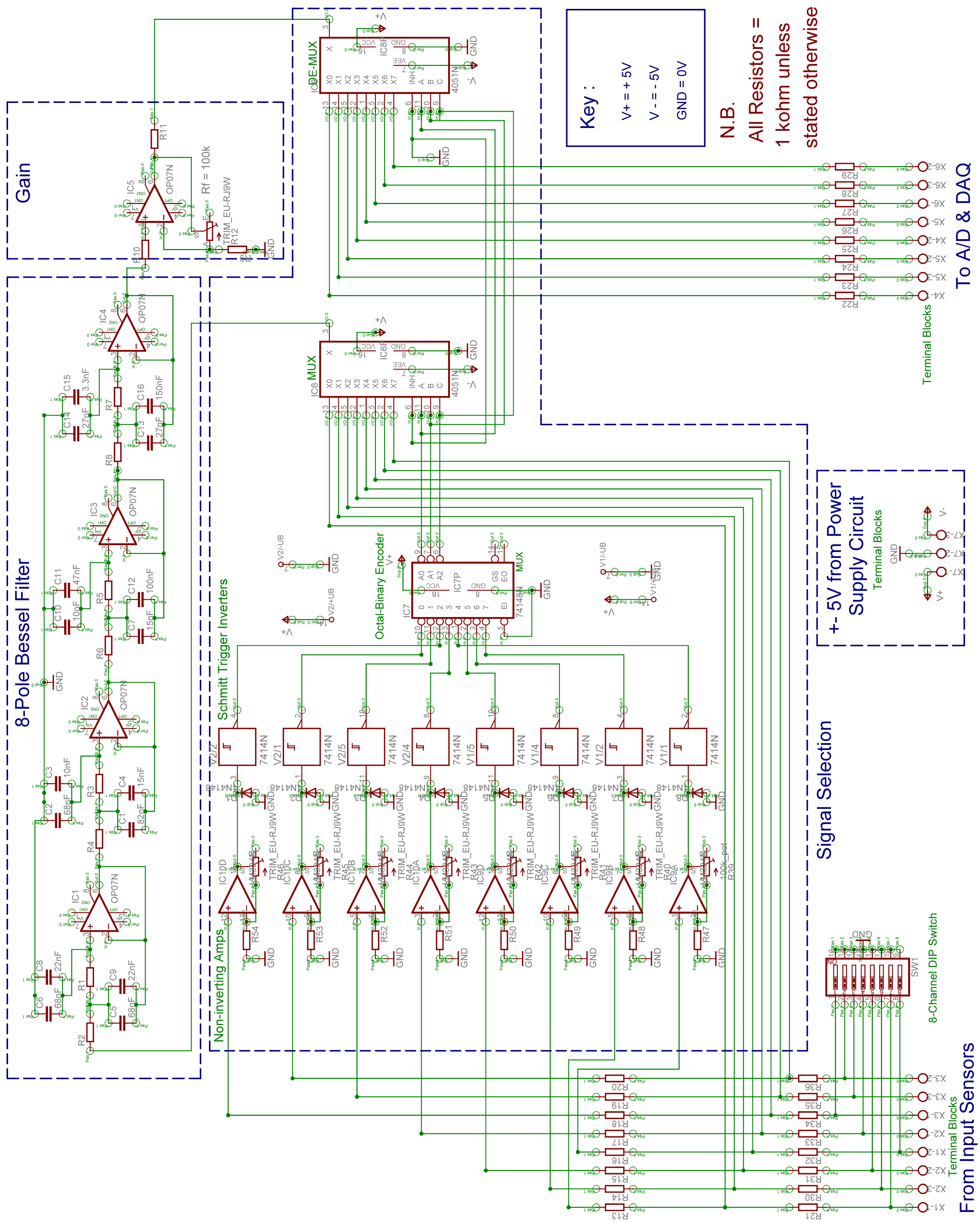


Figure F.7: The Input Sensor Signal Selection and Conditioning Circuit Schematic (Eagle CAD)

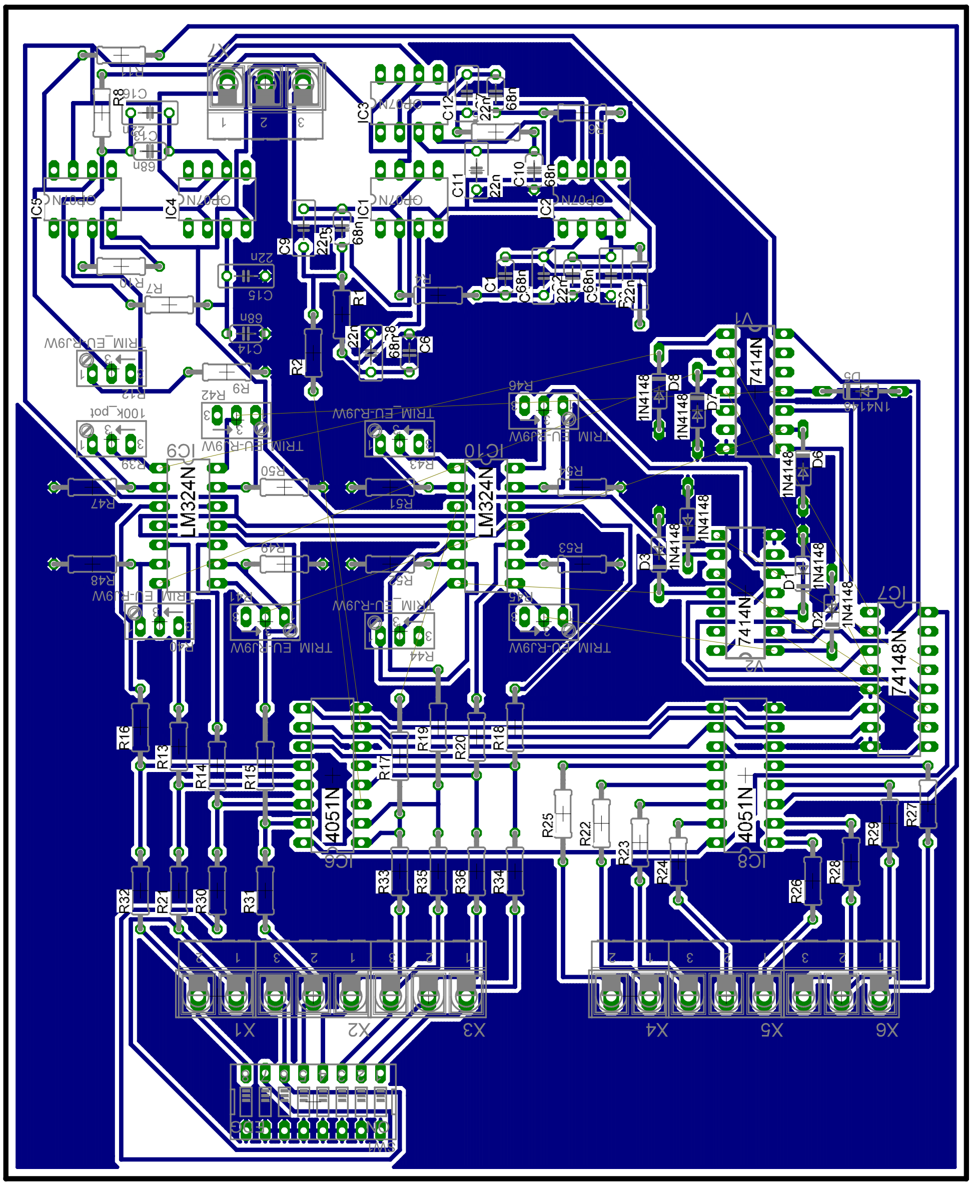


Figure F.8: The Input Sensor Signal Selection and Conditioning Circuit Board (Eagle CAD)

Appendix G

Ethics Approval

G.1 Approved Ethics Document

The signed ethics approval form is provided as a printed attachment on the following page.

UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG

Division of the Deputy Registrar (Research)

HUMAN RESEARCH ETHICS COMMITTEE (MEDICAL)
R14/49 Senatore

CLEARANCE CERTIFICATE	**PROTOCOL NUMBER MO61143**
PROJECT	Research into the design of technologies needed to assist people with severe motor disabilities
INVESTIGATORS	Mr F Senatore
DEPARTMENT	Engineering
DATE CONSIDERED	06.11.24
DECISION OF THE COMMITTEE*	APPROVED subject to submitting a revised

information sheet

Unless otherwise specified this ethical clearance is valid for 5 years and may be renewed upon application.

DATE 06.11.28 **CHAIRPERSON** ....
(Professors PE Cleaton-Jones, A Dhai, M Vorster, C Feldman, A Woodiwiss)

*Guidelines for written 'informed consent' attached where applicable

cc: Supervisor : Dr D Rubin

DECLARATION OF INVESTIGATOR(S)

To be completed in duplicate and **ONE COPY** returned to the Secretary at Room 10005, 10th Floor, Senate House, University.
I/We fully understand the conditions under which I am/we are authorized to carry out the abovementioned research and I/we guarantee to ensure compliance with these conditions. Should any departure to be contemplated from the research procedure as approved I/we undertake to resubmit the protocol to the Committee. **I agree to a completion of a yearly progress report.**

PLEASE QUOTE THE PROTOCOL NUMBER IN ALL ENQUIRIES

G.2 Participant Information Sheet

Dear potential participant,

Hi, I am Franco Senatore. I am currently performing research in order to obtain a Masters degree (MSc (Eng)) in Electrical Engineering.

You are invited to consider participating in a research study, which involves the development of a system which may assist subjects with a wide range of motor disabilities (e.g. quadriplegics) and with fairly diverse requirements in terms of performing basic tasks, such as turning a light switch on and off or controlling a wheel chair. You are among a variety of subjects, who have been asked to volunteer in order to test the system and its operation. Different subjects with different levels of motor disabilities are required in order to ascertain that the device can be customised to the particular subject, who may have a specific requirement in terms of controlling his/her environment.

You will be asked to perform a set of tests in the operation of the system using various sensors. These sensors may include a) touch sensors, b) movement/tilt sensors and/or a c) blink/wink switch (based on facial muscle movement). The sensors will be chosen on the basis that they are suitable for your particular movements. Placement of the sensors for the purpose of this study will be restricted to sensing arm or finger movements, head and neck movements and facial muscle movements i.e. eye-brow movement and winking actions. You will be asked to answer a questionnaire concerning your assessment of the performance and potential usefulness of the system. The only information that will be recorded will be the extent of your motor disability. The testing of the device has no risk to you and will not harm you in any way. You may also leave at any time throughout the testing procedure, without any adverse consequences to yourself or the system. The whole test procedure should not take longer than 1 hour to complete.

Thank you kindly for your participation.

Franco Senatore

G.3 Information Leaflet and Informed Consent

G.3.1 INTRODUCTION

You are invited to consider participating in a research study. Your participation in this study is entirely voluntary.

1. Before agreeing to participate, it is important that you read and understand the following explanation of the purpose of the study, and your right to withdraw from the study at any time without adverse consequences to yourself or the system. This information leaflet is to help you to decide if you would like to participate. You should fully understand what is involved before if you agree to take part in this study.
2. If you have any questions, do not hesitate to ask me.
3. You should not agree to take part unless you are satisfied about all the procedures involved.
4. If you decide to take part in this study, you will be asked to sign this document to confirm that you understand the study. You will be given a copy to keep.

G.3.2 PURPOSE OF THE STUDY

The study involves the development of a system which assists subjects with a wide range of severe motor disabilities (e.g. quadriplegics) with fairly diverse requirements in terms of performing basic tasks, such as turning a light switch on and off or controlling a wheel chair.

G.3.3 TESTING PROCEDURES

You will be asked to perform a set of tests in the operation of the system using various sensors (e.g. a touch pad) to control your environment e.g. switch a light on/off. These sensors will be placed in such as way as to efficiently detect residual movements that you may have:

For this study, the sensors will be restricted to the following:

- Touch sensors
- Movement sensors
- Blink/wink switch (facial muscle movement)

Placement of the sensors for the purpose of this study will be restricted to sensing arm or finger movements, head and neck movements and blowing or sipping actions through a straw. You will be asked to answer a questionnaire concerning your assessment of the performance and potential usefulness of the system. The only information that will be recorded will be the extent of your motor disability. Your identity will not be disclosed in any output, including reports, academic publications or thesis that may result from this work.

G.3.4 RISKS

No risks will be encountered throughout the testing procedure.

INFORMED CONSENT:

- I have received, read and understood the above written information (Participant Information Leaflet and Informed Consent) regarding the biomedical engineering study.
- I am aware that the results of the study, including personal details regarding my sex, age, date of birth, initials and diagnosis will be anonymously processed into a study report.
- I may, at any stage, without prejudice, withdraw my consent and participation in the study.
- I have had sufficient opportunity to ask questions and (of my own free will) declare myself prepared to participate in the study.

PARTICIPANT:

Printed Name	Signature / Mark or Thumbprint	Date and Time

G.4 Assistive Device Questionnaire

All sections are to be answered by the subject after performing various tests and will be filled out by myself.

I. SUBJECT DETAILS

Each subject will be assigned a number in order of testing performed.

SUBJECT NUMBER:___________________________

PARTICULAR MOTOR DISABILITY:____

____

DATE:_________________________________

II. INPUT SENSORS

i. USE OF THE INPUT SENSORS

a) The following table will be assessed and completed by the researcher, and concerns the use of various input sensors.

INPUT SENSOR	**IN USE?**	**SENSOR ATTACHMENT AREA**
Dual-axis accelerometer Y-axis		
Dual-axis accelerometer X-axis		
Touch sensor Channel 1		
Touch sensor Channel 2		
Touch sensor Channel 3		
Touch sensor Channel 4		
Touch sensor Channel 5		
Touch sensor Channel 6		
Blink/wink switch (SCATIR)		

ii. EASE OF USE AND COMFORT LEVEL OF THE VARIOUS INPUT SENSORS

a) How easy were the various sensors to use (on a scale of 1-5)? Mark the values below.

INPUT SENSOR	**RATING**					
Dual-axis accelerometer Y-axis	1	2	3	4	5	N/A
Dual-axis accelerometer X-axis	1	2	3	4	5	N/A
Touch sensor Channel 1	1	2	3	4	5	N/A
Touch sensor Channel 2	1	2	3	4	5	N/A
Touch sensor Channel 3	1	2	3	4	5	N/A
Touch sensor Channel 4	1	2	3	4	5	N/A
Touch sensor Channel 5	1	2	3	4	5	N/A
Touch sensor Channel 6	1	2	3	4	5	N/A
Blink/wink switch (SCATIR)	1	2	3	4	5	N/A

RATING KEY FOR SENSOR USABILITY LISTED ABOVE				
1 Impossible to use	2 Very difficult	3 Average	4 Easy	5 Extremely easy to use

b) How comfortable were the sensors to use (on a scale of 1-5)? Mark the values below.

INPUT SENSOR	**RATING**					
Dual-axis accelerometer Y-axis	1	2	3	4	5	N/A
Dual-axis accelerometer X-axis	1	2	3	4	5	N/A
Touch sensor Channel 1	1	2	3	4	5	N/A
Touch sensor Channel 2	1	2	3	4	5	N/A
Touch sensor Channel 3	1	2	3	4	5	N/A
Touch sensor Channel 4	1	2	3	4	5	N/A
Touch sensor Channel 5	1	2	3	4	5	N/A
Touch sensor Channel 6	1	2	3	4	5	N/A
Blink/wink switch (SCATIR)	1	2	3	4	5	N/A

RATING KEY FOR SENSOR COMFORT LEVEL LISTED ABOVE				
1 Very uncomfortable	2 Uncomfortable	3 Slightly uncomfortable	4 Comfortable	5 Extremely comfortable

iii. PERFORMANCE AND PREDICTABILITY OF THE INPUT SENSORS

How well did the various sensors perform in terms of providing predictable input readings based on your voluntary movement/s (on a scale of 1-5)? Mark the values below.

INPUT SENSOR	**RATING**					
Dual-axis accelerometer Y-axis	1	2	3	4	5	N/A
Dual-axis accelerometer X-axis	1	2	3	4	5	N/A
Touch sensor Channel 1	1	2	3	4	5	N/A
Touch sensor Channel 2	1	2	3	4	5	N/A
Touch sensor Channel 3	1	2	3	4	5	N/A
Touch sensor Channel 4	1	2	3	4	5	N/A
Touch sensor Channel 5	1	2	3	4	5	N/A
Touch sensor Channel 6	1	2	3	4	5	N/A
Blink/wink switch (SCATIR)	1	2	3	4	5	N/A

RATING KEY FOR SENSOR PERFORMANCE & PREDICTABILITY LEVEL LISTED ABOVE				
1 Very unpredictable performance (involuntary movements detected)	2 Unpredictable performance	3 Average performance	4 Good performance	5 Input sensor performs extremely predictably

iv. LEVEL OF FATIGUE EXPERIENCED

What level of fatigue did you experience during and after using the various sensors (on a scale of 1-5)? Mark the values below.

INPUT SENSOR	RATING					
Dual-axis accelerometer Y-axis	1	2	3	4	5	N/A
Dual-axis accelerometer X-axis	1	2	3	4	5	N/A
Touch sensor Channel 1	1	2	3	4	5	N/A
Touch sensor Channel 2	1	2	3	4	5	N/A
Touch sensor Channel 3	1	2	3	4	5	N/A
Touch sensor Channel 4	1	2	3	4	5	N/A
Touch sensor Channel 5	1	2	3	4	5	N/A
Touch sensor Channel 6	1	2	3	4	5	N/A
Blink/wink switch (SCATIR)	1	2	3	4	5	N/A

RATING KEY FOR THE FATIGUE LEVEL LISTED ABOVE				
1 No fatigue	2 Little fatigue experienced	3 Mild case of fatigue	4 Greater level of fatigue	5 Extreme level of fatigue

v. ADDITIONAL INFORMATION

Please state any additional information or concerns that you may have regarding the various input sensors and their functionality below:

III. THE OVERALL SYSTEM

i. CUSTOMISABILITY

a) In your opinion, how easily does the system accept various input sensors (on a scale of 1-5)? Mark the values below.

RATING OF THE SYSTEM CUSTOMISABILITY IN TERMS OF ACCEPTING VARIOUS INPUTS				
1 System accepts few to no input sensors	2 System accepts a few input sensors	3 System accepts some input sensors	4 System accepts most input sensors	5 System accepts all input sensors

b) In your opinion, how customisable is the system in terms of providing a specific output for a specific and voluntary input?

RATING OF THE OVERALL SYSTEM CUSTOMISABILITY LEVEL				
1 System is not customisable at all	2 System is only slightly customisable	3 System provides fair customisability	4 System is customisable to most inputs	5 System is fully customisable

ii. REPEATABILITY

In your opinion, how repeatable is the system i.e. how easily can the system repeat a specific task multiple times?

THE REPEATABILITY LEVEL RATING				
1 System does not repeat the various tasks accurately	2 System repeats the various tasks on a few accounts	3 System repeats the various tasks on some accounts	4 System repeats the various tasks on most accounts	5 System repeats all the various tasks accurately

iii. PERFORMANCE

In your opinion, what was the performance level of the system like i.e. did the system provide the required output for a specific input accurately, without complication and within an acceptable time frame?

THE PERFORMANCE LEVEL RATING				
1 System performs very poorly	2 System performs poorly	3 System performs adequately	4 System performs well	5 System performs excellently

iv. REQUIREMENTS

Based on the tests performed, do you think that the system is capable of satisfying some of your requirements, i.e. is the system specific enough for your needs?

REQUIREMENTS RATING				
1 The system is incapable of meeting any of my requirements	2 The system would meet very few of my requirements	3 The system could meet some of my requirements	4 The system could meet most of my requirements	5 The system would meet any or all of my requirements

v. ADDITIONAL INFORMATION

Please state any additional information or concerns that you may have regarding the overall system:

IV. POSSIBLE IMPROVEMENTS TO THE SYSTEM

Please state any additional functionality that you may require as well as any possible improvements to the system:

You have now reached the end of the questionnaire.

Thank you very much for your time!

Appendix H

User Manual

H.1 Introduction

This document is an operation guide for the "Generic Assistive System". The setup of the various input sensors as well as plugging these into the system are described here. This follows with information on using the signal selection and conditioning circuit as well as the *NI™ PCI 6221 DAQ* board and the *NI™ CB-68LP* terminal connector board. Lastly, extensive information is provided on setting up and using the software application to provide certain output functionality. The complete *Generic Assistive System* is shown in Figure H.1 below.

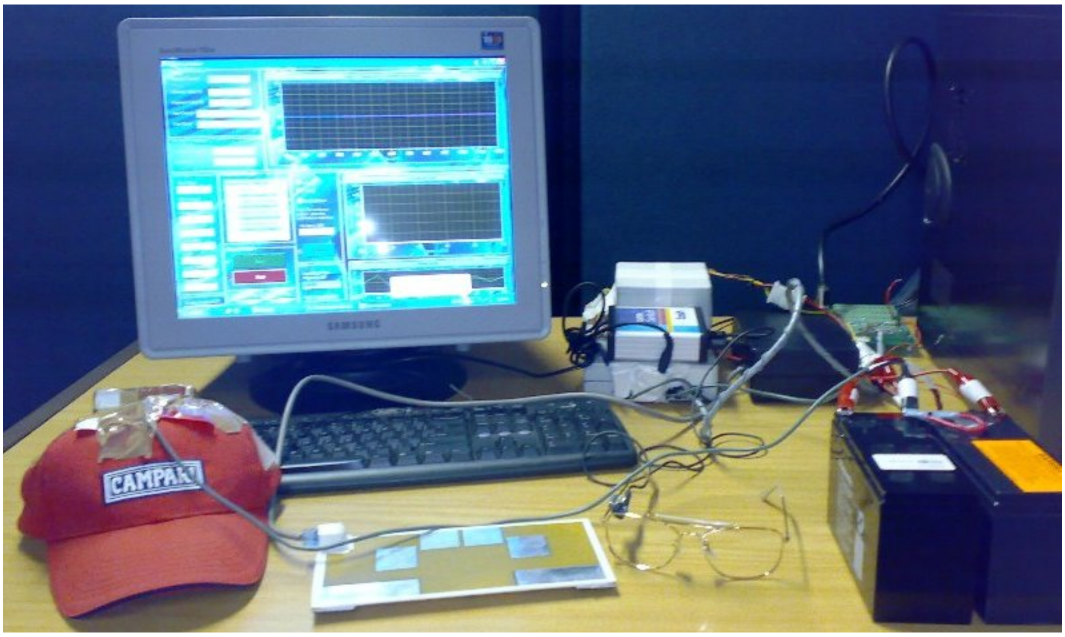


Figure H.1: The Generic Assistive System

H.2 Operating Procedure

H.2.1 Input Sensors

Prerequisites

All of the input sensors must have physical leads or wires, which provide a specific sensor signal. Positioning and setup of the sensors is also important to ensure correct and accurate operation.

Dual-Axis Accelerometer

This particular sensor may be used to detect movement/acceleration of the user. It may be attached to the head, neck or shoulder regions and may also be used a a tilt sensor, e.g. the user may wish to detect the level to which he/she had tilted sideways or up/down. The dual-axis accelerometer input sensor is illustrated in Figure H.2 below.

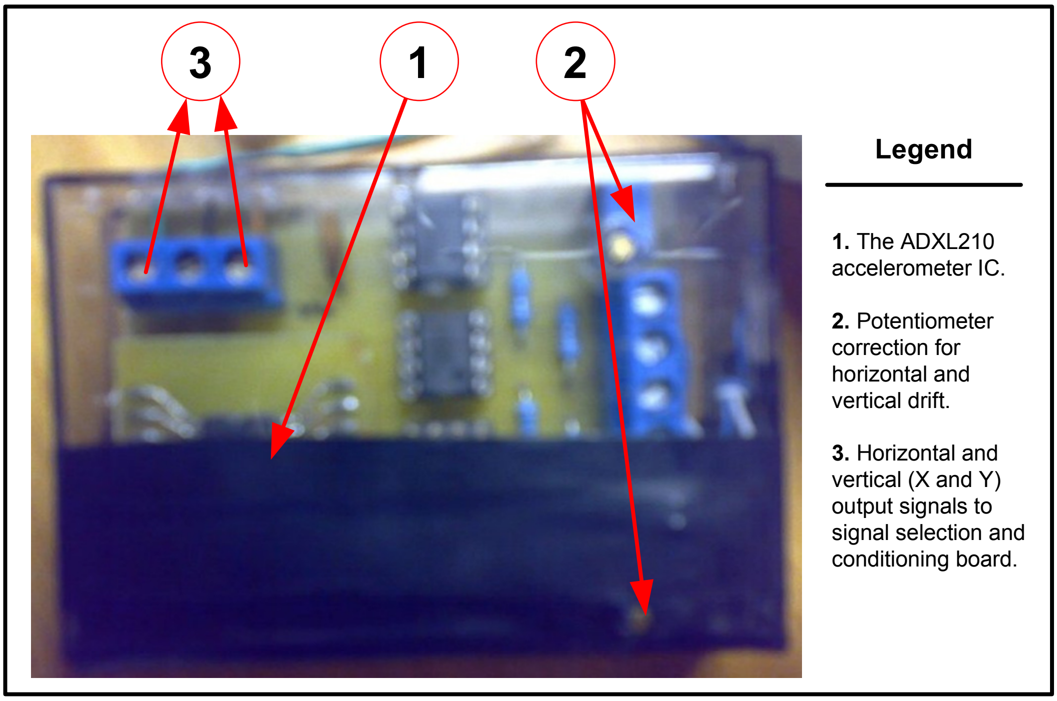


Figure H.2: The Dual-Axis Accelerometer

QT160 6-Key Touch Sensor

This sensor is used to detect slight touch by the user, which in this case is based on slight or residual finger movement. The touch sensor consists of a 6-key touch pad (Figure H.3) which is plugged into the *QT160D* control box as shown in Figure H.4.

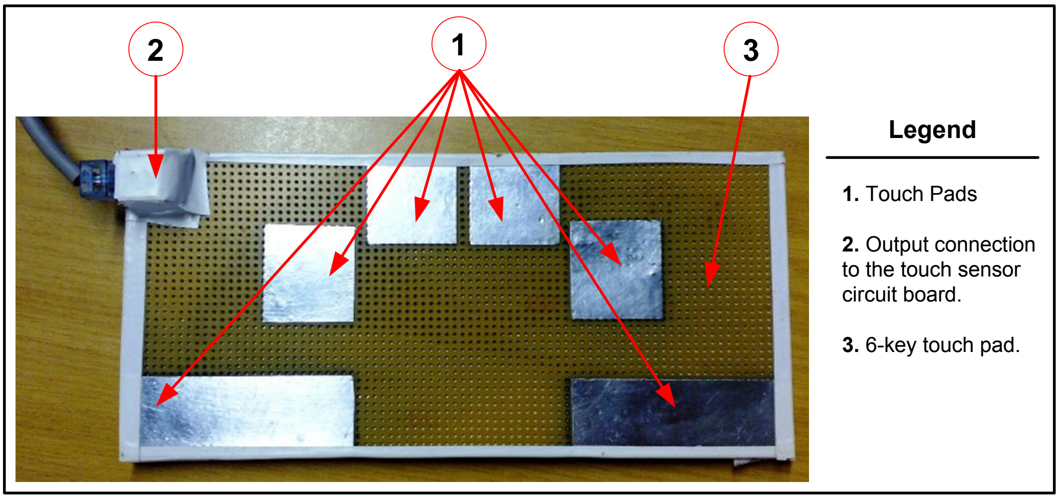


Figure H.3: 6-Key Touch Pad

Increasing and Decreasing the Sensitivity: The easiest way to change the sensitivity is to decrease or increase C_s (between $10\mu F$ and $47\mu F$), thereby decreasing or increasing the sensitivity respectively (Figure H.4). Other methods of changing the sensitivity of the sensor are discussed in Section 1.3.6 of the QT160 data sheet [24].

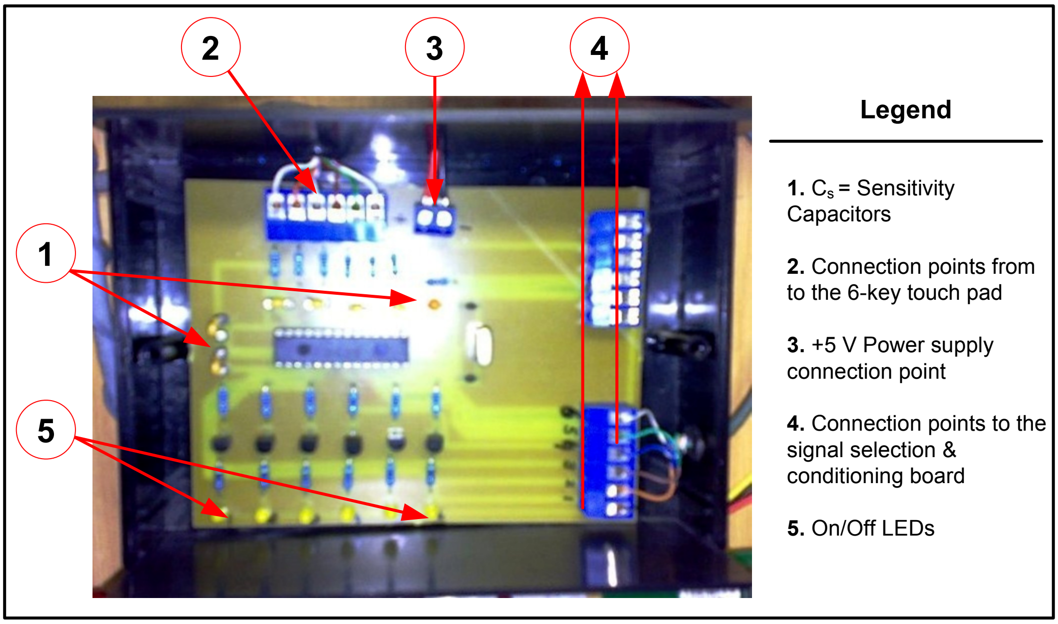


Figure H.4: The QT160 Touch Sensor Circuit

SCATIR Blink/Wink Switch

This is a blink/wink sensor which detects blink eye-movements and eyebrow movements and not the natural wink eye-movements. The full Tash blink/wink SCATIR switch system is shown in Figure H.5 below.

Sensor Detection: The input detection is provided via a retroreflective sensor attached to eye-glasses, which is consequently attached to the Tash SCATIR switch via an input jack connection, which activates a red LED (marked “Sens Pwr”). Blinking or moving one’s eyebrows up and down activates a green LED (marked “Relay On”), consequently provides an output signal(Figure H.6).

Output: The sensor output connects to the sensor signal selection and conditioning unit via a relay cable connected to the the “Relay Out” connection point of the SCATIR switch (Figure H.6).

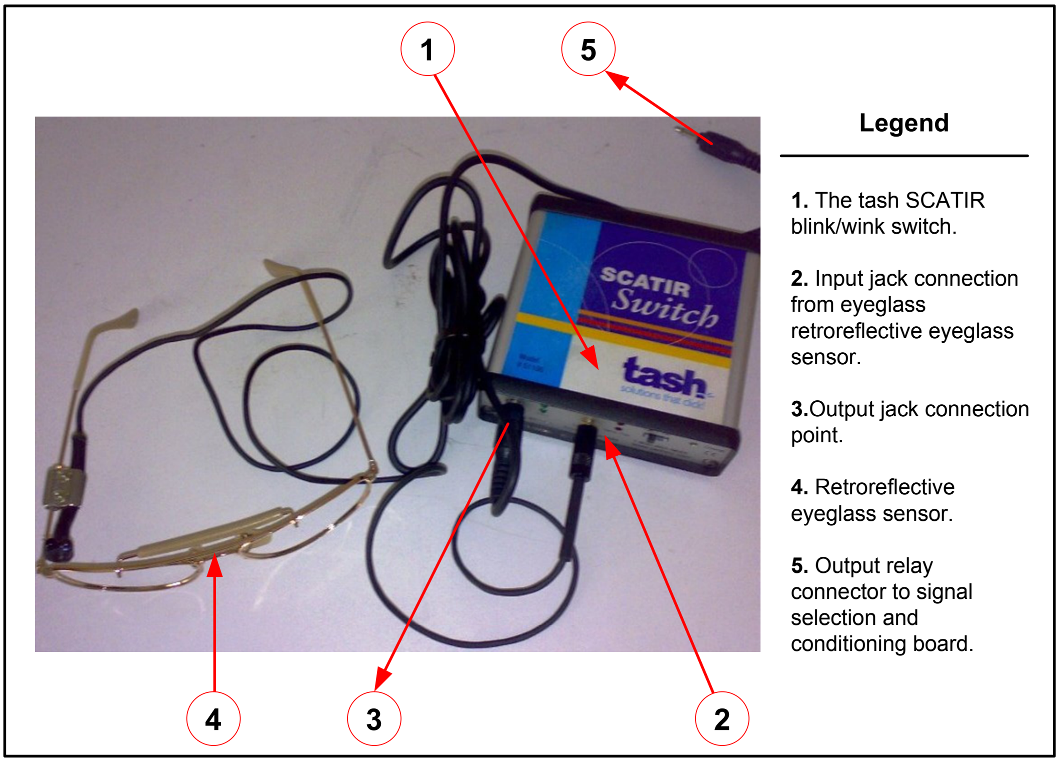


Figure H.5: The TASH SCATIR Switch with Eye-glasses

H.2.2 Signal Selection and Conditioning Board

The signal selection and conditioning board provides the functionality of conditioning the input sensor signals so that they may be in a form which is "acceptable" for the analogue-to-digital converter, which subsequently "connects" the analogue input signal to a software application. The full sensor signal selection and conditioning board is shown in Figure H.7.

Connecting the input sensors to the board

The various input sensors connect to this board via a series of 8 blue input channels/terminals, where up to 8-output signals are provided for these sensors at the output terminal block connection as shown in Figure H.8. These are subsequently connected to the *NI™ CB-68LP terminal connection board.*

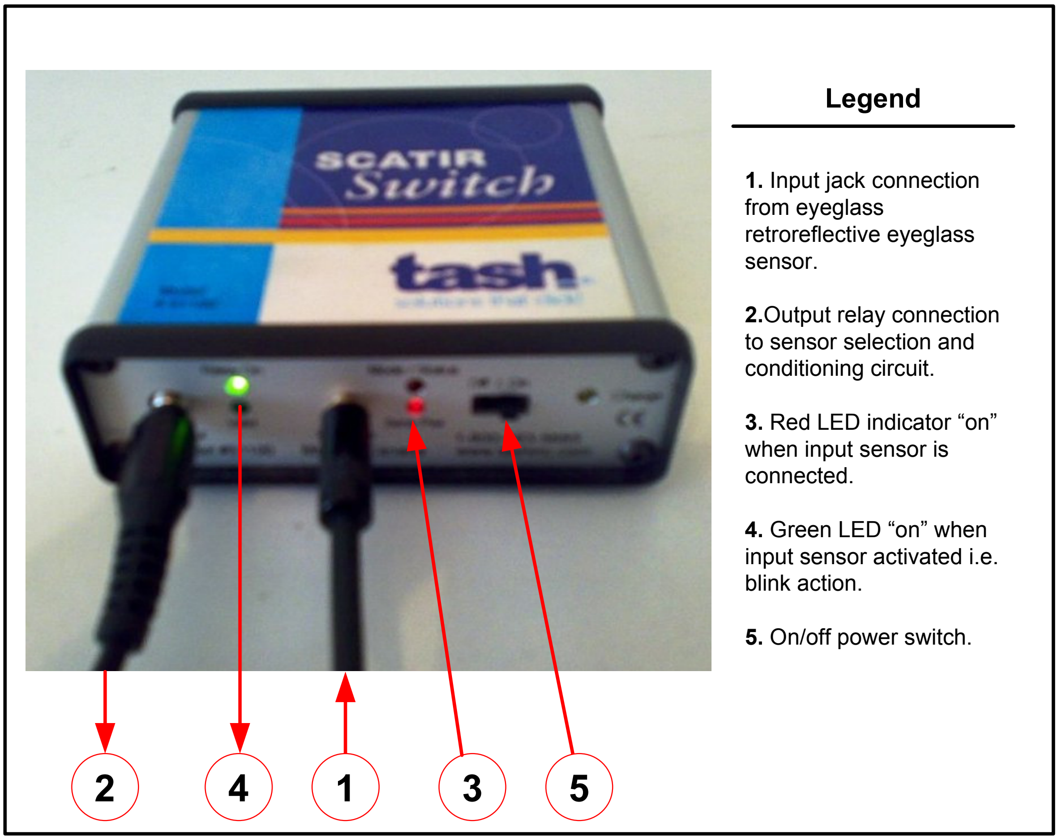


Figure H.6: SCATIR Switch Operation

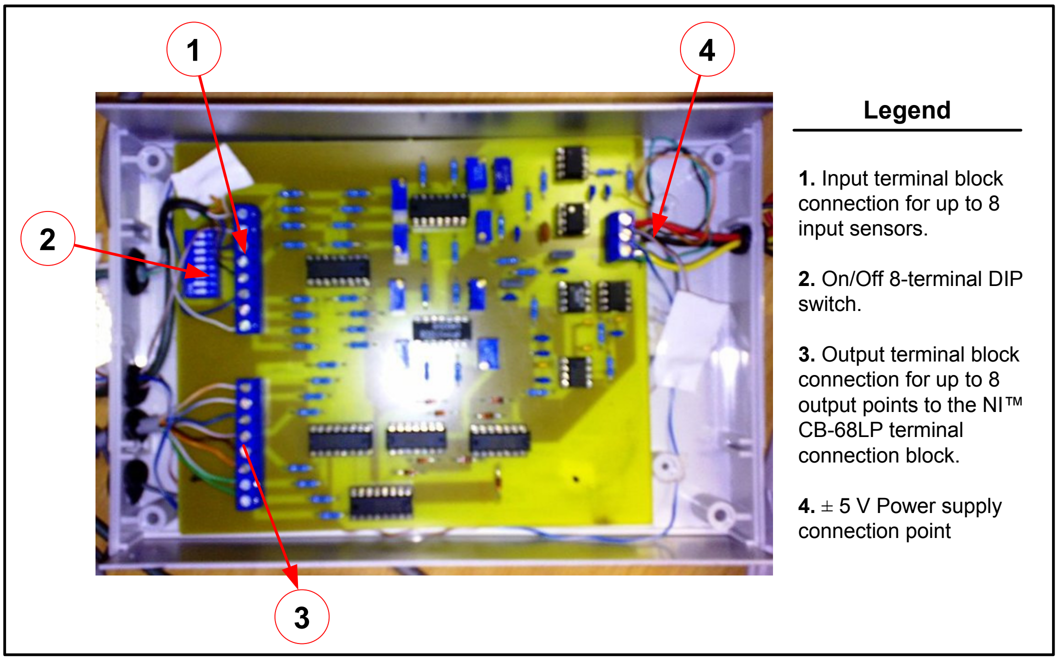


Figure H.7: The Signal Selection and Conditioning Board

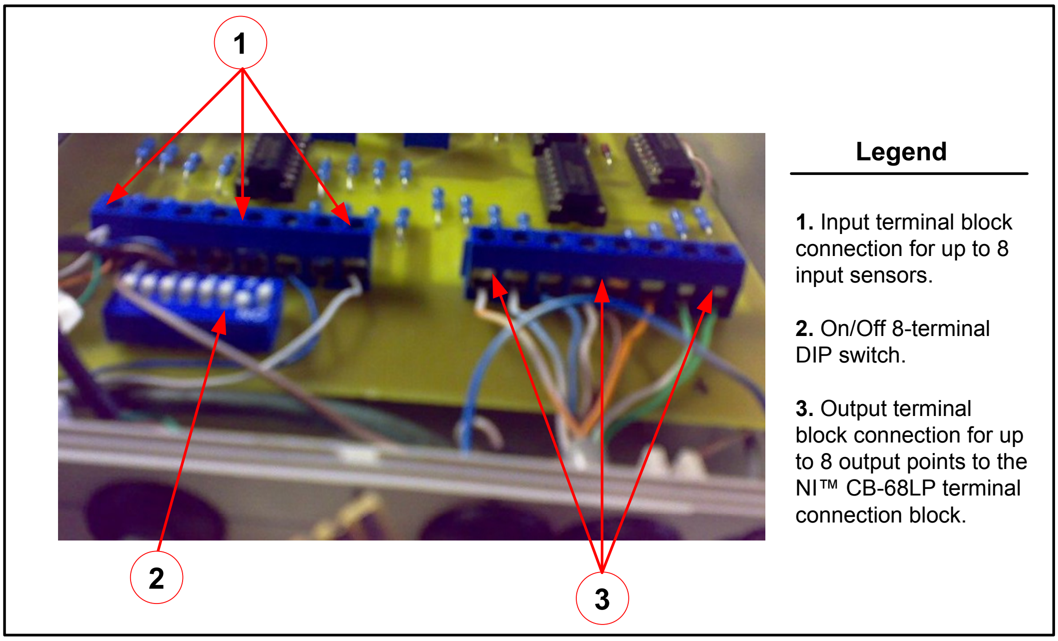


Figure H.8: Connecting the input sensors to the signal selection and conditioning board

H.2.3 NI PCI 6221 DAQ board and terminal board

Input Connections

Up to 8 single-ended signals are connected to the NI™ CB-68LP terminal connection board from the signal selection and conditioning board as shown in Figure H.9 below.

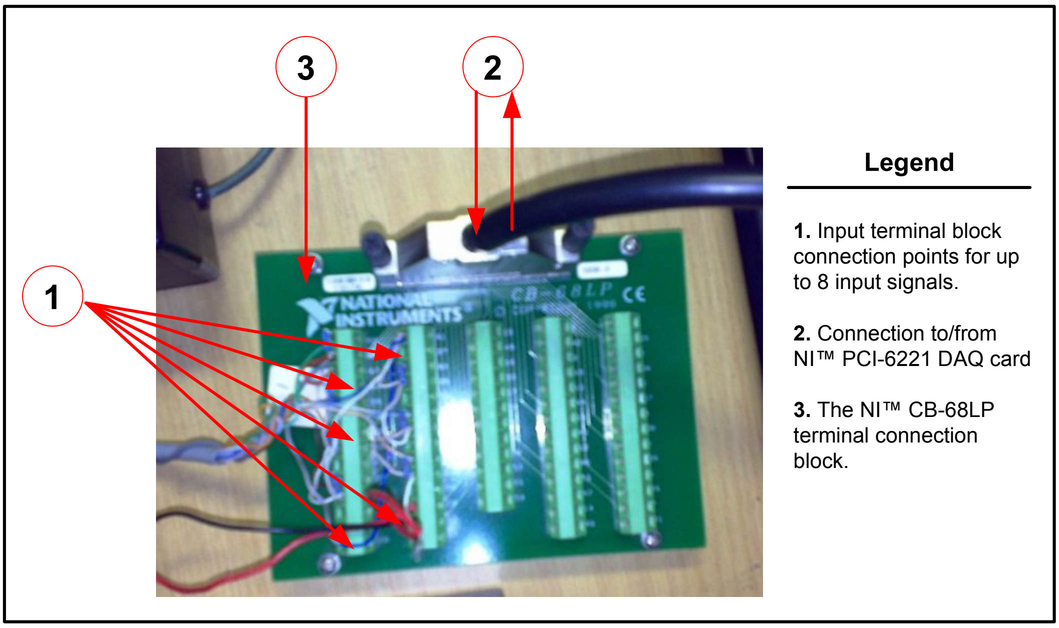


Figure H.9: The NI™ CB-68LP Terminal Block

Terminal Block to DAQ connection

The terminal block is plugged into the The NI™ PCI-6221 DAQ card via the connector shown in Figure H.10. The DAQ card is connected to the terminal block at the back of the PC as shown in Figure H.11.

Figure H.10: The DAQ Card Connector

Figure H.11: Connection to the NI PCI-6221™ DAQ card

H.2.4 Software Application

Opening the Application

Execute the application by selecting "GenAssDev2.exe". This opens the "Generic Assistive Device" software application. The main menu is shown in Figure H.12 below.

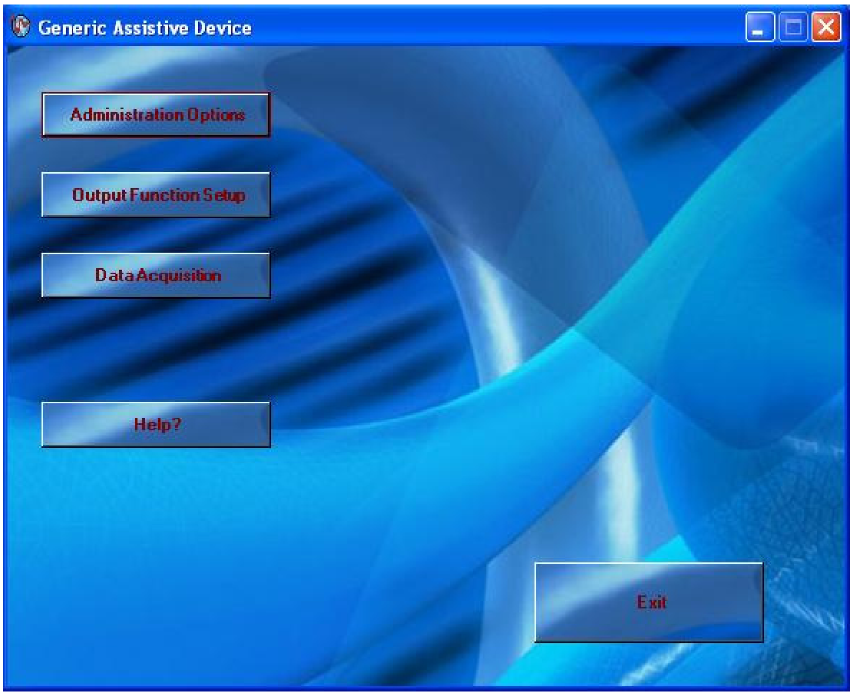


Figure H.12: Generic Assistive Device - Main Menu

Administrator Options

Select the "Administrator Options" button to open the "Administrator Options" dialog, shown in Figure H.13 below.

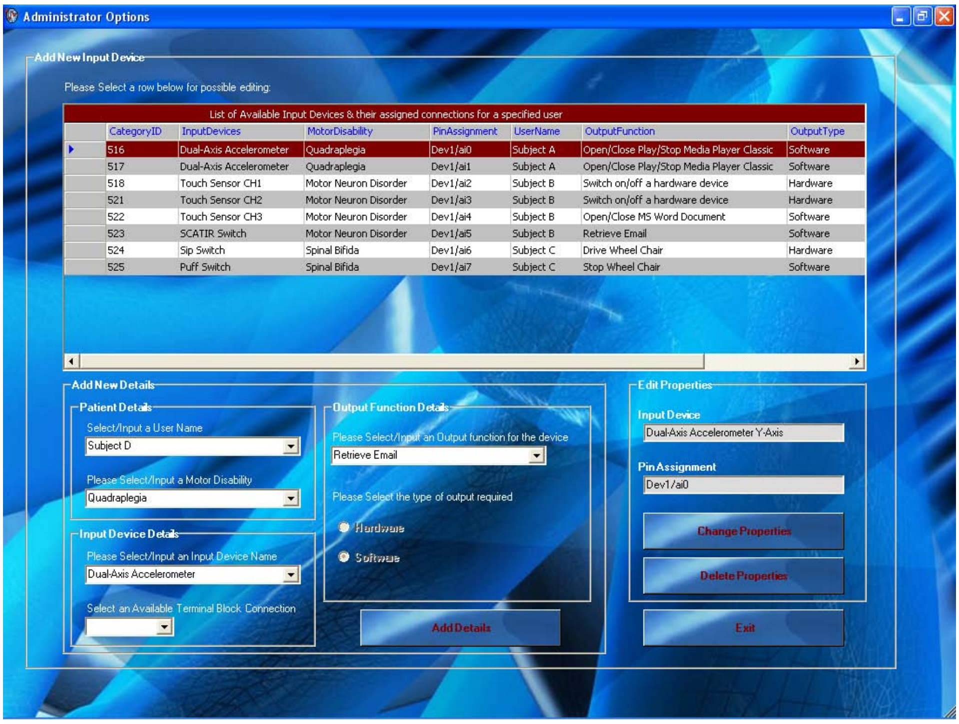


Figure H.13: Administrator Options Dialog

1. *Adding Details to the database*

   - Load the subject information including the name and type of motor disability.
   - Assign the various input device details, which include the input device/sensor name and the associated terminal block connection.
   - Provide the output function details (i.e. the output function name) and output type. Make sure to specify whether these are hardware or software output types.
   - Once all the details have been provided, the user/administrator must single left-click the "Add Details" button, which will populate a row consisting of the specified details to the "List of information regarding various subject, input device and output function details" data grid.
   - Repeat the above process for any other input channels, which have been connected to various input sensors. The added details for all 8 input channels is shown in Figure H.14 below.

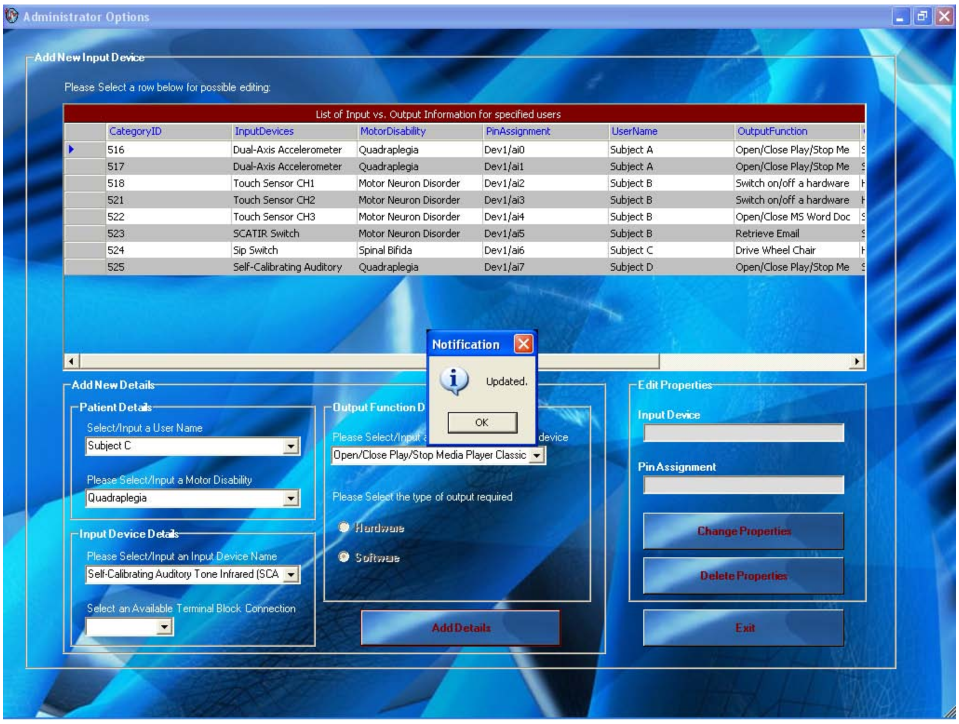


Figure H.14: Adding details to the datagrid/database

2. *Changing Details to the database*

   Should the user wish to change certain information in a specific row, he/she must select the row from the data grid and subsequently enter all the changes required in the "Add New Details" group box. Left-click the "Change Properties" button to save the changes made to the database, which subsequently updates the data grid with the new information for the selected row. An example of this is shown in Figure H.15, where some information has been changed for the selected "Dev/ai7" input channel.

3. *Deleting Details from the database*

   To delete a specific row of details the user must firstly select the row, followed by left-clicking the "Delete Properties" button to delete the data from the datagrid/database. This results in a confirmation message and thereafter the information is deleted from the data grid once "OK" has been selected. An example for input channel "Dev/ai7" is shown in Figure H.16.

4. *Exiting the "Administrator Options" dialog*

   To exit the "Administrator Options" dialog select the "Exit" button on the bottom right-hand corner of the window.

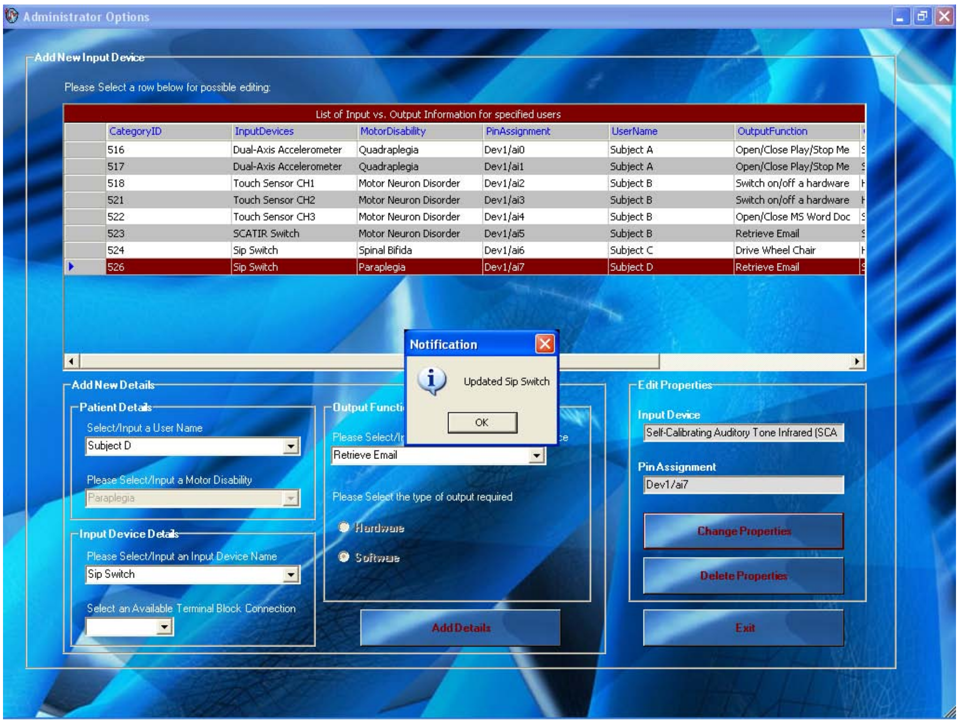


Figure H.15: Changing the details of the datagrid/database

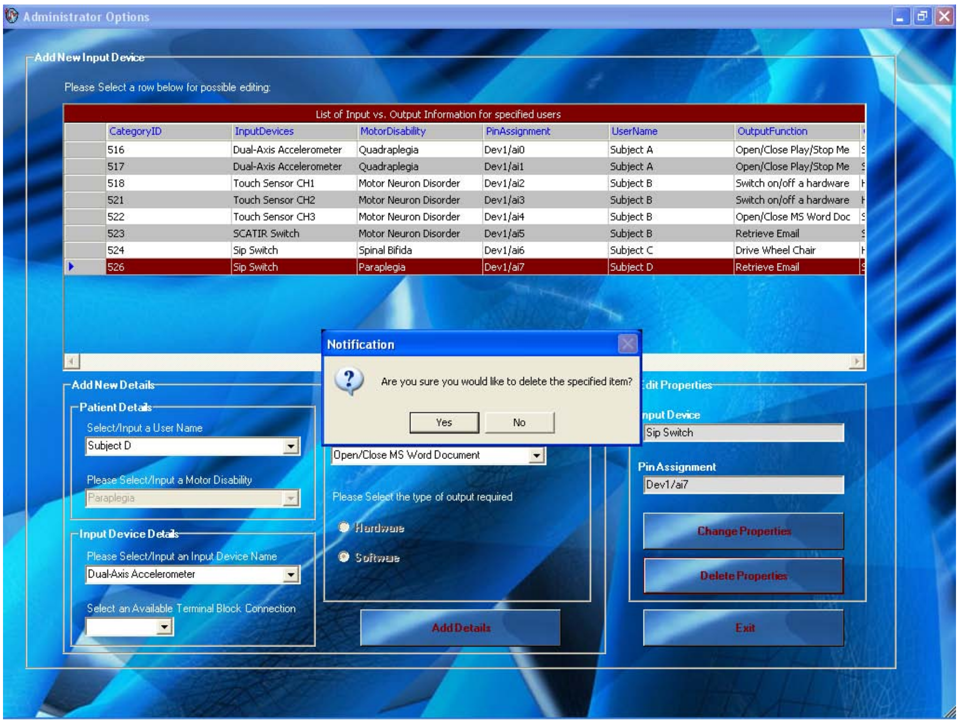


Figure H.16: Deleting certain details from the datagrid/database

Output Function Setup

Select the "Output Function Setup" button to open the "Output Function Setup" dialog, shown in Figure H.17 below.

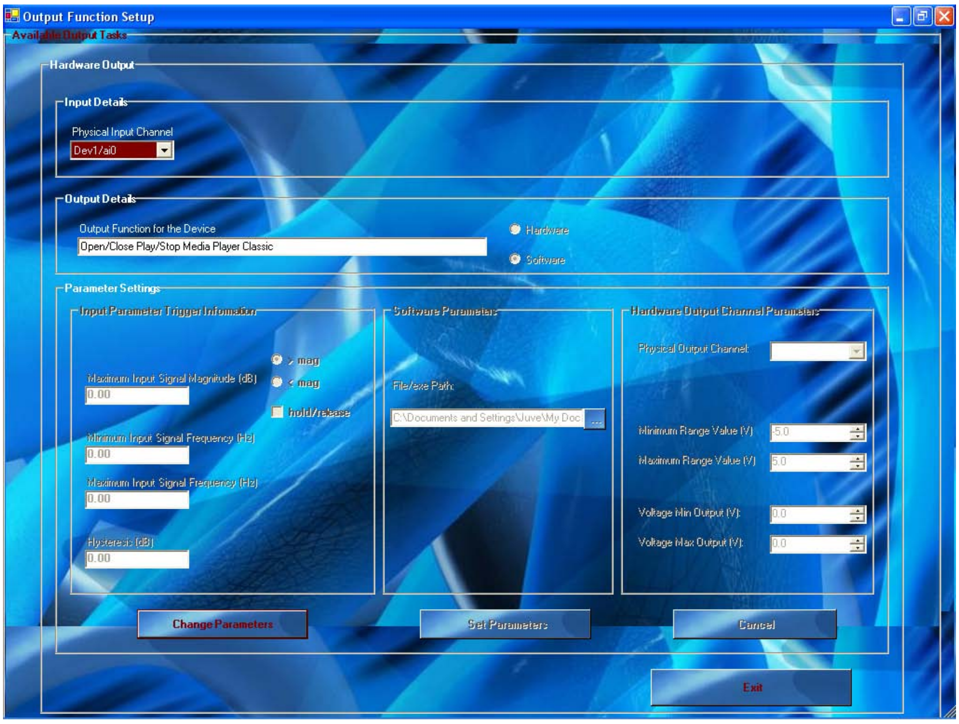


Figure H.17: The Output Function Setup dialog

1. *Adding/Changing input and output trigger parameters*

   - Select a channel, which will populate the "Output Function for the Device" group box with the output function and its associated type i.e. hardware or software.
   - Select the "Change Settings" button. This allows the user access to add/select parameter information to/from the "Parameter Settings" group box, which consists of the "Parameter Settings", "Software Parameters" and "Hardware Output Channel Parameters" group boxes.

   (a) *Input Trigger Parameters*

   The "Input Parameters Trigger Information" group box parameters specify at what levels the input signal would trigger in an output function (hardware or software). These details are populated in the list and are calibrated in the "Data Acquisition" window to be discussed later. The data acquisition process is discussed in further detail later in the document.

(b) *Software and Hardware Output Functions*

The administrator will only be allowed access to the "Software Parameters" and "Hardware Output Channel Parameters" group boxes, depending on whether the required output function is software or hardware in nature respectively.

- If the user has selected an input channel, which requires a software type of output, he/she must select the "" browse button, in order to search for a particular plug-in application executable file. Select the "Ok" button to save the file path to the database.
- If the output type, relating to the specified input channel, is a physical signal output (i.e. used to drive certain output devices), specify the output channel to be used as well as the voltage range and actual minimum and maximum voltage levels. The closer the min/max voltage range values are to the actual minimum and maximum input values, the better the resolution of the D/A conversion for the output signal.

2. *Saving/setting the input and output trigger parameters*

   - Select the "Set Parameters" button to save the data to the database.
   - Repeat the above process for the other available channels.

   An example of the saved settings is shown in Figure H.18 below.

3. *Cancelling the setting of parameters*

   If the "Change Parameters" button was selected erroneously, the administrator has the option of canceling this as well as any information entered into the various fields, by left-clicking the "Cancel" button.

4. *Exiting the "Output Function Dialog"*

   In order to exit the "Output Function Setup" dialog, the user must click the "Exit" button on the bottom right-hand corner of the dialog window.

Data Acquisition Setup

Select the "Data Acquisition" button on the main form to open window consisting of the functionality for the digital signal processing and data acquisition. This is illustrated in Figure H.19 below.

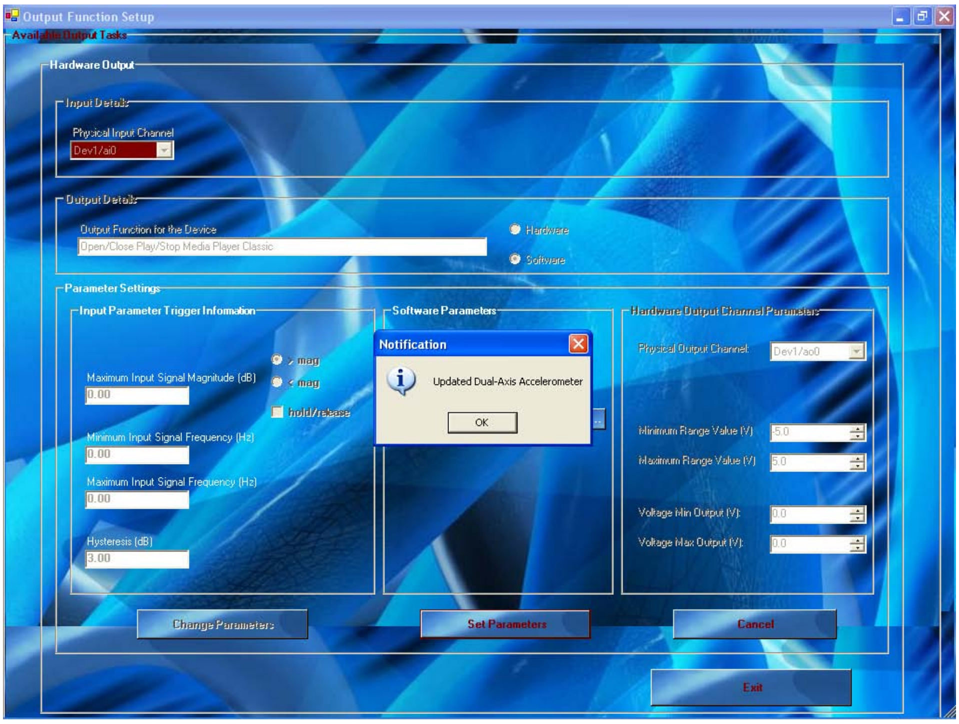


Figure H.18: Setting the input and output function trigger parameters

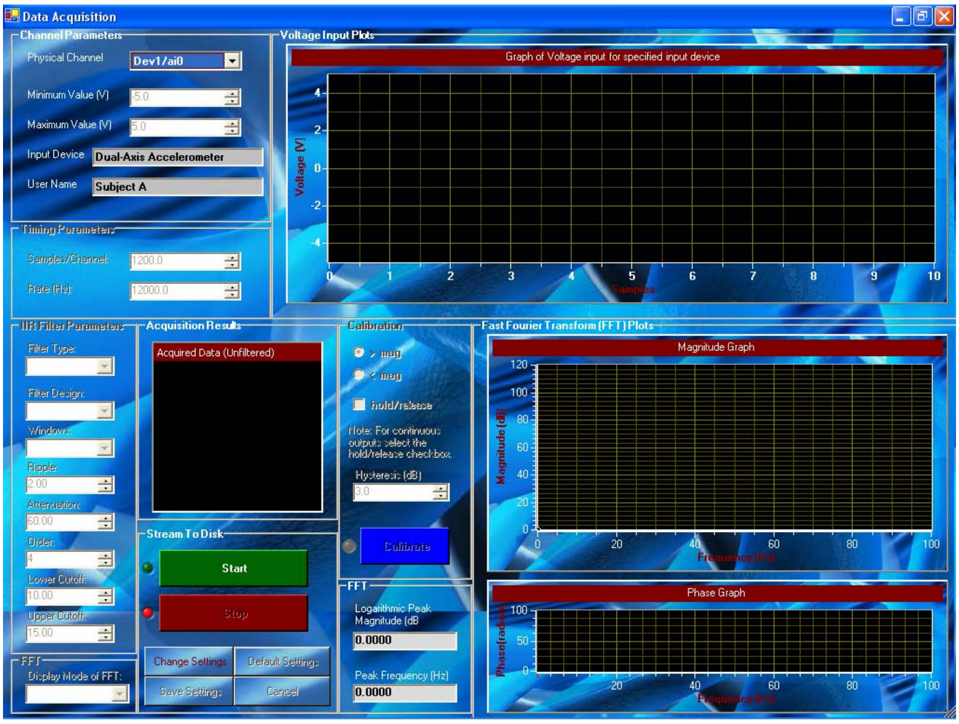


Figure H.19: The Data Acquisition dialog

1. *Digital Signal Processing Parameter Setup*

Select a physical input channel from the list. This must populate the "User Name" and "Input Device" with the required user name and input device information respectively.

(a) *DSP Information*

Select the "Change Settings" button and thereafter specify the digital filtering, timing, windowing and voltage range required for the input signal. The "Filter Type" requires the user to input a low pass, high pass, band pass or band stop filter type. For the purposes of this application, the type of digital filter required is a low pass. The filter design to be used is dependent on the input signal. The "Elliptic" and "Butterworth" are the best as they provide steep a steep roll-off, which is required for signals of specifically small frequency ranges, such as those being dealt with. A digital "Bessel" filter may be used, but a higher order filter may be required to provide the steeper roll-off, which may slow the performance of the data acquisition down. The "Ripple" and "Attenuation" give specific details for the filter performance, where the default values of 2 and 60 should be provided respectively. The "Upper Cutoff" and "Lower Cutoff" frequencies provide the upper and lower limit for the input signal. The lower cutoff frequency is the actual cutoff frequency provided, as used by the low pass filter, where the upper cutoff frequency ensures that the output function does not trigger at values above this frequency.

These values are of extreme importance as they provide the frequency range for which the input signal would trigger an output function.

(b) *Timing Information*

The "Timing Parameters" require the user to specify the "Samples/Channel" and the "Rate", which is the sampling rate/frequency at which the input signal is sampled. Note that the higher the number of samples per channel, the better the resolution of the input signal, i.e. information such as the frequency the FFT response becomes more accurate. The system has been designed for a sampling rate of 12 kS/s, which provides very accurate information for the input signal.

(c) *Input Range*

The minimum and maximum voltage range specifies the threshold levels for the input signal, which ensures that the A/D conversion is performed with the highest resolution. To ensure this, the min/max voltage range values must be as close to the actual input signal minimum and maximum voltages.

(d) *Calibration of input trigger parameters*

To start the calibration of the input trigger parameters, left-click the “Calibrate” button, which results in the button’s text changing to “Save Calibration”. Access will thereby be granted to the various calibration parameters.

Select whether the input parameter triggers above or below this calibrated value by selecting the “>mag” or “<mag” radio buttons respectively. The “hold/release” radio button may be selected in order to hold or release the state of the output function, with each subsequent output trigger action resulting in the output’s state being held or released i.e. toggled. This follows with a value being entered into the “Hysteresis (dB)” numeric up down box, typically a value of 3 or 4 dB. Hysteresis ensures that the output function is not triggered continuously while the input is at the required input trigger level. Selecting the “Calibration” button starts the processing of the input data, which are consequently displayed on the “Voltage Input Plot” graph along with data being displayed on the “Fast Fourier Transform (FFT) Plots”. The administrator must perform a trigger input action, which would be used later to perform the specified output function for the selected input channel. The magnitude and the frequency at which this input trigger action occurs are displayed numerically on the “Logarithmic Peak Magnitude (dB)” and “Peak Frequency (Hz)” boxes respectively. Once satisfied with the results, the input trigger information may be stored to the database by selecting the “Save Calibration” button. This should result in the various input trigger parameters being inaccessible. This also results in the button’s text changing back to “Calibrate”.

It must be noted that if the calibration is performed again (once the calibration has been performed for the specified channel), it must not result in an output function being triggered for the same input trigger action. This should only result once the “Start” button has been selected. This is discussed in further detail in Section 2 of “Data Acquisition”below.

(e) *Saving the information to the database*

Select the “Save Settings” button to save all the data for the specified channel to a database for later analysis.

(f) *Cancelling the setting of parameters*

Repeat the above process for all of the input channels. The user has the option of using default values, which are used when the selecting the "Default Settings" button.

2. *Data Acquisition*

   Repeat the above process for all of the input channels. The user has the option of using default values, which are used when the selecting the "Default Settings" button.

   (a) *Starting the data acquisition*

   Select the "Start button" to start the data acquisition process.

   The unfiltered and filtered input signal, as well as the windowing function is displayed on the "Graph of Voltage for specified input device" plot. A hidden legend for this plot comes into view once the administrator passes the mouse pointer over the plot.

   (b) *The Fast Fourier Transform (FFT)*

   Check the input trigger levels for each channel in terms of the FFT input signal magnitude and the associated frequency for which this occurs, which is provided graphically on the "Magnitude Graph" and numerically on the "Logarithmic Peak Magnitude (dB) and "Peak Frequency (Hz) respectively. The phase response is provided graphically on the "Phase Graph".

   An example of the data acquisition process is shown for a dual-axis accelerometer y input signal in Figure H.20 below. This particular example illustrates the input channel "Dev1/ai0" being triggered at $\approx 3.5V$, as shown on the "Acquired Data (Unfiltered)" data grid and the "Graph of Voltage for specified input device" plot. The red plot displays the unfiltered input signal, the green plot is the filtered input signal and the blue line illustrates the windowing function for the filtered signal. Here the peak magnitude of $\approx 62.4dB$, which occurs at a frequency of $5Hz$. However, the lower frequency cutoff is $4Hz$; therefore the peak magnitude should occur at a frequency which is below $4Hz$. To ensure better accuracy in terms of the fast Fourier transform and the resultant magnitude and frequency values, the number of samples/channel need to be increased above $2400 samples/channel$, which consequently increases the accuracy of the acquisition. This, however, reduces the performance of the acquisition dramatically, therefore a trade-off has been done between accuracy and performance, and it was found that values ranging between 2400 and 3000 samples/channel is suitable for the required purpose. For this example an output function would result with an input trigger magnitude level of $\approx$ 59 dB and a frequency range between $3Hz$ and $7Hz$, with an approximate hysteresis level of $3dB$.

It must be noted that the data acquisition has been performed simultaneously for all of the input channels, which have been specified in the "Administration Options" dialog.

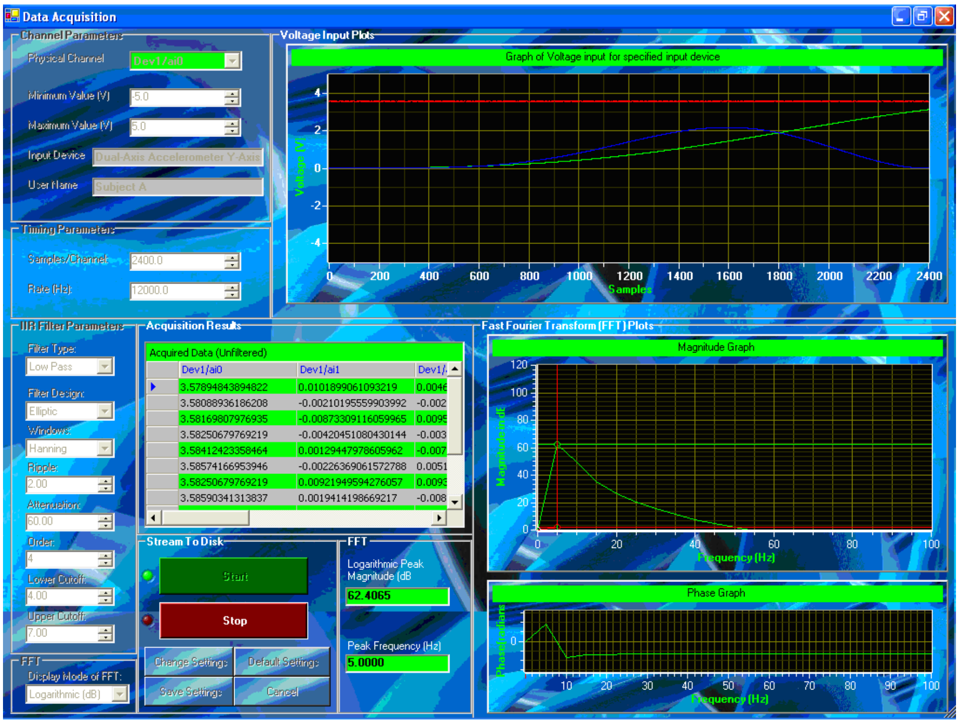


Figure H.20: The data acquisition process

(c) *Stopping the data acquisition*

To stop the data acquisition the user must select the "Stop" button, which also disposes of any tasks which may be running.

3. *Exiting the "Data Acquisition" dialog*

To exit the data acquisition dialog select the red crossed button in the top right-hand corner of the dialog window.

4. *Considerations for the performance of the data acquisition and triggering an output response*

- If an input trigger level is reached without an output trigger action, the user must stop the data acquisition by clicking the "Stop" button and recalibrating the channel by following the calibration routine described in Section 1d of "Data Acquisition Setup".
- Ensure that the data acquisition is working properly for all of the channels and ensure that a trigger action results in the correct output function for a specific input channel.
- Ensure that the lag of the data acquisition is minimal and that it performs the required functions efficiently.

H.2.5 Help

The "Help" button is selected in the main dialog window. This will open a new dialog window, consisting of a basic help text file covering all possible information that may be required as well as troubleshooting possible problems which may occur.

H.2.6 Exiting the application

The program is exited by selecting either the "Exit" button or the red crossed button in the top right-hand corner of the main dialog window.

H.3 Conclusion

The user manual provided detailed setup and operating information for the *Generic Assistive System* to be used on people suffering with various disabilities. This included information on the hardware and software devices being used for the system.

Appendix I

Contents of the Compact Disk

The attached compact disk (CD) contains additional resources that may be useful to readers of this document. The directory structure of the CD is shown in Figure I.1.

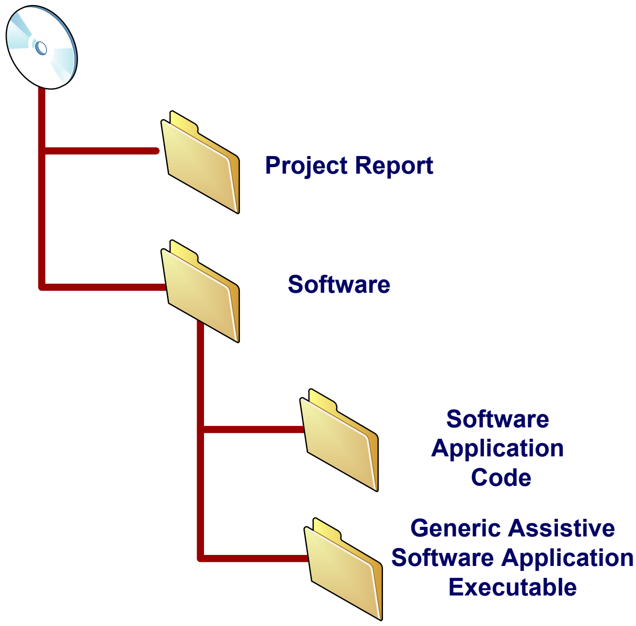


Figure I.1: Directory structure of attached CD

The contents of the CD are organized as follows:

- The Project Report directory contains an electronic copy of this document and requires Adobe Acrobat Reader to view the file.
- The Software directory contains the "Generic Assistive Software Application" executable as well as the code used.

References

[1] UAB Department of Physical Medicine and Rehabilitation, Washington DC, USA, *National Spinal Cord Injury Statistical Center*. http://images.main.uab.edu/spinalcord/graphicimages/bspine.gif. Last accessed 2005-08-29.

[2] B. Hamilton, J. A. Laughlin, C. V. Granger, and R. M. Kayton, "Interrater Reliability of the 7-level Functional Independence Measure (FIM)," in *Scandinavian Journal of Rehabilitation Medicine*, pp. 26(3):115–119, 1994.

[3] J. F. Ditunno, W. Young, W. H. Donovan, and C. G., *The International Standards Booklet for Neurological and Functional Classification of Spinal Cord Injuries.*, vol. 32. pp. 70-80.

[4] R. Jamal and R. Steer, "Filters," in *The Measurement Instrumentation and Sensors Handbook*, CRC Press LLC, 2000.

[5] American Spinal Injury Association, Chicago, IL, *Standards for Neurological Classification of Spinal Injury Patients*, 1996.

[6] D. L. Damiano and et al., "What does the Ashworth scale really measure and are instrumented measures more valid and precise?," in *Developmental Medicine and Child Neurology*, (Washington University - Human Performance Laboratory/Barnes-Jewish Hospitals, 4555 Forest Park Parkway, St. Louis, MO 63108, USA), pp. 44: 112–118, 2002.

[7] I. Bromely, *Tetraplegia and Paraplegia - A Guide for Physiotherapists*. Edinburgh: Churchill Livingstone, fifth ed., 1998.

[8] R. M. Hodapp, *Developement and Disabilities: Intellectual, Sensory and Motor Impairments*. Cambridge: Cambridge University Press, 1998.

[9] "Spinal cord 101," *Spinal Cord Injury Resource Centre*, 20 September 2007. http://www.spinalinjury.net/html/_spinal_cord_101.html. Last accessed 2006-07-09.

[10] R. S. Porter and J. L. Kaplan, “Brain, Spinal Cord, and Nerve Disorders: Spinal Cord Disorders,” in *The Merck Manual of Medical Information - Home Edition*, (Whitehouse Station, N.J. U.S.A.), Merck & Co., Inc., Merck Research Laboratories, 2004-2007.

[11] R. Schmidt, *Motor control and learning - A behavioural emphasis*. Champaign, Illinois: Human Kinetics Publishers Inc., 1982.

[12] V. Kumar, R. Tariq, and K. Ventkat, “Assistive devices for people with motor disabilities.,” in *Wiley Encyclopaedia of Electrical and Electronics Engineering*, 1997.

[13] R. N. Scott, R. H. Brittain, R. R. Caldwell, A. B. Cameron, and V. A. Dunfield, “Sensory-feedback system compatible with myoelectric control.,” in *Medical and Biological Engineering and Computing.*, ch. 18(1), pp. 65–69, 1980.

[14] A. H. Bottomly, “Myo-electric Control of Powered Prostheses.,” in *The Journal of Bone and Joint Surgery.*, pp. 411–415, 1965.

[15] V. Brooks, *The Neural Basis of Motor Montrol*. New York: Oxford University Press,, 1986.

[16] N. V. Thakor, “Biopotentials and electrophysiology measurement,” in *The Measurement Instrumentation and Sensors Handbook*, CRC Press LLC, 2000.

[17] “Motor Disabilities - Assistive Technologies,” 1999-2007. http://www.webaim.org/articles/motor/assistive. Last accessed 2006-09-02.

[18] R. J. Hansman Jr., “Characterisitics of instrumentation,” in *The Measurement Instrumentation and Sensors Handbook*, CRC Press LLC, 2000.

[19] J. Webster, “Reducing motion artefacts and the interference in biopotential recording,” in *IEEE Transactions on Biomedical Engineering*, pp. 31, 823–826, 1984.

[20] E. K. Antonsson and R. W. Mann, “The frequency content of gait,” *J.Biomech*, vol. 18, no. 1, pp. 39 – 47, 1985.

[21] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, “Implementation of a Real-Time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, pp. 39 – 47, January 2006.

[22] A. Burns and A. Wellings, “Real-Time Systems and Programming Languages (Third Edition),” in *Ada 95, Real-Time Java and Real-Time POSIX*, Addison Wesley Longmain, March 2001.

[23] "Low Cost $\pm 2g/\pm 10g$ Dual Axis ***i*MEMS®** Accelerometers with Digital Output - ADXL202/ADXL210," Data Sheet Rev. B, ANALOG DEVICES, One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A., 1999. World Wide Web Site: http://www.analog.com.

[24] "QProx™ QT160/QT161 6 Key Charge-Transfer QTouch™ Sensor IC," Data Sheet QT160/161 1.06/1102, QUANTUM Research Group, Corporate Headquarters, 1Mitchell Point, Ensign Way, Hamble SO31 4RF, Great Britain, 2002. World Wide Web Site: http://www.qprox.com.

[25] "SCATIR Switch User Guide," Data Sheet, Tash Inc., 3512 Mayland Court, Richmond, VA 23233 USA, June 2005. World Wide Web Site: http://www.tashinc.com.

[26] R. Pallas-Areny, "Amplifiers and signal conditioners," in *The Measurement Instrumentation and Sensors Handbook*, CRC Press LLC, 2000.

[27] "LM340/LM78XX Series 3-Terminal Positive Regulators," Data Sheet, National Semiconductor, U.S.A., August 2005. World Wide Website: http://www.national.com.

[28] "LM79XX Series 3-Terminal Negative Regulators," Data Sheet, National Semiconductor, U.S.A., November 1994. World Wide Website: http://www.national.com.

[29] "CD4051B, CD4052B, CD4053B - CMOS Analog Multiplexers/Demultiplexers," Data Sheet, Texas Instruments, August 1998.

[30] P. Horowitz and W. Hill, *The Art of Electronics*. Trumpington Street, Cambridge: Cambridge University Press, second ed., 1997. pp 2-54.

[31] A. B. Williams and F. J. Taylor, *Electronic Filter Design Handbook*. McGraw-Hill Publishing Company, second ed., 1998. pp. 2-54.

[32] "NI 622x Specifications," Data Sheet, National Instruments, June 2005.

[33] E. Loewenstein, "Analogue-to-digital converters," in *The Measurement Instrumentation and Sensors Handbook*, CRC Press LLC, 2000.

[34] "NI Developer Zone," Feb 2006. http://zone.ni.com/devzone/cda/tut/p/id/4807. Last accessed 2007/05/23.

[35] R. B. Panerai and et al., "Spectrum analysis and correlation," in *The Measurement Instrumentation and Sensors Handbook*, CRC Press LLC, 2000.

[36] "CEI IEC 61010-2-040 International Standard - Safety Requirements for Electrical Equipment," safety standard, International Electrotechnical Commission, International Electrotechnical Commission, 3, rue de Varembé, PO Box 131, CH-1211 Geneva 20, Switzerland, 2004-2005. Worldwide Website: http://www.iec.ch.

[37] "Low voltage electrical equipment - directive 73/23/eec," safety standard, CE Marking, 19 February 1973. Worldwide Website: http://www.CE-marking.org.

[38] "Electromagnetic Compatibility - Directive 89/336/EEC," safety standard, CE Marking, 3 May 1989. Worldwide Website: http://www.CE-marking.org.

[39] J. P. Bentley, *Principles of Measurement Systems*. Prentice Hall, third ed., 1995.

[40] "About Photoelectric Sensors," tech. rep., GlobalSpec, 350 Jordan Rd, Troy, NY, 12180, USA, 1999-2007. http://proximity-sensors.globalspec.com/specsearch/suppliers?comp=30. Last accessed 2007-04-30.

[41] A. V. Oppenheim and R. W. Schafer, *Discrete Time Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1999 ed.

[42] P. A. Lynn and W. Fuerst, *Introductory Digital Signal Processing with Computer Applications*. Chichester: John Wiley & Sons, 1989.

[43] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[44] F. J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," in *IEEE*, pp. 66: 51–83, 1978.

[45] J. C. Candy and G. C. Temes, "Oversampling methods for A/D and D/A conversion," in *IEEE Transactions on Circuits and Systems*, June 1987. Beginning discussion on the effects of oversampling on in-bound noise.

[46] H. V. Vliet, *Software Engineering Principles and Practice*. John Wiley and Sons, LTD, second edition ed., August 2002.