UNIVERSITY OF
BIRMINGHAM

FAULT TOLERANT TRAIN DOOR CONTROL

By

PAWEL ZUBRZYCKI

A thesis submitted to

The University of Birmingham

For the Degree of

MASTERS IN RESEARCH

School of Electronic, Electrical
and Computer Engineering
The University of Birmingham
October 2010

Table of contents

List of Figures

List of Tables

Chapter 1. Introduction

1. 1. Abstract

A fault tolerant train door controller was developed based on cost effective embedded system. Based on the research of the faults occurring in a door control process it was concluded approach to fault detection and fault prevention from condition monitoring point of view. Using maintenance statistics the most common faults were isolated and their detection implemented within the designed controller. The controller functionality was described and supported by conducted experiments. Further analytical approach was shortly introduced and summarised work done showed that small embedded system can be a powerful fault diagnosis tool and condition monitoring system. Further improvements were proposed.

1. 2. Project background

Nowadays, detecting incipient faults plays a very important role in planning maintenance works. Avoiding long out of service periods for trains is essential in operating a reliable, punctual railway. These statements lead to the conclusion that the implementation of a health monitoring system is very desirable. Continuous system monitoring allows the detection and timely repair of incipient faults before they cause more serious faults. This is called condition monitoring and it makes maintenance easier as the system indicates where the potential fault in the monitored equipment / device is. The information provided by such a system is very useful and saves both the maintainer’s time spent on looking for a fault and total train out of service time. If a fault is detected, the condition monitoring systems are able to stop device normal functionality in order to prevent unsafe operation. For example, in the case of a train door malfunction, a controlling device prevents the train from leaving a station or just blocks the door to prevent it from opening if safety precautions are not met. As embedded systems have become very cheap, the cost of system health monitoring has reduced. Modern technologies can provide low cost monitoring with the use of systems based on microcontrollers. It is becoming standard that such status monitoring embedded systems are present in all new solutions. They are essential especially where human life can be endangered in the event of a fault. Existing railway equipment is continuously being upgraded to increase its reliability and safety features to comply with new regulations. With embedded system monitoring, maintenance can be planned more efficiently. This planning leads to increased productivity and money savings. The information gathered while the device is working in good or faulty condition is often used for further research and to look for improvements in operational devices or to make new designs even more reliable.

The subject of this research is to design and develop a fault tolerant controller for electric train door. The designed device should demonstrate fault detection and safety features. The system should operate with a London Underground train door rig located in the University of Birmingham Railway Research laboratory. The train door is controlled through a drive screw attached to a DC electric motor manufactured by Minnesota Electric Technology (model 3C-5203172C**)**. The controller that is proposed should be more cost effective version of the existing controller designed by the Canadian company, Vapor Rail (model Vapor TP95). The Vapor controller solution is intended to be universal and portable with features that make it easy to use in different working conditions.

The designed controller was tested and its functionality was documented in the further chapters of this thesis. During tests, data was collected for normal and faulty operating conditions. The gathered information has been used to create an analytical model of the controller and to calculate parameters for fault detection algorithms. Furthermore, software implementation of a controller to ensure smooth door operation through correction of the door position based on feedback signals in a closed loop control has been developed.

The University of Birmingham has a rich history with various case studies related to the railway industry regarding fault diagnosis research (i.e. Fararooy and others, 1997, 2002; Lehrasab 1999; Dassanayake, 2001; Roberts and others, 2002). Those studies resulted in forming fault diagnosis algorithms that could be implemented in microcontroller based solutions which significantly decrease the cost of diagnosis.

1. 3. Motivation

In the field of railway engineering, it is very important to ensure that devices have a very high level of reliability. Especially when passenger safety is a concern, the condition of a train door operation has to be monitored constantly and well maintained. Newly designed door controller solutions are much more reliable than those used in existing, older train rolling stock. Train operators are investing in upgrading existing train door control systems to comply with new safety rules. Over time existing trains will be modernised and equipped with systems for door operation condition monitoring and data logging to increase reliability. This modernisation is extremely important, as can be shown by an online example (Train door flew open, 2007) of a situation which happened a few years ago when a passenger train was travelling from one station to the next with the door open. The train was equipped with a logging system, but the system was not accurate. During the incident the train driver received information that something was wrong but did not investigate the problem and overrode the error message. There was not enough detailed information given and although the driver was not sure what error he was overriding, he proceeded with the override. Luckily nobody was injured on this occasion. This case shows how important it is to equip trains with advanced condition monitoring systems that will not allow situations such as this to happen. On the other hand, the passengers just ignored the open door and nobody informed the driver about the danger or stopped the train. There are also lots of other reported issues related to train door control. For example, as in online article (Door horror stories, 2010), the door is closing too fast and not allowing all of the passengers to get into the train, or trapping them in between the door panels. To avoid such hazardous situations a correct, working train door condition monitoring system is a must. It has to give sufficient and detailed information to help prevent a fault or regarding a malfunction which has occurred. From the point of view of passenger safety, the doors are the most important part of the train. A malfunctioning door should be detected as soon as possible to avoid any rail service delays or trains out of service. The control system should also be able to put a single door out of service if a malfunction is detected. The system should keep the door closed and allow the train to continue normal operation instead of putting the entire train out of service. It is very beneficial to implement a system to monitor the door opening / closing time and, if there are any deviations, to make a required correction. For example, to avoid trapping hazards in the case of too fast door operation, adjust the power supply for electrically controlled doors or decrease the air pressure for pneumatic doors. These control adjustments can be carried out by embedding software controller into the control process. The proposal of such a controller will be introduced in further chapters.

1. 4. Summary

This chapter introduces the fundamentals of this research. It shows backgrounds and motivation to design and development of a proposed fault tolerant train door controller. It also highlights the importance of the presence of a condition monitoring system if controlled process concerns passenger safety and how information gathered by such a system can be further used.

The second chapter includes a literature review providing terminology overview related to fault analysis. Definitions of analytical, physical redundancy, fault detection, fault diagnosis and tolerance have been introduced. Failure modes were described. System reliability and methods of fault analysis were shortly discussed. Differences between open and closed loop system control were addressed. Faults for pneumatic and electric train door control were researched and identified. Methods of detection were described. Based on acquired maintenance statistics the most common faults were isolated and faults detectable by the designed controller are listed.

Chapter three starts from a short review of existing door control solutions. The design objectives and requirements are précised. Based on this information the controller was designed and its components chosen. A short description of design process is followed by a detailed description of the controller functionality.

Next step was to test the designed controller electrically and develop fault detection algorithms. The software development stages are described in details in the Chapter 4. The detection algorithms of the most common faults, which were defined in Chapter 2 implementation was undertaken.

In Chapter 5 an acquired data analysis is documented showing the designed controller functionality. The controller has been tested in normal and faulty conditions simulations. The acquired results were analysed and commented.

An analytical approach to fault detection for the electrical train door is developed in Chapter 6. A motor model is derived and simulated for the train door normal operation. Simulation results are verified with the results analysis from the previous chapter. Some further experiments are conducted to acquire nominal motor parameters. Implementation of additional regulators is proposed improving process control.

The last chapter summarises research work done and experiment results. The designed controller limitations are described. Further software and hardware improvements are proposed.

Chapter 2. Concept of fault tolerance

2. 1. Introduction

Before attempting to design a fault tolerant train door control, concept of a fault tolerance has to be defined. Information has to be gathered that give answers to following questions: what is a fault, how to detect it, how to diagnose, how to interpret fed back information, and more to continue with further research stages. These questions are answered below. This chapter also includes description of failure modes and its effects. System reliability is also discussed. Based on the research, potential train door faults were identified. From acquired list the most common faults were isolated which are backed up by collected maintenance statistics. Methods of detection of these faults in a closed loop system control are preceded by a short description of advantages of such a system over an open loop system control. This chapter therefore provides definition of the fault by means of literature review.

2. 2. Terminology

A *fault* is an abnormal condition / state or defect of a system according to definition found in ISO/CD 10303-226 document. In this research the system states as all components used in a train door control. This abnormal condition may lead to a *failure*. The failure limits the system functionality to perform normal operation.

Faults are divided into five main groups based on their behaviour (Isermann et al., 2000). The faults can be abrupt, transient, intermittent, noise or incipient. In mechanical systems occurring faults usually have incipient manner. Electrical system faults are more difficult to analyse. The system that monitors device parameters and based on that information makes a decision whether fault occurred or not is a *fault diagnosis* system. This system analyses effects of the fault and gives information where the fault can be located. Therefore, first function that the fault diagnosis system performs is a *fault detection* and then *fault isolation*.

Fault tolerant system (FTS) is a system that is able to prevent anticipated faults and has implemented some tolerance for unanticipated faults as well (Netten, 1997). By prevention is meant that the system has implemented some methodologies, technologies in design process which allow avoiding the fault. The system has to be also robust for unanticipated faults. The fault tolerance can be realised by an implementation of hardware or software redundancy. Therefore, the FTS may be based on *analytical*, *physical redundancy* or both (Abzug et al., 2002).

To pursue analytical redundancy (AR) implementation, first a system model has to be developed. Methods of AR implementation are discussed further in this chapter. In general this approach is based on analysis of measured outputs and signal outputs generated from a mathematical model of the system (Osder, 1999; Frank, 1990). Result of a comparison of these signals is referred as *residuum*. The AR has two forms (Chow, Willsky, 1982) - direct redundancy which is relationship between instantaneous outputs of sensors, and temporal redundancy that defines relationship between histories of sensor outputs and actuators inputs.

To realise the physical redundancy additional system components are used that have the same functionality as already existing components. For example, system with fitted two sensors monitoring the same quantity compares signals from them and analyses response. The fault is detected if there is a difference between acquired signal values (synchronisation is a must between compared signals to avoid non existing fault detection). In some systems, with redundant components, the control process can be switched from malfunctioning component to a redundant one realising the same function. *Condition monitoring* system reports then a fault but the control process is still functional. Unfortunately, the physical redundancy has some disadvantages that lead to higher system costs and extra space required to fit additional components.

Implementing fault tolerance within the system increases its *reliability*. Therefore, the system is less likely to develop a fault. The Institute of Electrical and Electronics Engineers (IEEE 1332-1998) defines reliability as a system ability to perform its intended functions under stated conditions for a specified period of time. Because reliability is a probability the system still have chance to develop a failure. Overall system reliability is dependent on the reliability of all the system components. Therefore, to determine the system reliability a model has to be constructed based on data analysis of each system component time-to-failure operation (ReliaSoft, 2007). At this stage, relationships between components are considered and the least reliable ones can be replaced to improve reliability of the whole system.

For the train door control reliability is a very important asset. Higher reliability means that fewer trains are put out of service and gives higher quality service for passengers. Based on researched faults, described in further section, and maintenance statistics (WMATA, 2004) for the train door system it can be clearly seen that mechanical components develop more faults which become simply worn due to constant use. However, the main factor affecting the train door system reliability is operating environment. The environmental factor (i.e. excessive heat or cold, humidity, moisture) causes more than 60% of all occurring faults.

2. 3. Failure modes

Nowadays, demand for high quality, reliable products is increased. New developed products have more capability and functionality. That makes maintaining quality and reliability a very complex problem as these developed systems parameters are achieved through extensive testing and model simulations in late development stages. To analyse potential reliability and quality problems in early development stage a Failure Modes and Effects Analysis (FMEA) methodology comes very handy. The FMEA is a procedure (FMEA, 1940) employed by engineers to improve system reliability if consistently used throughout the design process. Using it is easier to take actions if hardware modifications are necessary. The FMEA procedure is being periodically updated to reflect changes in all product development stages. It is a tool that provides information about risks in design process and which one of them are the greatest concerns. Therefore, its proper use allows dealing with a problem before it arises.

The process of conducting FMEA, as describes Langford (1995) and Mikulak et al. (1996), has three main phases. These phases are preceded by a pre work, which gives detailed system description. This step simplifies further analysis as it provides good understanding of the system by helping engineer with a proper identification of system function use. The main phases are:

- Severity;
- Occurrence;
- Detection.

The first phase, severity, identifies *failure modes* based on functional requirements and describes their effects. A failure mode is a manner in which component, subsystem, system etc. failure is observed; it generally describes the way the failure occurs (Mikulak et al., 1996). To each failure mode a rank is given that defines whether the failure effect has a minor or critical impact to the system or a user. The highest ranked failures may also cause injury, therefore must be eliminated by design change or implementation of protection from their effects. Potential failure modes for the train door control are:

- Corrosion;
- Door friction change that gives current increase;
- Sensors or encoder failure;

and effects:

- Increased operational noise;
- Door misaligned;
- Door unable to move or abnormal door movement.

In the second phase, occurrence is defined how often failure modes occur. Each failure mode has assigned an occurrence ranking (numbers between 1 and 10 or in %), where higher rank means the failure occurs more often (Mikulak et al., 1996). In the last phase cause of each failure mode has to be identified. Inspection methods have to be chosen that prevent failure modes from occurring or detect before reach customer. To each failure mode then is given rank (Mikulak et al., 1996) that defines how likely it is for a failure to be identified or detected on time through planned tests or inspections and caused defects removed. If the assigned rank is high, it is likely for a failure to not be detected. Potential causes of failure modes within the train door control are:

- Excessive voltage;
- Improper door panel alignment;
- Improper operating conditions;

- Obstruction;
- Poor maintenance;
- Worn components.

At the end to each fault mode there is a risk priority number (RPN) calculated that is multiplication of ranks received from above three phases:

RPN=S*O*D

Where: S – Severity rating, O – occurrence rating, D – Detection rating.

Below is an example of FMEA worksheet:

Function	Failure mode	Effects	S	Cause	O	Current controls	D	RPN	Recommended actions	Responsibility & target completion date	Action taken
Door closing	Increased current	Unable to close the door	5	Debris / Obstruction	10	N/A or no sensor to check for obstruction	1	50	Perform cost analysis of adding additional sensor or torque monitoring	Pawel Z. 07/10/10	Added to next controller version

Table 2-1.Typical FMEA worksheet.

The failures with the highest RPN should be dealt first as indicated areas may need the design changes to be safe for use.

2. 4. Open loop door control systems

The most common door control method found in UK trains is based on pneumatic control. This type of door control was already in use in early 1920s. From then on pneumatic control was constantly modernised. Most trains are still equipped with pneumatic door control, although modern systems in use do not much resemble the old systems. They had to be adapted to improve maintenance and comply with present day safety standards.

The pneumatic door control is an open loop system. The control process does not supply feedback to the control system as in figure 2-1.

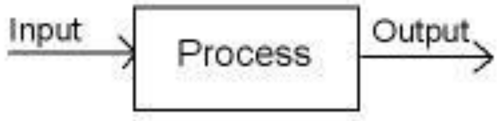


Figure 2-1. Open loop system.

Therefore, the feedback has to be supplied from lots of sensors fitted within the system. The door controller then reacts according to the situation which has occurred.

The advantages of this solution are its cost and simplicity. Disadvantages are that implementation of the fault detection or safety features requires use of additional components supplying feedback about current system state.

2. 5. Closed loop door control systems

The designed control system is a closed loop system, meaning that signals from the output are fed back to the input and used in a process of control as shown in figure 2-2.

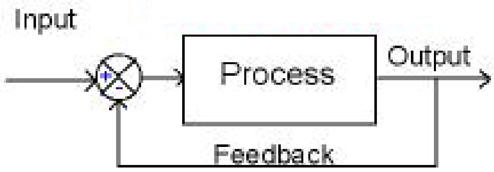


Figure 2-2. Closed loop system.

The signals that are fed back come from the motor optical encoder. The encoder monitors the motor shaft position. Therefore it supplies information to the system regarding the position of the door and, while in motion, velocity. The encoder feeds back three signals. Two signals are in quadrature and based on them the current door position can be calculated. The third signal allows measurement of the motor velocity. All those signals are fed to the microcontroller quadrature encoder inputs and used in the control process. To ensure the accuracy of the encoder outputs, all the signals are filtered against noise by internal microcontroller filters. The signals fed back from the optical encoder help to implement fault detection as the exact door position and the current door speed is known throughout the control process. The signals give information such as the door velocity increase or if the door is not fully open / closed that also makes easier to implement safety features.

2. 6. Common faults and methods of detection

Faults within the train door and the train door control were identified. The most common faults were focussed upon. The result of this research is presented in the next two subsections. The research was backed up by maintenance statistics regarding occurring faults acquired from London Underground [United Kingdom] (Peacock, 1990), New York City and Washington Metropolitan Area Transit Authority [United States] (WMATA, 2004). As the research is related to the electrically operated train door, the electric faults are described in more detail, although some of the data available concerned pneumatic operated train doors (UK statistics).

The first step was to identify which components of the train door may develop faults and to find out how to avoid potential failures by introducing proper fault detection and maintenance in a timely manner to make the control system more reliable. The causes of failures and failure modes were researched. This includes situations that may occur in real life, such as safety precautions, which are necessary if passengers interfere with the operation of the door (obstruction detection). Identified faults were divided into two groups – related to mechanical and electrical components. Later on potential failures, hazards or safety issues were simulated in the research lab to test the functionality and fault detection reaction of the designed controller.

Pneumatic train door

A typical pneumatic door actuation system is shown in the figure 2-3 below (Central Line Tube Stock Maintenance Manual, 1990). The door control mechanism is fitted at the bottom of the door panels. Its detailed operation description can be found, for example, in the thesis of Dassanayake (2000, p36). In the early stages, when the pneumatic systems were fitted into trains, no fault tolerance or fault detection was implemented. Slowly, and after some time, improvements were introduced. Later on, redundant solutions were implemented to increase system reliability and passenger safety. Improvements are still being made today. This includes upgrades necessary to bring old control systems up to date to comply with new standards, as nowadays most UK trains in use have a door control system based on pneumatic actuators fitted, which was manufactured a few decades ago.

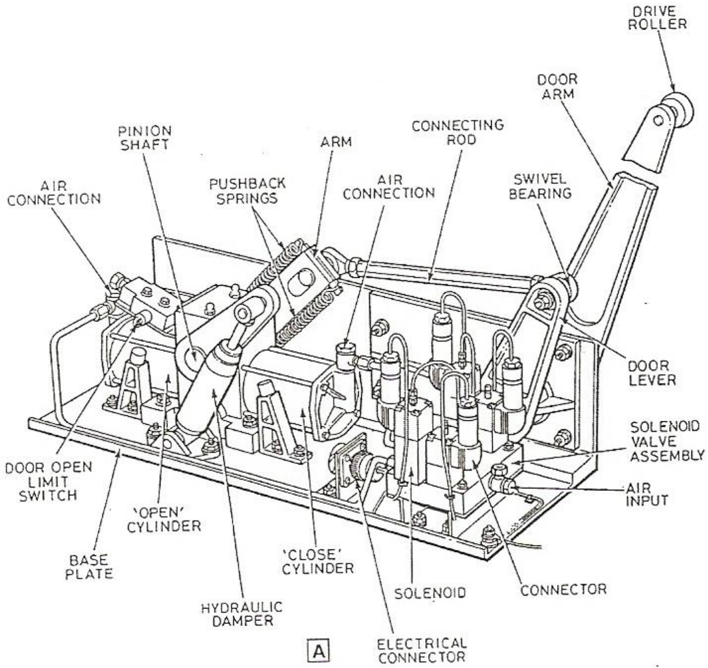


Figure 2-3. Pneumatic door actuator construction.

In general, pneumatic control is an open loop control system with no feedback. System monitoring can only be implemented by fitting additional components (i.e. air flow sensors, position, pressure sensors) to monitor the operation of the door actuators.

Faults in pneumatic door control are related, in most cases, to mechanical failures (Peacock, 1990). The components directly exposed to or interacting with passengers are found to fail more frequently. Moving parts of the door panels that exhibit wear while in constant use also fail more frequently.

The identified faults include:

- Decreased air pressure that can lead to a velocity loss which is detectable by the door opening / closing time measurement deviation. It could be detected by the use of two sensors where one indicates the door fully open and the other that the door is fully closed,
- Drive roller wear, swivel bearing wear,
- Spring wear,
- Sliding door track worn,
- Sensor malfunction,
- Door open /close buttons wear (passenger exposed component in constant use),
- Door panels misalignment i.e. caused by loose door arm roller guide assembly,
- Door closing or opening too fast / slow.

Electric train door

The faults occurring in electric train door control can be related either to the electrical or to the mechanical system. Research indicated that the mechanical systems develop more faults (WMATA, 2004). Typical mechanical faults are related to wear due to friction between the components in constant use, corrosion or accumulated dirt. Many faults can be avoided by preventative maintenance that involves maintenance scheduling and condition monitoring. For example in some devices there may be redundant or additional (such as sensors to monitor status of other components) components fitted, performance of the device is analysed against normal conditions model or visual inspections are pursued. Based on this information condition monitoring system detects potential faults and prevents monitored device serious failure.

Electrical faults are more difficult to predict and diagnose. To simplify diagnosis and potential repairs, in many cases, the electrical system is designed as a *modular system*. The modular system is based on the circuit boards mounted in racks, which makes servicing or upgrades simpler. If a fault occurs whole module is replaced by a new one.

The panel fixings of an electric operated door have very similar construction to these with pneumatic control. Therefore, the mechanical faults related to the door panels are assumed to be the same. The parts exposed to passengers are also the same and the faults related to them as well (normal wear or due to vandalism).

A schematic of an overhead electric door control system is shown in the figure 2-4 (Vapor Rail technical proposal TP95-15, 1995):

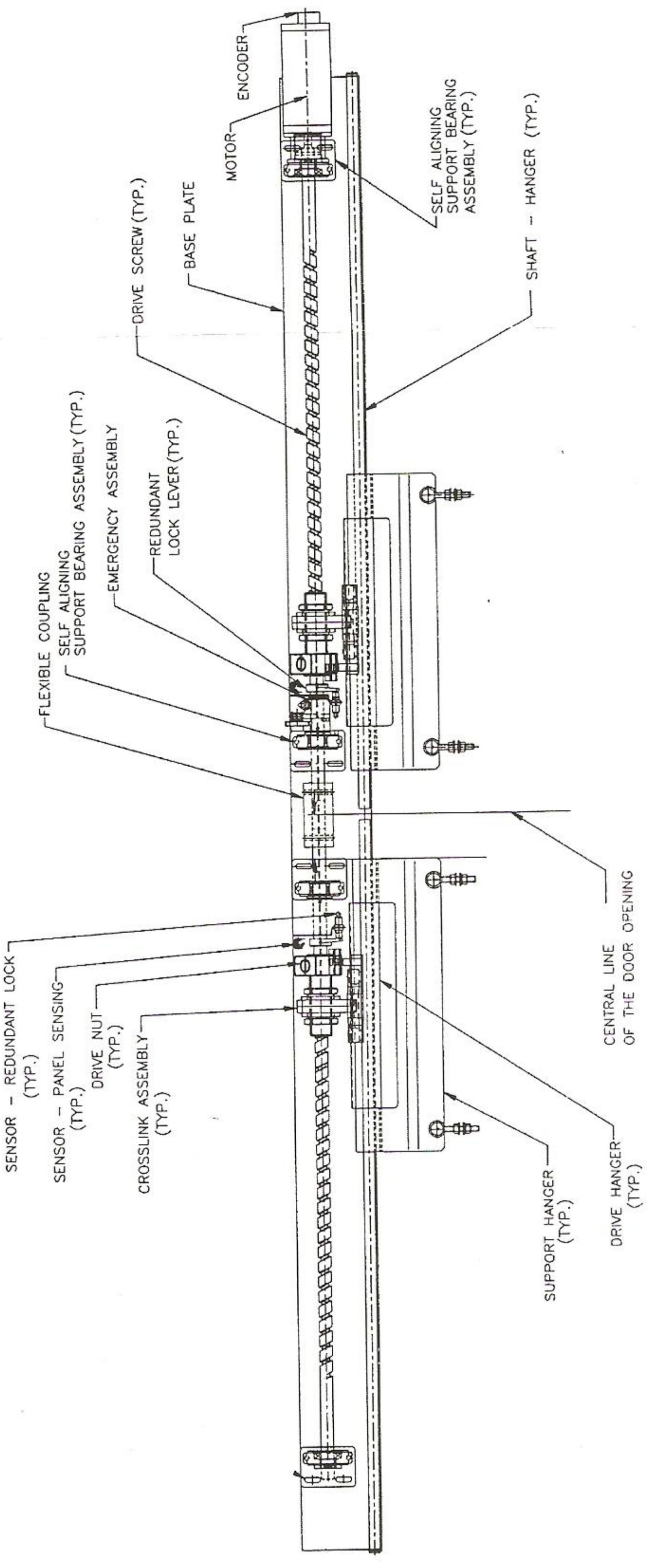


Figure 2-4. Door drive schematic.

Possible mechanical faults are enclosed in a table 2-1 below:

Failure mode	Fault effect	Method of detection
Roller faults	These faults could be related to wear or due to the collection of debris. Both cases cause friction forces to increase. This could result in the situation that the rollers are blocked or have difficulty moving smoothly across the door guide. In extreme situations the motor may be not able to move door panels resulting in the door being put out of service or it could increase the door opening / closing time.	The roller faults can be detected through current monitoring or measurement of the door movement time. The rollers wear can also be detected by a distance sensor fixed on chassis that monitors the door panel lateral movement. To prevent occurrence, the fault condition monitoring system could issue a warning to change rollers on time.
Linear shaft assembly	Friction increases, which cause the linear shaft to misalign and cause further difficulties in door movement.	This could be detected with the same methods as roller faults.
Push button fault	Due to constant use mechanical parts of the open / close buttons malfunction causing the button to not work when pressed.	Difficult to detect by the system but can be monitored by scheduled maintenance checks. It can be verified if a train operator is able to open / close the door.
Ball bearing fault	Bearings indicating wear emit more heat to the system and increase noise of the door movement.	This fault can be detected by monitoring the system temperature or current drawn change (increased friction).
Door panels misalignment	Door fixings become loose i.e. due to constant operation causing banging noise and lateral movement.	It can be detected by a distance sensor monitoring the door panel lateral movement or current drawn

		change (increased friction).
System overheat	Door erratic behaviour or the door controller shut downs.	System temperature monitoring sensor.
Motor brushes wear	This fault may cause arcing, motor temperature to increase or make the motor not functional.	The motor temperature can be monitored by a temperature sensor. Faulty motor can be detected from optical encoder readings if the door panels didn't change position. To avoid failure -monitor brushes wear on regular maintenance checks.

Table 2-2. Mechanical faults.

Possible electrical faults:

Failure mode	**Failure effects**	**Method of detection**
Sensor fault	Differences between sensor measured values and actual values due to sensor malfunction or increased noise in the system (i.e. interference or caused by dirt on infrared sensor).	The fault can be identified from other factors such as the motor optical encoder, which makes possible to verify the states of the sensor.
Wrong motor control signal or no signal	This fault causes the system erratic behaviour or no door movement (i.e. Pulse Width Modulation {PWM} signal error).	Internal microcontroller functions can be used. Procedures to ensure passenger safety should be implemented in case this fault occurs.
Optical encoder failure	The encoder gives wrong door velocity and position readings, which causes	Encoder readings can be verified by response from additional

	the door erratic behaviour. Measurement error can occur due to high sampling speed or noise.	sensors in the system.
Defective sound alert	No sound information for passengers when the door is about to close.	Difficult to detect by the system but can be monitored by scheduled maintenance checks.

Table 2-3. Electrical faults.

Further information regarding the approach and fault detection implementation methods can be found in the Chapter 3.

From above faults researched, the most critical ones within an electrical door control system were isolated backed up by the maintenance statistics (Peacock, 1990; WMATA, 2004). Methods for detecting them that will be implemented in the controller are:

- An increase in friction caused by roller malfunction, which is detected by a current sensing circuit from the feedback signal from the motor drive (higher current drawn from the driver circuit, which can cause longer opening / closing times than those set by the program) or decreased velocity readings from the optical encoder feedback.
- Detection of potential obstacles between the door while the closing operation is executed by monitoring the increase of friction that causes velocity decrease and higher current flow to the motor than in normal operations.
- Motor electronic drive functionality monitoring through a voltage sense circuit that detects any voltage changes supplied to the motor.
- Motor electronic drive overheating and overcurrent monitoring and protection. Overheating protection is already built into the H-Bridge and can be easily monitored through the output pin provided. Testing this situation in the lab would be very difficult as this should never occur.

- Optical encoder fault detection when the wrong motor velocity or position of the lead screw is read. It can be detected by installing some additional redundant sensors to verify the lead screw position from another source. The velocity can be monitored through door opening / closing time measurement with the use of a microcontroller timer. Readings from the encoder are also loaded with error that can be compensated by an integrated microcontroller noise filter or software proportional-integral-derivative (PID) controller.

Faults due to poor maintenance are not considered in the project in fault tolerant procedures as it is impossible to verify functionality due to the short period of tests of the controller in the lab.

2. 7. Methods of fault diagnosis and identification

Fault diagnosis (FD) techniques allow analysing effects and symptoms of a fault to properly identify failure mode, time and its location. The FD is also used to predict and control the performance of a system. Its techniques assure high performance over a wide operating range for nowadays control systems. The systems complexity increases constantly and the FD becomes one of the main problem solution technologies used to meet design requirements on system reliability.

There are many methods available for the fault diagnosis. The diagnostic procedure is based on the knowledge about the control process and observed and heuristic symptoms (Isermann, 2000). Modern widely used FD methods are model based and are using mathematical process and signal models. The FD method suitable for a control system is selected depending on the requirements. Then the method is implemented to simulated mathematical model of the control system.

Model based fault diagnosis started already in early 1970s. In 1990, Frank published a survey paper *Automatica* which classified model based FD methods into three groups:

- Observer based;
- Parity space and
- Parameter identification based methods.

More information regarding these FD methods and classification can be found in Patton et al. (1997), Isermann (2000) and Ding (2008).

Following online article (Intelligent Fault Diagnosis 1997) FD methods can also be classified into groups below:

- Time series analysis - use collected data to predict future events;
- Fuzzy logic methods - fuzzy diagnosis matrix, clustering, evaluation, segmentation and learning;
- Neural networks - system adaptation by self learning, black box identification;
- Multi-source & multi-sensor data fusion - combination of numeric, logic, linguistic information;
- Case based reasoning - heuristic method using experience and previously collected data;
- Probability reasoning - Bayesian networks (Koski et al., 2008);
- Hybrid methods - combination of above methods.

In the next subsection is a short description of a FD method that is suitable for implementation within developed controller software. This method belongs to the fuzzy logic methods group.

Takagi- Sugeno reasoning

The Takagi-Sugeno (TS) fuzzy controller (Brdys, 2009) uses fuzzy rules, which are linguistic If / Then statements involving fuzzy sets. It is suitable for non linear process control. The TS fuzzy rules can be formed based on information supplied describing the process. A typical rule has the following form:

$$If\ x\ is\ A_i\ \ then\ y_i = f(x_i), i = 1..K$$

Where A_i is a fuzzy set and y_i is an output of the i^{th} linear subsystem, x is input and K is subsystem count.

Dividing the non linear process into linear regions allows the use of linear subsystems to control the process. Therefore TS reasoning can be considered as a generalisation of a gain-scheduling concept (Tanaka et al., 1992; Korba et al., 1997). The reasoning can have use, for example, in motor current control where a few piecewise linear regions can be easily isolated. This example of fuzzy controller implementation is continued further in Chapter 5 where software development is described.

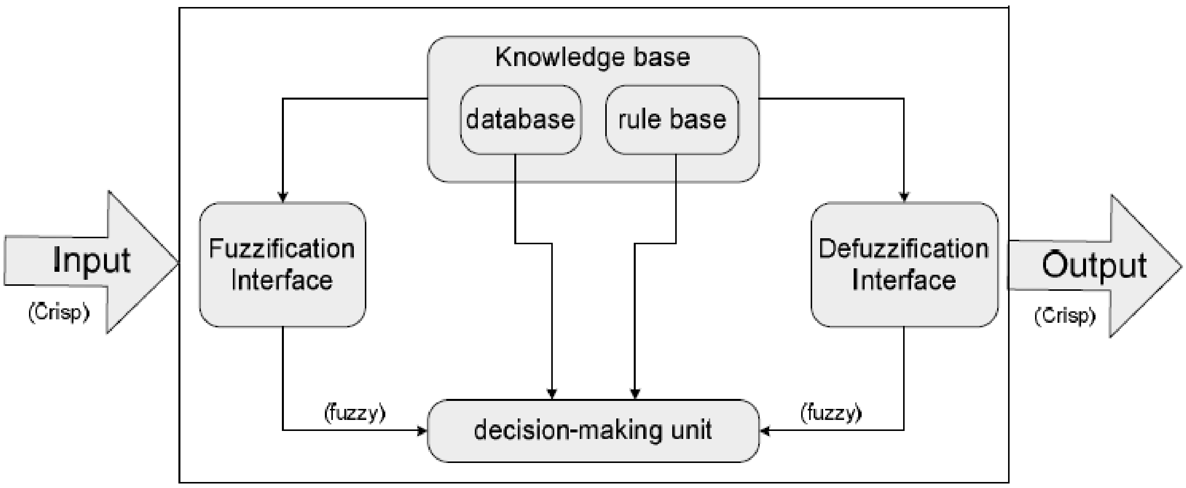


Figure 2-5. Fuzzy interface system.

In figure 2-5 a schematic of fuzzy system is shown. It is composed of four blocks. The knowledge base block is build of database and rule base block (it contains a number of fuzzy if / then rules). The fuzzification interface transforms crisp inputs into degrees to match with linguistic values. The decision making unit performs inference operations on the rules and defuzzification interface transforms the results of the inference into a crisp output (Mehran, 2008). In TS controller the decision making unit, based on the input value and the rules defined, chooses the right linear regulator to improve performance of the whole system assuring smooth change from one regulator to another.

2. 8. Summary

In this chapter the literature was reviewed. The fault tolerance was explained. Faults within the pneumatic and electric door control system were identified. How the faults may affect the system functionality was described and methods of their detection were discussed. The fault diagnosis methods were introduced that can be further implemented within a developed controller model in Chapter 6 to improve system performance. Information gathered in this chapter gives the fundamentals to proceed with a controller design and development of the fault tolerance described in further chapters. Based on the above information (fault detection methods approach) and motor parameters, hardware specifications were concluded and the design processed.

Chapter 3. Door controller design

3. 1. Introduction

This chapter is dedicated for controller design. At first there are enclosed short reviews of existing market solutions for a train door control. A Vapor Rail (VR) controller is described in detail as its functionality directly reflects to the designed controller. Project objectives and requirements are specified, led by research from the previous chapter regarding critical fault detection implementation. The components are being chosen to fulfil these requirements and design a fault tolerant controller. The controller design process is explained and its functionality introduced.

3. 2. Market solutions

Existing solutions

There are many door control solutions available currently on the market. Some of the solutions can work either with the older, widely used pneumatic train door control systems or with electric train door control systems (Vapor Rail, n.d.) and others designed specifically for one type of the door control (Train Door Solutions, n.d.). Many of them are offered as an upgrade for rolling stock to replace old control systems. Thanks to these upgrades old trains are being refurbished and equipped with new features and can still be used for transporting passengers. Door control solutions are manufactured by companies, for example, Train Door Solutions (TDS) (Train Door Solutions, n.d.), or at an international level, Siemens (Siemens, n.d.), Faiveley Transport (Faiveley Transport, n.d) or Vapor Rail (part of Wabtec Corporation).

The first company, TDS, deals mainly with pneumatic door control. It specialises in maintenance and technical improvements of existing door systems. TDS offers upgrades of the old air systems currently fitted in trains in use to new, more reliable components and door control systems adding safety and condition monitoring features to old rolling stock (Class 158/170 fleet and others).

The second company, Siemens, is a German manufacturer of various parts for trains as well as whole trains and locomotives. Siemens manufacture advanced control and monitoring systems based on the Simatic PCS7 (Siemens, n.d.) process control systems used to control various aspects of a train.

Faiveley Transport (FT) is a world leading manufacturer of various railway systems. It specialises in door systems technology for over 70 years (Faiveley Transport, n.d.). The newest designs are based on electrical systems. The FT door solutions claim to be world leaders on the market to conform to the highest safety standards and innovations in train access for passengers with reduced mobility. This company introduced first electric door control with electronic door controller (year 1979) and used an optical encoder for door positioning (year 1990) (Faiveley Transport, n.d.).

The other widely known company also with long railway history is the Canadian company, Vapor Rail. This company provides a wide range of products for door control. These products include air and electric operated systems for buses and trains. The controller is widely used for either pneumatic or electric train door control. It is a very sophisticated device.

This controller was designed to be flexible and to be able to fit in various train models. Vapor Rail (VR) produces, as mentioned earlier, the door controller 'The Third Generation Electric door operator' that originally operates the door in the research lab. The controller that is subject to this research is a simpler and more cost effective version of the VR controller.

The VR controller solution is intended to be universal and portable with features that make it easy to use in different working conditions, such as a wide voltage supply range, different current requirements, compatibility with different lead screw drives and various communication protocols to work as a part of distributed system. Due to its wide adaptability, the VR controller is more complex and therefore more expensive than many other controllers. In spite of its high price it gives increased functionality, including compatibility with older existing train door control systems and it could also be used in future solutions where it can be adapted with very little development time. The VR controller functions are divided into three modules (Calamatas, 1997). The first part is a superior module (PC104 board with Intel CPU) that controls the two other modules and has implemented data logging functions. The second part is responsible for door control and obstruction detection. That part is based on a separate digital signal processor (DSP LM629, n.d.) and is able to control the door even in the case of failure of the superior circuit (it is able to finish the currently executed operation). The third part is responsible for monitoring and I/O functions. A new door lock design that prevents the door from opening in case of controller failure is also implemented.

Innovations and future plans

Demand for highly reliable systems increases. The reliability of old systems used in rolling stock needs improvement. Existing and new door system solutions must conform to new safety standards and allow train access for passengers with reduced mobility. Innovations are also driven by other factors such as scheduling proper maintenance leading to

Condition-Based Maintenance (CBM) that allows cutting costs and improving system reliability. Such equipment condition monitoring analysis determines whether maintenance is necessary. The railway industry is searching for solutions to minimise the equipment located on train tracks near train stations also to prevent it from being vandalised. This includes part that is responsible for triggering the door opening /closing mechanism when the train stops at the station, or for keeping its arrival / departure time and position monitoring. For example, a few years ago there was an experiment in England on Southern trains (Satellite fault, 2004); a system controlling the train door movement through Global Positioning System (GPS) was tested. At the early stages there were some issues with the system as it did not always let the doors open at the stations, leaving passengers stuck inside, but those issues were later overcome. Now the system is operational and has been implemented on other trains. Other innovations include improving door service by preventing loss of interior train heat (Train Heat Issues, n.d.) while the train waits at stations for departure. Each door is equipped with an open / close button that allows passengers to open the door they want to use instead of a train driver opening all of them, which helps to retain interior heat. Existing door control systems are also being upgraded. The control systems are being equipped with additional safety and monitoring features, in many cases redundant, to increase overall system reliability.

This year (September 2010), a fleet of trains, the new class 380 trains were introduced in Scotland (ScotRail, 2010). These trains are fully compliant with Technical Specification for Interoperability for Persons with Reduced Mobility (PRM-TSI) regulations in respect of the door access (PRM-TSI, 2008). The improvements include easier access for mobility impaired passengers and also benefits for the loading and unloading of prams and buggies.

Figure 3-1. New ScotRail class 380 trains.

Some of the existing systems are being replaced by new technologies. According to ScotsRail online news (Scotsrail, 2010) the train door will feature “a distributed control system to replace centralised electrical systems thus improving reliability through the removal of electro-mechanical control gear, and providing diverse routing in control architecture”. Communication between the nodes of the distributed system will be based on a databus bus, which allows the transfer of different data types between railway carriages through a single connector (Databus, 2010). In addition, the Train Management System (Tendulkar, 2006) will be providing information on system performance to the maintainer to avoid any serious faults and to schedule proper maintenance. The new class of trains will also have a Selective Door Opening (SDO, n.d.) system that allows use of the trains at stations with shorter platforms and therefore providing safe control by preventing release of the door on carriages not at a platform. The train will be additionally equipped with more sensors to improve obstruction detection compared to previous train classes.

3. 3. Project objectives

The project development was divided into three main stages. It was realised according to the following steps:

- Design a controller for a lead screw driven train door;
- Develop a fault detection approach;
- Implement fault tolerant procedures within the control algorithm.

For stage one, the controller electrical requirements were elicited based on research and manufacturer information (see next subsection 3.4). After this step, components were chosen and then the controller designed and assembled. The main source of the information for this step was the specification of the VR controller, which is able to operate the same train door as that in the research laboratory. It should be noted that the VR controller has a much broader general specifications due to its adaptability. It was not designed for a specified environment. The designed controller is not as universal as the VR controller. The controller specification was tailored to use it with the motor drive lead screw located in the research laboratory. The controller is able to work with a supply voltage of up to 55 V (In this class of trains the maximum power supply for equipment is 52 V DC. In newer constructions the power supply is increased to 110 V). Therefore, it is able to supply maximum allowed voltage for the motor if necessary (52V DC). The designed controller functionality, reliability and safety features are modelled based on VR controller features.

In the second stage, research was undertaken into ways of realising critical fault detection, defined in Chapter 2, with the use of embedded systems supplying required feedback signals. The fault detection methods to be implemented were selected.

In stage 3, the fault tolerant procedures were implemented into the controller algorithm. The procedures ensure timely fault detection and prevention of more serious faults. Fault tolerance includes additional infrared sensors that supply a redundant signal to increase controller reliability and fault detection, current and voltage sense signal change monitoring and a feedback signal from optical encoder. Additionally H-Bridge temperature monitoring can be added with use of bridge thermal sense output.

3. 4. Project requirements

The sources of information defining the controller power requirements to sustain the motor demands were Dr H P B Dassanayake's thesis (Dassanayake, 2001) and Philip J. Calamatas' design review (Calamatas, 1997). As already mentioned, the Vapor Rail (VR) controller was designed to support a wide range of products, including earlier generations of electronic door controls (i.e. pneumatic or relay based systems). Therefore its specifications are much higher than the designed controller. The main design purpose for the VR controller was to ensure very high reliability and flexibility with a cost effective system. Due to its flexibility and compatibility with previous door controls, the cost has increased, however, as a product supporting a very wide range of door controls; its price is very competitive. A new approach to the "door closing and locking sequence" (Calamatas, 1997) was introduced with the VR controller. This approach allowed switching relays, which were used as a protection shell in earlier door control constructions, to be removed. The designed controller should match the level of reliability of the VR controller and be more cost effective in relation to single implementation.

Based on the information acquired, a general specification for the train door control is:

- To work with a single power supply of up to 52 V DC, equal to motor nominal power supply;
- To be able to draw current up to 6 A peak and sustain 3 A continuous current in the operational area;
- The door opening time should be 3.5 sec or less and should be user programmable through software changes (PWM pulse width control) or supply voltage change;
- The door closing time should be between 3.5 and 4.0 sec and should be user programmable (PWM pulse width control) or supply voltage change;
- To show the controller status (door position, drawn current etc.) and currently executed operation on an LCD display;
- The speed of door motion should not exceed 1.5 m/sec;
- Acoustic noise from the sliding door should be decreased to a minimum (especially when the door is fully open or fully closed);
- Quick obstruction detection that, when detected, releases the door by a minimum of 50 mm;
- Based on a simple, small and low cost embedded system.

Together with this information and knowledge from the previous chapter regarding the most critical fault detection methods (to determine if fault detection methods implementation require any additional components), the design process could be moved on to the next step. The hardware specifications were concluded and the controller components selected.

The controller has a modular design and is made of two modules as shown on figure 3-2. The main component of the first module is a microcontroller. It is responsible for fault tolerant door control. At the heart of the second module is a full H-Bridge that is responsible for delivering the required power to the motor drive. It is very important for the project to monitor the voltage and current on the motor windings and feedback signals coming from the motor optical encoder. Monitoring of these signals is required to assure the stability of a closed loop system through controller operation and implemented fault detection. Mentioned values help detect malfunction in the operation of the motor drive or controller itself. From those signals increased friction (obstruction detection or failing mechanical parts) can be detected, velocity changes or the exact position of the drive shaft monitored.

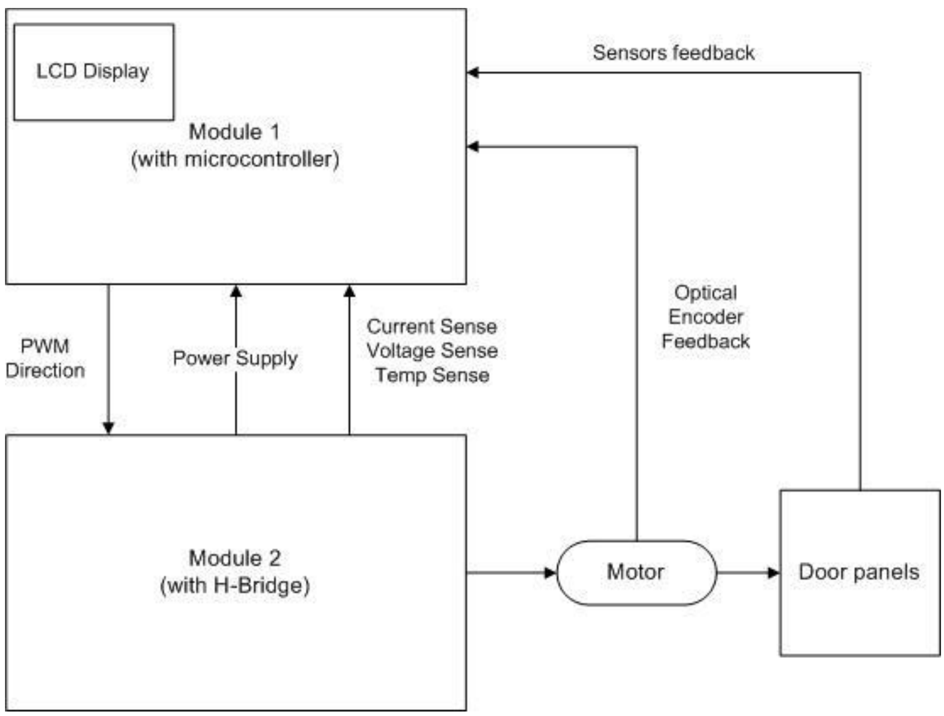


Figure 3-2. Controller block diagram.

There are two additional infrared sensors used. These sensors provide redundant information, which can be obtained from other sources. This information is used for the control process status verification purposes. The information is significant to increase controller reliability and is useful to detect the door misalignment and other component failures, particularly when working with an optical encoder where there are errors in transmission or a noisy environment.

3. 5. Controller design

Choice of components

The components for the controller were selected based on the desired system specification and to assure critical fault detection. The parameters of the motor had to be taken into consideration when choosing the H-Bridge as these parameters give specific power demands for its circuit. To work with the motor the LMD18200T from National Semiconductor was chosen, which fulfils these requirements. The control circuit is driven by a Microchip PIC18F4431 microcontroller. The fault detection is software implemented and based on monitoring of signals coming from infrared sensors, a motor optical encoder, current and voltage sense outputs and optional H-Bridge thermal sense output.

Below is the list of the all main components used:

- PIC18F4431 micro-controller,
- LMD18200 full H-bridge,
- LM2576HVT voltage regulator,
- HXS 10-NP current transducer,
- LM258N operational amplifier,
- PC1602LRU LCD display,
- Sharp G20 series infrared sensors.

A full list of components is enclosed in Appendix C.

The choice of microcontroller was led by the requirements of the process control. Therefore, the selected one has to be quick enough to detect an obstruction within a few milliseconds, to be able to sample monitored signals required to realise desired fault tolerance and be cost effective. An 8-bit microcontroller fulfils these requirements preferably with built in quadrature encoder (to work with the motor optical encoder). The microcontroller also has to be able to generate a PWM signal with frequencies assuring smooth motor operation (up to 10kHz). The PWM signal frequency then puts requirement on a sample rate of an analogue to digital converter (ADC) where H-Bridge voltage and current sense outputs have to be sampled with twice higher frequency. Taking these requirements under consideration a Microchip 8-bit devices family was selected.

Further, to include the quadrature encoder (QEI) a PIC18F2X31 or PIC18F4X31 devices family was chosen (QEI is used to monitor motor shaft movement through the motor optical encoder. Therefore, the door panel position is known at all times). There was also a requirement to have a minimum of four analogue to digital converter (ADC) inputs able to acquire data independently and continuously to monitor fed back signals. As QEI shares pins with ADC, making it difficult to monitor all required signals at the same time, the PIC18F2X31 devices family is eliminated as only 2 ADC inputs out of 5 are accessible. These steps lead to the last family of microcontrollers, PIC18F4X31 devices.

The chosen device family has 9 ADC inputs (also 3 excluded for QEI). As the microcontroller is only able to sample up to four analogue signals at the same time, this puts limits on all additional sensors used. The sensors have to give a digital output to ensure their states are monitored continuously.

The microcontroller also has to be fast enough to sample and analyse all the required signals. It has to react as quickly as possible in case of a fault or obstruction detection to assure passenger safety and stop the door movement immediately. However, this requirement was not of high importance in the laboratory environment as the chosen microcontroller does not work at its maximum frequency.

It should be noted that in this project the choice of an H-Bridge was not optimal, as it was based on insufficient information. In spite of working with the motor in some specific conditions (long time exposure to high friction forces i.e. obstruction may damage it) it is not able to supply enough power. However, it does not affect the work of the controller in normal conditions, as software precautions have been implemented to limit current in such situations. The H-Bridge is able to supply up to 3 A continuous current and can sustain up to 6 A peak current. The average current that the motor draws is approximately 2.2 A with door panels attached. As the motor starts and overcomes static friction forces, the current drawn from the source can reach up to 6A at supply voltage level required to acquire door movement speed defined in system specifications. The higher supply voltage may increase the current drawn up to 10A peak. The H-Bridge additionally benefits from embedded current sense output, thermal flag output signalling overheating and emergency brake input.

The LCD is used as a communication device between the controller and a user. It displays the status of the controller and in the design and testing process was used for debugging purposes. It indicates the current task executed and possible errors occurred. Current measurement and the door position are also displayed.

The voltage regulator makes available additional voltage levels required to power up all low voltage components. It converts voltage supply to the +5 V level required by the microcontroller, optical encoder and sensors. The transducer is fitted as an additional device to monitor the motor current to the LMD18200 built-in current sense output. It is a redundant device to assure reliable current measurement and monitoring.

All the high voltage components were selected to be able to work at up to a +60 V DC to allow the motor to operate with full power if necessary and ensure compatibility with the original train power line (+52 V DC).

The Design

The design process was divided into two parts as the controller is built of two separate modules. The first module includes a microcontroller and LCD display. There are additional power connection points to connect the power supply to the optical encoder and sensors and also inputs for feedback signals. This module is made on a universal board. Below are drawings of the designed controller:

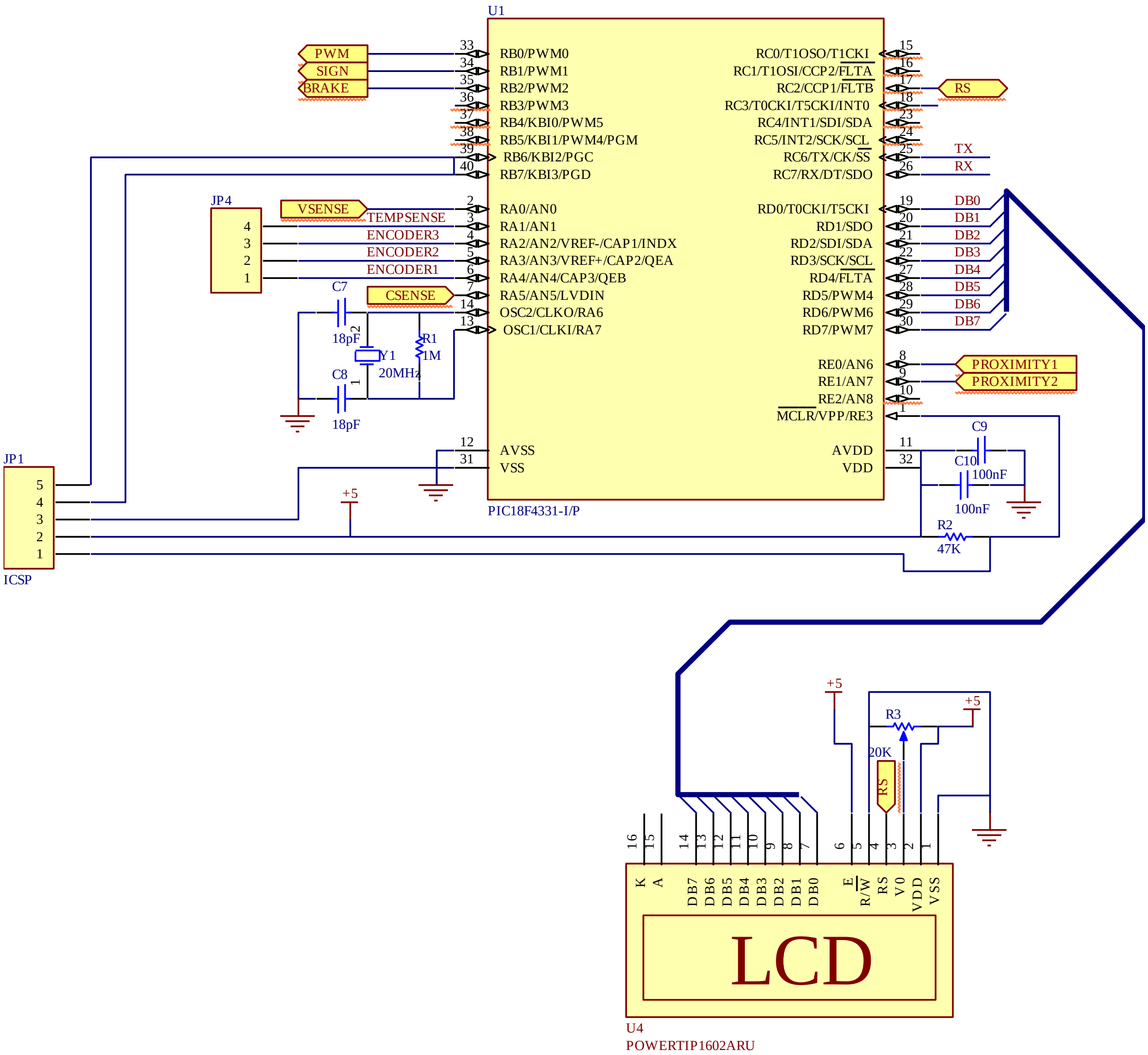


Figure 3-3. Microcontroller module schematic.

A circuit of the second module including the H-bridge was designed in Protel software (schematic and a PCB design). Later on, the PCB was made. The board additionally includes a transducer and components used to convert the main power supply voltage into a voltage suitable (TTL) for the module with a microcontroller. The module also includes an operational amplifier used to ensure the right voltage levels of the voltage sense output signals to apply them safely to the ADC inputs of the microcontroller.

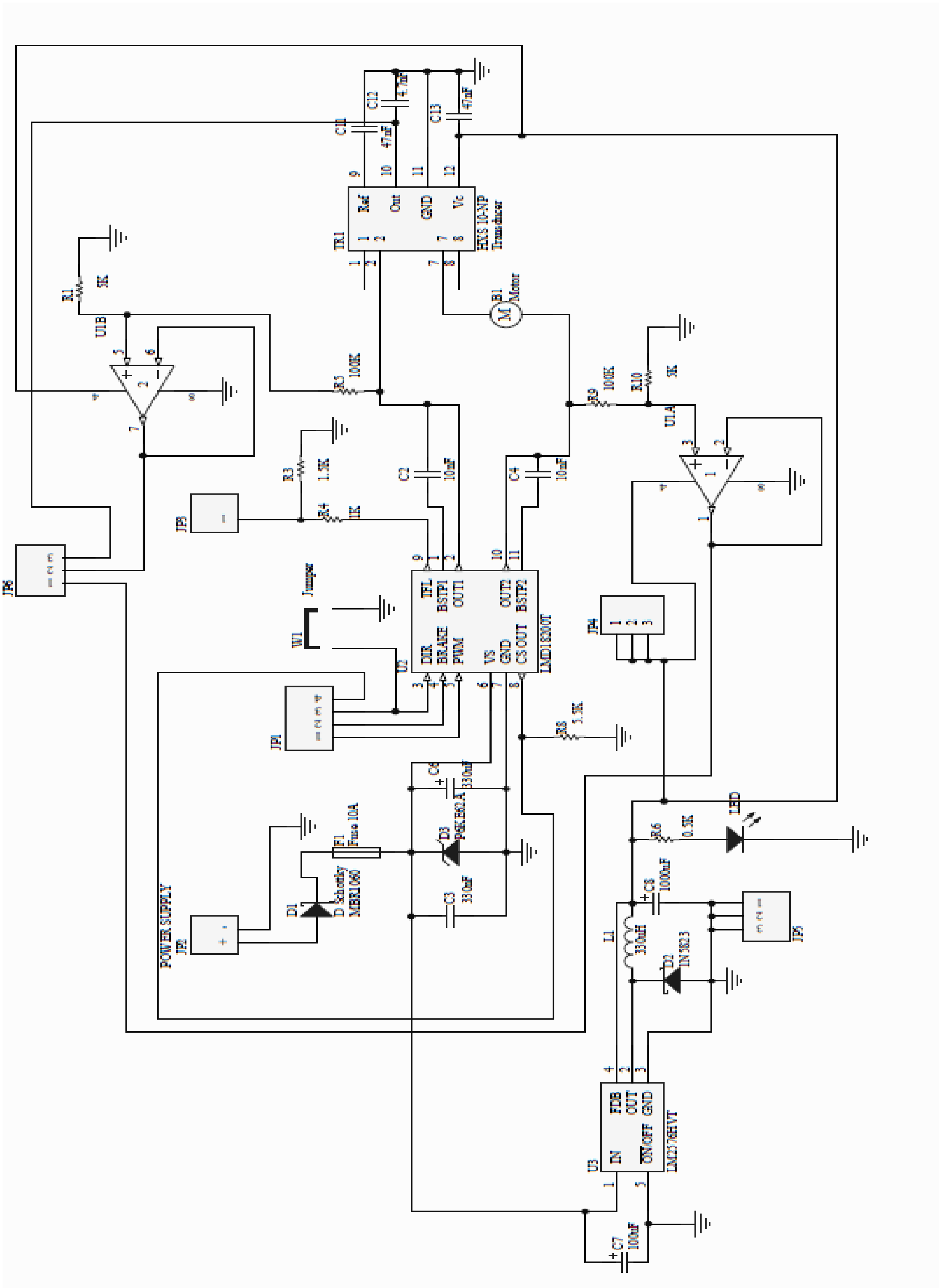


Figure 3-4. H-Bridge module schematic.

In the schematic of the second module three parts of the circuit can be isolated. The first is the H-Bridge with a protective circuit for high voltage. The second is the circuit converting the higher voltage to a +5 V voltage. A LM2576HVT converter and inductance are used for DC-DC conversion. This converter uses the PWM conversion [52 kHz] method and its output can be used as an external PWM generator for emergency door closing / opening in case of the microcontroller failure, to avoid opening the door with maximum motor speed equal to the power supply. In the third part there are circuits responsible for conversion of the measured quantities to the signals applicable for ADC inputs of the microcontroller. The resistors in this part were carefully selected to achieve the full scale of measured values for the ADC inputs. There are two sensors used. One indicates that the door is fully open and the other that it is fully closed. Both sensors are fitted on the door frame. The sensors are directly connected to microcontroller ADC inputs. They were specifically chosen to work with a 5 V range within the analogue to digital converter operating voltage range. The optical encoder and sensors are powered up from the microcontroller module. The encoder is connected to the module (microcontroller QEI inputs) with the use of shielded cable to minimise signal noise. Both modules are equipped with connectors for feedback signals, simplifying their transfer from one board to another. There are also some additional low power supply connectors. The low voltage module also has an In Circuit Serial Programming connector (ICSP), used to program the microcontroller without removing it from the circuit.

Controller functionality

The controller can be supplied from the power supply up to +60 V DC. In the lab all the tests were made with a +32 V DC supply that is sufficient to obtain the required door speed. All the software settings then refer to that voltage level.

The motor control is based on a pulse width modulation (PWM) signal. The control signal is generated by the microcontroller. The pulse width is dynamically changed throughout the control process.

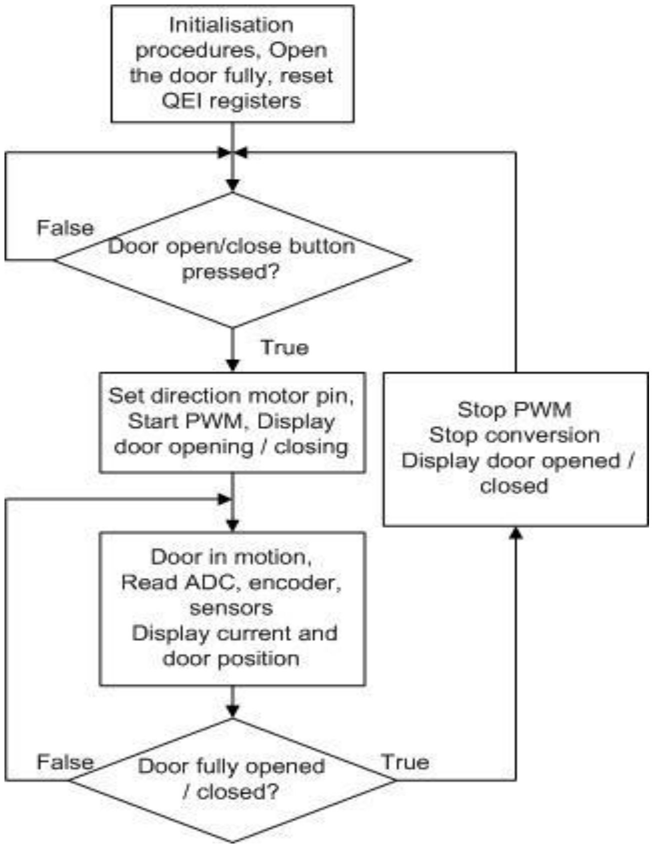


Figure 3-5. Controller functional algorithm (no fault detection shown).

Through the pulse width modulation, when the motor starts / stops, the current limiting function is implemented (more details in the next chapter). The pulse width is changed from zero to approximately 87.5% where the motor works with the maximum desired speed assuring the required door opening time of approximately 3 seconds. The closing time is a bit longer and lasts 3.6 seconds with a pulse width of 75%. The PWM signal is generated by the PWM0 output and its frequency is set to 9.6 kHz. The frequency can be increased or decreased in software code (however, to ensure that the motor has a fairly smooth operation the frequency has to be set above 2 kHz). The door opens / closes when the push-button located next to the microcontroller is pressed. The LMD18200T H-Bridge needs only two control signals to be supplied to operate properly. The first one is the PWM signal. The second is a direction signal, which is responsible for setting the motor motion direction (one I/O pin is used for this purpose). The current door position is acquired from the motor optical encoder readings. Detailed description of the encoder operation can be found Dr HPB Dassanayake thesis (Dassanayake p.50, 2001). The readings are filtered through internal microcontroller filters to reduce overshoot. The controller uses encoder signals to always set itself to the same start conditions, which is the door fully open. When the power is on, the controller opens the door fully if the door is in any other position and then it becomes ready to use. The motor stops the door movement when the infrared sensor fitted on the outer side of the door frame sends the information to the microcontroller or when a large overcurrent is detected (signifying that the door has reached the desired position). When this happens, the optical encoder position readings are cleared. This operation is done to ensure the same starting position for the controller in the case of power loss while in normal operation and to ensure that the door position is always interpreted correctly. Once the controller program starts, the data from the encoder is sampled continuously. It is very important not to omit any measurements related to the motor shaft position as it may cause an unpredicted door movement. When the door is in motion, the registers relating to the QEI are updated accordingly, independent from other functions realised by the microcontroller.

The other monitored signals (current and voltage sense) are sampled by the microcontroller ADC. The microcontroller is working with a 20 MHz external oscillator and the converter works with nearly maximum conversion speed. It is able to collect approximately 100 k samples per second with conversion result every 10 µs. For the purpose of the program, the first conversion is triggered by an internal microcontroller timer count. Results are read every ~50 µs (with no LCD support as it delays reading conversion results). This is the minimum time for a fault to be detected (it can be decreased by increasing oscillator clock). Any necessary corrections, such as keeping constant speed (in a specific range only due to power supply limitations), or eliminating position errors in case of a minor fault, are realised through the software controller implemented in the microcontroller algorithm program.

The analogue signals are connected to the ADC inputs 0,1,6 and 7, respectively current sense, infrared sensor output, voltage sense (door opening) and voltage sense (door closing). The second infrared sensor (giving information that the door is fully closed) has digital output and is connected straight to the I/O pin of the microcontroller. For the purpose of the LCD display, port C of the microcontroller is used. The LCD display is programmed to work in 4-bit transmission mode.

3. 6. Summary

After short review of the market solutions for the train door control the controller requirements and design process was presented. The controller was assembled and its functionality briefly described. This chapter explained design process up to software development stage. Therefore, the next chapter continues the controller development process by describing how the fault tolerance is realised within implemented algorithm in details.

Chapter 4. Methods for fault tolerant door control

4. 1. Introduction

In this chapter methods of fault detection implementation and each software development stage are described in detail. Functional algorithms that lead to detection of critical faults are explained and illustrated. Fault tolerance is being implemented. The software was developed with the use of Microchip MPLAB suite. The microcontroller was programmed through an ICSP connector fitted on the board with the use of Microchip PICKit2 USB programmer.

4. 2. Development stages

At the first stage of the software development process, the procedures responsible for controlling train door operations (opening / closing) were implemented. This was done to test whether the controller is working correctly. At first, the door movement distance was defined by a timer count. In the later stages, when communication with the motor optical encoder was successfully made, the exact door position was read from it. After this step, when the controller could perform door movement operations, procedures assuring smoother door movement and fault tolerant procedures were implemented. Throughout the controller testing process, data was set for the purpose of program fault analysis. All of the fault analysis implemented is based on analysis of the signals fed back from the H-Bridge module, the H-Bridge itself, sensors and the motor optical encoder.

During the early stages, a LCD display was used for debugging. Data received from the ADC, the optical encoder and the results of executed functions were presented on the display. At the final stage, the display was used to present the controller status (or errors occurred) and the current value in and door position, both in hexadecimal formats (presented current value is not pre scaled and does not show real current value).

The controller performs the door opening / closing operation. Depending on the state of the two switches connected to the microcontroller, the program executes the procedure of train door movement. These switches simulate the operation of opening / closing door buttons which are usually fitted in trains accessible for passengers. After the controller is powered up and the program starts, the door is fully open as shown in figure 3-5. This is done to assure that the starting conditions for the program algorithm are the same each time. Then the program waits for the user buttons to be pressed. In the software there is one function, which is responsible for the door movement in both directions. The status of the direction pin distinguishes if the door is being opened or closed. The program algorithm stays in the main loop from now on and waits for further tasks. Once the button is pressed, the microcontroller sets the direction pin respectively and the door movement is started. When the door movement is finished, the program algorithm goes back to the main loop and awaits the next task.

4. 3. Fault tolerance implementation

Fault tolerance is a complex problem. It is very important as it concerns safety and affects equipment functionality (fault prevention or detection). In the lab environment not all faults can be simulated as there is only a part of a train carriage available. Faults which appear very rarely, related to train driver control or communication with other train parts, were not considered. If a fault is detected, the controller gives information about it on the LCD display and continues in operation if possible. In some cases, if possible, the controller tries to minimise the influence of the fault, or changed circumstances, on its operation through implemented software correction algorithms.

Some of the most common faults were simulated in the laboratory environment. The list below shows what types of faults are detectable by the software. This is followed by a description of how the fault was dealt with in each case.

Detectable faults:

a) Overcurrent monitoring and obstruction detection;
b) Supply voltage changes;
c) Pushbutton malfunction detection (if pressing the button does not start any operation);
d) Increased friction detection while the door is in normal operation detected through a longer opening / closing time and slightly higher motor current requirements (this could indicate e.g. worn rollers, motor brushes);
e) Door not fully closed or not fully open, or door panels misaligned (infrared sensors);
f) Door velocity changes;
g) Motor optical encoder malfunction;
h) PWM signal fault detection.

Additionally there are implemented functions minimising current overshoot while the motor starts and stops that are also responsible for the door movement noise reduction.

Overcurrent monitoring and obstruction detection

The current monitoring signal is supplied either from the H-Bridge current sensor or the transducer output (the transducer current sensor output needs to be amplified, as in the operational range of the motor it only gives output values with a 0.2 V threshold). The current value is monitored throughout the door control process. Its hexadecimal value is read after conversion has finished from the ADC register. There are eight previously stored current values. The present conversion result is added to seven out of eight stored conversion results and then divided by eight (the highest current value is eliminated from the addition as single value is interpreted as noise and rejected but it is stored for future analysis. The next high current value conversion result confirms whether overcurrent has occurred and triggers fault detection). The acquired average result is compared with the values set in the algorithm. There are two thresholds. The first value corresponds to approximately 2.4 A. If the average result is below this level, normal operation is continued. If the result is in between the first and second (2.7 A) threshold value then a potential fault has been detected that causes an increased frictional force (i.e. rollers wear). A warning is reported on the LCD display. The controller remains fully functional. If the average value falls above the second threshold, then a potential obstruction or serious fault has been detected. In this case, the door will be reopened for a short distance (approximately 100 mm, defined in the program). The controller will perform further three attempts to close the door. If all attempts are unsuccessful, the controller will stop operation and report a fault on the LCD. The door is then out of service. Above method detects increased current value and stops the door in case of high increase is detected based on defined thresholds comparing to the standard operating value. This approach is not a very accurate form of obstruction detection while the motor accelerates and draws more current. Therefore for this purpose the current analysis starts when the motor reaches constant speed. Functional algorithm is shown in figure 4-1 on the next page.

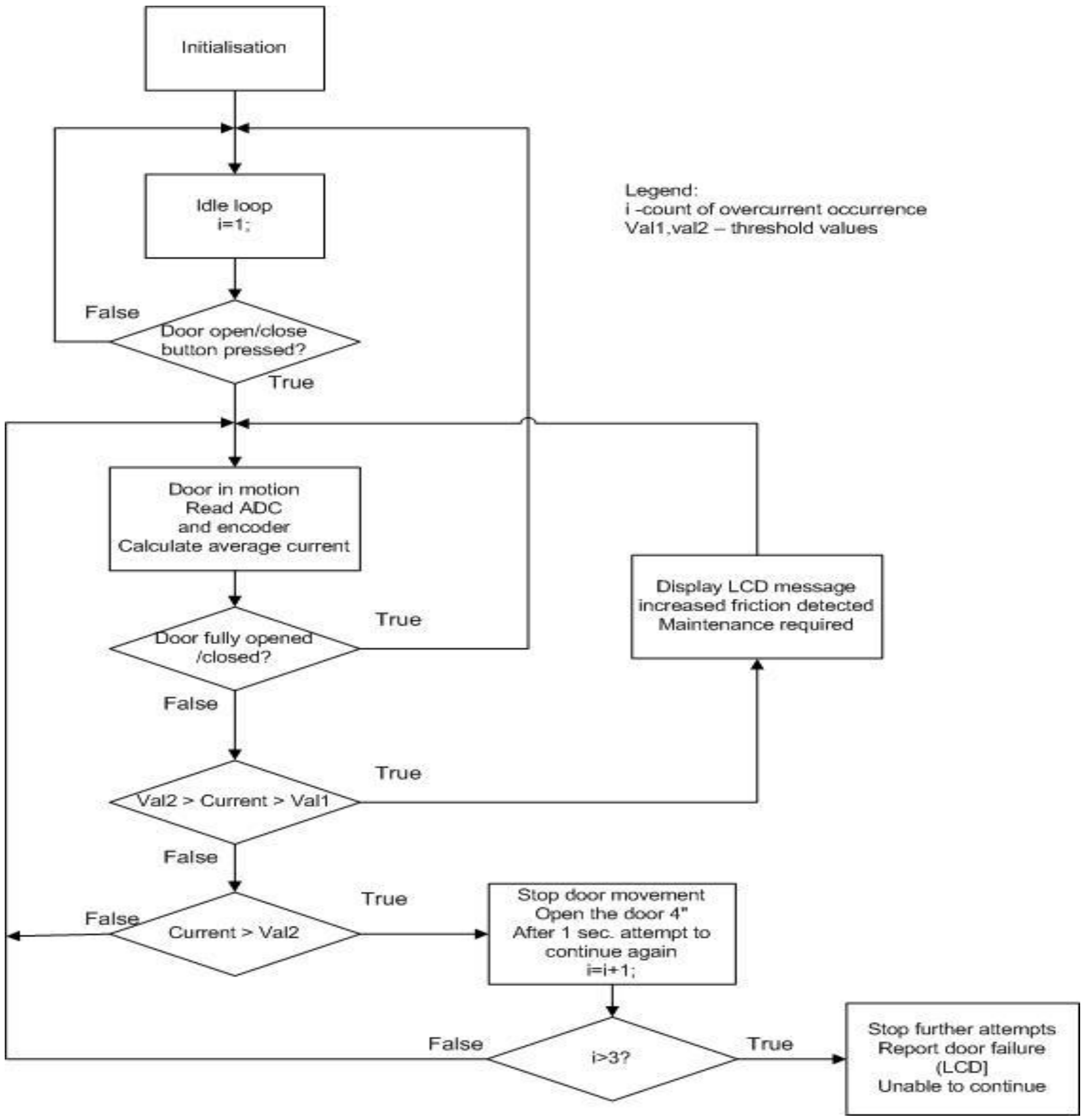


Figure 4-1. Overcurrent monitoring algorithm.

To detect a potential obstruction while the motor draws high current its velocity changes have to be analysed from optical encoder feedback signal. If any deviations are observed (i.e. velocity does not rise as expected) then the obstruction would be detected.

Supply voltage changes

Voltage is monitored separately for door opening and closing operations. If there is any voltage change on the motor windings, the controller displays a warning on the LCD. The warning is shown if the voltage is out of a defined range. The program allows ±1 V change from a nominal 32 V power supply without generating a warning. If there is higher change detected, implemented algorithm will try to keep the door velocity constant (equal to the velocity of normal operation) by adjustment of the PWM signal pulse width (through increasing or decreasing the PWM duty cycle, depending whether a lower or higher voltage was detected). The adjustment level is limited to the level of the power supply (for 32 V in normal operation the PWM pulse width is set to 75%). Therefore, if the power supply decreases, it can only be adjusted within a very limited range. Then the velocity of the door will be lower than in normal operation (for supply lower than 24V). If the supply voltage increases, it can be adjusted within a very wide range, even exceeding the motor nominal power requirements. This control process is a software implementation of a proportional regulator. The software regulator tries to keep the motor velocity constant by changing the voltage gain through dynamic pulse width modification. The measured voltage sense value used in calculations is derived in the same way as the current sense value (average from the last 8 measurements with elimination of the highest measurement result [noise peak voltage]). The voltage sense must be sampled with a twice higher frequency than the PWM signal to calculate its average value correctly).

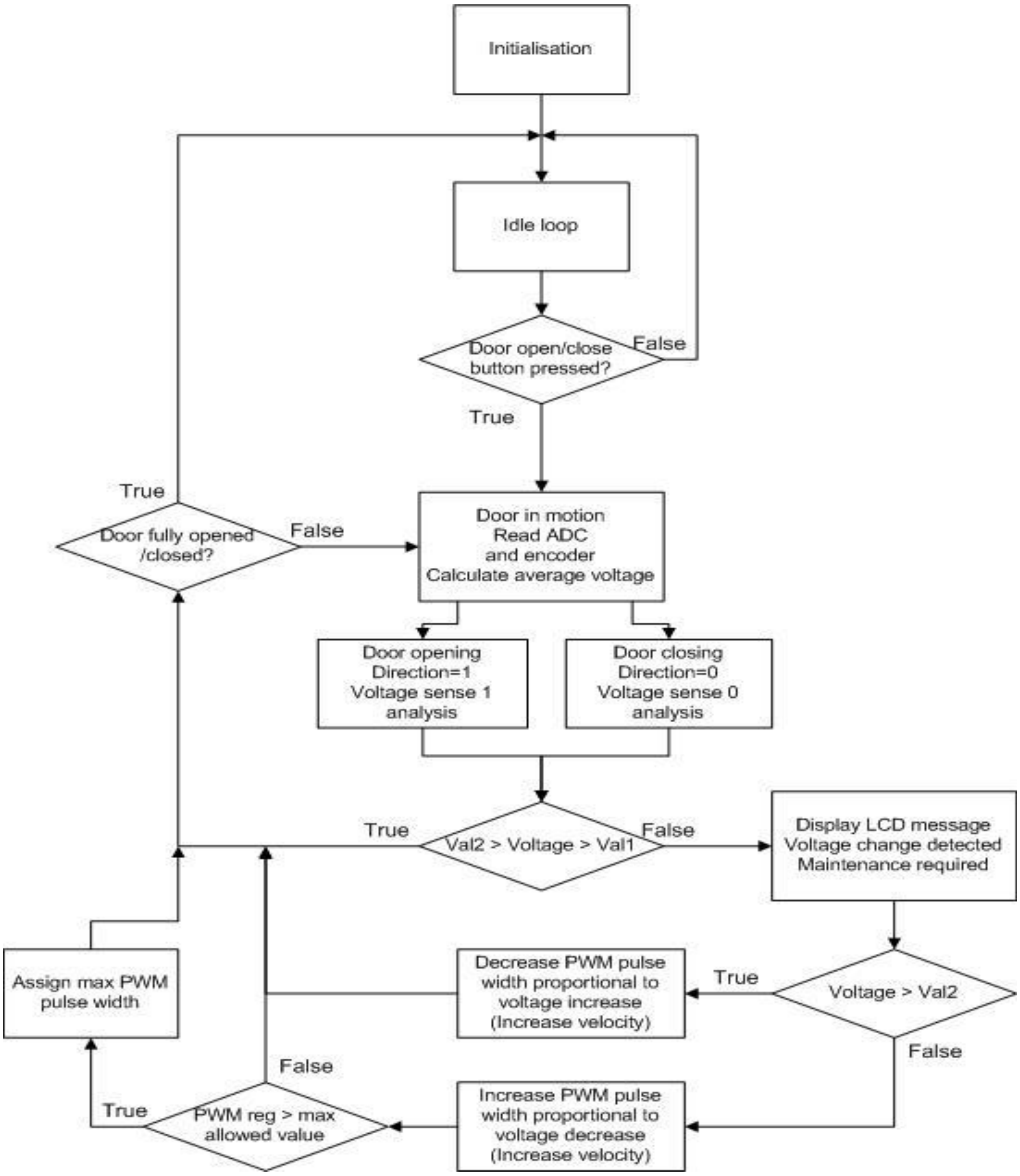


Figure 4-2. Voltage changes detection algorithm.

The implemented algorithm is able to keep a constant door velocity within a 24-55 V range, although voltages above 40 V were not tested. Below a supply voltage of 24 V, the velocity of the door will decrease due to a 100% PWM pulse width limit. The voltage on the motor windings will then be equal to the supply voltage. At 24V voltage the door opening velocity will be the same as the door closing velocity.

Pushbutton malfunction detection

This function is limited only to detection if the operation is executed after pressing the button. The fault is detected if there is no door movement after the button is pressed. Detection is made with the use of infrared sensors (no change of state after a period of time measured by an internal microcontroller timer indicates a fault). If the door was part of a whole train, additional safety features could be implemented. For example, if the door is controlled by a train operator and it does not react to the closing command, a control signal could be sent that prevents the train from leaving the station. This signal could give details to the train operator about the malfunction which has occurred, specifying a failing door and the potential fault (why the door was not able to close). To allow the train to continue in service, there could be mechanical leverage to close and lock the door, preventing future opening (override the control system to not use that door).

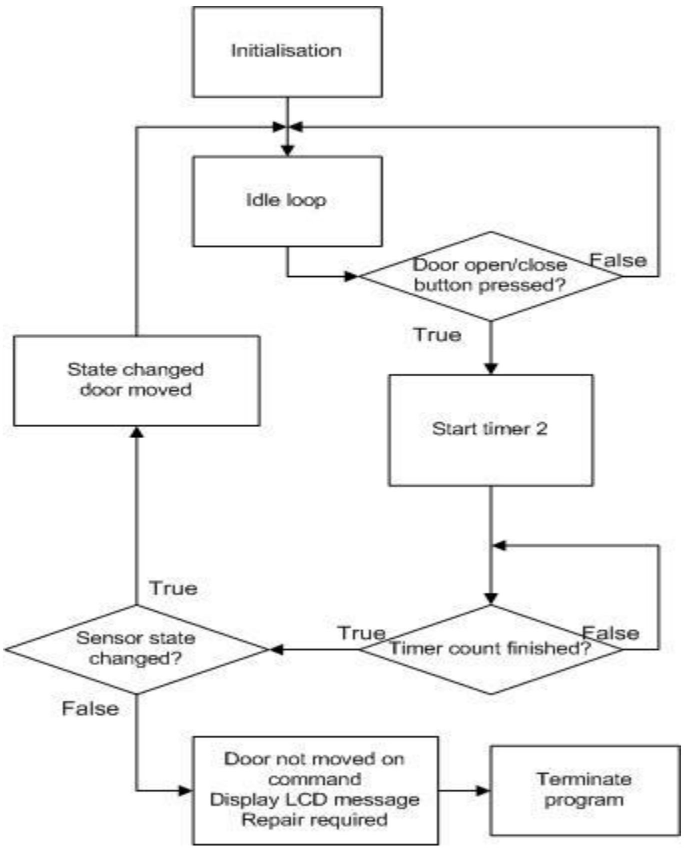


Figure 4-3. Door movement failure detection algorithm.

This algorithm can also be based on information coming from the optical encoder to determine if the door after pressing the button moved.

Increased friction detection

This fault can be detected in two ways. The first approach is the same as a), where increased current was detected and its value falls between the first and second threshold value defined in software. A message is then shown on the LCD display, as seen in figure 4-1. The second approach is based on door opening /closing time monitoring. The opening time is set to approximately 3.6 seconds and the closing time is 3.0 seconds. The door movement time is measured by an internal timer of the microcontroller. If the measured time is longer than expected, the controller indicates a warning that there is a minor fault, although it does not specify the malfunctioning component (insufficient information to isolate a fault). To determine whether increased friction caused operation time change other factors have to be analysed such as supply voltage and the motor velocity changes. The controller is still operational while detecting this fault.

Door not fully closed / open

There are currently two sensors installed that indicate the position of the door. One indicates when the door is fully open and the other when it is closed. Information provided from them is for verification of the readings from the optical encoder and is also used for calibration purposes (optical encoder start position error). Software functions responsible for door movement rely only on the encoder readings. When the controller is powered up, the start position is verified with sensor readings and the position registers value (QEI registers) is reset. At the end of the command execution, the position is also verified with the state of the sensors (i.e. when the door is closing). If it does not match, further movement is forced to ensure that the door is fully closed, until its position is confirmed with the sensor state (the door position is also verified with the encoder reading if the door panels are in the desired place). One of the infrared sensors also allows the distance to be measured. Using this sensor, a lateral movement of the door panel is monitored to detect whether the panel is loose. There is a threshold defined, within which each panel is allowed to move (average voltage level monitoring from sensor ADC input). If the sensor readings are out of range, a fault is detected. Detection of the door panel misalignment is signalled by a message on the LCD display. This approach can also detect if there is any dirt / debris accumulation on the rollers track at the bottom of the door panel that causes the panel to be pushed out. The rubber roller wear can also trigger this fault message as it causes the door panel to be closer to the sensor.

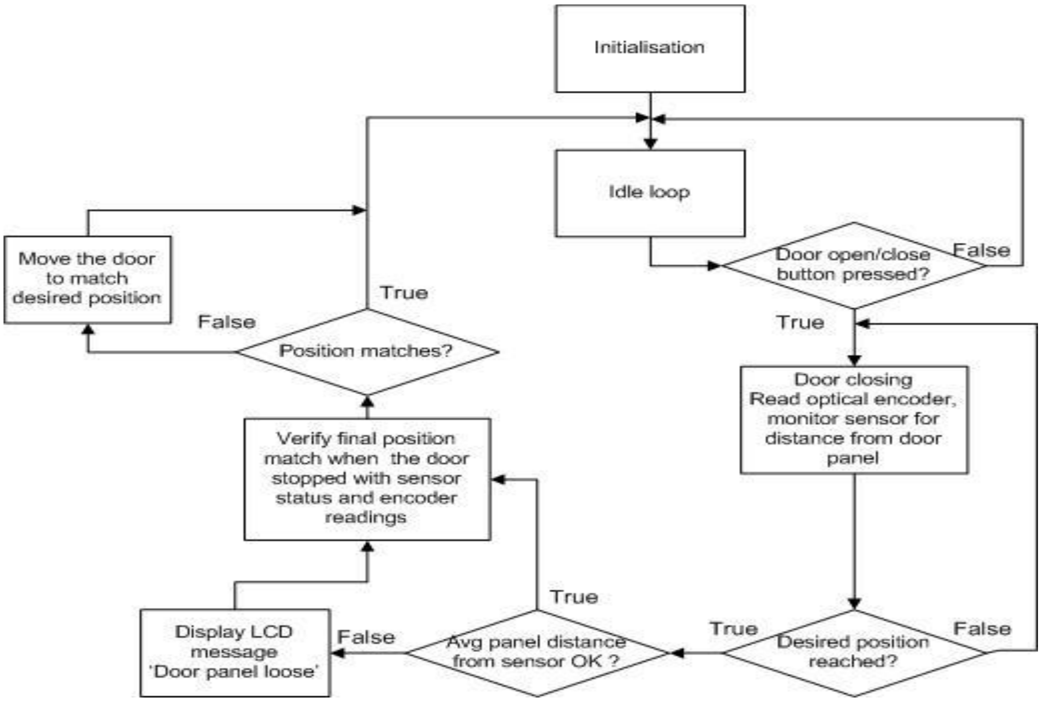


Figure 4-4. Door closing position verification algorithm.

The monitoring of panel distance from the sensor is realised only for the bottom right part of the door panel. To implement any undesired panel movement detection, a further two sensors are required to complement the existing sensors giving total of four per each door panel, two at the top and two at the bottom.

Door velocity changes

Velocity readings are obtainable from the optical encoder index output. However, readings from the encoder are not used in the implemented fault analysis algorithms due to very low amplitude of the signal and noise. Therefore, the velocity monitoring is carried out only through the door opening / closing time monitoring as described in d). There are procedures implemented to keep the velocity at a constant level by keeping the door opening / closing time constant. If the door movement is too long there will be a message shown on the LCD display. Door movement which is too fast does not occur, as the velocity depends on the supply voltage level and if it increases it is corrected by the software algorithm, i.e. the PWM pulse width is adjusted to the right level (pulse width decreased) as described above in b) and figure 4-2.

Optical encoder malfunction

The optical encoder faults are detected with use of infrared sensors. The distance travelled by the door is stored in two 8-bit QEI registers. If the door does not reach the desired position after executing the door opening / closing function (at the end of the operation the door position is verified with the sensor state), the fault is reported on the LCD display. When the door is fully opened the registers indicate 0xFFFF. When the door is fully closed the registers indicate 0x2630. The distance decreases when the door is closing. QEI registers are zeroed at each controller power up cycle after the door is fully open. There is no need for other calibration procedures in the laboratory environment. However, if the controller works for longer periods of time such procedures are required. The optical encoder malfunction usually causes erratic behaviour of the door if the encoder readings are wrong. The only way to detect encoder malfunction is to use some additional devices to verify its indications. For example, if a faulty encoder indicates too early that the position for door closed is reached, if there is no verification, the door can be left open and passengers put in danger. Therefore, using sensors or other methods to verify the position of the door panels is essential in commercial use. The verification algorithm is shown in figure 4-4.

PWM signal fault detection

The signal controlling motor movement is monitored through the microcontroller fault detection features. If there is a fault detected the PWM output is disabled and does not generate any output. In case of PWM failure, the H-Bridge PWM input must be kept in a low state, as any other state makes the door movement.

Current minimising

There are procedures implemented to minimise the current drawn from the source while the motor starts and given back to the circuit while the motor stops. The difference between the controller operation with and without these procedures can be seen in the next chapter. These procedures work through dynamic change of the voltage on the motor windings. The supplied voltage depends on the PWM pulse width and is monitored throughout whole control process. If a higher voltage is detected while the motor starts, procedures are adjusted accordingly, as in b). The pulse width is increased / decreased in steps over a period of time until the motor reaches full speed when the door starts moving or smoothly slows down to the destination point. The procedures are calibrated to not exceed a 3 A current draw from the source when the motor is moving the door (this also keeps the H-Bridge fully functional throughout the control process by not exceeding its maximum allowed continuous current value [3 A]) and to prevent giving the current back to the source when the motor is braking. Therefore, while the motor starts, there are five steps increasing the PWM pulse width gradually until it reaches the desired value and an additional step for the door opening procedure (for faster door opening). While the motor brakes there are three steps implemented. This approach also minimises noise when the door is fully open or fully closed as the motor slows down smoothly. The noise occurring when the panels meet each other in the middle is decreased and when the door is opening it does not hit the limiter with maximum motor velocity. A higher PWM frequency also decreases noise coming from the motor drive. Tests were made starting from a frequency of 1.2 kHz and reaching 9.6 kHz PWM frequency, as that is sufficient to be used in the control process.

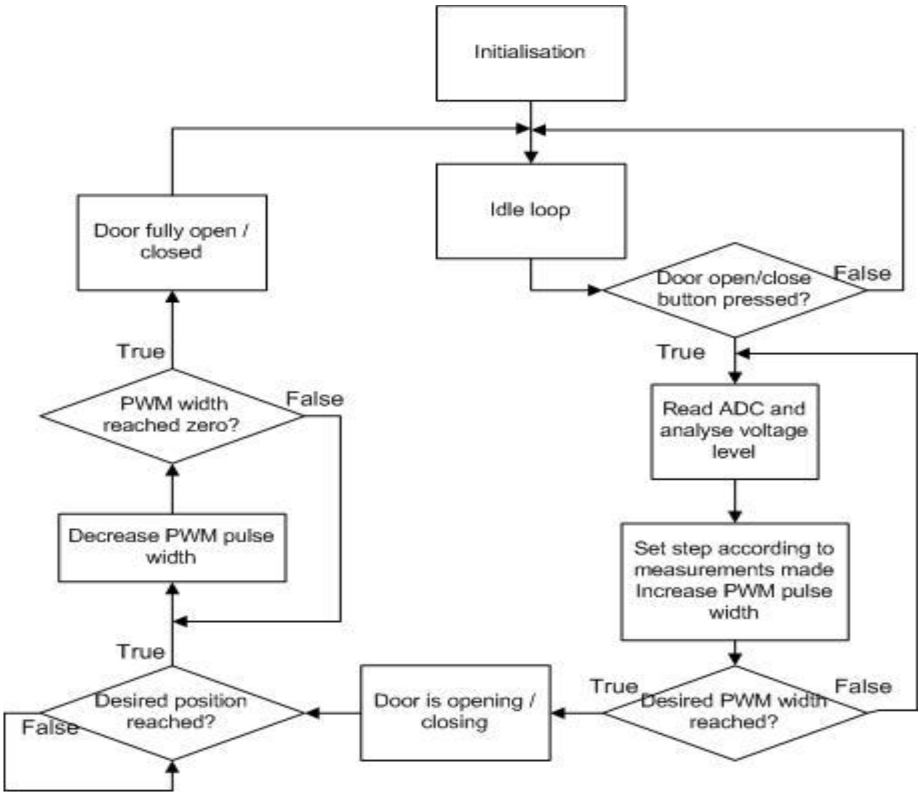


Figure 4-5. Current minimising algorithm.

The current limiting procedures also decrease the heat emitted by the H-Bridge circuit, therefore prolonging its life by not working at construction limits and decreasing the possibility of the module overheating.

4. 4. Summary

The implementation of the controller fault tolerance was explained. To increase its functionality and improve fault detection additional sensors can be fitted. Based on the information provided by these sensors, more details can be collected regarding the fault, which allow exact identification. At the current stage (when the fault is detected) there are always a few reasons which could call out the message displayed on the LCD. Methods are available to narrow potential fault sources, for example, distance sensors can be fitted indicating wear of the rollers if the door panels are closing up to the train chassis on each side. Monitoring motor speed in a closed loop from the optical encoder output would improve the controller functionality. Especially when the motor accelerates by only monitoring current any fault related to its increase at that time can't be detected. An intelligent algorithm could be implemented to control the PWM pulse width dynamically when the motor starts / stops based on actual current measurement to decrease current peaks as well.

Abnormal behaviour	**Monitored signal**	**Possible faults**	**Action**
Increased friction, obstruction detection	Current sense signal slightly increase	Rollers worn, motor brushes worn, debris, dirt on the door panel track etc.	LCD message, no other action
	High current sense signal increase	Potential obstruction detected or other, not able to close the door	LCD message, attempt to continue operation, if failed, stop all
Supply voltage changes	Voltage sense signal change	Voltage increased / decreased on motor windings, possible slower door movement	LCD message, PWM pulse width adjusted to keep the door velocity as in normal condition
Pushbutton malfunction detection	Sensor indications	Sensor status not changed, door not moved, possible motor or power supply malfunction	LCD message, no other action

Door not fully closed / open	Sensors indication if encoder shows position for the door fully opened / closed	Optical encoder failure, encoder readings not calibrated with sensors statuses	LCD message, further door movement forced to fully close / open to match with sensor indication
Door panels misaligned	Door panel distance from sensor	Lateral distance changed, possible loose panels, dirt in the door panels route, rollers worn	LCD message, no other action
Door velocity monitoring	Door opening /closing time measurement	Opening / closing time exceeded, voltage changed	LCD message, no other action
Peak current minimising while motor starts / stops, noise reduction	N/A	Based on simulations to receive desired results regarding noise and current overshoot reduction,	PWM duty cycle changed gradually while motor starts / stops
Motor optical encoder malfunction	Sensors, optical encoder	Door erratic behaviour, controller unable to operate properly	LCD message, force to close the door until sensor indication, controller failure
PWM signal fault detection.	Internal microcontroller function	No PWM signal generated	LCD message, controller failure

Table 4-1. Summarised implemented fault detection.

As mentioned above, the door position calibration occurs when the controller is powered up. For the laboratory experiments this procedure is sufficient. If this controller were part of a train control system, the optical encoder readings would have to be verified on a regular basis, as in longer periods of controller operation a small displacement may occur (in the main loop of the software program with infrared or other sensor readings). This could easily be done with the use of the written software with a small modification. The supplied power could then be cut off from the controlling module each time the train moves and turned back on when it stops. This approach would prevent the door from unauthorised opening (door only released manually in an emergency situation) and would also implement energy saving features.

Further step, after the fault tolerance implementation is to document the controller functionality and behaviour in case of a fault. Therefore, data is being collected. Test stage and fault simulation procedures are described in the next chapter.

Chapter 5. Analysis of results

5. 1. Preface

In this chapter the designed controller functionality is documented. Normal working conditions and reproducible faulty conditions are tested. The test stages show how are working in practice the fault detection algorithms described in the previous chapter. Gathered results are carefully analysed and commented.

5. 2. Test stages

Before attempting to connect power to the controller and H-Bridge boards, each module was checked electrically to ensure that the connections were correct according to the schematic circuit and that no unintended short circuit had occurred in the assembly process. Both controller modules were then connected together and tested. The next stage was to verify whether the voltage level of signals fed back to the microcontroller were within the desired range and whether it was safe to connect them to the microcontroller. Particularly extreme conditions (for example when overcurrent occurs) were tested to prevent circuit damage in the future. After implementing simple door movement functionality, the control signals generated by the microcontroller were checked on an oscilloscope to ensure that they were correct and the controller was then connected to the motor. When all the above steps had been successfully carried out, fault tolerance procedures, as described in the previous chapter, were implemented. Then the data acquisition process was started.

5. 3. Data acquisition and analysis

The data was collected using a NI USB-6008 data acquisition device from National Instruments. For the purpose of data acquisition, a LabVIEW Signal Express application was used. The acquired results were then exported to a Microsoft Excel spreadsheet. Depending on which PWM frequency was set, the data sampling rate varied in range from 100 Hz to 10 kHz (maximum sampling rate for this device). The acquired data includes the voltage and current sense output signal values. Some of enclosed below plots (figures 5-5, 5-7 and 5-11 to 5-15) do not show original signal frequency but only illustrate when duty cycle changes over time. The signals were monitored separately while the door was opened and closed. The data was gathered in normal operating conditions and also with simulated fault conditions. The current value was collected from both the H-Bridge current sense output and the transducer output. Based on the PWM pulse width measurements over the door movement time, the velocity profile was derived. The velocity measurement could not be acquired directly from the optical encoder, as was mentioned before, because of an encoder index output malfunction (the encoder generates an unexpected output).

All the acquired results are pre-scaled from measured quantities to real ones. Scaling factors are based on used resistor values, on information found in H-Bridge technical notes (National Semiconductor Corp., 2002) regarding the current sense output and on transducer technical notes (LEM, n.d.). As the monitored signals are analysed through the microcontroller's ADC their value had to be altered to always be between 0 V and 5 V. A 32 V power supply is considered to be nominal in all tests to assure the door movement velocity specified in requirements.

Fault detection time

Data sampling time is determined by the time of ADC conversion. The first conversion is started immediately after the door movement button is pressed and the last one finished when the door reaches desired position. The analogue to digital converter is set to work at the maximum speed allowed with 20 MHz oscillator in a single mode conversion. The conversion time is then approximately 50 µs to acquire samples of each monitored feedback signal. Once the conversion is finished, an ADC interrupt is serviced. The interrupt service takes quite a long time and the delay is mainly caused by the LCD display support. Each time the conversion is finished, current and a door position register values are displayed. To show these measurements (plus a small amount of time for instruction execution) approximately 500 µs is required. Once the interrupt service routine is finished, another conversion is started. Then if, for example, an obstruction occurs, a minimum detection time is a sum of the above, which equals approximately 550 µs. The algorithm does not detect the obstruction if sample indicates overcurrent only once (single current peak is treated as noise and ignored). Detection has to be confirmed by another conversion result. Therefore, the minimum time for the obstruction detection is 1100 µs. The detection time for any other faults is the same. Replacing the LCD with any device that can display messages faster such as a serial link to a PC running at high baud rate (or by removal of being displayed values) could decrease detection time significantly (currently the obstruction detection time is comparable with the Vapor Rail door controller (Dassanayake p. 51-52, 2001). To identify a fault properly the algorithms have to analyse more stored samples from previous measurements and also some threshold values should be changed over the control process as not all monitored signals are linear. This approach lengthens detection time but would identify faults more accurately.

Normal door operation

Current analysis

The current was measured for various PWM frequencies and various power supply voltage levels to observe the motor behaviour in different conditions. Normal and faulty conditions were tested and the best working conditions for continuous operation were chosen. The lowest PWM frequency was set to 1.2 kHz and the highest frequency was 9.6 kHz. Other frequencies between these two values were also tested, although the results are not shown here as the controller represents intermediate states between these values. Based on the tests, the 9.6 kHz frequency is chosen for normal door operation. All further tests are pursued with this frequency. The data sample rate is also increased for the higher PWM frequency. Current characteristics for the voltage supply at lower and higher level are shown in figures 5-1 to 5-4 below. Each current sense graph also includes a trend line showing the average current value from collected samples.

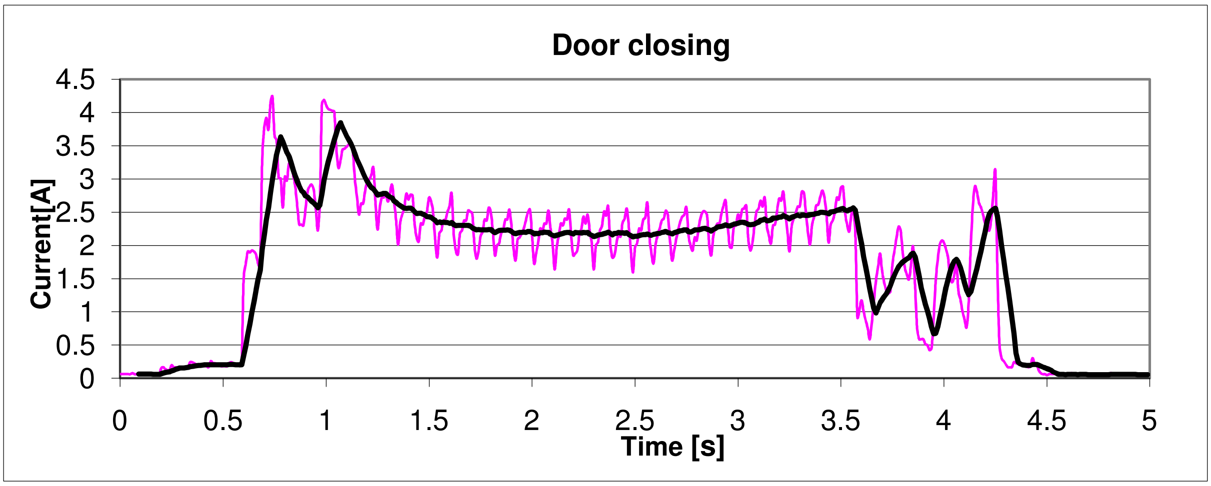


Figure 5-1. Current sense for door closing @ 1.2 kHz

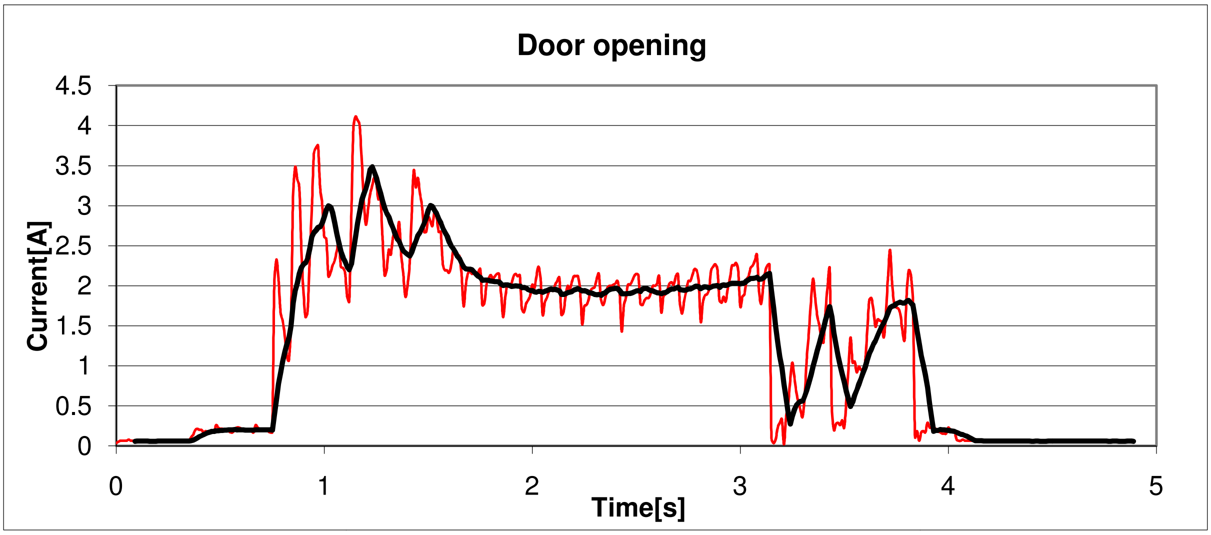


Figure 5-2. Current sense for door opening @ 1.2 KHz

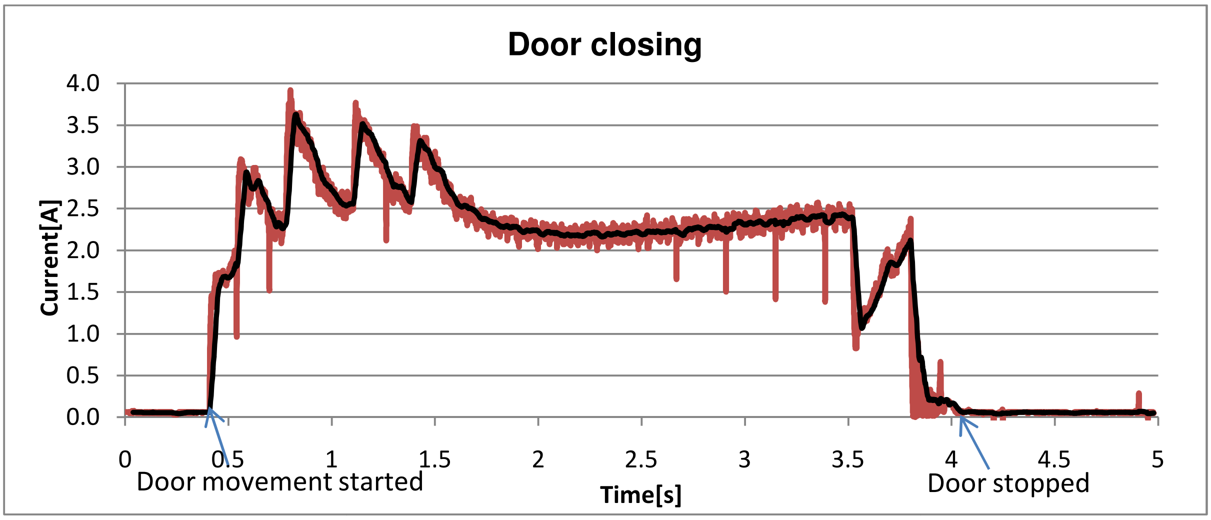


Figure 5-3. Current sense for door closing @ 9.6 KHz

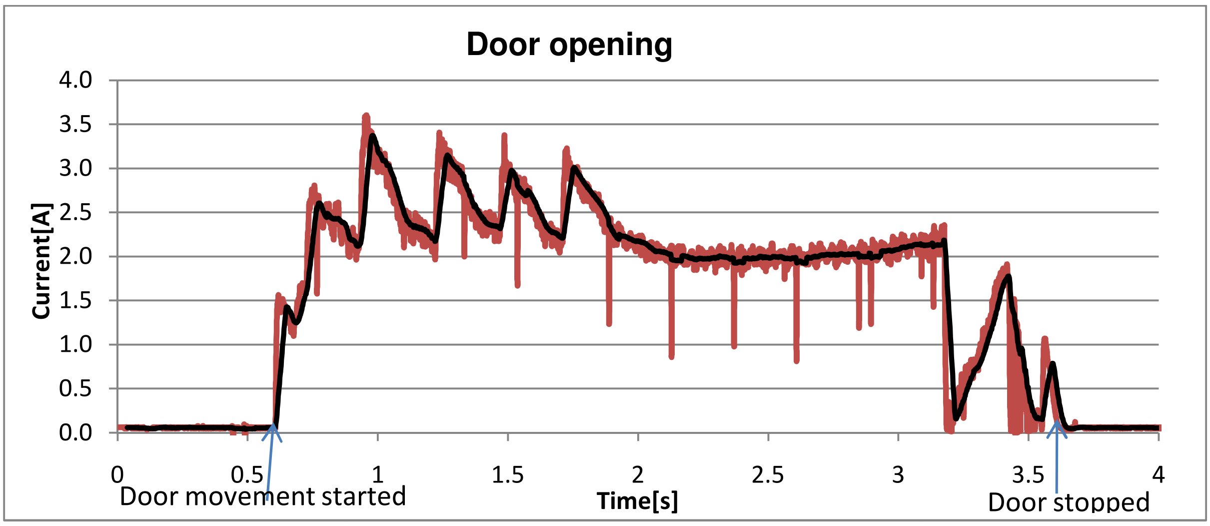


Figure 5-4. Current sense for door opening @ 9.6 KHz.

Based on the above plots (figure 5-1, 5-2 comparing with figure 5-3 and 5-4) it can be seen that increasing the PWM frequency gives a more stable output signal and decreases current overshoot (when the motor starts, current drawn from the source dropped by approximately 0.5 A [figure 5-1 and 5-3]). With higher PWM frequency the motor also operates more smoothly and noise coming from it is being minimised (the motor operation can be barely heard). The current draw is slightly higher when the door is closing, by approximately 0.2 A (on average 2.25 A for a closing operation and 2.1 A for opening [trend lines on plots 5-3 and 5-4]). With higher average current there are higher current peaks when the motor starts in the door closing profile. The current peaks indicate the function execution in the software, which increases the PWM signal pulse width to ensure a smoother motor start and also to limit current drawn to maximum 3.5A from the power source. The time of each PWM pulse width change and amount of changes (for the door opening procedure there is one more pulse width change) can be clearly distinguished. After the desired duty cycle is reached the current drawn from the source is stabilising for the rest of the door movement operation. At the end of the door control process, when the motor is braking, the PWM duty cycle also drops down gradually. There is a higher current drop when the door is opening that occurs due to the door higher velocity acquired in this profile. The procedure of decreasing the PWM pulse width gradually is also implemented to avoid situations when the motor brakes and gives the current back to circuit. However, the motor works as a generator only when braking from high speeds for supply voltage closer to nominal one. If the motor produces power, accumulated power is emitted by the H-Bridge as heat to the surrounding area. This situation is not desirable, although it does not damage the controller. In some cases, however, it could increase the temperature of the whole controller and may cause the control module to overheat if no precautions are taken. Slowing the motor down gradually also reduces panel impact noise. When the door starts moving it gains velocity smoothly and when it stops it loses speed gradually. This approach prevents a sudden sharp movement of the door panels when the motor starts and when it stops impact with the maximum velocity.

The acquired data also shows that the door opening and closing times are within the range specified in the requirements. For the opening operation it is approximately 3 seconds and for the closing operation 3.6 seconds. Movement time differences indicate that the door is being opened with a slightly higher velocity (additional step increasing the PWM duty cycle on figure 5-4). Figure 5-5 below acknowledges the current sense analysis indications.

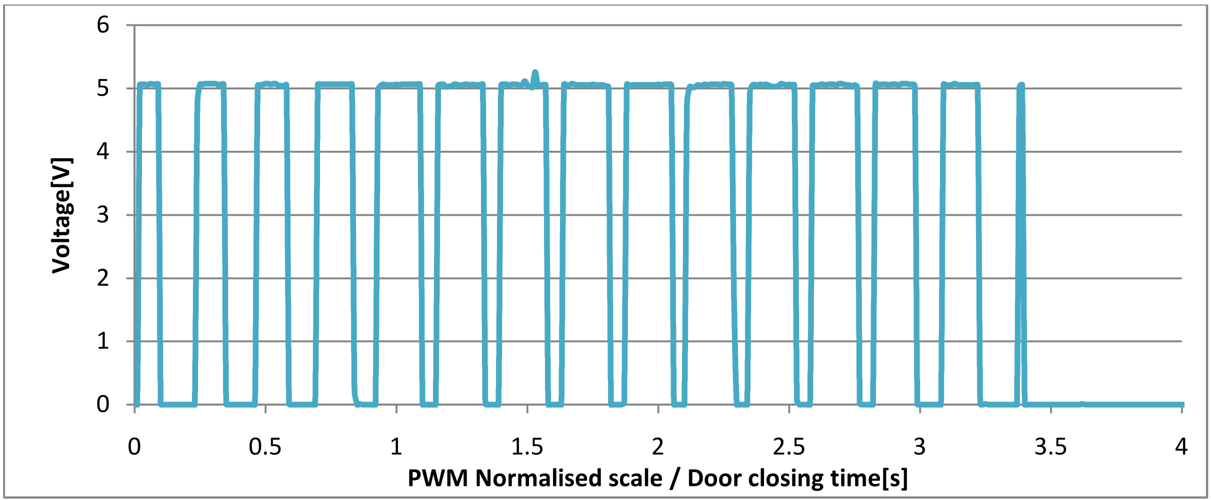


Figure 5-5. Normalised PWM control signal for door closing.

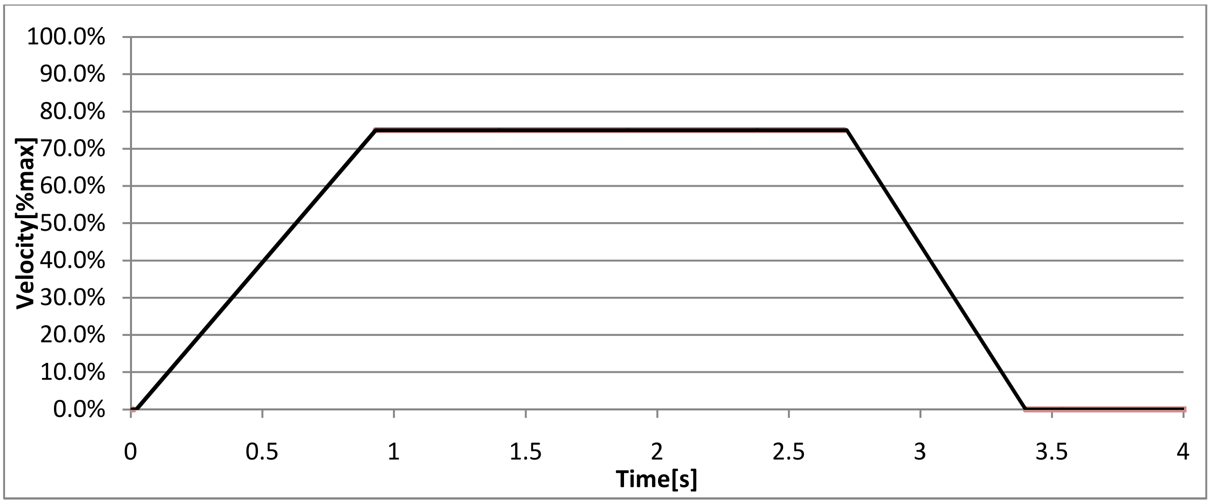


Figure 5-6. Velocity profile for door closing.

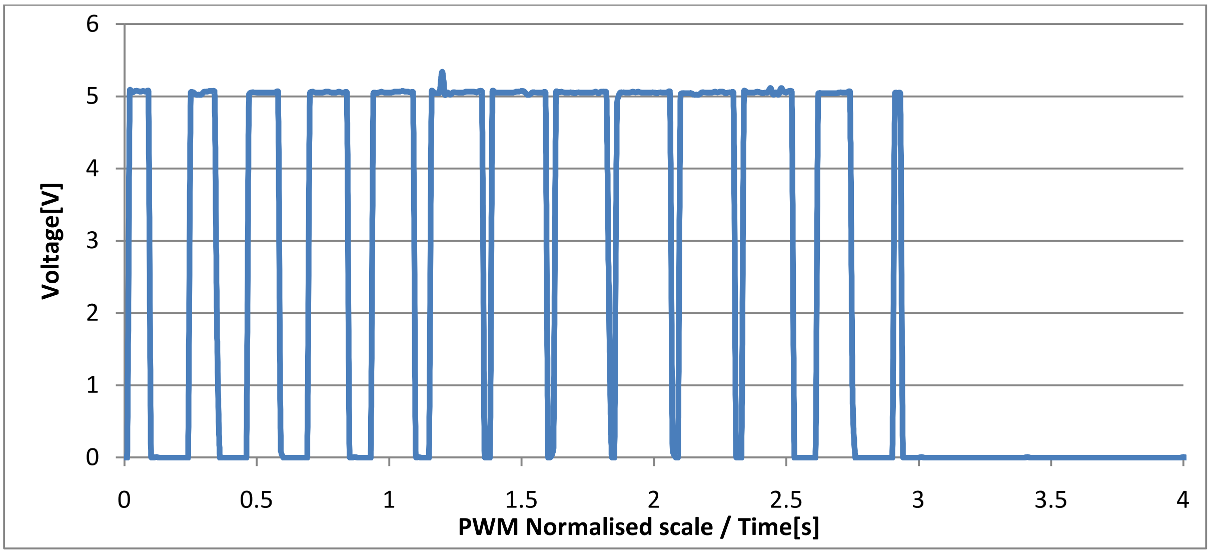


Figure 5-7. Normalised PWM control signal for door opening.

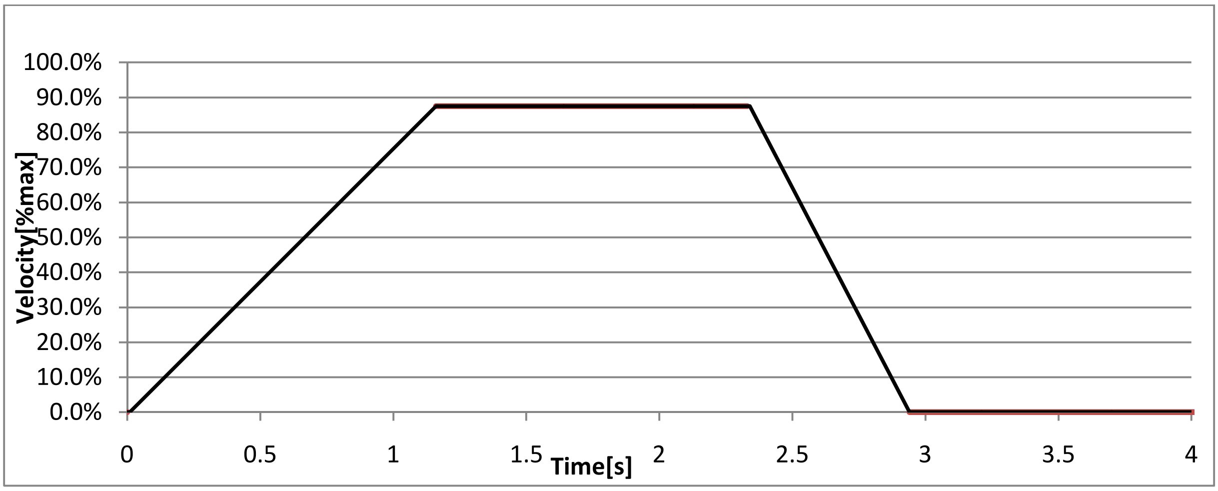


Figure 5-8. Velocity profile for door opening.

The PWM signal is generated for 3.4 s while the door is closing and 2.9 s while opening (plus motor braking inertia give the opening / closing times mentioned above). Figures 5-5 and 5-7 enclose a normalised PWM signal to demonstrate when pulse widths are being changed. Based on the PWM signal pulse width, velocity profiles were derived. Maximum velocity is reached for the widest pulses generated. The velocity is pre-scaled as a part of the maximum velocity that could be achieved with 100% PWM pulse width for a 32 V power supply. The motor speed rises for approximately 1 s and drops down 0.6 s when the door is being closed. This is indicated on figure 5-3 by the time at which the current starts rising to the time at which the last current rise occurs, where the motor velocity reaches the maximum value (75% PWM pulse width). Velocity profile for the door opening has slightly longer slopes as there is one additional step changing the PWM duty cycle. This step gives a bigger pulse width (87.5% PWM pulse width) resulting in increased motor acceleration. This is also indicated in figure 5-4 where the last current raise occur 1.2 s from the time the push button was pressed. All the above relates to the control process where functions limiting current overshoot are active. These procedures cause the motor to start and stop over a longer period of time but ensure that the current value does not exceed 3 A and noise reduction. Figure 5-9 below shows current drawn when functions increasing the PWM pulse width gradually are disabled.

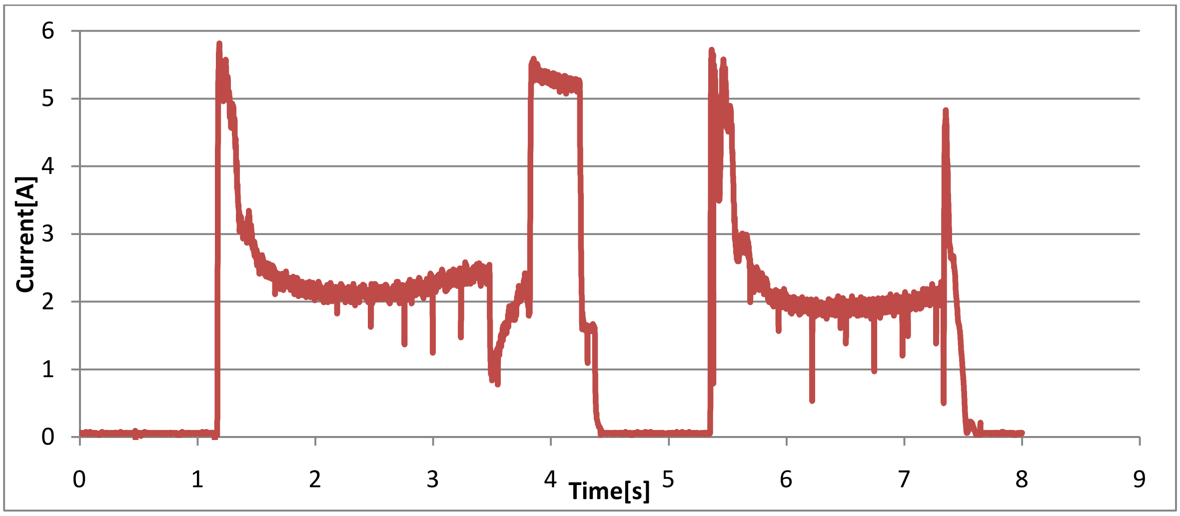


Figure 5-9. Current sense for door movement with constant PWM duty cycle.

Eliminating functions to operate the door smoothly nearly doubles the current drawn from the source as when the motor starts, a peak reaches nearly 6 A (to acquire higher velocity result in an even bigger current peak). To carry out this test, overcurrent detection was disabled due to its occurrence at the end of the door movement profile. The door panels are moving over the whole distance with constant speed (75% PWM duty cycle) and they hit each other at high speed when the door is closing or when the opening panels hit a door limiter fitted on the chassis, as the motor velocity does not drop gradually. This causes the high current to occur again when the door reaches its destination. In the software, when the function changing the PWM duty cycle is enabled, it is activated when the door reaches the desired position and then velocity starts to drop down gradually until the door comes to a complete stop making the control process smooth.

In figure 5-10 below the opposite situation to described previously is presented. The number of PWM duty cycle changes to reach the maximum velocity was doubled.

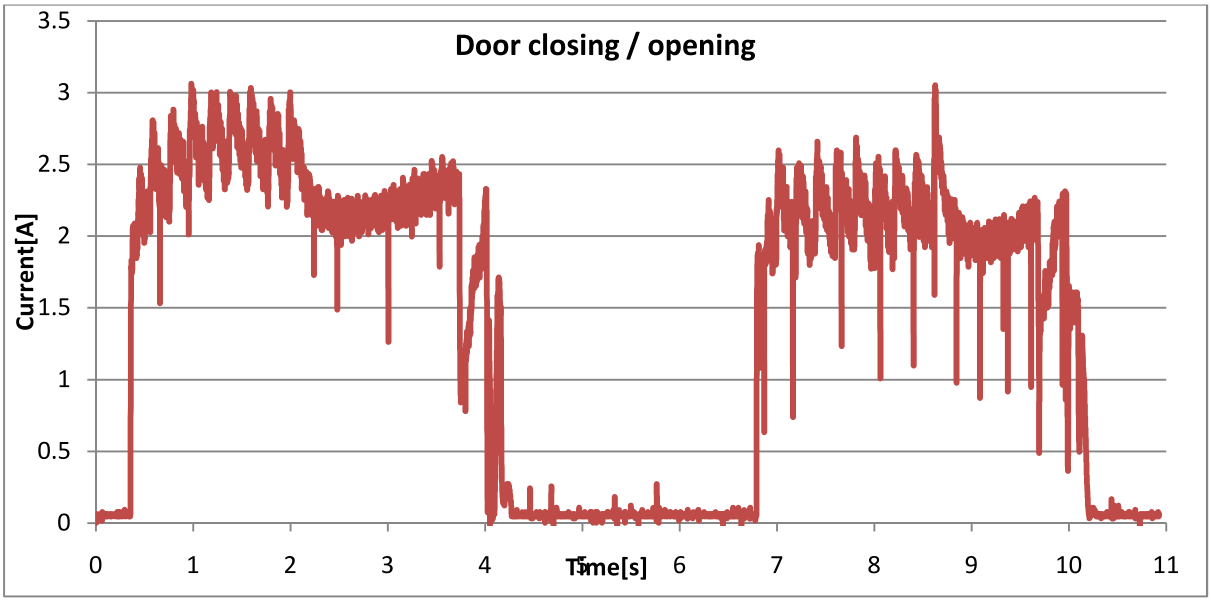


Figure 5-10. Current sense for door closing / opening with doubled PWM duty cycle changes.

Due to the greater number of steps, the current drawn from the source supply (3 A max.) is decreased compared to the number of steps chosen for a normal door operation profile, as described above. A compromise must be made between the current drawn and the velocity profile, as increasing amount of PWM duty cycle changes is keeping the current as low as possible but causes the velocity to increase more slowly over a longer period of time until reaches constant level (each change of duty cycle has a minimum execution time). Therefore the door movement time lengthens, reaching nearly 4 s for closing and 3.5 s for opening. Implementing more steps can make the current characteristics smoother and reduce overshoot. However, to keep the door opening /closing time within a specified range requires increasing the defined maximum PWM duty cycle, resulting in a velocity increase. A higher velocity may require even more steps to move the door smoothly, which also incurs more time needed for the motor to gain the required speed.

An implementation of a software controller can be used to minimise current peaks. Its task would be to increase the PWM pulse width based on the current sense signal value over time to ensure the best characteristics. This controller would decide how many steps are required to not exceed the previously specified door opening / closing time and reach the desired velocity whilst also keeping the current drawn from the source as low as possible.

Voltage analysis

Measured voltage is divided to ensure the right voltage level on the ADC microcontroller input. It is altered to sustain up to a 52 V maximum motor voltage rating and power supply as in an original train power source line.

The voltage profile is similar to the velocity profile as the motor velocity depends on the voltage level. A higher voltage on the motor windings results in a faster door movement. Although after implementation of the voltage control algorithm there is only a slight change of the door movement speed in case of voltage increase, which is barely noticeable. Even this little difference can be eliminated through tuning the algorithm parameters. Figures 5-5 and 5-7 illustrate PWM duty cycle changes generated by the microcontroller over the door control process time. The voltage sense signal is the same PWM signal generated by the microcontroller amplified by the H-Bridge. On the motor windings is its average value.

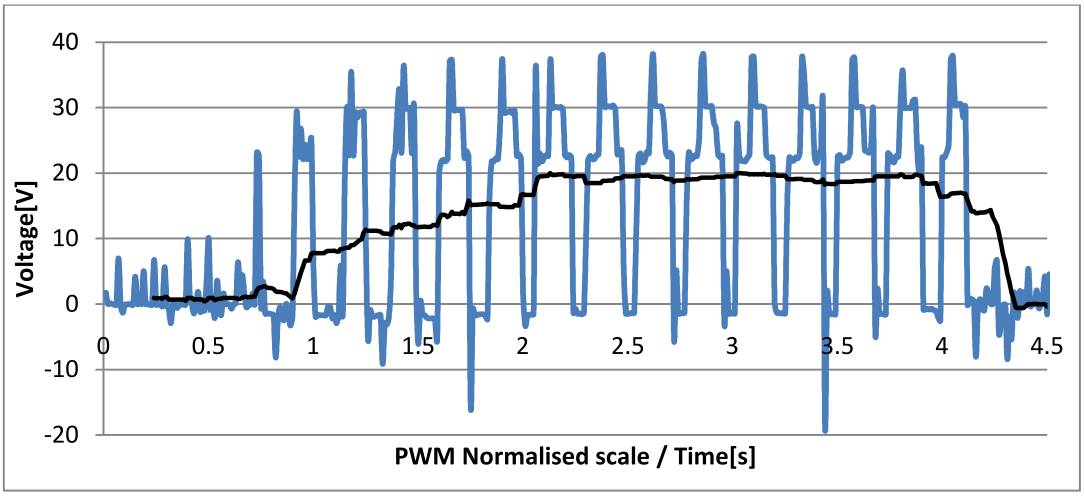


Figure 5-11. Voltage on motor windings for door closing.

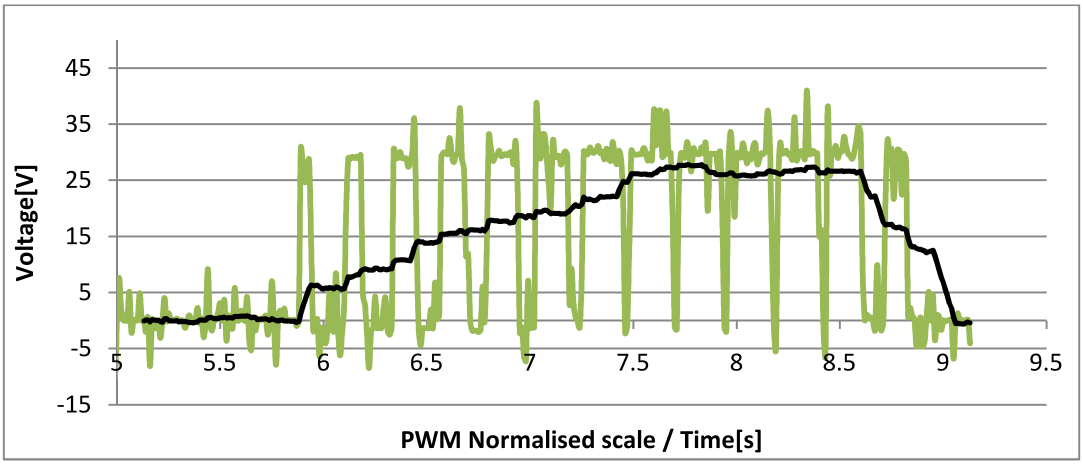


Figure 5-12. Voltage on motor windings for door opening.

The trend lines in figures 5-11 and 5-12 show the average voltage based on acquired samples that is present on the motor windings. This average profile matches the velocity profile. Normalised PWM plots in figures 5-11 and 5-12 (and also plots in figures 5-13, 5-14, 5-15 below) were derived based on data collected with sampling rate lower than the PWM signal frequency, therefore lots of samples are missed in the analysis.

The voltage sense output monitoring (that is based on the PWM frequency) puts the requirements for the sampling rate. Therefore, to avoid omission of any sample the sampling rate should be a minimum of the doubled PWM frequency (Nyquist criteria), which is 19.2 kHz. Then the voltage sense has to be sampled every 50 μs. As was mentioned above, the whole sampling process with code execution time takes 550 μs. It means that there is only one sample collected out of 11. Therefore, the monitored signal is not recognised correctly. Then it is difficult to set the conditions in the velocity regulation algorithm for a high voltage situation where the PWM signal changes value frequently, which can result in incorrect voltage level interpretation.

For a higher voltage, the velocity is maintained at the same level. With an implemented voltage control algorithm, the PWM pulse width is adjusted resulting in the average voltage on the motor windings being lower, in spite of a higher power supply. The normalised motor control signal for a 36 V power supply is shown in figures 5-13 and 5-14 below. The trend line indicates the average voltage on the motor windings throughout the control process. Comparing the trend lines with those in figures 5-11and 5-12, it can be seen that velocity is kept at the same level, close to normal door operation as with nominal voltage.

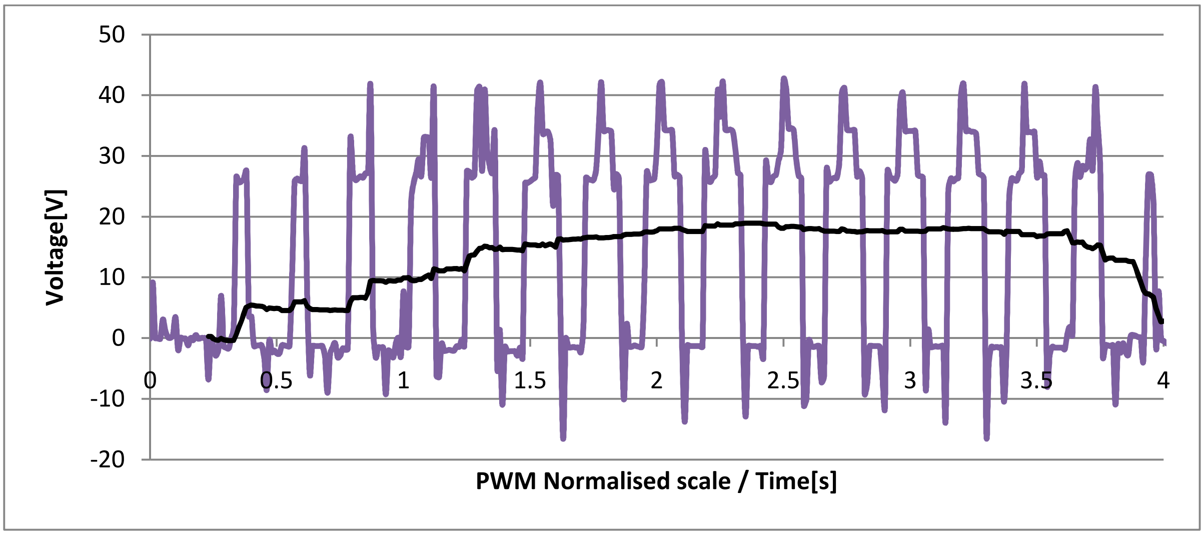


Figure 5-13. Voltage on motor windings for 36 V supply, door closing profile.

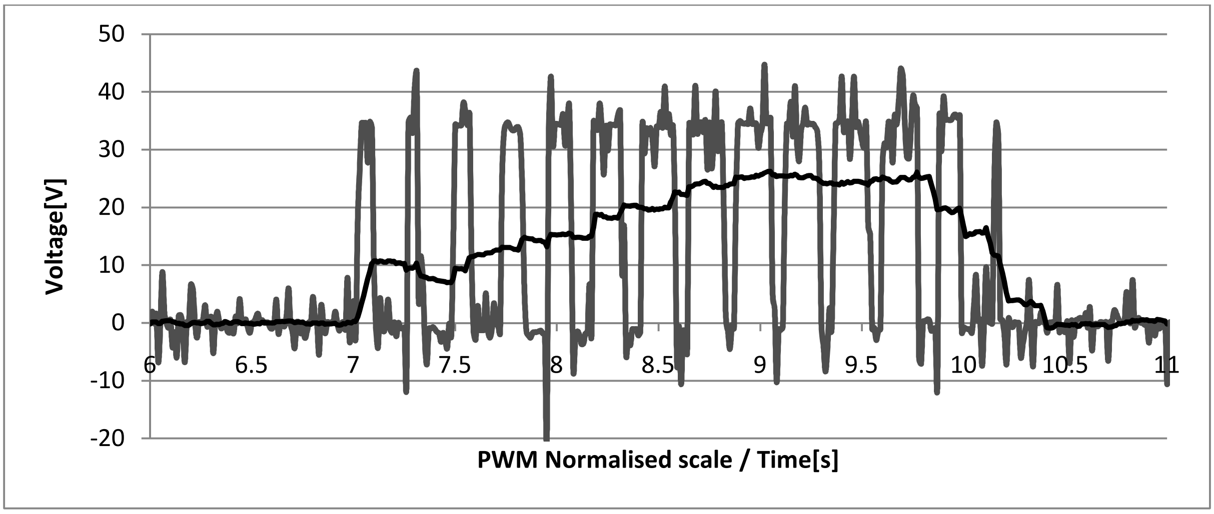


Figure 5-14. Voltage on motor windings for 36 V supply, door opening profile.

The voltage sense signal profiles in figures 5-13 and 5-14 above are for a 36 V power supply. When the voltage supply level falls below the nominal value (32 V) the PWM pulse width is increased up to 100%. The voltage loss is then compensated down to 24 V. Below that value, the door opening / closing time is lengthened as the maximum duty cycle is reached.

Fault emulation

Faults tested in the laboratory environment include simulation of increased friction (as in rollers wear), obstruction detection, push button failure, loose door panels and velocity changes (caused by voltage supply change) detection.

Increased friction detection

Roller wear is detectable through increased friction (although this is the most common reason for increased friction, dirt accumulation on the door panel route or ball bearing faults can also cause friction increase, which will also be detected by the same algorithm). In this condition the motor draws more power from the source to be able to move the door and overcome higher friction. If increased current is detected, the controller continues normal operation and displays a message on the LCD screen.

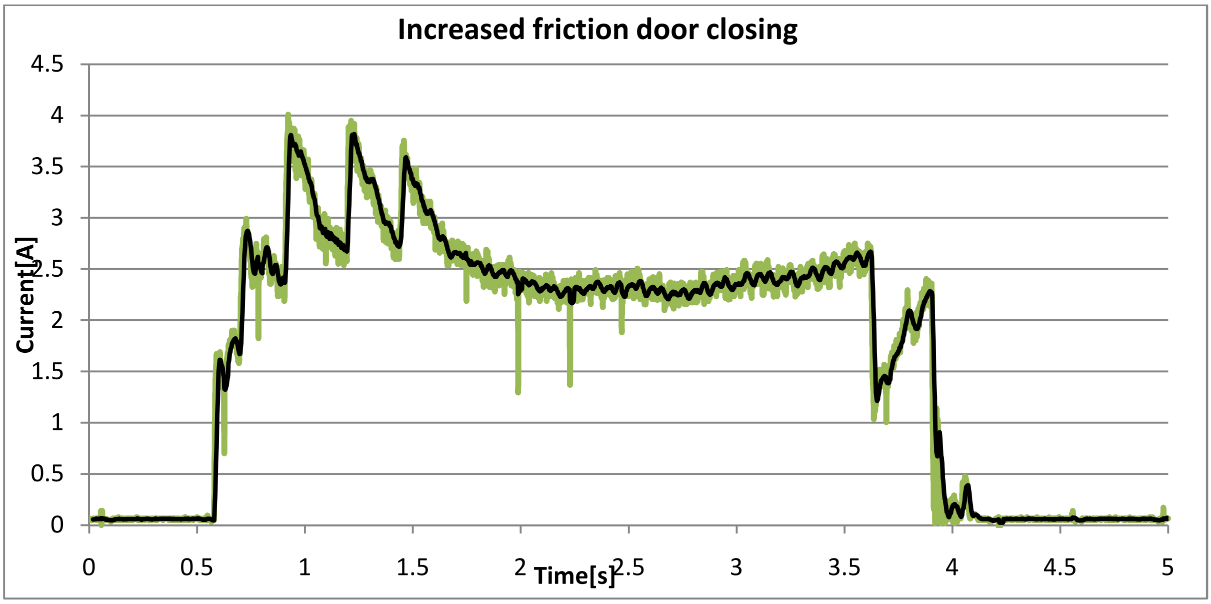


Figure 5-15. Current sense with increased friction for door closing procedure.

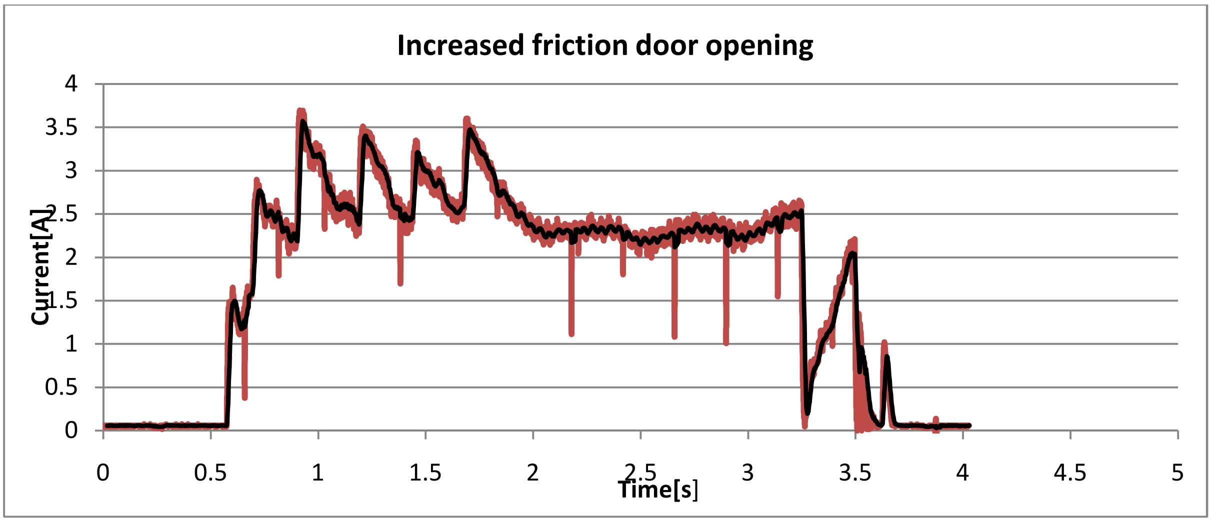


Figure 5-16. Current sense with increased friction for door opening procedure.

The current drawn is slightly higher for both profiles compared to the door normal operation current drawn shown in figures 5-3 and 5-4. Detected change is approximately 0.2-0.4 A. The threshold set in the software reflects a 0.2-0.6 A current rise and any current sense value in between is interpreted as a detection of increased friction.

This test was carried out by putting a foreign object between the rollers and the door panel. It was causing the movement of the rollers to be more difficult resulting in current rise and falling in between the defined threshold. Leaning on the door panel is also detected by the software algorithm as a friction increase and the message is shown on the LCD display.

Obstruction detection

The second simulation was obstruction detection. This test is also based on monitoring the current value to detect higher current changes that fall above 2.8 A. Once obstruction is detected a timer measuring the door movement time is disabled.

Figure 5-17 shows the whole process that is undertaken by the controller if the obstruction is detected, including a successful attempt that the controller makes afterwards to close the door.

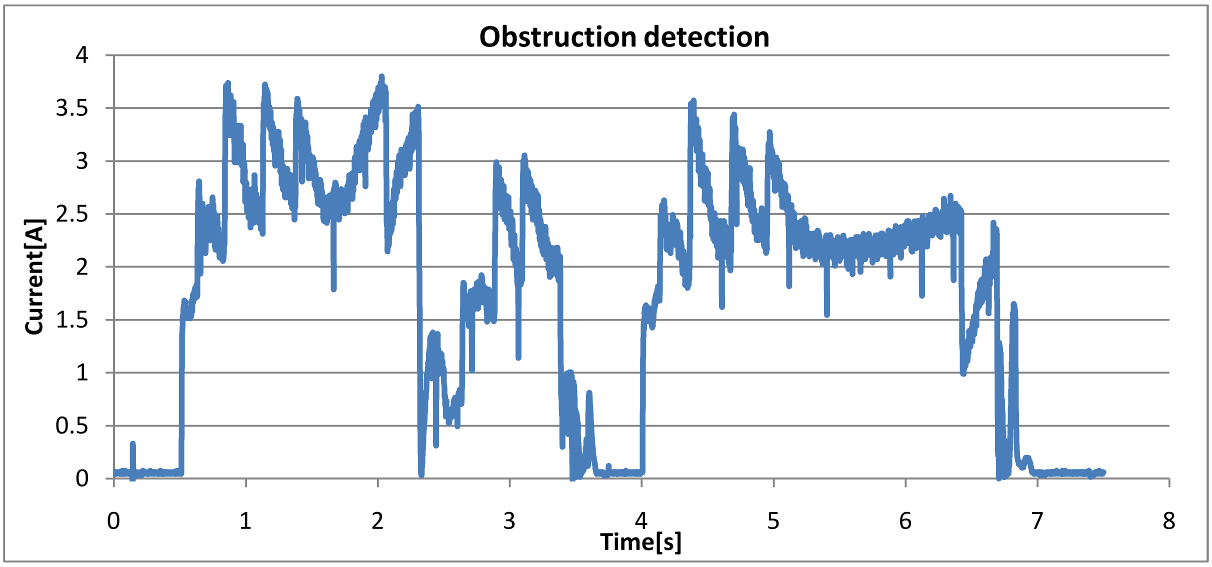


Figure 5-17. Current sense while obstruction detected.

When there is an obstacle preventing the door to be closed, the current drawn from the source may increase significantly (up to 6 Amps peak, although all current values above 3 Amps trigger the door to be stopped). As seen above, the controller reaction for obstacle detection is relatively short. When the current value grows above the second value set in the software, the obstacle is detected, the door is stopped and then reopened for a short distance (in figure 5-17 the current changes immediately after obstruction detection, the door opens with a lower speed than normal door closing profile). Then between the 3rd and 4th second the controller gives time for the obstruction to be removed (the most common being a passenger in between the door panels). After the time passes (timer1 count in the software program) the controller attempts to close the door again. Figure 5-17 shows the occurrence of a single obstruction and then the door is fully closed (4th to 7th second) after the way was cleared.

Voltage changes

The next step was simulation of voltage changes using a regulated power supply. This was simulated through changing the power supply voltage by approximately 20% up and down from the nominal value (32 V). This was done after disabling implemented voltage regulation procedures, as with them active voltage change cannot be well observed.

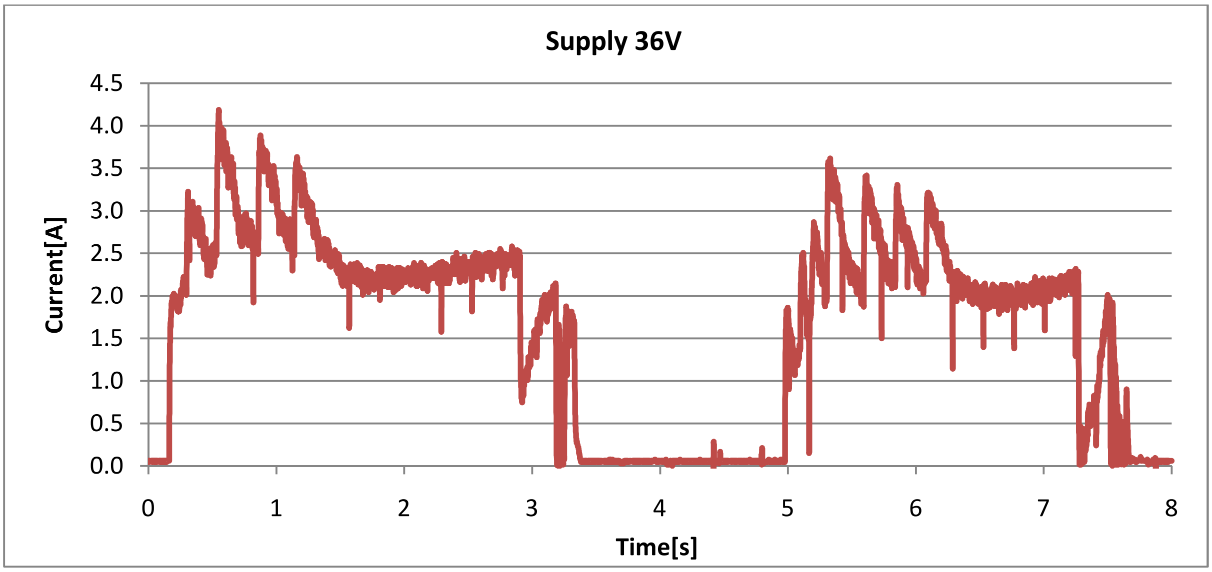


Figure 5-18. Current sense for 36 V power supply.

There are two profiles shown in figure 5-18, the first is for the door closing and the following for the door opening. It can be clearly seen that increasing supply voltage decreases door opening time (2.7 s compared to 3 s previously) /closing time (3.2 s compared to 3.6 s previously). Velocity therefore increases and the current drawn while the motor starts also increases due to the higher speed that now has been reached. Enabling the PWM pulse width dynamic change software implementation eliminates all the changes occurred in this profile. When the voltage regulation is enabled and the voltage level changes more than 10%, a message is displayed on the LCD and the door movement velocity is kept constant.

For a lower power supply, the door opening / closing time rises and the velocity decreases. The motor also draws less current.

Loose door panel detection

To test detection of lateral movement of the panels the panel fixing has to be loosened. This test is realised based on the measurements from the infrared distance sensor. The panel lateral movement is detected if the distance between the sensor and the panel changes during the door opening / closing procedure. The measurement result is correct if it falls within a declared algorithm threshold. The average value is calculated and after the door movement finishes (all samples are collected) it is checked to see if it falls within the desired range. This procedure detects not only the panel movement but also dirt or any objects in the way of the rollers which cause the door panel to move away from its normal route (or when rollers are wearied the panel is nearer the sensor). The algorithm functionality was tested by moving (approximately 3 mm) away the infrared sensor from the door panel. Doing that causes an error message to appear on the LCD display.

All other fault detection is based on the data analysis and cannot be illustrated as characteristics. For example, the push button failure was tested simply by disconnecting the PWM signal from the H-Bridge and connecting it to the ground. After the time passes for the door to be opened, the sensor readings and the door position are checked. If the door has not moved (sensors and QEI registers return the same value) a message is displayed on the LCD.

When the door movement starts, the timer starts to measure the opening / closing time. If the voltage supply is lowered and the velocity drops down a message is shown on the LCD display.

5. 4. Summary

Implemented fault tolerant algorithms were discussed and approach to fault detection explained. The controller operation in normal and faulty condition was tested. Based on data collected its functionality commented. Based on the tests pursued it can be concluded that the system requirements stated in section 3.4 of Chapter 3 are realised and fault tolerance defined in section 4.3 of Chapter 4 is implemented successfully.

The designed controller:

- Is able to work with the supply up to 52 V (up to 40 V was tested) ;
- Sustains current up to 6 A peak and 3 A for continuous operation;
- Performs the door closing and opening operation within required time and the time is user programmable;
- Shows status and report faults on the LCD display;
- Moves the door with speed not exceeding 1.5 m/s;
- Minimises acoustic noise from moving panels through software procedures;
- Detects obstruction or reports a fault within milliseconds;
- Levels supply voltage changes (within limited range only).

Some of the implemented fault tolerance was not tested such as PWM signal failure or loose panel detection. Also voltage changing function does not always calculate average voltage correctly. This is caused due to too low voltage sense output sampling rate.

Chapter 6. The development of an analytical approach to fault detection

6. 1. Introduction

In this chapter stages leading to a model development are introduced. As was mentioned already in Chapter 2 the model development is a first step to make possible to realise analytical fault detection. Based on the tests made in the previous chapter and a few additional experiments described below the motor model was derived and simulated in Matlab software. The simulation results and literature review related to the motor model development are briefly presented in subsections below.

6. 2. Motor friction

To start the door panels movement the motor has to overcome force. This force is a friction force. It is essential to understand how the friction affects the control process as it determines its behaviour. Therefore, it is a very important parameter for the motor model development. The friction has three components (Brdys, 2009) that can be seen on the figure below:

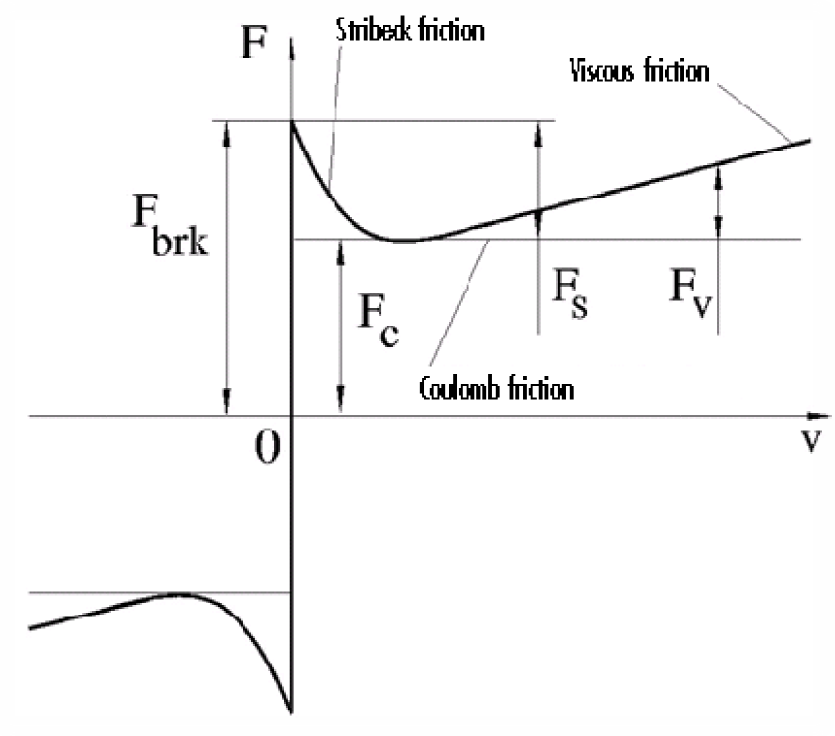


Figure 6-1. Friction components.

Where:

F_c – Coulomb friction,

F_s – Stribeck friction,

F_v – viscous friction,

F_{brk} – breakaway friction (sum of Stribeck and Coulomb friction)

Stribeck friction (stiction force or static friction) occurs only when the motor starts at slow velocities. Later on for higher velocities only Coulomb and viscous friction occurs. The Coulomb friction is constant and does not depend on velocity. The viscous friction is directed opposite to the motor movement and it is proportional to the velocity.

Total friction force in a steady state is given by the equation:

$$F(v) = (F_c + (F_{brk} - F_c) \cdot e^{-|\left(\frac{v}{V_s}\right)|^2}) + F_v \cdot v$$

Where: v is the motor velocity, V_s is the Stribeck velocity,

F_c – Coulomb friction, F_v – viscous friction, F_{brk} – breakaway friction.

Static friction only occurs when the motor is starting to move, therefore is not modelled as it only affects the control process unnoticeable.

6. 3. Motor parameters calculation and analysis

The subject of the modelling is a DC motor with permanent magnets. The model is based on measurements made and tests carried out in the laboratory. To acquire the motor model parameters some of the tests should be done with the door panels removed (no load). Therefore, they were not pursued. Necessary results not acquired from tests are taken from Dr HPB Dassanayake's thesis (ref. Chapter 3, section 3.7).

The measured motor parameters with no load are:

Resistance: R_a=2.2 Ω.

Nominal current: I_a=1.8 A.

A simplified mathematical model of the DC motor is described by following equations:

Motor voltage: $v_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K_e \omega(t)$

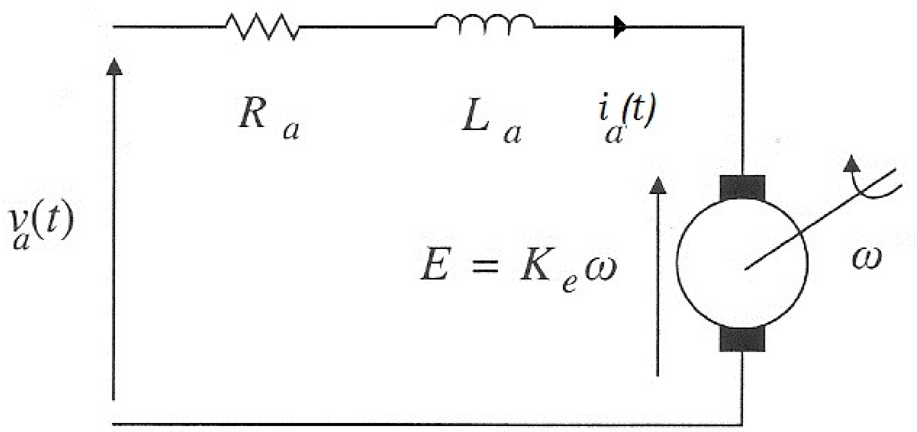


Figure 6-2. Model of a DC motor.

Where:

R_a, L_a are armature resistance and inductance respectively;

ω is the motor angular velocity;

K_e is the back e.m.f. constant (electro motive force constant);

v_a, i_a are armature voltage and current respectively.

Torque: $$T_{em} = T_2 + T_0 + J\frac{d\omega(t)}{dt}$$

Where:

T_{em}, T_2, T_0 are driving, load and no load torque respectively;

J is the machine rotational moment of inertia.

In the DC motor there is also generated magnetic field from the motor magnets (Wang et al., 2009). As the motor used in the control process is a permanent magnet DC motor the field generated is constant.

Equation parameters calculation

In each of the tests done the motor was powered from a 32 V DC power supply controlled by a 9.6 kHz PWM signal (75% / 87.5% duty cycle). This should give a 24 V voltage on the motor windings for door closing and 28 V for the door opening operation if there are no voltage losses within the H-Bridge. The actual voltage measured on the motor windings is 21.9 V for the door closing and 26.1 V for the door opening (measurement includes losses in the system). The nominal motor voltage is 24 V and the current 2 A (ref. manufacturer's note at speed ~720 RPM). For these values the motor velocity is approximately 75 rad/s @ 24 V and 86 rad/s @ 28 V (~840 RPM) with no load. At these parameters the motor has ~70% efficiency.

The measured current value with the door panels installed is 2.13 A for the door opening and 2.27 A for the door closing profile with the actual measured voltage levels mentioned above.

Armature resistance, R_a, can be derived based on the current measurement across a resistor connected in series with the motor. Based on the manufacturer's technical notes $R_a \approx 1.67\ \Omega$ for a cold motor (room temperature) and $R_a \approx 1.9\ \Omega$ for a hot motor (40-60ºC). Measured resistance is a bit higher and equals $R_a \approx 2.2\ \Omega$ due to the brush effect (drop of voltage between brushes and a copper commutator surface as described in Kenjo et al. (1985). As the motor is not in constant use it can be assumed that $R_a \approx 1.7\ \Omega$ (works in temperature slightly higher than room temperature).

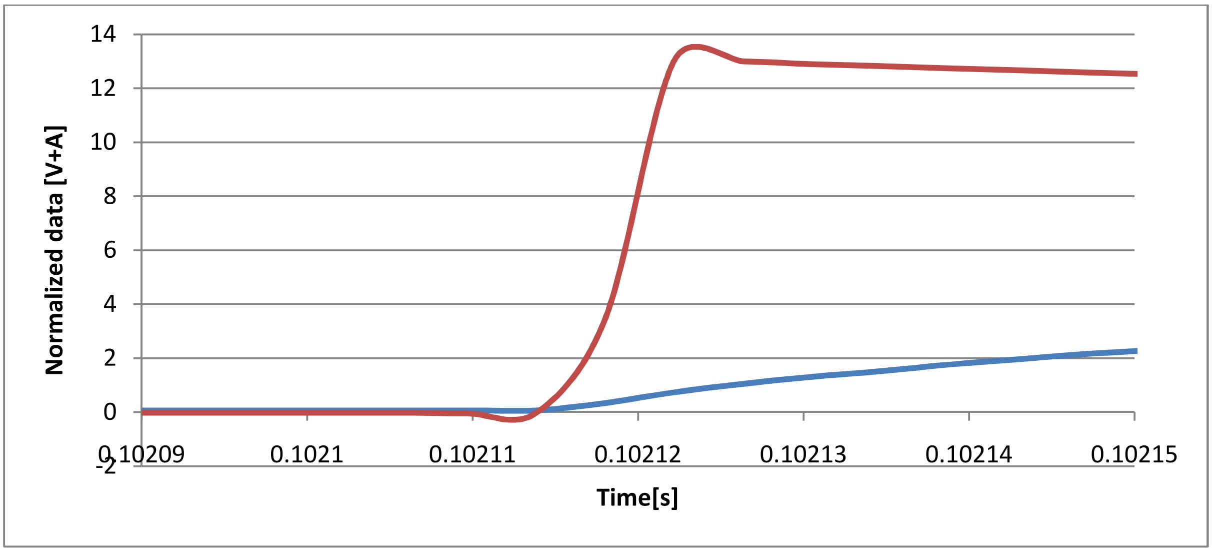


Figure 6-3. Voltage and current characteristics when motor starts with load.

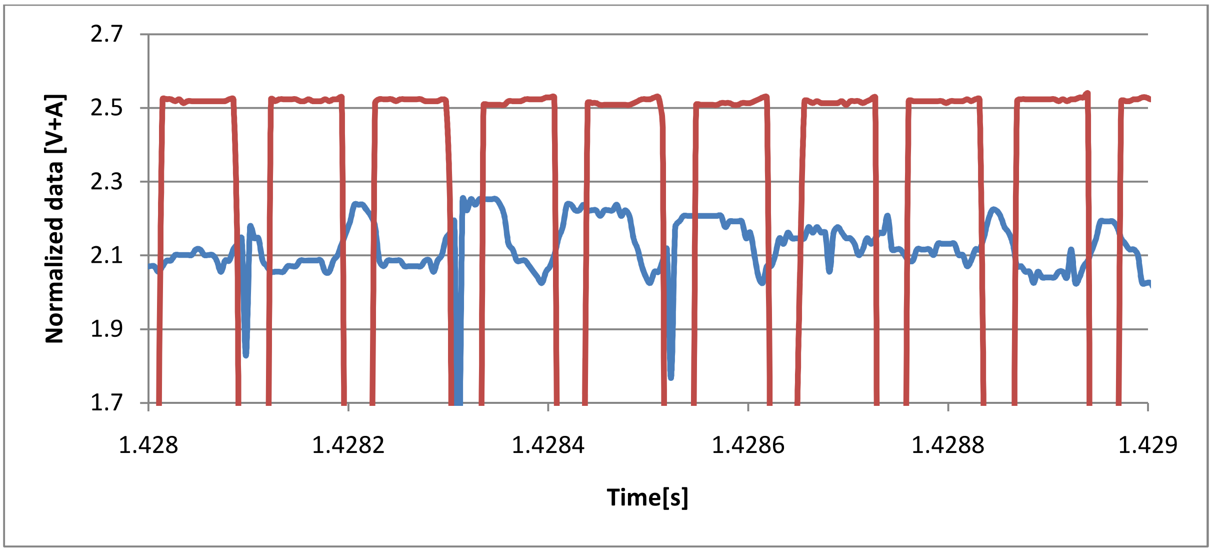


Figure 6-4. Current ripple @ 9.6 kHz PWM frequency.

From the plot in figure 6-4, which represents part of the door closing profile, a current ripple occurring for each PWM pulse can be read. The approximate value of the ripple is $\Delta i \approx 250$ mA. This is quite high value, which therefore makes fault detection based on the current analysis more difficult (wider thresholds have to be defined). The ripple should therefore be decreased by further increase of the PWM frequency (based on tests done with a lower PWM frequencies, for example, at 1.2kHz the current ripple was nearly 1A [ref. figure 5-1 in Chapter 5], a 9.6kHz frequency decreased the ripple to 0.25A).

Armature inductance, L_a, was calculated from the relationship between the motor time constant and armature resistance: $\tau_e = {}^{L_a}/_{R_a}$

This constant defines time when the motor reaches 63% of the steady state current (when the current reaches 63% of 2 A, that gives a 1.2 A value after a 13 V voltage step). The value of the constant can be read straight from the figure showing current rise over time (as figure 6-4 but with no load). However, this test was not pursued. The τ_e was then estimated, based on information from Dr HPB Dassanayake's thesis (2001).

For voltage step 13 V $\tau_e \approx 2.5\ ms$. Inductance $L_a \approx 17$ mH.

Knowing armature resistance R_a, in a steady state the relationship characteristics between voltage and velocity, based on the measurements for specific voltage levels, can be deducted. The electro motive force constant K_e is a received characteristics gradient. The voltage is given by the equation below and transformed.

$$V_a = R_a I_a + K_e \omega \;\; \gg \;\; K_e = (V_a - R_a I_a)/\omega$$

Therefore $K_e \approx 0.28$ Vs/rad at a steady state for nominal values (estimated). All the parameters in the first equation describing the motor voltage are now known for the modelling.

For the purpose of the model development dynamic friction values are assumed to be as in Dr HPB Dassanayake thesis (2001), (ref. p. 67 table) where in normal operation:

$\mathrm{F_c} = 0.73\ \mathrm{Nm}, \mathrm{F_v} = 0.002\,\frac{\mathrm{Nm}}{\mathrm{rad}/\mathrm{s}}$ for door closing profile,

$\mathrm{F_c} = 0.66\ \mathrm{Nm}, \mathrm{F_v} = 0.0019\,\frac{\mathrm{Nm}}{\mathrm{rad}/\mathrm{s}}$ for door opening profile.

Inertia calculation

Inertia is related to the sum of all the friction forces working within the motor and the door panels and is complex to calculate. It is defined as the time that the motor needs to move from a standstill state (to overcome friction forces and starts rotating) once given a voltage step. Based on estimates made by Dr HPB Dassanayake (2001) for various loading conditions, inertia with two door panels attached equals (in seconds):

Door opening – 0.0046 s,

Door closing – 0.0045 s.

As the motor is a permanent magnet DC motor, a magnetic field is fixed and the motor voltage and torque equations can be written as follows:

$$\frac{di_a(t)}{dt} = -\frac{R_a}{L_a} i_a(t) - \frac{K_e}{L_a}\omega(t) + \frac{1}{L_a} v_a(t)$$

$$\frac{d\omega(t)}{dt} = \frac{K_T}{J} i_a(t) - \frac{1}{J} B_f \omega(t) - T_{frc}$$

Where $T_0 = B_f \omega(t), B_f$ - friction coefficient and T_{frc} is a frictional torque

K_T electrical torque constant ≈ 0.262 [Nm/A] (referred from manufacturer's note).

$T_{em} = T_2 + T_0 = K_T I_a$ for a steady state.

Further transforming the above equations:

$$sI_a(s) = -\frac{R_a}{L_a} I_a(s) - \frac{K_e}{L_a} \Omega(s) + \frac{1}{L_a} V_a(s)$$

$$s\Omega(s) = \frac{K_T}{J} i_a(s) - \frac{1}{J} B_f \Omega(s) - T_{frc}(s)$$

Now model transfer functions can be derived from these equations and the motor can then be modelled.

6. 4. Model of the motor

Based on the calculated motor parameters and characteristics from the previous subsection, the model was derived and implemented in a Matlab Simulink. The designed model has the following structure:

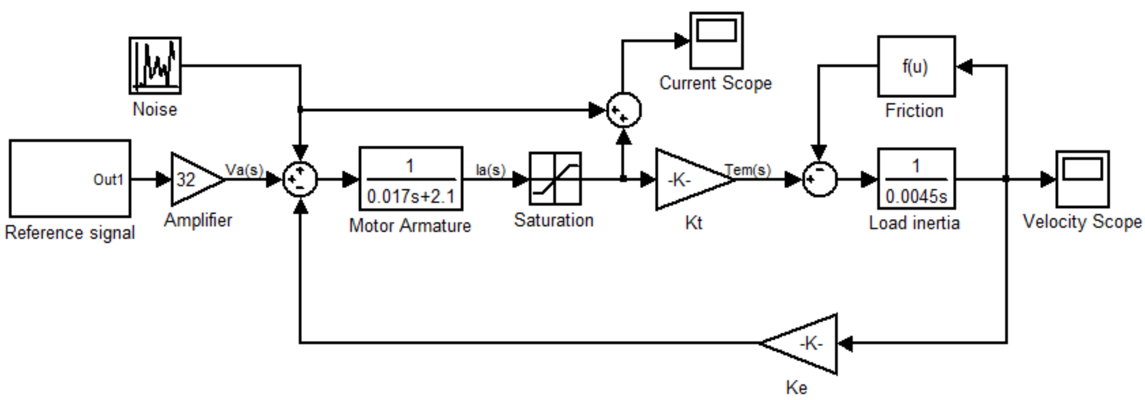


Figure 6-5. Simplified motor model.

Figure 6-5 shows a simplified model. In a fully modelled controller for normal conditions there are present signals supplying feedback. Therefore, no fault tolerance was implemented. The above model simulates the door closing operation only. For the door opening simulation model parameters have to be changed respectively. Pulse width of generated PWM reference signal increases gradually, as in the simulated original object (where the PWM duty cycle increases in five steps). The controller model generates the PWM signal with an amplitude of 1 V and then the signal is amplified 32 times to simulate a 32 V power supply. Maximum PWM pulse width reached is 75%. Using the motor parameters, which had been previously derived the controller response was simulated.

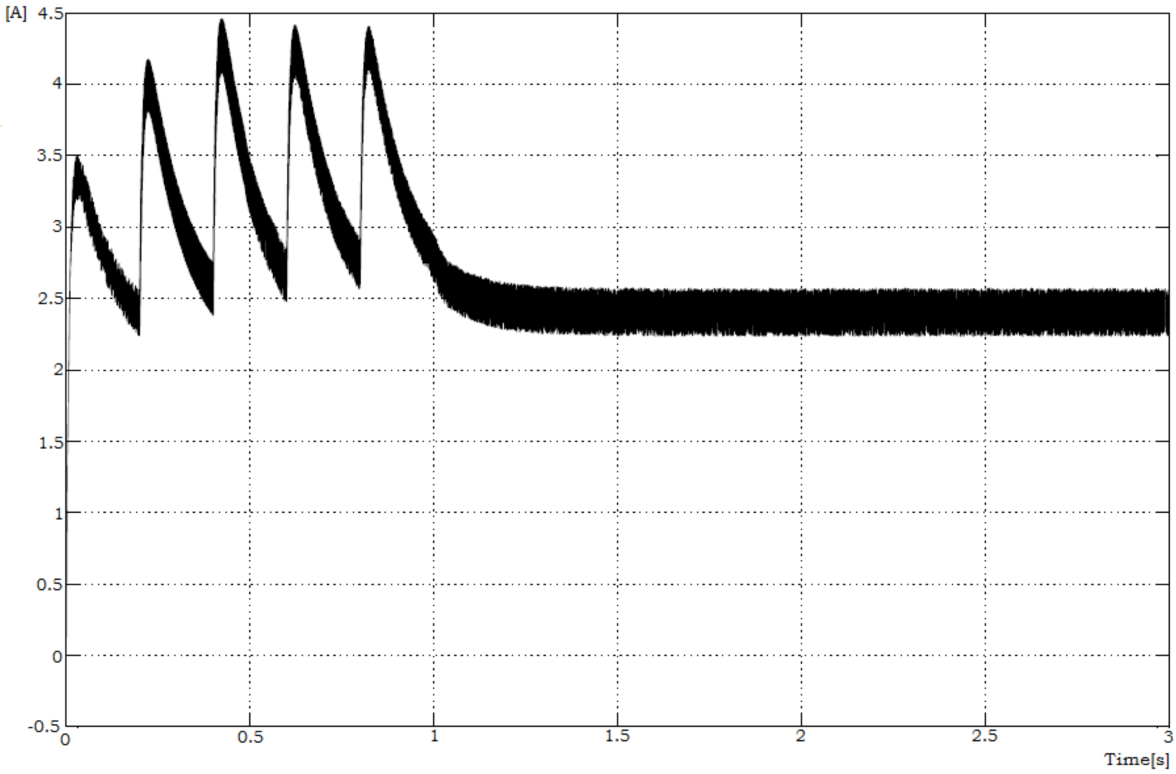


Figure 6-6. Modelled motor current sense output.

The motor current characteristic shown in figure 6-6 is similar to those seen during measurement tests for normal motor operations [figure 5-3]. Differences have arisen because there are no voltage losses implemented either for the H-Bridge or the motor or all noise sources. Therefore the modelled controller block gives a higher supply voltage to the modelled motor, causing a higher current to be drawn and also a higher velocity on the output. The PWM signal is generated continuously.

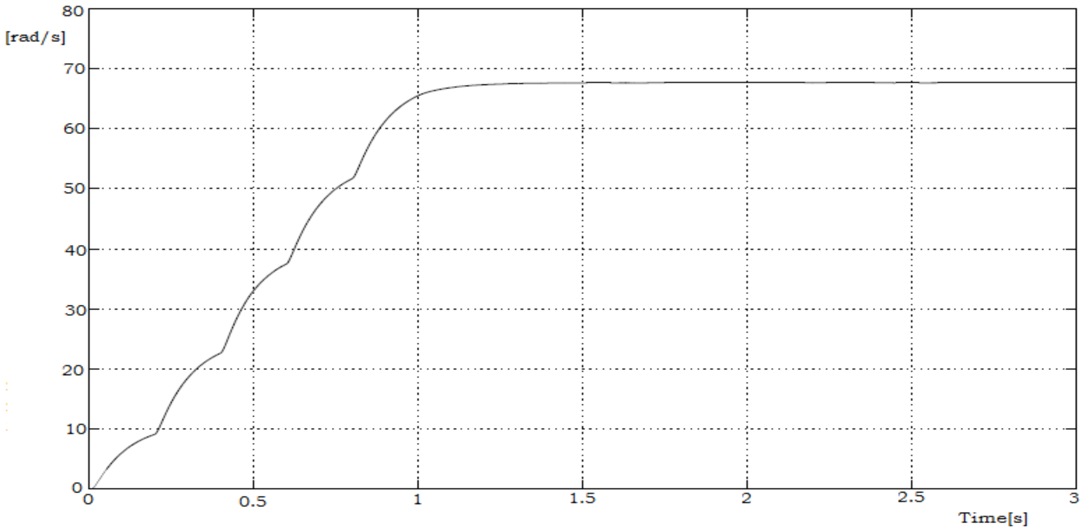


Figure 6-7. Modelled motor velocity.

The motor velocity rises gradually as the PWM duty cycle is increased every 0.2 s. The motor reaches speed at approximately 68 rad/s with implemented motor friction.

On the figure 6-8 is response of the analytical model where as input is used data collected and analysed in previous chapter.

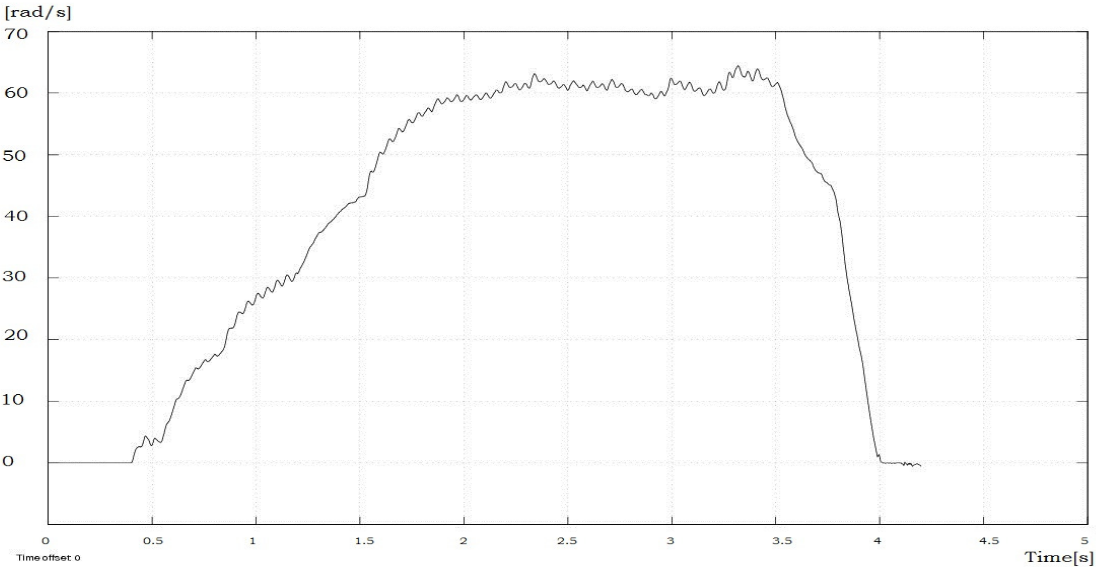


Figure 6-8. Motor velocity based on data collected.

6. 5. Analytical fault diagnosis approach

After developing the model of the system an analytical fault diagnosis can be implemented. As was mentioned already in Chapter 2, the analytical fault diagnosis is based on the information provided from the monitored device. Using mathematical methods, the collected data is then analysed. Based on the data, healthy and simulated faulty model profiles can be derived. This approach enables analytical models to be developed based on mathematical data analysis with the implementation of various fault detection algorithms. Depending on the process requirements right analytical method is selected. To derive the model as close as possible to the original object data has to be sampled with frequencies twice as high as the monitored signal frequency to allow the reconstruction of the exact monitored signals structures. In addition, if data is collected from the analysed object in lots of various working conditions, a better model can be derived and then simulated. By analysing simulation results and model behaviour, the most optimal analytical fault detection methods can be selected. The resultant algorithms can later be implemented into the motor controller, improving fault detection and diagnosis through software and some hardware modification if any additional feedback signals are required.

Model based diagnosis gives a powerful tool to simulate object (control system) behaviour to improve physical system development. The model outputs can be compared to the object output and if any differences occur in the output behaviour in normal conditions, further diagnosis can be conducted and optimisation made. The model based diagnosis lead also to a more detailed fault identification.

6. 6. Proposed software controllers

Proportional regulation

The source program already has some simple software controller implementation. This is a proportional (P) regulator that adjusts the PWM duty cycle (ref. Chapter 5) depending on the measured voltage level on the motor windings over the control process. Implementation of the controller of this type is sufficient to assure smooth door operation.

As mentioned in Chapter 4, a software PI/PID regulator could be implemented to minimise: current overshoot, time needed by the motor to increase the velocity to desired level and noise. By minimising current overshoot regulator would prevent the circuit from damage and prolong its durability as it will be working with smaller current values. This software controller would have to assure smooth door speed increase / decrease and noise of the door operation minimum possible. It would have to consider the current level (which is dependent on the voltage while the motor starts) and also the regulation time, not exceeding the maximum allowed as defined by the door opening / closing time, as minimising the current means that the motor velocity raises more slowly. Therefore, based on the monitored current sense output and fed back motor velocity signal, a regulator gain would be derived giving the best compromise to ensure a small overshoot and a reasonably short period of velocity rise (note that each PWM duty cycle change requires a specified minimum time [the microcontroller instruction execution time and H-Bridge transistors delay]).

Takagi-Sugeno reasoning

Following the description of the TS logic in Chapter 2 there can be a few situations distinguished in the designed controller where it can be used. Based on figure 5-3, within the measured current values, three periods of time can be distinguished. Therefore, to monitor overcurrent adequately for proper fault detection, a separate controller could be designed for each period. The first would work when the motor starts until it reaches a maximum desired velocity (current rising). Then, while the velocity is constant, another regulator would be enabled as the current has a constant value over this period of time. The third regulation would be enabled when the motor is slowing down, when the current rises and then drops down. For this approach, it would be appropriate to implement a fuzzy logic multi-regional controller that allows smooth switching from one state to the other by changing a local regional linear controller, depending on identification of the process operating region. The correct regulator is therefore chosen based on the current sense signal monitoring.

This logic can also be used for the classification of potential faults detected based on the friction force changes. Four levels of friction changes can be differentiated: normal; slightly increased (i.e. wear of the rollers); highly increased (obstruction detected); or lower friction level (door removed).

6. 7. Summary

The approach to the system model development was introduced. The simplified model was derived and simulated. Simulation results acknowledge the model regularity. The model can be further developed with implementation of fault detection existing in the designed controller software algorithm. The analytical fault detection methods can be implemented to further improve the designed controller fault tolerance and reliability. First step could be to implement the algorithms realising Takagi Sugeno reasoning described in previous subsection or simulate PI/PID controller to derive its most accurate parameters for developed door controller.

Chapter 7. Summary

7. 1. Conclusions

Based on the research and the information which was gathered, the most common faults within the electric door control were identified. These faults are the result of poor maintenance and cause friction to increase. Monitoring any friction changes are therefore very important in the process of door control. This was the most important requirement for the controller which was designed. Overall controller functionality was described in detail in previous chapters. The controller is successfully realising implemented tasks. All algorithms responsible for fault detection are working properly as desired, increasing passenger safety and system reliability. The controller was made with some minor design mistakes; however, overall it is fulfilling the specified system requirements through the implementation of additional software functions (the design mistakes were corrected by the software program limiting the current drawn by the motor). Tests on the controller were carried out in normal working conditions for the door and in various simulated faulty conditions. Based on the acquired data analysis, it is functioning as expected. All of the targets set for the controller during this research were also successfully realised. The controller is a simpler and cost effective version of the VR controller (i.e. less components used, one of the VR solution components [LM629] costs as much as most of the components of the designed controller), giving similar functionality for the test rig located in the lab. Based on the acquired data, motor model parameters were derived, however, measurements were limited only to tests with a load (both door panels attached to the drive screw of the motor). All accessible feedback signals were correctly monitored by the microcontroller and it gave enough information to carry out implemented fault detection. These signals allow the detection of any current or velocity changes which have occurred as a result of an increase in friction forces. The main dynamic feedback signals for this purpose are provided from the motor optical encoder and the H-Bridge current sense output. Many faults, including the most commonly researched faults, can therefore be detected as they cause friction to change. The controller can identify a group of possible faults. Fault occurrence is indicated by a message on the LCD screen. The information given does not indicate the specific fault which has occurred, but it indicates which measured value is not within its normal range, as set in the software program (threshold values are set as a result of tests). The fault which is detected can be further investigated within the system by implementation of one of the earlier mentioned model based fault diagnosis methods. After a closer study of the fault diagnosis methods, the right one can be chosen depending on the system requirements, as some methods provide tools to diagnose physical faults, others are handy to provide fault isolation and others are useful for the control of linear / non linear processes. All of these methods are model based and after simulations / computations / data analysis have been carried out, simple software solutions can later be implemented into the physical control system.

The system which has been designed operates on part of a real size train. All tests and experiments were carried out on a system which closely resembles a commercial door system, i.e. used in London Underground rolling stock.

7. 2. Limitations

As mentioned above, the controller is a simpler version of a VR controller. Implementing simplicity puts limitations on the controller. The specification defined in the second chapter limits the usage of the controller to work only with the laboratory test rig. Therefore it is not as easily adaptable as the VR controller to different conditions. It can only control a train door if its control method is similar to that used in the lab.

In the lab environment, all faults which can be detected by the implemented algorithms were simulated, except those which occur very rarely (i.e. motor overheating, wear of the motor brushes, PWM signal failure [microcontroller failure]). Due to the limited time available for testing faults related to components, wear could not be simulated either (wear of the rollers or ball bearings). In the lab there is only part of a train available, with one set of doors. Therefore the control system is not part of a distributed system as in existing trains; it is only a standalone module. There is no communication with the train operator or the other distributed system parts involved in the project. Therefore any possible data transfer issues are also excluded from the tests. The requirements regarding transmission protocol to communicate with other system parts are also not known. Therefore with the choice of the microcontroller transmission protocols were not considered as the controller was not designed to be part of a distributed system. Additionally, in the laboratory environment it is not possible to simulate passenger – door interaction or real environmental effects which can occur during the use of the rolling stock.

The controller which was designed has a poorly chosen H-Bridge. The continuous current drawn can be maximum of 3 A and 6 A for current peaks. When an obstruction or a fault is detected which causes very high overcurrent for long periods of time the circuit can be damaged or it can cause the H-Bridge to overheat and shut down. This situation can occur if no precautions are taken, as the motor can draw up to 10 A while it starts or if friction greatly increases (i.e. an obstruction that prevents the door panels from moving). Some software steps were therefore undertaken to limit overcurrent and, if it occurs, stop the operation of the motor. The current analysis starts when it stabilises at approximately 2.2 A and the motor velocity is constant, as in the software only one threshold level is defined. To enable current analysis when the door movement starts, there have to be at least 2 different defined thresholds that would then be swapped over time (ref. Chapter 5 Takagi Sugeno reasoning example). Based on the observation of figure 5-3 characteristics current stabilises at ~2.3A when the door is being closed to rise to higher level at the end of the door movement. Therefore when this rise occurs the threshold values should be adjusted respectively to not detect any non existing fault as the current increase at this stage is a normal condition. There is also only one set of thresholds defined for the door opening and closing operation. As there is already a 0.2 A current difference between those profiles, a greater friction increase triggers fault detection while the door is opening.

Another limitation is a lack of a logging system where possible faults or error messages could be stored for later view. The only communication with the user or maintenance technician is through the LCD display. The display which is used to show monitored signals is not big enough. Therefore not all signals can be displayed at the same time. Changing them is only possible through software modification. In addition, the controller uses too much of its operational time in displaying messages on the LCD compared to the code execution time of the remaining part of the software. Displaying many messages slows down the fault detection time and also causes the analogue to digital converter to miss monitored signal samples. However, as part of the distributed system this limitation would no longer be a disadvantage because the LCD display would be removed and potential error messages sent through a transmission protocol to the main device.

A supply voltage, where PWM duty cycle is set near to 100% (for 32V is set already to 75%), results in a limited range of voltage correcting functions operation. For example if the supply voltage is too low a situation can occur when increasing PWM pulse width to maximum still results in too slow door movement. The controller is able to work up to a

55 V DC supply, therefore a laboratory power supply limits the operation range of voltage functions. In case of a power supply fault, the best regulation range would be reached when the PWM duty cycle is set at 50% for the door closing operation in normal working conditions, which requires a 48 V power supply. These functions would not be used often in normal conditions, as only small voltage changes occur in the power sources of a train. Therefore it is sufficient to have an adjustment of only ±5 V to maintain a constant door velocity. The voltage correction functionality could also be implemented, together with a friction detection algorithm. If a small increase in friction is detected, which causes the door speed to drop down a little, by increasing the given voltage to the motor windings the velocity could still be kept constant. To adjust the voltage within a small range, floating point arithmetic has to be used to calculate a new PWM duty cycle based on the measurements. The chosen microcontroller is insufficient to carry out that function. In a limited way, this function can be implemented as a table into the memory of the microcontroller.

The microcontroller which was used does not have enough computational power to implement within it more complex analytical fault detection methods and acquired data filtering algorithms. The data is currently only filtered by very simple FIR filter implementation. It would be more useful for the door control process to have a floating point capable microcontroller or a DSP to get a higher degree of accuracy of the performed calculations and analysis.

An optical encoder working index output, attached to the motor, would make analysis of the door velocity profile and detection of any changes easier. Wear of the rollers could be detected not only through increased current but also with the use of an additional distance sensor, indicating a gap decrease in between the shaft assembly and the door panels. This sensor could also detect any debris accumulation due to poor maintenance, which could also cause the door to move away from a chassis or loose panel.

Due to a low PWM frequency, the measured signals have quite a high ripple, causing their monitoring to be more difficult, as the defined threshold must have a wider range so as not to execute any fault detection algorithm accidentally. Therefore increasing the PWM frequency improves control and the stability of feedback signals.

Most of the limitations mentioned above do limit the use of the designed control system to a single door panel operation, although they do not interfere with its functionality. To make this control system suitable for commercial use, it has to be redesigned, although it can be based on the same or slightly modified components. The changes would mainly concern implementation of the logging system and the use of the LCD display, if still required.

7. 3. Further work

At the current development stage, the controller performs the door movement operations and displays messages on the LCD regarding any potentially detected fault or error. The controller does not give enough information regarding the fault which has occurred, therefore fault identification needs improvement so that faults can be properly classified for passenger safety and further train use.

Improvements can be made either to the software (altering the code) or hardware (i.e. adding some sensors) depending on the approach required, as changes in both fields are possible. The controller was designed for simple door control. It undertakes given tasks successfully, therefore any improvements proposed would make it a more sophisticated device.

Proposed hardware improvements

If in commercial use, the controller would no longer require an LCD display (or it could be used to show messages for passengers only and not for diagnosis). Communication could be carried out through a data transmission protocol with another part of the distributed system. Transmission protocols should be investigated to avoid the situation which currently occurs in the microcontroller, whereby displaying the message takes longer than the execution of other parts of the software code. This will also improve fault analysis as the sample rate would be increased and samples not missed.

Moving from the current microcontroller to DSP would be advantageous as mathematical operations and estimations made over the door movement process could be completed more quickly and the samples collected could be filtered from noise at the same time. Floating point operations for the estimation of parameters could be very useful, as PIC does not operate very well on the float type. By increasing the PWM frequency, the ripple of measured feedback signals would be decreased, however, the sampling rate would then have to be increased. The maximum sampling rate for the chosen microcontroller is approximately 50 ksps at 40 MHz clock. Therefore 25 kHz is the limit for the PWM frequency. For higher PWM frequencies, a faster external ADC has to be used. A higher clock rate requires the use of an external oscillator for internal timer modules to measure the door opening / closing time, as with a speed of 20 MHz microcontroller clock the maximum measured time is already slightly shorter than the normal closing time.

For commercial use, infrared sensors would have to be replaced by induction sensors or any other more tolerant to work in heavy duty environment (more resistant to dirt). This would be particularly important for those sensors that are located at the bottom of the door panel, where they are more exposed to dirt accumulation and therefore unable to carry out proper detection. The number of sensors could also be increased to four per panel, as at the current stage the two sensors which are installed only monitor the position of the right panel of the door. Four sensors would allow the detection of any door panel misalignment and position change. Installing additional sensors could also improve fault detection and allow better fault identification.

The H-Bridge which has been used appears to realise its task properly with software current limiting functions, although it does not allow the system behaviour to be tested without them when the motor draws current higher than 6 A. It cannot, therefore, be used for deriving some of the motor parameters needed later on to build the software motor model. In the case of a microcontroller failure (not the PWM output as it will be disabled by the controller if it occurs) there could be a situation where the motor could try to draw more power than the H-Bridge is able to sustain, causing the H-Bridge to be damaged or overheat. It would be useful to implement additional door control and motor braking in the case of an operational microcontroller also including an interface for a passenger to operate in emergency situations. This solution could use a 5 V power supply already accessible on the module board, decreased by a voltage divider to prevent door movement with maximum motor velocity.

Proposed software improvements

By implementing some hardware improvements the motor model could be updated and supplemented to include feedback signals. Then not only normal operation but also fault detection can be simulated. The software program should be optimised to acquire signal samples faster and decrease the code execution time for the LCD display. By increasing the PWM frequency, the threshold ranges can then be tightened and obstruction detection time shortened, minimising the risk of passenger injury. It would also be beneficial to implement a dynamic controller, as proposed in the previous chapter, to change the PWM duty cycle while the motor starts to assure a smoother current draw from the source. Now duty cycle changes are being done in a timely manner until the desired speed is reached. Better software data filtering would also result in quicker fault diagnosis and more samples taken for analysis would increase the accuracy of identification. The velocity profile should also be updated to decrease the motor speed over a longer distance only while the door is closing, as this is when most obstructions caused by passengers' bodies occur, to minimise the risk of passenger injury.

List of References

1. Dassanayake, H.P.B. (September 2001), '*Fault diagnosis of a new generation of intelligent train door systems*' thesis for the degree of Doctor of philosophy.
2. Calamatas, Philip J. (August 1997), *'Third Generation Mass Transit Door Controller'*, Preliminary Design Review. Vapor Rail.
3. Central Line Tube stock maintenance manual (1990), *'Vapor Doors and Door Control Systems'*, Issue 2.
4. Messina, P., Pacher, Ch., Sullivan, T. (2004), *'Rail Car Door System Analysis Project'*, United States.
5. WMATA (March 2004), *'Rolling Stock Equipment Technical Forum – Door project'*, Field questionnaire, Washington Metropolitan Area Transit Authority.
6. Kenjo, T., Nagamori, S. (1985), *'Permanent Magnet and Brushless DC motors'*, Clarendon press Oxford, UK.
7. Microchip (2003), *'PIC18F2331/2431/4331/4431 Data sheet'*. Available from: www.microchip.com [Accessed 15 October 2009].
8. Powertip (n.d.), '*PC 1602-L LCD Display Data sheet'*. Available from: www.powertipusa.com/pdf/pc1602l.pdf [Accessed 27 October 2009].
9. LEM (n.d.), *'Current Transducer HXS 10-NP/SP3 Data sheet'*. Available from: www.lem.com/docs/products/hxs10np_sp3_e.pdf [Accessed 2 November 2009].
10. National Semiconductor Corporation (2002), *'LMD18200 55V 3A Full H-Bridge Data sheet'*. Available from: www.national.com/pf/LM/LMD18200.html [Accessed 12 November 2009].
11. National Semiconductor Corporation (2004), *'LM2576HVT-5.0 Simple Switcher 3A step down voltage regulator Data sheet'*. Available from: www.national.com/pf/LM/LM2576HV.html [Accessed 20 November 2009].
12. Minnesota Electric Technology Inc. (n.d), *'3C-5203172C DC motor Data sheet'*. No more available, attached as Appendix C.
13. Brdys, M. (2009), *'Takagi, T. and Sugeno, M. reasoning', 'Friction modelling'* – Intelligent Systems lecture notes.
14. Wang, J., Hillmansen, S. (2009) *'DC motors'* - Electrical Energy Conversion Systems lecture notes.
15. Mathworks Inc. (n.d.), '*SIMULINK - Dynamic System Simulation for MATLAB'* User manual Version 3.
16. Mathworks Inc., MATLAB online help. Available from: www.mathworks.com/help/ [Accessed April 2010].
17. Train Door flew open at 80mph (2007). Online news. Available from: http://www.northamptonchron.co.uk/news/local/train_door_flew_open_at_80mph_1_916746 [Accessed 15 September 2010].
18. Door horror stories (July 2010). Available from: http://www.newsshopper.co.uk/news/8264350.BEXLEY__More_Southeastern_passengers_come_forward_with_door_horror_stories/ [Accessed 15 September 2010].
19. ISO/CD 10303-226, Failure definition according to International Standard Organisation (ISO), available online on ISO website.

20. Isermann, R., Schwarz, R., and Stölz, S. (2000). *'Fault tolerant drive-by-wire systems: concepts and realisations'*. Proceedings of the 4th IFAC symposium on Fault Detection, Supervision and Safety for Technical Process – SAFEPROCESS 2000, Budapest, Hungary.
21. Netten, B.D., (September 1997), *'Application Of Artificial Intelligence In Process Control'*. Presentation given at ERASMUS Intensive course, Valencia.
22. Abzug, M.J., Larrabbe, E.E. (2002) *'Airplane Stability and Control – second edition'*. Page 88-89.
23. Osder, S. (1999) *'Practical View of Redundancy Management Application and Theory'* in Journal of Guidance, Control, and Dynamics, Vol.22, No. 1.
24. Langford, J. W. (1995) *'Logistics: Principles and Applications'*, McGraw Hill, page 488.
25. Mikulak R.J., McDermott R., Beauregard M. (1996) *'The Basics of FMEA'*, (Fault Methods and Effects Analysis).
26. Tanaka, Sugeno, M. (1992) '*Fuzzy systems theory and its applications'*.
27. Patton, R.J., Chen, J. (1997) *'Fault detection, supervision and safety for technical processes',* SAFEPROCESS'97 : a proceedings volume from the 3rd IFAC symposium, Kingston Upon Hull, UK.
28. Isermann, R., (2000) *'Fault Diagnosis Systems – An Introduction from fault detection to fault tolerance'*. Available from online Google books: http://books.google.co.uk/books?id=6yUfoZhGMY0C&printsec=frontcover&dq=fault+diagnosis+isermann&hl=en&ei=qKfQTMjwFMqdOt7VnccF&sa=X&oi=book_result&ct=result&resnum=1&ved=0CDIQ6AEwAA#v=onepage&q&f=false [Accessed 5 October 2010].
29. Intelligent Fault Diagnosis (1997). Available from: www.zaptron.com/automation/diagnosis.htm [Accessed 10 October 2010].
30. Korba, P., Frank, P.M. (1997) '*A model-based fuzzy control: LMI-based approach'* University of Duisburg publication, Germany.
31. Mehran, K. (2008) *'Takagi-Sugeno Fuzzy Modelling for Process Control'*, Newcastle University.
32. IEEE 1332-1998, (1998), '*Standard Reliability Program for the Development and Production of Electronic Systems and Equipment'*, **E-ISBN:** 0-7381-1411-1.
33. Ding, S. (2008), *'Model based fault diagnosis techniques – Design Schemes, Algorithms and Tools'*, Springer Verlag Berlin, **ISBN**: 978-3-540-76303-1.
34. Frank (1990), *'Automatica'* – survey paper, referred from page 9 of Ding, S. (2008) 'Model based fault diagnosis techniques'.
35. Koski, T., Noble, J. (2008), *'Bayesian networks – An Introduction'*.
36. Train Door Solutions (n.d.), a door systems website, (TDS). Available from: www.traindoor.com/ [Accessed 06 June 2010].
37. Vapor Rail (n.d.) previously, now Wabtec door systems website, (VR). Available from: www.wabtecrail.co.uk/doorsystems.htm [Accessed 08 June 2010].
38. Siemens door systems website (n.d.). Available from: www.mobility.siemens.com [Accessed 08 June 2010].
39. Faiveley Transport (n.d.), manufacturer of railway equipment. Available from: www.faiveley.fr/uk/accueil.php [Accessed 09 September 2010].
40. Simatic PCS 7 Distributed Process Control System. Available from: https://www.automation.siemens.com/mcms/process-control-systems/en/distributed-control-system-simatic-pcs-7/Pages/distributed-control-system-simatic-pcs-7.aspx [Accessed 01 September 2010].

41. DSP LM629 (n.d.), Precision Motion Controller Data Sheet. Available from: www.national.com/pf/**LM**/**LM629**.html [Accessed 05 June 2010].
42. Satellite fault shuts train doors (2004), Southern trains. Available from: news.bbc.co.uk/1/hi/england/southern_counties/3602494.stm [Accessed 05 June 2010].
43. Train heat issues (n.d.). Available from: http://www.tubeprune.com/rollingstock.html #DoorSystems [Accessed 05 June 2010].
44. Scotrail class 380 trains (2010). Available from: www.scotrail.co.uk/newtrains [Accessed 15 June 2010].
45. PRM-TSI (2008) information. Available from: www.publications.parliament.uk/pa/ld200708/ldselect/ldmerit/119/11906.htm [Accessed 10 June 2010].
46. Scotsrail (2010), news cited. Available from: www.scotsrail.co.uk/news [Accessed 20 June 2010].
47. Databus (2010)bus connectors designed for railway industry. Information available from: www.machinebuilding.net/p/p3455.htm [Accessed 24 June 2010].
48. Tendulkar, P. (October 2006), 'How *Train Management System Works?*'.Available online from: trainweb.org/railworld/TMS2/ [Accessed 20 June 2010].
49. SDO (n.d.) Selective Door Operation system. Available from: www.railsigns.co.uk /sect21page4/sect21page4.html [Accessed 20 June 2010].
50. ReliaSoft Research (2007), *'System Analysis: Reliability, Availability and Optimization Online Reference'*, electronic book. Available from: www.weibull.com/systemrelwebcontents.htm [Accessed 12 October 2010].
51. Vapor Rail technical proposal TP95-15 (1995), *'Vapor linear overhead door operator and controls for application to Mexico cities metro rail transit cars'*. MRN 95032-1.
52. Peacock S. (1990), Victoria Line Tube Stock [1967] Refurbished Pneumatic Door System maintenance worksheets.

Appendix A: C source code

```
#include <p18F4431.h>
#include <delays.h>

// *** function prototypes ***
void nop();
void writestring(char * wskaznik);
void LCD_clear(void);
void InterruptVectorHigh(void);
void InterruptVectorLow(void);
void InterrruptHandHigh(void);
void change_smooth_dcycledownPWM1(void);
void change_smooth_dcycleupPWM1(void);
void checkIandVifOK(void);
void convertbyte(int result, char* str);
void convert(int result, char* str);
void lcd_position(unsigned char line, unsigned char character);
void movedooralittle(void);
void door_move(void);
void change_dutycyclePWM1(unsigned int newdutycycle);
void checkdoormovement(void);
//******* global variables/strings declaration
short arraycnt,arrcntv1,currenable;
//const short loopcnt=5; //always for one more sample - one is rejected (the biggest value)
short oldcurr[9]; //equals loopcnt
short oldvolt1[9];
const int pwmperioddown[5]={0x2bd,0x27b,0x239,0x1f7,0x1b5};//decreasing pwm for higher voltage
const int pwmperiodup[5]={0x333,0x366,0x399,0x3cc,0x3ff}; //incrreasing pwm for lower voltage
int positionflag,distance,overccount,movementflag, motorvoltage,newpwmperiod, motorbrakes;
int dooropensensor,i,x;
char* str; //to dispaly all strings
char dooropen[13]={'D','o','o','r',' ','O','p','e','n','i','n','g',0}; //for lcd while opening button will be pressed
char doorclose[13]={'D','o','o','r',' ','C','l','o','s','i','n','g',0};
char dooropened[13]={'D','o','o','r',' ',' ','O','p','e','n','e','d',0};
char doorclosed[13]={'D','o','o','r',' ',' ','C','l','o','s','e','d',0};
char welcome[6]={'H','e','l','l','o',0};
char warning[9]={'L','o','w',' ','v','o','l','t',0};
char warning2[9]={'h','i','g',' ','v','o','l','t',0};
char warningrolls[13]={'R','o','l','l','s',' ','w','e','a','r','e','d',0};
char error[13]={'D','o','o','r',' ','f','a','i','l','u','r','e',0};
char timeexceeded[16]={'T','o','o',' ','l','o','n','g',' ','o',' ','t','i','m','e',0};
char timeexceeded2[16]={'T','o','o',' ','l','o','n','g',' ','c',' ','t','i','m','e',0};
char buttonorpwmfail[14]={'B','u','t','t','o','n',' ','N','O','T',' ','w','.',0};
char misalignment[13]={'M','i','s','a','l','i','g','n','m','e','n','t',0};

// info: 20MHz clock gives 1 cycle every 50ns - 1 instruction executed in 50ns
#pragma config OSC = HS // automatically timer onwhich gives delay until EXt osc will be stable.
#pragma config FCMEN = OFF
#pragma config IESO = OFF
#pragma config BOREN = OFF
#pragma config WDTEN = OFF
#pragma config MCLRE = ON
#pragma config PWRTEN = ON
#pragma config STVREN = OFF
#pragma config LVP = OFF // low voltage programming off
#pragma config DEBUG = OFF
```

```
#pragma config CP0 = OFF   // memory blocks protection bits
#pragma config CP1 = OFF
#pragma config CP2 = OFF
#pragma config CPB = OFF
#pragma config CPD = OFF
#pragma config PWMPIN=ON
#pragma config LPOL=HIGH
#pragma config HPOL=HIGH
#pragma config FLTAMX=RD4

//interrupt vector table and handling
#pragma code InterruptVectorHigh = 0x08 //interrupt vector
void InterruptVectorHigh(void)
{
        _asm
        goto InterrruptHandHigh
        _endasm
}

#pragma code
#pragma interrupt InterrruptHandHigh

void InterrruptHandHigh(void)
{
int j;
//timer 0 interrupt
if (INTCONbits.TMR0IF)
{ //measure door movement time when measurement finished jumps here and set flag
//if this is done before door is closed then time is exceeded
T0CONbits.TMR0ON=0; //disable timer0
INTCONbits.TMR0IF=0; // clear interrupt flag

if (overccount==0)
                        {
                        LCD_clear();
                        lcd_position(0,0);
                        if (positionflag==0) writestring(timeexceeded);//time exceeded door opening
                        if (positionflag==1) writestring(timeexceeded2); //exceeded door closing time
                        checkdoormovement();
                        }
}
//ADC interrupt
if (PIR1bits.ADIF){
int temp;
oldcurr[arraycnt]=ADRESH; //only 8MSB of ADC conversion considered
if (currenable!=0)
{
arraycnt++;
if (arraycnt==9) arraycnt=0; //start overwriting buffer from the begining
i=0;
temp=0; //result that compared later is a sum of four samples from the array
for (j=0;j<9;j++)
        {
        i=oldcurr[j]+i; //calculate sum of all samples
        if (oldcurr[j]>temp) temp=oldcurr[j]; //catch the highest sample value
        }  //
i=i-temp;//reject highest sample from sum
i=i>>3; //divide by 8 related to (loopcnt-1) divider if changed in the future
//read voltage value on current sense from LMD18200
lcd_position(1,0);
```

```
convertbyte(i,str); //display average current
writestring(str);
//distance=POSCNTH; //display distance commented out to speed up conversion
//distance=distance<<8;
//distance=distance+POSCNTL; //to show on lcd
//convert(distance,str);
//writestring(str);
if (i>0x70)
	{ //overcurrent appeared stop everything at the moment later restart after some time
	//max is about 4.45V voltage level measured on ADC from LMD18200 current sense (377uA per AMP that gives 0.679 )
	//could be lower value but during first door move for some reason current is higher :(
	if (overccount==x) overccount=overccount+1; //meaning that previous overcurrent was serviced
	}
if (i>0x5f & i<=0x70) writestring(warningrolls); //rolls wearied and needs changing (friscion slightly increased)
	//works message on display and door still operational only message on LCD
} //end of current analysis
//write on LCD second conversion result from sensor that detects if door is closed
dooropensensor=ADRESH;
if (dooropensensor>0x23) //meaning that panel is pushed out of its normal route due to dirt or sth
{
lcd_position(0,0);
writestring(misalignment); //inform user about a fault
}

i=ADRESH; //door closing only analysis - otherwise negative potential
if (positionflag==1 & motorbrakes==0) //condition that door close button pressed
	{
	oldvolt1[arrcntv1]=i; //then copy to array
	arrcntv1++;
	if (arrcntv1==9) arrcntv1=0;
	i=0;
	temp=0;
	for (j=0;j<9;j++)
	{
		i=oldvolt1[j]+i; //calculate sum of all samples
		if (oldvolt1[j]>temp) temp=oldvolt1[j]; //catch the highest sample value
	} //
	i=i-temp;//reject highest sample from sum
	i=i>>3; //divide by 8 related to (loopcnt-1) divider if changed in the future
	convertbyte(i,str);
	writestring(str);

	if (i>0x88)
	{
		lcd_position(1,0);
		writestring(warning2); //voltage out of set range
		if (i<0x8c) newpwmperiod=pwmperioddown[0];//each threshold equals 2V power supply increase
		else if (i<0x95) newpwmperiod=pwmperioddown[1];
		else if (i<0x9e) newpwmperiod=pwmperioddown[2];
		else if (i<0xa7) newpwmperiod=pwmperioddown[3];

	}
	/*if ((i<0x7e) & (POSCNTH>0x80))
	{
	lcd_position(1,0);
	writestring(warning); //voltage out of set range (too low)
	if (i>0x79) newpwmperiod=pwmperiodup[0];		//add for faster door opening than closing
	else if (i>0x70) newpwmperiod=pwmperiodup[1];
```

```
		else if (i>0x67) newpwmperiod=pwmperiodup[2];
		} */
		}
//*******************************************************************************
i=ADRESH; //door opening only analysis - otherwise negative potential
if (positionflag==0 & motorbrakes==0) //condition that door open button pressed
		{ //can be used the same array because oother condition not fulfilled
		oldvolt1[arrcntv1]=i; //then copy to array
		arrcntv1++;
		if (arrcntv1==9) arrcntv1=0;
		i=0;
		temp=0;
		for (j=0;j<9;j++)
		{
			i=oldvolt1[j]+i; //calculate sum of all samples
			if (oldvolt1[j]>temp) temp=oldvolt1[j]; //catch the highest sample value
		}  //
		i=i-temp;//reject highest sample from sum
		i=i>>3; //divide by 4 related to (loopcnt-1) divider if changed in the future
		convertbyte(i,str);
		writestring(str);
		if ((i>0x88))
		{
			lcd_position(1,0);
			writestring(warning2); //voltage out of set range trying to set to the right velocity
			if (i<0x8c) newpwmperiod=pwmperioddown[0];	//add for faster door opening than closing
			else if (i<0x95) newpwmperiod=pwmperioddown[1];
			else if (i<0x9e) newpwmperiod=pwmperioddown[2];
			else if (i<0xa7) newpwmperiod=pwmperioddown[3]; //assign new pulse width for pwm
		}
		/*if ((i<0x7e) & (POSCNTH<0xa0))
		{
		lcd_position(1,0);
		writestring(warning); //voltage out of set range (too low)
		if (i>0x79) newpwmperiod=pwmperiodup[0];		//add for faster door opening than closing
		else if (i>0x70) newpwmperiod=pwmperiodup[1];
		else if (i>0x67) newpwmperiod=pwmperiodup[2];
		//else if (i>0x5e) newpwmperiod=pwmperiodup[3]; //assign new pulse width for pwm
		//else if (i<=0x67) newpwmperiod=0x37f; //maximum to not exceed pwm duty cycle in accelerating
functions
		} */
}

PIR1bits.ADIF=0; //clear interrupt flag
ADCON0bits.GO_DONE=1; //start next conversion
}
//triggering ADC conversion
if (PIR1bits.TMR2IF){
ADCON0bits.GO_DONE=1; //start conversion
T2CONbits.TMR2ON=0; //no more timer2 used
}
}
// end of interrupts handling

#define LCD_E_SET() PORTCbits.RC0=1;   // write enable
#define LCD_E_CLR() PORTCbits.RC0=0;   //write disable
#define LCD_RS_SET()PORTCbits.RC1=1;   //high write data
#define LCD_RS_CLR()PORTCbits.RC1=0;  //low write instruction

#pragma code
```

```
//init all required ports I/O and registers
void initpic(void)
{
//LCD port init - uses 6 i/o, 4 for data, 2 control
TRISD=0x00; // first 4 as output data for LCD , rest not used
TRISC=0x0C; // outputs only RC0 and RC1 used for LCD, low inputs for the pushbuttons [RC2 & RC3)
TRISA=0x3F; // ADC port ALL Inputs except pins 6,7
TRISE=0x07; // set port E RE0-2 as analogue inputs for ADC conversion
}
void nop() //do nothing for one cycle
{}

void initPWM(void)
{
PDC0L=0x00;//set to zero as no output - motor stopped
PDC0H=0x00;//0x2ff is 75% duty cycle
PWMCON0=0;
OVDCOND=0xff; //not necessary?
PTCON0bits.PTMOD1=1; //for continues mode
PTCON0bits.PTMOD0=1; // single shot mode //if both enabled continuous up/down mode
PWMCON0bits.PWMEN2=0; //pwm1 and  3 enabled
PWMCON0bits.PWMEN1=1;
PWMCON0bits.PWMEN0=1; //on reset is 1 so it has to be changed to 0 to disable other PWMs
PWMCON0bits.PMOD1=1; //pwm2,3 independent
PWMCON0bits.PMOD0=1; //pwm0,1 complimentary mode
PWMCON1=0;
PTPERL=0xff; // frequency of PWM period 12bit register max value:0x0fff  then minimum frequency
PTPERH=0x00;  // with 20MHZ should be about 1.2kHz with 0x0fff,  now set to 9.6MHz
DTCON=0x01; //small deadtime 100ns@20MHz
FLTCONFIG=0x80; // pwm fault detection by microcontroller defined here
PTCON1bits.PTEN=1; //enable PWM
}
// to control voltage on motor windings through PWM width
void change_dutycyclePWM1(unsigned int newdutycycle)
{
	PWMCON1bits.UDIS = 1; //disable updates from duty cycle and period buffers
	PDC0L = newdutycycle;  //change duty cycle
	PDC0H = (newdutycycle >>8) & 0x3f; //14bit register
	PWMCON1bits.UDIS = 0; //enable updates back
}
void change_smooth_dcycleupPWM1(void) //increasing duty cycle - starting motor
{
//function lasts 0.65second + instructions execution time,duty cycle is zero when starts function
motorbrakes=1;
newpwmperiod=0x2ff; // 75% duty cycle for 32V
change_dutycyclePWM1(0x0df);
Delay10KTCYx(30*2);//delay 100ms multiplied by 2 to get right delay (50ns*10k*2=1ms)
change_dutycyclePWM1(0x150);
Delay10KTCYx(180*2);
change_dutycyclePWM1(newpwmperiod-0xff); //half duty cycle
Delay10KTCYx(200*2);
change_dutycyclePWM1(newpwmperiod-0x70);
Delay10KTCYx(190*2);
change_dutycyclePWM1(newpwmperiod); //maximum duty cycle and max speed then
Delay10KTCYx(60*2); //delay cut from other steps
if (positionflag==0) change_dutycyclePWM1(newpwmperiod+0x80); //a bit faster door opening 90% duty
cycle- not tested
Delay10KTCYx(210*2);//to delay overcurrent measurement
motorbrakes=0; //enable voltage control
```

```
 }
void change_smooth_dcycleupPWM0(void) //increasing duty cycle - starting motor
{  //other function as above with different amount of duty cycle changes
motorbrakes=1;
//newpwmperiod=0x2ff;
change_dutycyclePWM1(0x120);
Delay10KTCYx(40*2);//delay 100ms multiplied by 2 to get right delay (50ns*10k*2=1ms)
//change_dutycyclePWM1(0x140); //bylo 13f i ok
//Delay10KTCYx(40*2);
change_dutycyclePWM1(newpwmperiod-0x170);
Delay10KTCYx(70*2);
//change_dutycyclePWM1(newpwmperiod-0x140);
//Delay10KTCYx(40*2);
change_dutycyclePWM1(newpwmperiod-0xff); //half duty cycle
Delay10KTCYx(90*2);
//change_dutycyclePWM1(newpwmperiod-0xc0);
//Delay10KTCYx(50*2);
change_dutycyclePWM1(newpwmperiod-0x80);
Delay10KTCYx(80*2);
//change_dutycyclePWM1(newpwmperiod-0x40);
//Delay10KTCYx(40*2);
change_dutycyclePWM1(newpwmperiod); //maximum duty cycle and max speed then
Delay10KTCYx(80*2); //delay cut from other steps
if (positionflag==0)
{
change_dutycyclePWM1(newpwmperiod+0x80); //a bit faster door opening 90% duty cycle [was +0x40 when
below enabled]
Delay10KTCYx(70*2);
//change_dutycyclePWM1(newpwmperiod+0x80); //a bit faster door opening 90% duty cycle
//Delay10KTCYx(50*2);
}
Delay10KTCYx(50*2);//to delay overcurrent monitoring and wait until current stabilises
motorbrakes=0; //enable voltage control
 }
void change_smooth_dcycledownPWM1(void) //decreasing duty cycle -stopping motor
{
//tested working fine ,current OK steps sufficient amount
//function lasts 0.34-0.38 second - is shorter than starting - no need for big current while braking
motorbrakes=1; //to stop using voltage correction functions
/*if (positionflag==0)
{
change_dutycyclePWM1(newpwmperiod-0x60); //extra step for door opening slowdown
Delay10KTCYx(40*2);
}*/
change_dutycyclePWM1(newpwmperiod-0xbb);
Delay10KTCYx(190*2);
change_dutycyclePWM1(newpwmperiod-0x1d0);
Delay10KTCYx(130*2);
change_dutycyclePWM1(0x08f);
Delay10KTCYx(30*2);
change_dutycyclePWM1(0x0000);
motorbrakes=0;
}
void movedooralittle(void)
{ //function used to release the door in case of obstruction detection
motorbrakes=1; //disable voltage control
change_dutycyclePWM1(0x0bf);
Delay10KTCYx(50*2);//delay 100ms multiplied by 2 to get right delay (50ns*10k*2=1ms)
change_dutycyclePWM1(0x13f);
Delay10KTCYx(190*2);
```

```
change_dutycyclePWM1(0x1bf);
Delay10KTCYx(180*2);
change_dutycyclePWM1(0x23f);
Delay10KTCYx(250*2);
change_dutycyclePWM1(0x18f);
Delay10KTCYx(150*2);
change_dutycyclePWM1(0x09f);
Delay10KTCYx(30*2);
change_dutycyclePWM1(0x000);
motorbrakes=0; //reenable voltage control
}
void checkdoormovement(void)
{
if (positionflag==0) //door opening button pressed
	if ((POSCNTH<0x30) & (POSCNTH>0x25)) //meaning door didn't move, dooropensensor can be used
here too
			{
			lcd_position(1,0);
			writestring(buttonorpwmfail);
			}
if (positionflag==1) //door closing button pressed
	if ((POSCNTH>0xfa) || (POSCNTH<0x01))//meaning door didn't move
			{
			lcd_position(1,0);
			writestring(buttonorpwmfail);
			}
}
//******************** PWM end *****************************

//********************* door control ******************
void QEIinit(void)
{
QEICON=0b00111011; // velocity mode enabled bit 7,no overflow bit 6,forward bit 5,
					//mode x4 update bits 4-2[101 index pulse or 110 period
match],velocity reduction ratio 1:1
					//when index pulse present it starts counting impulses
//configure noise filters
DFLTCON=0b00111000; //all filters enabled at each input
//POSTCNT is starting counting when motor starts and counts down until matches MAXCNT register
POSCNTH=0;
POSCNTL=0; // on reset unknown value so it has to be set zero position
VELRL=0;
VELRH=0; // velocity measure registers (now not used de to encoder malfunction)
MAXCNTH=0xff; //counts down to this value
MAXCNTL=0xff; //
}
void door_move(void) //door move
{
T2CONbits.TMR2ON=1; //enable timer 2
currenable=0;  //disble current monitoring while motor starts
change_smooth_dcycleupPWM0();
//change_dutycyclePWM1(0x3ff); //for voltage step testing full duty cycle set
currenable=1; //enable monitoring when constant speed already
}

void TMR0countinit(void) //measure how long lasts door movement
{
T0CON=0b00000111; //set 3 low bits 1:256 prescaler, [setting bit7 enables timer],
INTCONbits.TMR0IE=1; //enable timer0 interrupt
//approx 3.35sec: max time measurement 0xffff
```

```
}
void TMR2countinit(void)
{
// trigger first ADC conversion - later  remove it
T2CON=0b01111011; // bits 0,1 prescaler 1:16, bit 2 start/stop timer, bits 3-7 postscaler 1:16
            // those settings give max delay possible by timer
PR2=0xFf; // maybe not necessary because initially is 0xFF -to this value timer counts
TMR2=0x00; // start counting value
PIE1bits.TMR2IE=1; //enable interrupts INTCON register configured in Timer 0 init function
PIR1bits.TMR2IF=0; //clear flag
}
void initADC(void)// configure adc + interrupts when conversion finished to check if I and V are OK
{//TRISA register configured already as inputs pins 0-6, pins 2-4 used for optical encoder
ADCON1=0b00011100; //Vref+=Vdd, Wref-=Vss, overwrite unread conversion result , bit 4 enable FIFO
ADCON2=0b00111001; //left justified, Acq time: 16Tad (must be longer than 12Tad) and cclock FOSC/8 for
accuracy
ANSEL0=0b11000011; //enable ADC pins 7,6,1,0 - 4 needed (two for current and two for voltage)
ADCHS=0b01000100; //Groups choice: D pin7, B pin1, C pin6, A pin0 of ADC used ADC:0,1,6,7 in order
ABCD
PIE1bits.ADIE=1; //enable interrupts
PIR1bits.ADIF=0;
ADCON3=0b10010000; //enables interrupts (after 4th word is written to buffer)
ADCON0=0b00010101; //enables single shot conversion, multichannel sequential mode 2, bit 0 enable ADC
//access to other conversion results through change of two last bits of ADCON1 register later in interrupt
//automatically incremented when ADRESH read
}
void stoptimerandADC (void)
{
            ADCON0bits.GO_DONE=0; //stop conversion if was started as single shoot mode
            T0CONbits.TMR0ON=0; //disable measuring opening /closing time if still being measured
}
// ************************** LCD support ***************8
//all functions work - tested
void writetoLCD(char znak)
{
LCD_E_SET();
nop();
PORTD=(znak & 0xf0)>>4; //write high 4 bits to port D from znak
nop();
LCD_E_CLR();
nop();
LCD_E_SET();
nop();
PORTD=(znak & 0x0F); //write low 4 bits to port D from znak
nop();
LCD_E_CLR();
Delay10TCYx(40); // delay 37us  time to write one instruction to LCD
}
void writecommand(char znak)
{
LCD_RS_CLR();  //set RS low to write instructions to LCD
writetoLCD(znak);
}
void writedata(char znak)
{
LCD_RS_SET(); //set RS high to write data
writetoLCD(znak);
}

void initLCD(void) // done according to ST7066U datasheet
```

```
{
int i;
  LCD_RS_CLR();
  LCD_E_CLR();
  Delay10KTCYx(50*2);//delay 50msec
  LCD_E_SET();
  PORTD=0x03; // init lcd
  Delay10TCYx(10);
  LCD_E_CLR();
  Delay10KTCYx(2*2);
  writecommand(0x28); // interface 4-bits, 2-lines, sign 5x7
  Delay10KTCYx(2);
  writecommand(0x28); // send twice  ref. ST7066U datasheet
  Delay10KTCYx(2);
  writecommand(0x0c); // enable LCD functions
  Delay10KTCYx(2);
  writecommand(0x01); //clear LCD , wait
  Delay10KTCYx(2*2);
  writecommand(0x06); // no shift
  Delay10KTCYx(2);
 }
void LCD_clear(void)
{
writecommand(0x01);
Delay10KTCYx(4*2); //delay more than 1.52ms (minimum time for clear LCD)
}

void writestring(char * wskaznik)
{
while(*wskaznik) // until reaches 0
  {
  writedata(*wskaznik); //write 'wskaznik' to LCD
  wskaznik++;
  }
}
void lcd_position(unsigned char line, unsigned char character)
{
   writecommand((line*0x40+character) | 0x80);
}
//**************************MAIN PROGRAM*****************
#pragma code
void main(void)
{
int j;
initpic();
initADC(); //normalADC init for single shot mode
initLCD();
lcd_position(0,0);
writestring(welcome);
TMR0countinit(); //initialise timer 0 for door movement measurement period
TMR2countinit(); //initialise timer 2
initPWM(); //initialise PWM
QEIinit(); //initialise QEI and enable door position monitoring
positionflag=20;
movementflag=0;
motorvoltage=0x84 ; //when power supply is 32V
x=0; //for overcurrent recognition
i=0;
newpwmperiod=0x2ff; //set starting value for 32V
overccount=0;
```

```
for (j=0;j<9;j++)
{
        oldcurr[j]=0x50; //fill in array with starting values for current sense average
        oldvolt1[j]=0x82; //as above for voltage measured for 32V power supply - door closing
}
arraycnt=0; //clear counter for the current sense array
arrcntv1=0;
//set the door to fully open no interrupts used
        PORTCbits.RC7=1; //door open direction
        change_smooth_dcycleupPWM1(); //start PWM
        ADCON0=0b00011001;  //multichannel mode,ch AN0 and AN1 only, start conversion
        ADCON0bits.GO_DONE=1;//ch AN0 Csense, Ch AN1 infrared sensor
        while ((i<=0x60))
                                {
                                        while (ADCON0bits.GO_DONE);//wait until conversion is finished
                                        i=ADRESH;//read conv results
                                        dooropensensor=ADRESH; //sensor value- while door in motion is
bigger than zero (object not detected)
                                        if (dooropensensor<=0x05) i=0x70; // stop the motor
                                        convertbyte(dooropensensor,str);
                                        lcd_position(1,0);
                                        writestring(str);
                                        ADCON0bits.GO_DONE=1;//set next conversion
                                }
        change_smooth_dcycledownPWM1(); //stop PWM - door is fully open
Delay10KTCYx(1000);
POSCNTH=0x00; //for door fully open then it decreases if closing
POSCNTL=0x00; //clear enconder registers when door fully open
//to assure always the same starting position when program starts
//continue program if the door is open verify then with sensor if active then door fully open
ADCON0=0b00010101; //go back to init ADC settings all other were not changed
INTCONbits.PEIE=1; //enable interrupts
INTCONbits.GIE=1;
lcd_position(0,0);
writestring(dooropened);

//############### Main program loop ############### //
while(1)
        {
// check if button is pressed
if (PORTCbits.RC2==0) //check if button pressed to open door
        {
        PORTCbits.RC7=1; //set direction to door open for LMD18200 direction pin
        LCD_clear();
   lcd_position(0,0);
        writestring(dooropen); //write on LCD current operation
        positionflag=0; //door open button pressed indication
        door_move(); //open the door
        checkdoormovement(); //check if door moved after pressing the button
        TMR0H=0x76;//cd;
        TMR0L=0x27; //set timer to measure opening time
        T0CONbits.TMR0ON=1; // enable opening time measurement
        }
if (PORTCbits.RC3==0) //check if button pressed to close door
        {
        PORTCbits.RC7=0; //set direction state to door close
        LCD_clear();
        lcd_position(0,0);
        writestring(doorclose); //write on LCD operation
        positionflag=1; //door close button pressed indication
```

```
            door_move();// close the door
            checkdoormovement(); //check if door moved after pressing the button
            TMR0H=0x4b;
            TMR0L=0x27;
            T0CONbits.TMR0ON=1; // enable closing time measurement
            } //end if
if (positionflag==1 & POSCNTH<=0x3c)
    {
            if (PDC0H!=0)change_smooth_dcycledownPWM1(); //assure that door is in motion from PWM
register
            if (POSCNTH<=0x37) //meaning that there was no obstruction and door is closed
                        {
                        stoptimerandADC();
                        lcd_position(0,0);
                        writestring(doorclosed);
                        positionflag=2; //allow again door operation
                        overccount=0;
                        if (POSCNTH>0x29) movedooralittle(); //to assure that door is fully closed if lower closing
velocity
                        x=0;
                        } //else obstruction detected
            }
if (overccount>x) //only enter if overcurrent occured
            {
            INTCONbits.PEIE=0; //disable  all interrupts
            INTCONbits.GIE=0;
            x=overccount;
            if (PDC0H!=0) change_smooth_dcycledownPWM1();
            PORTCbits.RC7=1; //set direction to door open for LMD18200 direction pin
            movedooralittle(); //open the door
            PORTCbits.RC7=0; // set again direction to close the door
            Delay10KTCYx(1000*2); //wait 1 second until next attempt
            PIR1bits.ADIF=0; //adc int flag clear
            PIR1bits.TMR2IF=0; //timer2 int flag clear
            INTCONbits.PEIE=1; //enable interrupts
            INTCONbits.GIE=1;
            if (overccount>=3)
                        {
                        LCD_clear();
                        lcd_position(0,0);
                        writestring(error);
                        T2CONbits.TMR2ON=0; //stop timer to not allow further attempts to close
                        positionflag=2; //to avoid further door closing
                        }
            else
                        {
                        positionflag=1; // to enable to enter condition in main loop
                        door_move(); //enable door operation again and curr monitoring
                        }
            }
if (positionflag==0)
            {
            if (POSCNTH>=0xe7) //starts slowing down when reaches this value
            {
            if (PDC0H!=0)change_smooth_dcycledownPWM1();
            stoptimerandADC();
            //read value from timer register here if overccount==0 only
            lcd_position(0,0);
            writestring(dooropened);
            positionflag=2;
```

```
            overccount=0; //reset timer for next door closing count
            }
            }
}//while
}//end of main

//**************************************************************************
// for converting into ASCII for LCD display 16bit value
// changes value into ascii string which can be presented on LCD display
void convert(int result, char* str)
{
   int temp;
   int j;
            for (j=0;j<4;j++)
            {
                        temp=result>>4*j;
                        temp=temp&0x00f;
               //temp=temp>>12;
                        if (temp<=9)
                                    str[5-j]=temp+48;
                        else
                                    str[5-j]=temp+55;
            }
            str[0]='0';
            str[1]='x';
            str[6]=' '; // toseparete results on screen
            str[7]=0;
}

void convertbyte(int result, char* str)
{//for converting into ASCII for LCD display 8bit value
   int temp;
            temp=result>>4;
            result=result&0x0f;
                        if (temp<=9)
                                    str[2]=temp+48;
                        else
                                    str[2]=temp+55;
                        if (result<=9)
                                    str[3]=result+48;
                        else
                                    str[3]=result+55;
            str[0]='0';
            str[1]='x';
            str[4]=' '; // to separate results on screen
            str[5]=0; //to indicate string finish at the end zero
}
```

Appendix B: List of components

C.No	Quantity	Component Name
1.	1	PIC18F4431 microcontroller
2.	1	330µH Inductance (3A)
3.	1	LMD18200T full H-bridge
4.	1	LM2576HVT voltage regulator
5.	1	HXS 10-NP current transducer
6.	1	LM258N operational amplifier
7.	1	PC1602LRU LCD display
8.	1	GP2Y0D805Z0F Sharp infrared sensor
9.	3	1k Resistor
10.	2	11k Resistor
11.	1	1.8k Resistor
12.	2	100nF Capacitor
13.	1	1000µF Capacitor 80V
14.	2	18pF Capacitor
15.	1	20MHz Oscillator
16.	1	MBR1060 Schottky Diode
17.	1	1N5823 Diode
18.	3	0.5K Resistor
19.	2	LED Diode
20.	1	Fuse 2A
21.	1	Fuse 10A
22.	1	P6KE62A Diode
23.	2	10nF Capacitor
24.	1	330µF Capacitor 10V
25.	1	1.5k Resistor
26.	2	47nF Capacitor
27.	1	4.7nF Capacitor
28.	1	1-20k Potentiometer
29.	1	47k Resistor
30.	1	Voltage Inverter (-5V) for LCD Display
31.	1	Operational Amplifier
32.	10	Pin Connector
33.	20	Pin Socket

Appendix C: DC Motor Datasheet

TE 24 Volts DC 2 Poles

MET PMDC Motor Design

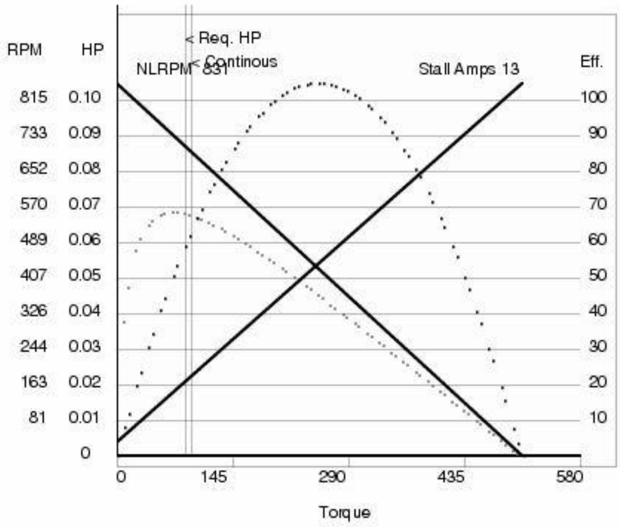
DC Motor Designer (12/14/09)
Printed: 9/17/2010

Customer: Pawel Z.
Model: 3C-5203172C
Dwg #:
Rev: *
Designed by: Kurt Witbeck

COLD Magnet: 22 deg C Copper: 22 deg C

Stall Current: 13 Stall Torque: 506 Oz-in KT: 37.17 Oz-In/Amps KE: 30.47 Volts/KRPM Resistance: 1.6632 Ohms

HP	Torque	Speed	Amps	Eff	Duty
NL	0	831	0.3	0	CONT.
0.01	7	817	.5	34.8	5.7
0.01	11	811	.6	43.9	4.8
0.01	15	804	.7	50.3	4.1
0.02	19	798	.8	55.1	3.6
0.02	23	791	1.0	58.7	3.2
0.02	27	785	1.1	61.5	2.9
0.02	31	778	1.2	63.6	2.6
0.03	35	771	1.3	65.3	2.4
0.03	40	764	1.4	66.5	2.2
0.03	44	757	1.5	67.5	2.
0.04	48	749	1.6	68.3	1.9
0.04	53	742	1.8	68.8	1.7
0.04	58	734	1.9	69.2	1.6
0.05	62	726	2.0	69.4	1.5
0.05	67	718	2.1	69.4	1.4
0.05	72	710	2.3	69.4	1.3
0.05	78	702	2.4	69.2	1.3
0.06	83	693	2.6	69.0	1.2
0.06	88	684	2.7	68.7	1.1



(Approx. 80 RPM per horizontal line.)

HOT Magnet: 40 deg C Copper: 60 deg C

Stall Current: 12 Stall Torque: 430 Oz-in KT: 35.89 Oz-In/Amps KE: 29.42 Volts/KRPM Resistance: 1.9078 Ohms

HP	Torque	Speed	Amps	Eff
NL	0	860	0.3	0
0.01	7	843	.5	34.0
0.01	11	836	.7	43.0
0.01	15	828	.8	49.4
0.02	18	821	.9	54.1
0.02	22	813	1.0	57.7
0.02	26	805	1.1	60.4
0.02	30	797	1.2	62.5
0.03	35	788	1.3	64.1
0.03	39	780	1.4	65.3
0.03	43	771	1.5	66.2
0.04	48	762	1.7	66.8
0.04	52	753	1.8	67.2
0.04	57	743	1.9	67.5
0.05	62	734	2.1	67.6
0.05	67	724	2.2	67.5
0.05	72	713	2.4	67.3
0.05	78	702	2.5	66.9
0.06	83	691	2.7	66.5
0.06	89	679	2.8	65.9

Notes:

EC 29/22 B DM4A 323-01025-00 Ext 16-05 S 2.875 F 0.980000257015228 HA 1638 FM 6.16/6.01 DM

MET - 1507 First Ave Mankato, MN 56001 - PHONE (507)625-6117 - FAX (507)625-1485