CODEN:LUTEDX/(TEIE-5230)/1-92/(2006)

Industrial Electrical Engineering and Automation

DSP Controller for Power Electronic Converter Applications

Daniel Martinsson

Magnus Lind

Dept. of Industrial Electrical Engineering and Automation

Lund University

DSP Controller for Power Electronic Converter Applications

Master Thesis work, 2006 at
Industrial Electrical Engineering and Automation, LTH

Daniel Martinsson, E-00
Magnus Lind, E-02

Abstract

Control of electric drive systems with Digital Signal Processors (DSP) is today a very common task. Faster and more functional units have made it possible to base the entire control system around the DSP without the need for additional components. As the processor lacks the ability to supply the outer world with desired power, an interface to external equipment, such as frequency converters, is required. In this project, such a universal interface circuit board was developed, built and tested. In order to generalize the solution, different input/output voltages can be used. This solution enables both CMOS level input transistor drivers and other less frequently used signal levels, such as the laboratory setup at IEA. In order to test the interface prototype a control system with a PMSM motor was developed. The control algorithm was implemented using C/C++. Maximum utilization of the interface card gives multiple controller ability; and can therefore control numerous motor applications simultaneously.

Keywords: DSP, ADC, PWM, Motor Control

Preface

This thesis is the final presentation of the development of a Digital Signal Processor (DSP) Controller for Power Electronic Converter Applications. The total work has been carried out in IEA Laboratory 7 and Laboratory 4, located in the M-building at the faculty of engineers at Lund University.

The workload has been shared equally between the authors without any distinctive differences. A more detailed specification is therefore, not deemed necessary.

WARNING: Before usage of this system please refer to notes under *External Connections, p 43*.

Acknowledgements

The authors would like to acknowledge the following people at the department of *Industrial Electrical Engineering and Automation* at Lund University; *Per Karlsson*[1] *Gunnar Lindstedt*,[2] and finally *Getachew Darge*[3] for their outstanding help and guidance throughout this entire project. Also all other personnel at the department deserve credit for their positive behavior towards all students alike. Lastly, a special thanks to Christina Dege for her help in proofreading this thesis.

[1] Research Associate
[2] Associate Professor
[3] Research Engineer

Table of contents

1 Project Outline *1*

1.1 Introduction **1**

2 Digital Signal Processor *2*

2.1 Basic facts **2**

2.2 Code Composer Studio and JTAG **3**

2.3 Input/Output signals **4**

2.4 Event Managers **4**

2.5 Timer module **4**

2.6 Pulse Width Modulation **5**

2.7 Analog to Digital conversion **5**

3 Interface *6*

3.1 Background **6**

3.2 Design Outline **7**

3.3 Component selection **8**

3.3.1 LM311 (Comparator) 9
3.3.2 LM339 (Comparator) 11
3.3.3 7407 (Buffer/driver) 12
3.3.4 LF347 (Standard performing OP) 13
3.3.5 EL2044 (High performing OP) 15
3.3.6 LT1230CN (High performing OP) 16
3.3.7 MAX 326 (Analog switch) 17

3.4 Electronic Interface **18**

4 Printed Circuit Board *19*

4.1 Design and development **19**

4.2 Schematics **19**

4.3 Layout **19**

5 Motor Control *20*

5.1 Introduction on modulation **20**

5.2 Coordinate Transformation **20**

5.3 Carrier Wave **22**

5.3.1 DSP Implementation 23

5.4 Modulation **24**

5.4.1 Basic facts 24
5.4.2 Sinusoidal Modulation 26
5.4.3 Symmetrical Modulation 27

5.4.4 Reduced Switching Modulation ... 28
5.5 Space Vector Control; theory and implementation ... 29
5.5.1 Basic facts ... 29
5.5.2 Vector Time Control ... 30
5.6 DSP Implementation ... 31
6 Power in a generic three-phase system ... 33
6.1 General information ... 33
6.2 Current ripple ... 34
7 Speed and Torque Controller ... 35
7.1 Current Controller ... 35
7.1.1 PI Controller ... 35
7.2 Torque Controller ... 38
7.2.1 PI Controller ... 38
7.3 Program development ... 39
7.4 Code Composer Studio ... 40
8 Interface card Settings and Layout ... 41
8.1 Laboratory setup ... 41
8.1.1 External Connections ... 43
8.2 Result/Conclusion ... 44
8.3 Improvements ... 46
9 References ... 47
9.1 Books ... 47
9.2 Datasheets components ... 47
9.3 Datasheets DSP ... 48
9.4 Figures ... 48
Appendix A Circuit characteristics ... A-1
A.1 LM311 ... A-1
A.2 LM311 with/without pull-up ... A-2
A.3 LM339 ... A-3
A.4 LM339 with/without pull-up ... A-4
A.5 7407 ... A-5
A.6 LF347 ... A-6
A.7 TL074 ... A-7
A.8 EL2044 vs. LM347 ... A-8

A.9 LT1230 ..... **A-9**
A.10 Total Delay in PWM and ADC Circuit **A-10**
Appendix B Three-phase and Coordinate Transformation ***B-1***
B.1 Power invariant **B-1**
B.2 Amplitude invariant **B-2**
B.3 Coordinate Transformation **B-3**
Appendix C C-Code ***C-1***
C.1 C-Code **C-1**
Appendix D Circuit Schematics and PCB layout ***D-1***
D.1 Schematics **D-1**
D.1.1 ADC Circuit D-1
D.1.2 PWM Circuit D-2
D.2 Layout **D-3**
D.2.1 Power Electronic Card D-3
D.2.2 Extension Card for EUDON Connector type F D-4
D.3 Component list **D-5**
Appendix E Internal and External connection layout ***E-1***
E.1 Front Connections **E-1**
E.2 Rear Connections **E-2**
E.3 Internal Connections **E-3**
E.3.1 DSP E-3
E.3.2 Layout of the interface card E-5

Figures and Tables

List of Figures

Figure 2.1-1: Principal Sketch of the current dSPACE system 1
Figure 2.1-2: Controller outline for the DSP system 1
Figure 2.1-1: Picture of the development card (black case to the right holds the DSP) 2
Figure 2.2-1 Snapshot of CCS programming interface 3
Figure 2.2-2: Block diagram of the development card [9.41] 4
Figure 3.3-1: Used configuration for a LM311 9
Figure 3.3-2: Used configuration for a LM339 11
Figure 3.3-3: Used configuration for a TL074 13
Figure 3.3-4: Used configuration for an EL2444 15
Figure 3.3-5: Used configuration for a LT1230 16
Figure 3.3-6: Used configuration for a MAX 326 17
Figure 3.4-1: ADC circuit 18
Figure 3.4-2: PWM circuit 18
Figure 5.2-1: Three-phase converter [9.42] 20
Figure 5.2-2: Three-phase system voltages [Ua,Ub,Uc]-frame 21
Figure 5.2-3: Three-phase representations in a [α, β]-frame 21
Figure 5.2-4: Three-phase systems represented in a [d, q]-frame 21
Figure 5.3-1: Carrier wave comparisons, basic sketch. 22
Figure 5.4-1: Switching principal [9.42] 24
Figure 5.4-2: Sinusoidal Modulation [9.44] 26
Figure 5.4-3: Symmetrical Modulation [9.45] 27
Figure 5.4-4: Reduced Switching Modulation [9.46] 28
Figure 5.5-1: Possible switch combinations [9.47] 29
Figure 5.5-2: Space Vector visualization 29
Figure 5.5-3: Visualization of switching period 30
Figure 5.6-1 Vector changes during one period 32
Figure 6.1-1: Generic 3-phase system [9.48] 33
Figure 7.2-1: PMSM Controller layout 38
Figure 7.3-1: Controller flowchart 39
Figure 8.1-1: The laboratory setup 41
Figure 8.1-2: Complete laboratory setup excluding motors. 42
Figure 8.2-1: Reference value of Omega equals 40 rad/s 44
Figure 8.2-2 : Reference value of Omega equals 80 rad/s 45
Figure A.1-1: Rise time for a LM311 A-1
Figure A.1-2: Fall time for a LM311 A-1
Figure A.2-1: A LM311 without pull-up A-2
Figure A.2-2: A LM311 with pull-up A-2
Figure A.3-1: Rise time for a LM339 A-3
Figure A.3-2: Fall time for a LM339 A-3
Figure A.4-1: A LM339 without pull-up A-4
Figure A.4-2: A LM339 With pull-up A-4
Figure A.5-1: Rise time for a HC4707 A-5
Figure A.5-2: Fall time for a HC4707 A-5

Figure A.6-1: Rise time for a LF347 _____ *A-6*
Figure A.6-2: Fall time for a LF347 _____ *A-6*
Figure A.7-1: Rise time for a TL074 _____ *A-7*
Figure A.7-2: Fall time for a TL074 _____ *A-7*
Figure A.8-1: An EL2044 vs. a LM347 working as a follower _____ *A-8*
Figure A.8-2: An EL2044 vs. a LM347 at 0.9 MHz _____ *A-8*
Figure A.9-1: Internal delay of LT1230CN _____ *A-9*
Figure A.10-1: Total delay in an ADC Circuit (Rise time) _____ *A-10*
Figure A.10-2: Total delay in an ADC Circuit (Fall time) _____ *A-10*
Figure A.10-3: The total delay in a PWM channel (Fall time) _____ *A-11*
Figure A.10-4: The total delay in a PWM channel (Fall time) _____ *A-11*
Figure B.3-1: Voltage U represented in the [d, q]-frame _____ *B-3*
Figure D.1-1: ADC Circuit _____ *D-1*
Figure D.1-2: PWM Circuit _____ *D-2*
Figure D.2-1: Power Electronic Card, Top layer _____ *D-3*
Figure D.2-2: Power Electronic Card, Bottom Layer _____ *D-3*
Figure D.2-3: Power Electronic Card, Top silk _____ *D-4*
Figure D.2-4: Extension Card, Bottom layer (left) and Top silk (right) _____ *D-4*
Figure E.1-1: Front view panel with numbered blocks _____ *E-1*
Figure E.1-2: Schematics of LEDs and Button connections _____ *E-1*
Figure E.1-3: Arrangement of measurements loops on front panel _____ *E-2*
Figure E.3-1: PCB layout of the DSP development board [9.41] _____ *E-3*
Figure E.3-2: Important spots on the interface card _____ *E-5*

List of Tables

Table 1: Component list _____ *D-5*
Table 2: Extension Card (left), Power Electronic Card (right) _____ *E-2*
Table 3: Block diagram of the connector P5 _____ *E-3*
Table 4: Block diagram of the connector P7 _____ *E-4*
Table 5: Block diagram of the connector P8 _____ *E-4*
Table 6: Block diagram of the connector P9 _____ *E-4*
Table 7: Block diagram of PWM A and PWM B connections from DSP _____ *E-6*
Table 8: Block diagram of the connector PWM A that leads to extension card _____ *E-6*
Table 9: Block diagram of the connector PWM B that leads to extension card _____ *E-6*
Table 10: Block diagram of the ADC connection. _____ *E-6*

1 Project Outline

1.1 Introduction

Today DSPs are a common component in different control applications. This is due to the low cost and good performance features. The controller part of the laboratory setup in the motor-lab consists of a system called dSPACE. This system is used in course laboratory sessions due to its good visualization of signal responses. The limitations of this system lie in the inability to expand and adapt to other setups of power electronics. One of the major reasons for this is the limited amount of input/output-signals. Throughout the years, different experiments with different types of DSPs have been carried out at IEA to come across this problem. Early DSPs required many extra circuits to work (such as fast AD-converters). Today DSPs have much more calculation power and can therefore, potentially be the heart of a controller system without any need for external signal components.

The aim of the project is to provide the first step for introducing improved DSP motor control in the lab. The main task was therefore to build an adapter card between the DSP and the lab frequency converter. The utilization of 12 PWM channels and 16 ADC channels generalizes the card. In order to test the interface a controller algorithm was also to be implemented.

The following figure describes the present dSPACE control system and the desired DSP based system.

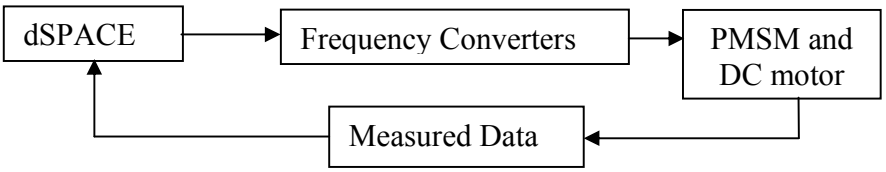

Figure 1.1-1: Principal Sketch of the current dSPACE system

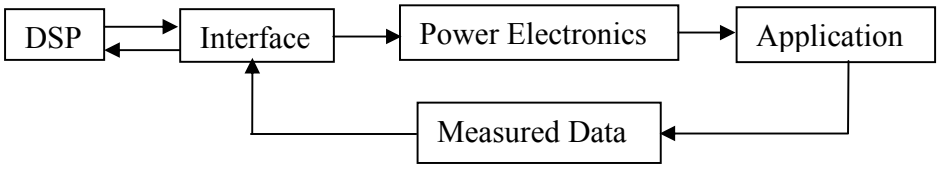

Figure 1.1-2: Controller outline for the DSP system

2 Digital Signal Processor

2.1 Basic facts

Under normal design conditions, consisting of a control system with certain demands and specifications, a controller is selected based on the given parameters. However, in the current case, the DSP was already selected. The brief presentation of the given DSP that follows below only includes the features that were used in this project.

Name:	**TMS320F2812 (Manufacturer: Texas Instruments)**
Clock speed:	**150 MHz**
Memory:	**Expandable up to 1Mb**
Analog to digital-conversion:	**16 Channels, 12-Bit resolution, 25 MHz sample rate**
Pulse Width Modulation signals:	**16-channels, space vector capability**
Input/0utput[1]-pins:	**Up to 56**
Signal levels:	**[0, 3.3] V, (0-3 V on ADC-pins)**

The given DSP was shipped with a development card (eZdsp^{TM} F2812 from Spectrum Digital, Inc). The purpose of this card is to provide a platform for easy and fast evaluation of the DSP features. The onboard I/O-pins easily enable connections to the outer world (soldering pads equally spaced 2,54 mm (industrial standard)) see Figure 2.1-1 for layout.

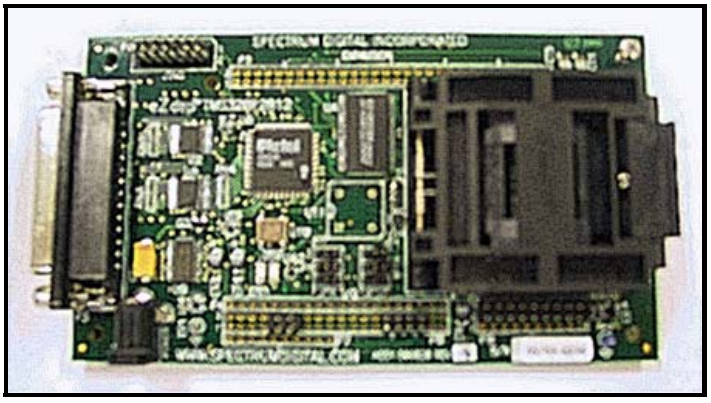

Figure 2.1-1: Picture of the development card (black case to the right holds the DSP)

If the DSP is to be used without the development card, more knowledge about internal signals is needed. In order to get a system running, detail studies of datasheets would be vital and external components will be needed.

[1] Denoted I/O from here on

2.2 Code Composer Studio and JTAG

The code development is carried out through an application suite called Code Composer[1] Studio™. It is made by Texas Instruments and is free of charge. It supports all their families and variations of DSPs. The key feature of the CCS is to serve the whole development chain. It supports programming in both C and Assembler. In Figure 2.2-1 the outline of the programming interface is stated. The CCS consists of four main parts: Project window, Program window, Watch window and Output window.

In the Project window all files in an open project is shown in a directory structure. A Double-click on a file will open it in Program window and make it editable. When choosing the compile command, potential errors will show up in the Output window. The watch window enables tracking of variable values while running the program.

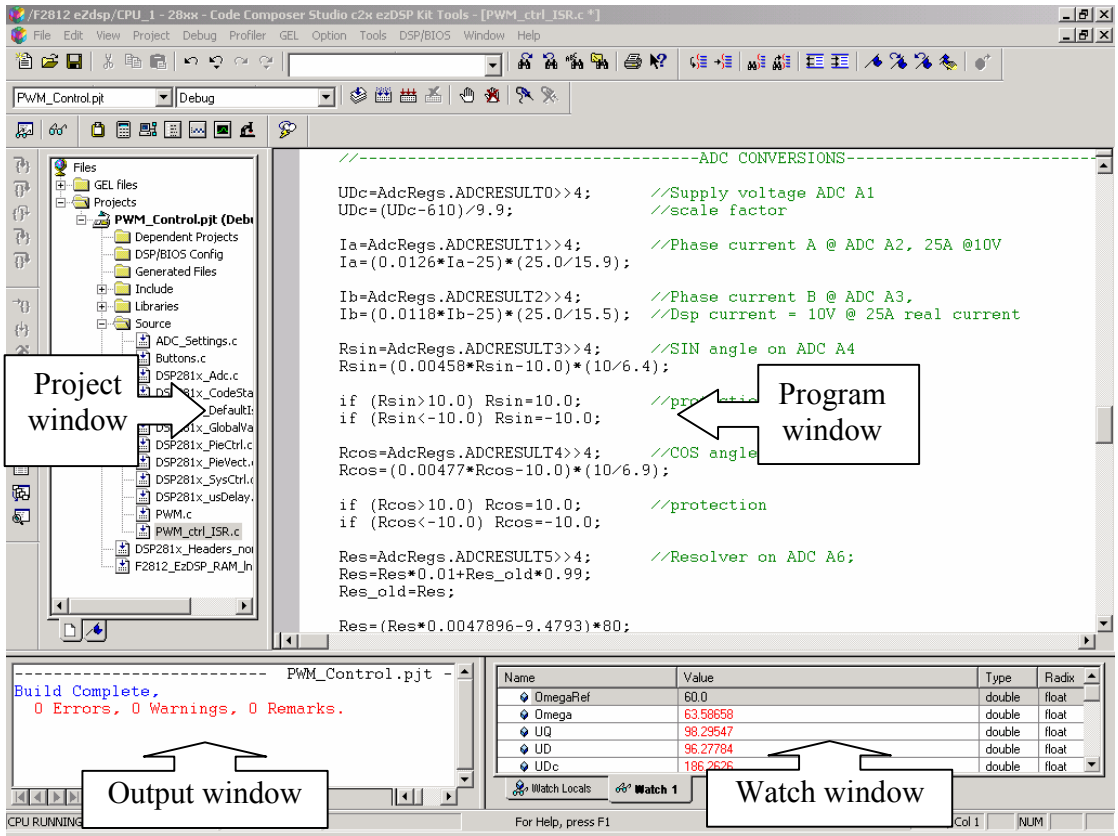

Figure 2.2-1 Snapshot of CCS programming interface

The CCS communicates with the DSP via the computers standard parallel port. The communication is handled by the standardized Joint Test Action Group [2]-interface (IEEE 1149.1). Onboard there is also a 14-pin header connector for JTAG. See Figure 2.2-2 for

[1] Denoted CCS from here on
[2] Denoted JTAG from here on

block diagram. JTAG is a debugging system for embedded systems. The DSP contains a JTAG module that enables CCS to access the registers of the DSP while in running mode.

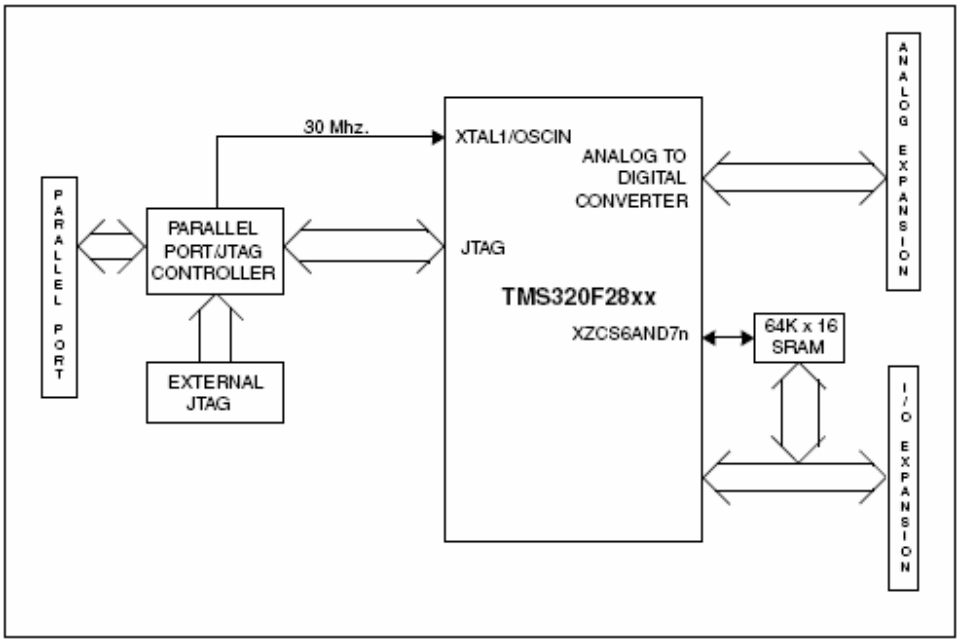

Figure 2.2-2: Block diagram of the development card [9.41]

2.3 Input/Output signals

The number of I/O-pins is in total 56. Most of these pins have multiple functions set by flags in registers. Each pin can be set as a standard I/O-pin, reading or writing digital data. Special functions such as Pulse Width Modulation[1] and Analog to Digital Conversion[2] can also be assigned to specific predefined pins.

2.4 Event Managers

On the DSP, special functions are separated into two, different "Event Managers " (named EVA and EVB). The EVx are identical and contain a broad range of special functions used in motion control and motor control (Timers, PWM and ADC). Each EVx is individually programmable and works in parallel with each other. Please refer to [9.33] for further details.

2.5 Timer module

The DSP is equipped with four independent timers (two for each EVx). The timer is useful for time critical applications. It works by counting up/down a 16-bit value and generates an interrupt at a predefined value, indicating, "Time is up". Please refer to [9.33] for further details.

[1] Denoted PWM from here on
[2] Denoted ADC from here on

2.6 Pulse Width Modulation

The DSP has 16-channels for pulse width modulation divided likewise on the Event Managers. Each channel can be programmed individually or in pairs. The PWM-module uses the built in timer circuit for generation of the PWM pulses. The PWM-module also contains a pulse pattern generator for programmable generation of symmetric and asymmetric PWM waveforms.

A PWM signal is defined as a signal with constant amplitude, meanwhile the pulse width is allowed to vary. The pulse width is defined as a value between 0 and 1, or 0 to 100%. The value is named *Duty cycle*, the name refers to the time a specified output will be used.

2.7 Analog to Digital conversion

There are 16 Analog to Digital Conversion-channels divided into two groups of 8. Each channel has a resolution of 12-bit and a sampling frequency of maximum 25 MHz. The input signal must be between [0, 3] V. The outcome of the ADC is then calculated as:

$$\text{Digital Value} = 4095 \cdot \frac{(\text{Input Analog Voltage - ADCLO})}{3}$$

ADCLO is a pin on the processor where the analog ground should be connected (value is zero). The value 4095 is derived from the resolution of 12-bits minus one LSB:

$2^{12} - 1 = 4095$

The digital value is then stored in the 12 MSB of a special result register for easy access. Please refer to [9.32] for further details.

3 Interface

3.1 Background

In a general application with PWM controlled transistors, each output channel controls one and only one transistor. In a general three-phase application, six transistors are needed, two for each phase. This design will thereby require six separate output channels to control all three phases or more generally, two channels per controlled phase. This is however not always the case!

In the existing laboratory[1] setup, only three output channels are needed due to the internal design. Because the new system should be able to handle multiple applications, an electronic adaptation is required. Depending on the controller, design measurements of different voltage and current levels are needed. In most applications, only one voltage and two current levels, respectively, are utilized, while all other needed levels are calculated based on these values.

A program suite called dSPACE, with a DSP system located on a plug-in board in an external computer connected to the system, controls the current lab system. The PWM controller in the dSPACE environment generates three independent output channels. Because only three channels are present even though six transistors are used, an internal solution within the frequency rectifier is vital. Each incoming channel is separated into an inverted respective non-inverted signal. The onboard solution provides thereby the system with six output channels. However, these are not independent; only three different transistors can be controlled while the other three act as slave transistors.

The design of this lab system must therefore be taken into consideration. A general system with six different PWM channels should be as controllable as a system with only three PWM channels. It is vital that the output levels are adequate regardless of the outer attached system. In order to guarantee that the interface is able to deliver a specified voltage two different levels are used. The two voltage levels are chosen, taken into account the available systems on the market, and should therefore be accurate in most cases, even though other voltage values certainly are possible. The different voltage levels are stated below.

Lab system specification
$Output\ level = \pm 15V$ (Transistor driver input levels in frequency converter)
$Input\ level = \pm 10V$ (Measurement levels)

DSP system specification
$Output\ level = [0,3.3]V$ (Output levels on DSP)
$Input\ level = [0,3]V$ (Input level on ADC pins)

[1] Denoted Lab system from here on.

The voltage levels above are the required values for the respectively transistors. Note that the current is represented in volts. In the case of the lab system, a switch between transistors takes place when the input voltage reaches above +10 V respectively drops below -10 V. The onboard electronic must therefore transform [0,3.3] V to ±15 V, also ±10 V to [0,3] V. These are the voltage levels used in the lab system. In other systems [0,15] V is used to switch between the transistors (standard CMOS-levels), thereby also a transformation between [0,3.3] V to [0,15] V is needed, with the same input level as in the previous case, no further action regarding the value is required.

3.2 Design Outline

Since the interface card should be able to handle different types of converters different output levels will be needed. Given PWM specification requires [-15, 15] and [0, 15] V. The DSP output lies in the range of [0, 3.3] V, herein exists the need for transformation. One must therefore either transform the output or use it to control higher voltages. The latter solution is used in this project. A comparator is therefore a suitable choice as the output is either high or low. One benefit with a comparator is the feature of different supply voltages. One comparator can thereby be set to work within different boundaries.

The two different voltage levels may be achieved by changing the negative supply to either 0 or -15V. The output toggles as fast as the input equals or exceeds a reference signal on the second input[1]. As the comparator is controlled by the DSP, input two is set to 1.6 V.

Each comparator circuit is followed by an operational amplifier[2] working as a follower i.e. gain=1. This configuration is carried out as an impedance separation between the measurement object and the controller system. Finally, system output voltage must be able to be zero as a safety requirement. In order to fulfill this task an analog switch is used, which is assembled after the follower. This is not the only solution that has been tested during the project; it is however the only one that fulfills both safety and technical requirement.

In the given system, ADC specifications require a transformation between ± 10 V to [0,3] V.

An inverting amplifier with a variable offset/gain performs the transformation.
Usage of one OP with given features inverts the signal i.e. two circuits are needed. The input circuit is an inverter thus gains =-1, meanwhile the output stage holds a gain of ≤ -1.

[1] Please refer to adequate datasheet for pin configuration etc.
[2] Denoted OP from here on.

3.3 Component selection

During the startup phase, different solutions to achieve adequate signal levels were tested in a test bench. The main goal was to find the fastest possible circuit for each task. The selection process is based primarily on a few critical parameters.

The parameter *slew rate* was given a high priority. The value of this sets the minimum time that the circuit will need to toggle between two internal levels. The unit V/us indicates how fast a given step will be for a certain voltage level. However, the slew rate is not static on both flanks, tests during the project development indicate that a difference on respectively, $T_{RiseTime}$ and $T_{FallTme}$ is present. One can therefore not choose a circuit based only on this parameter; test has revealed that a good value often is a signal of a circuit with good overall performance. A good value in this case is a high number, which thereby will give a short $T_{RiseTime}$ and $T_{FallTme}$. Note that slew rate is only defined for an analog circuit; in a digital circuit the switching times T_{LH} respectively T_{HL} set the performance. The terms respectively, $T_{RiseTime}$ and $T_{FallTme}$ might be misleading; a more adequate term would be *response time* but due to deviations on the flanks each flank time is specified.

A list of tested circuits will follow below, each circuit is specified with a full solution schematics and detailed explanation. Each component and corresponding circuit has been empirically evaluated and analyzed. Both pros and cons were noted in all configurations.

The selection of components was carried out during the early stages of the development phase. Based on experience both prior to and during the project different types of components were chosen. Each component was chosen due to its own individual properties based on technical criteria.

It is under no circumstances proven that the selected circuits are in any way optimum; clearly both faster and better components are available. However based on financial and technical aspects the following selection was made.

Comparator:	LM311 and LM339
Buffer/Driver:	7407[1]
Operation Amplifier:	LF347, TL074, EL2044 and LT1230
Analog Switch:	MAX326

[1] Used as an operational amplifier in the test application

3.3.1 LM311 (Comparator)

The comparator LM311 is a common standard circuit. It is average performing and therefore suitable in a wide area of applications at a low cost. The special feature of a comparator contra a standard OP is its ability to do a fast switch to its max/min voltage levels, when input signal is above or under a certain compare value.

The *LM311* also provides an external controller pin named *enable*, when the enable input is grounded the output will be active, as the enable toggles from low to high the output will be turned off. Conversely, this function performed significantly poorly during the development phase, but since the function is imperative, an exclusion of the *LM311* became unavoidable.

The following values are empirically determined; please refer to Figure A.1-1 and Figure A.1-2.

$T_{RiseTime} = 0.4\mu s$

$T_{FallTime} = 0.1\mu s$

A typical value of the response time is 200 ns according to the datasheet, this value is however only valid under specified test conditions. Please refer to [9.24] for further details. The pull-up resistor after LM311 was empirically determined to get the right value. A final value of 2,2 kΩ was chosen based on numerous tests with different values in the range of 1 to 300 kΩ. The selected resistor presented the best tradeoff between $T_{RiseTime}$ and $T_{FallTme}$. Studies have proven that the presence of a pull-up resistor is vital to achieve the adequate output characteristic. Please refer to Figure A.2-1 and Figure A.2-2.

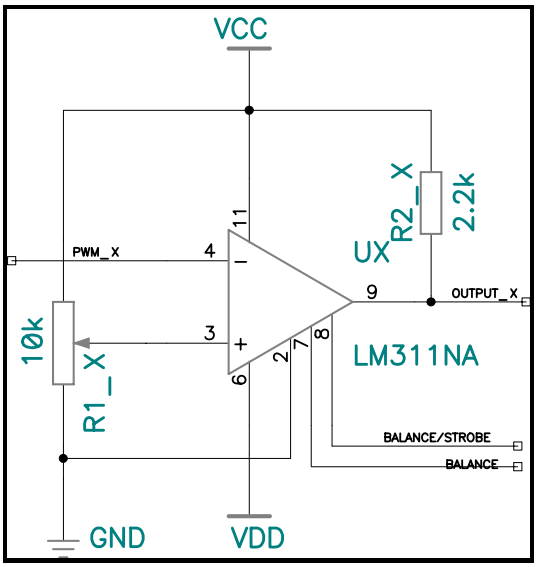

Figure 3.3-1: Used configuration for a LM311

A resistor with a power rating of 0.25W was chosen. This is based on the following estimation.

Lowest power consumption under operation;

$$P(t) = U(t) * I(t) \Leftrightarrow P(t) = \frac{U(t)^2}{R} \Rightarrow p(t) = \frac{15^2}{2200} = 0.10W$$

Equation 3.3-1

Absolute worst-case scenario;

$$P(t) = U(t) * I(t) \Leftrightarrow P(t) = \frac{U(t)^2}{R} \cdot \Rightarrow p(t) = \frac{30^2}{2200} = 0.41W$$

Equation 3.3-2

Yet, this is not the average power consumption as the output toggles between different levels based on controller settings[1]. Thereby 0.25W should to be an acceptable value. In order to minimize potential risk for overheating all pins are preferably set to toggle continuously i.e. power will only be consumed in the pull-up resistor, when the voltage equals [-15,0][2].

[1] [0, 15] V or [-15, 15] V
[2] Circuit has only been tested in CMOS-mode

3.3.2 LM339 (Comparator)

Preferably, a LM339 should be used because of the lower price per unit, since the LM339 contains four equal comparators. However, the LM311 is superior in the ability to disable the output; this feature is possible due to the enable pin. A LM339 requires an external circuit to fulfill the same task. This, on the other hand, is not a serious problem; the only drawback to this solution is that a second circuit is needed; conversely, a faster circuit than the LM311 may be used. The following values are empirically determined; please refer to Figure A.3-1 and Figure A.3-2.

$$T_{RiseTime} = 0.2\mu s$$
$$T_{FallTme} = 0.5\mu s$$

The LM339 also requires a pull-up resistor for best performance, exactly in the same way as a LM311. The pull-up resistor after the LM339 was empirically determined to get the right value. Just like the LM311 a final value of 2,2 kΩ was chosen, once again based on numerous tests with different values in the range of 1 to 300 kΩ. Please refer to Figure A.4-1 and Figure A.4-2.

The same power as in a LM311 is consumed.

$$P(t) = U(t) * I(t) \Leftrightarrow P(t) = \frac{U(t)^2}{R} \Rightarrow p(t) = \frac{30^2}{2200} = 0.41W$$

Equation 3.3-3

No output difference between a LM311 and a LM339 is noted. Thus, the same assumptions are adequate. The R1_X resistor[1] is used to set the switching level, to enable different levels a potentiometer is used instead of a constant value, please refer to Figure 3.3-2.

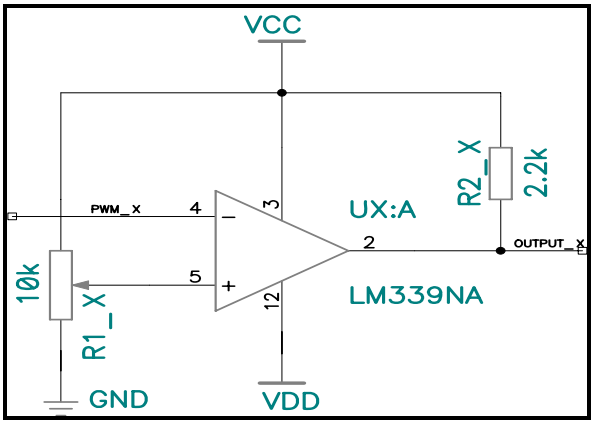

Figure 3.3-2: Used configuration for a LM339

[1] Same potentiometer is used for all comparators.

3.3.3 7407 (Buffer/driver)

The given circuit was chosen due to its excellent rise and fall time. As stated below, a significant time difference between the two flanks was noted. The 7407 was the fastest available circuit among the selected ones, despite the given time difference of ΔT. Even thought the rise-time of 500 ns is 10 times higher than the exceptionally low fall-time, is it still considerably fast compared to others alike. Nevertheless, due to the fact that the 7407 is a digital circuit one must also be aware of the switching times as the output changes. The following values are empirically determined. Please refer to Figure A.5-1 and Figure A.5-2.

$T_{RiseTime} = 0.5\mu s$
$T_{FallTime} = 0.05\mu s$
$T_{PLH} = Not\ \ possible\ to\ measure$
$T_{LHL} = Not\ \ possible\ to\ measure$

T_{PLH} is per definition the time delay that occurs between an input step and input toggle occurs. Likewise T_{PHL} represents the delay as the output toggles from high to low i.e. circuit delay.

A typical value for the T_{PLH} and T_{PHL} according to the datasheet is 6 ns. These values are only valid during specified test conditions. Please refer to [9.21] for further details.
The major drawback with the 7407 is it lacks the ability to handle negative voltage levels. Due to this severe limitation, the 7407 was excluded from further development.

3.3.4 LF347 (Standard performing OP)

The *LF347* is one of the most frequently used OPs on the market of today. The slew rate of 13 V/µs is often acceptable in a standard application without high demands of rise and fall-times, respectively. The low price per unit is definitely an advantage over other similar components. If a smaller bandwidth is acceptable, one can choose the TL074 that has the same characteristics; apart from the bandwidth, at an even lower price. The following values are empirically determined. Please refer to Figure A.6-1 and Figure A.6-2.

$$T_{RiseTime} = 1.2\mu s$$
$$T_{FallTime} = 1.2\mu s$$

$$Slew\ rate: 16V / \mu s$$

The following values for a TL074 are also empirically determined. Please refer to figures Figure A.7-1 and Figure A.7-2.

$$T_{RiseTime} = 1.5\mu s$$
$$T_{FallTime} = 1.5\mu s$$

$$Slew\ rate: 13V / \mu s$$

Compared with the 7407 both the LF347 and TL074 have approximately 20 times longer rise time and 2 times the fall time, according to the datasheet both circuits have a rise time of 1µs under other test conditions, please refer to [9.22] and [9.23] for more details.

As the ADC circuit is not time critical in the same way as a PWM output, slower circuits will be used. Based on the low price and acceptable performance the TL074 was selected.

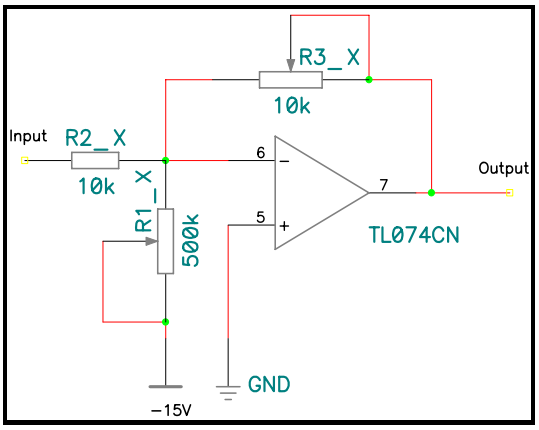

Figure 3.3-3: Used configuration for a TL074

The resistor configuration around the TL074 stated in Figure 3.3-3 is an inverting amplifier with a DC offset capability. By adjusting R3_X the gain of the signal will change. By turning the potentiometer R1_X, the offset voltage is adjusted. For further details, please refer to [9.2] . The choice of resistor values enables numerous input levels. As long as DSP input is sustained between [0,3] V any input signal may be used (the supply voltage to the OP sets the limit).

3.3.5 EL2044 (High performing OP)

This is the most expensive OP among the selected ones; the price for an EL2044 is roughly three times higher than other options, for example the LM347. Although the high price certainly deters implementation, some benefits are possible to find. The EL2044 can deliver a slew rate of about 250 V/µs, which by far is better than the LM347 with only 13 V/µs.

To enhance the superior feature of the EL2044 one can compare the bandwidth of 120 MHz with the ordinary LM347, which only has a bandwidth of 3 MHz!
In the final design the EL2044 was used as a follower, it will thereby decrease the load at the output of the LM339.

As stated earlier, none of the selected circuits is undoubtedly state of the art, for example is it possible to purchase a replacement for the EL2044 with a bandwidth of 200 MHz. Under the available test conditions, no difference between the EL2044 and the LM347 could be detected in the range of 0-100kHz; however, at frequencies above these a small delay will be present. Please refer to Figure A.8-1 and Figure A.8-2.

Nevertheless, as the replacement circuit only is an updated version of the EL2044 no exchange was needed. The quality criteria were certainly fulfilled with the outdated circuit, which thereby was used in the final design phase due to the financial saving. Yet, lack of possible retailers excludes the EL2044 from the concluding design.

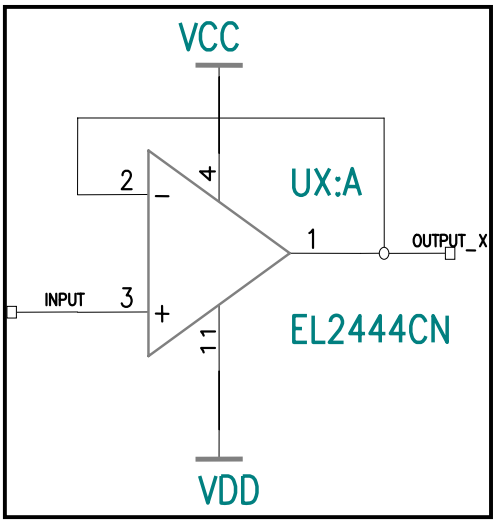

Figure 3.3-4: Used configuration for an EL2444[1]

The configuration is straightforward with no needed extra components.

[1] Quadruple version of EL2044

3.3.6 LT1230CN (High performing OP)

As the EL2044 was excluded from the project, a replacement circuit was needed. Based on background research the LT1230 was selected. This was for the reason that the LT1230 is pin compatible with both the TL074 and the EL2444; as a result, no changes within the design were needed. However, as the LT1230 not is an optimal buffer circuit, action needed to be taken to prevent high current within the circuit. As a direct result was a resistor of 2.2 kΩ assembled in the feedback circuit, please refer to Figure 3.3-5

The following values are empirically determined.

$$T_{RiseTime} = Not\ possible\ to\ measure$$
$$T_{FallTime} = Not\ possible\ to\ measure$$

As the PWM, channels need significantly fast circuits the LT1230 was used. The slew rate of 1000 V/µs enables an even faster switching frequency than needed.

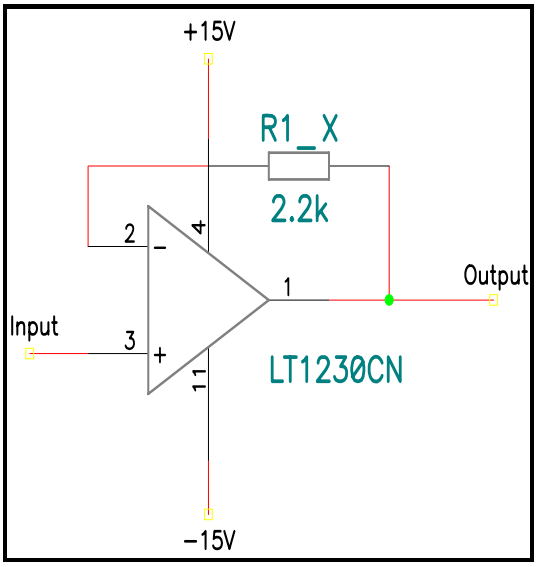

Figure 3.3-5: Used configuration for a LT1230

The value of 2.2 kΩ was determined empirically. Nevertheless, other values may be possible. According to the datasheet, [9.27] heat sinks may be needed to prevent circuit damages. As an extra caution the interface card features heat sinks.

3.3.7 MAX 326 (Analog switch)

As the LM339 was chosen, one would need an extra circuit to enable control of the output, since it is vital that the output can be zero regardless of the input signal. The circuit MAX 326 was selected; this was based on the same criteria as the other components.

The chip is an ON-OFF-switch with no mechanical parts, i.e. transistors do the job! It is used in the system to generate the zero-level output stage on both transistors. The switch is controlled via standard [0, 5] V-logic (3,3V will also be acceptable) and can thereby easily be controlled by the DSP please refer to Figure 3.3-6.

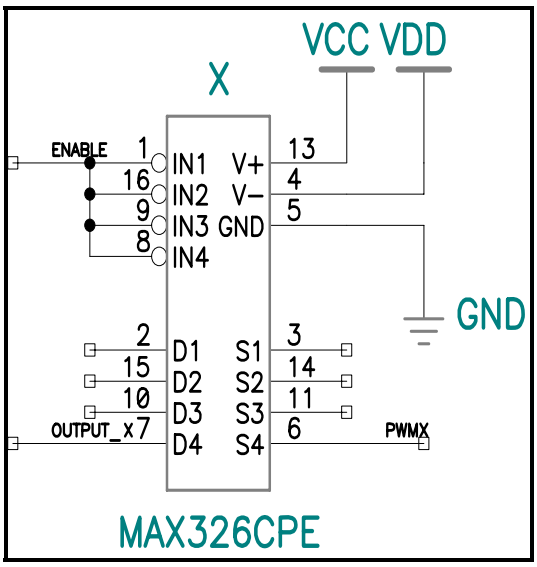

Figure 3.3-6: Used configuration for a MAX 326

The control of the circuit is carried out via the *enable* pin. The circuit provides the system with the needed output control, thus no other components are needed.

3.4 Electronic Interface

As a result of the development phase, the following two circuits were chosen for the final design. Please note that the final design for each individual circuit may vary from the ones stated below, also note that only one channel per circuit is displayed, as well as no decoupling capacitors are included. For complete schematics, please refer to the section *Schematics, pD-1.*

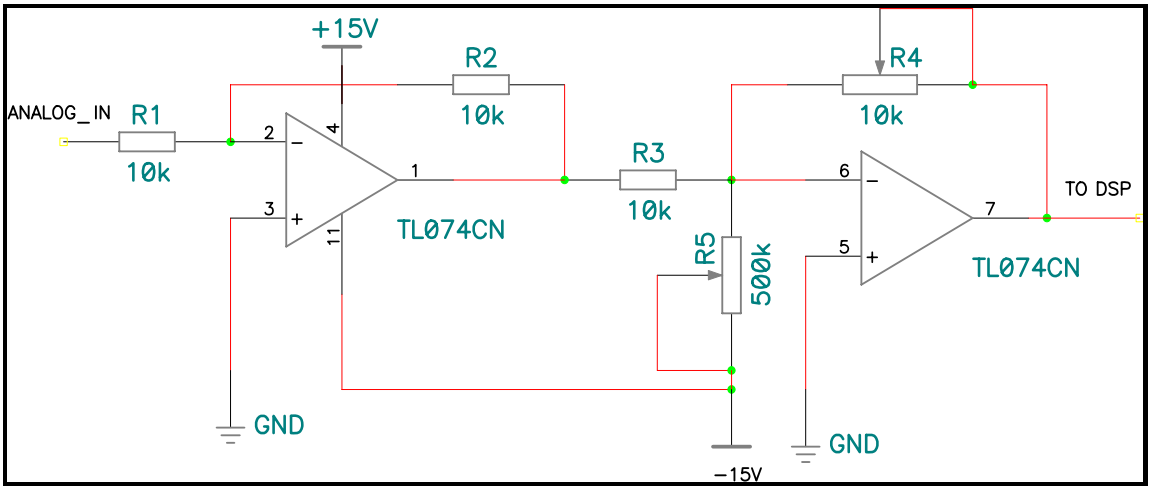

Figure 3.4-1: ADC circuit

The ADC circuit will transform $\pm 10V$ to [0, 3] V, while the PWM circuit will convert [0, 3.3] V to [-15, 15][1] or [0, 15][2]V, depending on the controller mode.

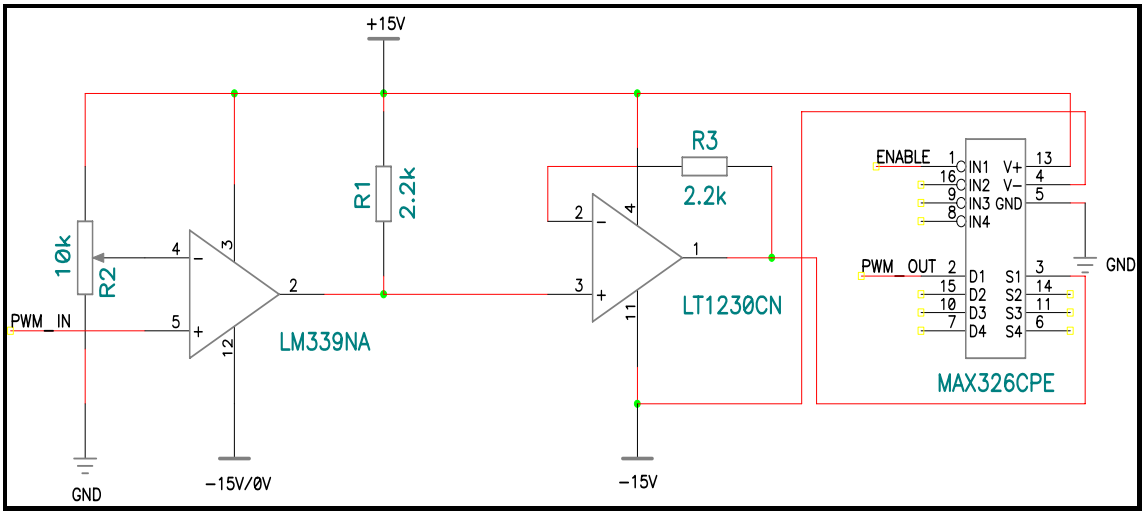

Figure 3.4-2: PWM circuit

The total delay in respectively circuit can be seen in *Total Delay in PWM and ADC Circuit, pA-10.*

[1] Existing Lab system
[2] General CMOS voltage levels

4 Printed Circuit Board

4.1 Design and development

This section will describe all the details concerning the development of the Printed Circuit Board[1]. The complete design has been carried out via the P-Cad 2000 program. This program includes parts for both electrical and physical card design. The P-Cad 2000 contains two major tools, Schematics and PCB. The Schematics tool enables the user to perform a design based on the electrical aspects while the PCB program enable a design based on visual aspects. However, a change in the schematics part of the programs will immediately affect the other one, while on the other hand, a change in the PCB layer will not change the connections in the Schematics, changes will only affect components and connections onboard the PCB layer. There are also other tools used for design of single components, a detailed specification of these areas will not be covered as they have not played a fundamental role in the design development.

4.2 Schematics

As both the PWM and the ADC circuit should be located at the same PCB, one design schedule schematics was used. This solution was possible due to the built in function of multiple design sheets. The function of a specified sheet is to enable a hierarchical structure of the overall design, which thereby clearly enhances the visibility. The total schematic will follow under Schematics*, pD-1*.

4.3 Layout

The final version of the completed PCB design will follow under Layout*, pD-3*. Please note that both the PWM and ADC circuit are located onboard.

[1] Denoted PCB from here on

5 Motor Control

5.1 Introduction on modulation

A modulated signal is also another name for a voltage over time-controlled signal. The control principle is based on a comparison between a predefined value and an actual level within the system. The predefined value may be set to change over time; as a result, the output signal will also be time dependent. Different methods to generate reference values are at hand. Each method has its own pros and cons, which thereby increases the available alternatives.

The output toggles as often as a match between the levels occur. The frequency of this depends on the modulation method. *Sinusoidal, Symmetrical* and *Reduced Switching Modulation* are all common methods in applications that are more general. These different methods among others will be covered in the section Modulation, p24.

5.2 Coordinate Transformation

In a symmetric three-phase system as displayed in Figure 5.2-1 ua(t), ub(t) etc are time dependent. An output voltage U that depends on the angle ω(t) can be seen in Figure 5.2-2

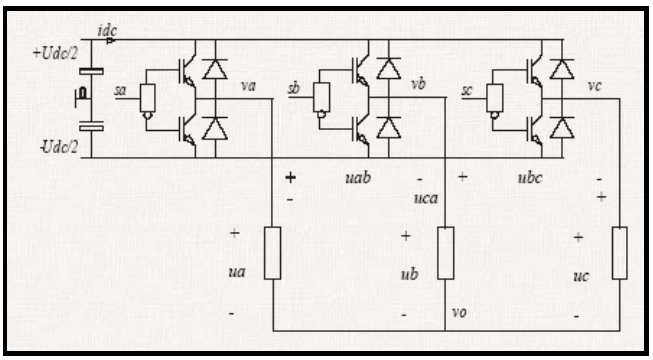

Figure 5.2-1: Three-phase converter [9.42]

In order to eliminate changes in the output voltage due to the angle dependency coordinate transformation is used. The principal is based on the idea that a three-dimensional coordinate system may be expressed in a two-dimensional. The transformation is carried out in several steps; the details of this can be seen in the section *Coordinate Transformation, pB-3.*

The system voltage U represented in the origin three-dimensional coordinate system.

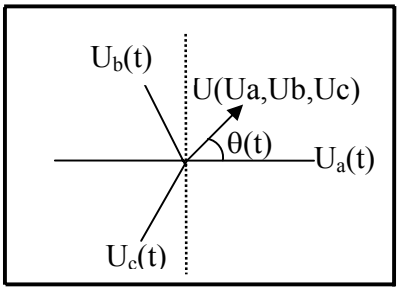

Figure 5.2-2: Three-phase system voltages [Ua,Ub,Uc]-frame

Figure 5.2-2 can be depicted as below. As seen, angle dependency is still present.

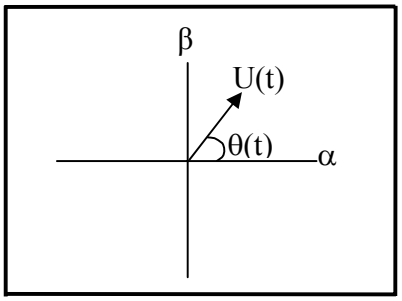

Figure 5.2-3: Three-phase representations in a [α, β]-frame

The final system is represented in a [d, q] frame where no angle dependency is at hand; this is possible due to the technique called coordinate transformation. The [d, q] frame rotates with the angle ω and appears thereby as quasi-stationary.

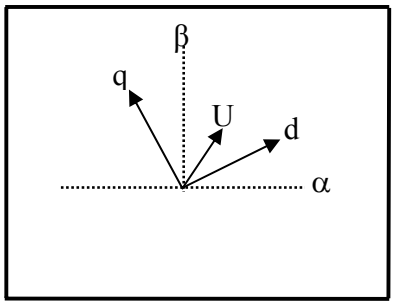

Figure 5.2-4: Three-phase systems represented in a [d, q]-frame

5.3 Carrier Wave

An internal reference signal is made within the DSP. This signal is used as a controller whenever a toggle of the output should occur. The *carrier wave* is controlled via the implemented regulator. This solution enables different shape and frequencies, as different applications need different carrier waves.

The fundamental idea of the carrier wave is to decide when an output toggle should take place. The control is based on a comparison between a calculated reference value and a given compare value. The calculated reference value is determined by the choice of modulation technique, while the compare value is set by the carrier wave. The shape of the carrier wave is either saw-tooth or triangular shaped. The output will change as often as an intersection between the two values occurs

By changing between positive/negative slopes different output characteristics will be present. A basic sketch is shown in Figure 5.3-1, note that two switching periods are displayed. The triangular shaped carrier wave intersects a predefined value; *Compare* which causes the output to change

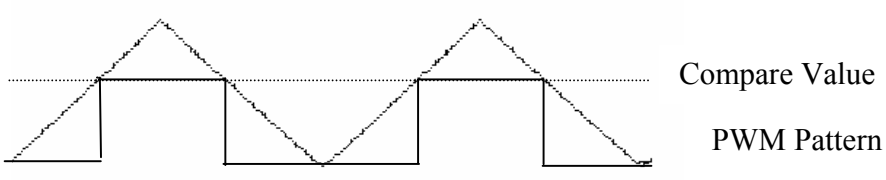

Figure 5.3-1: Carrier wave comparisons, basic sketch.

The supply voltage Udc sets the amplitude of the carrier wave, i.e. the amplitude equals half Udc. A term called modulation index *m* is defined by Equation 5.3-1.

$$m = \frac{2 \cdot U_{i,ref}}{u_{dc}}, \quad i = 1,2,3$$

Equation 5.3-1

A value $m \leq 1$ is truly acceptable, however m=1 may cause the controller to desire a higher value than possible due to the fact that $\hat{A}_c = \hat{A}_{ref}$

When $m \geq 1$ intersection between the two curves expires, as a result output will be constantly high; i.e. control signal saturation.

5.3.1 DSP Implementation

Every given wave type can be implemented in the F2812. The user can set desired shape and frequency. In order to enable such an operation, different aspects needs to be considered.

- Selection of EVx
- Corresponding timer period bit must be set i.e. $\frac{1}{\text{switching frequency}}$
- Corresponding timer control bit must be set according to desired output
- Corresponding timer counter must be set to a default value

The following pseudo code generates a triangular shaped carrier wave on EVA.

```
/*-----------Prior Code ends here----------*/
Init_EVA(); //Event manager is initialized
EvaRegs.T1PR  = 0xFFFF; //Desired switching frequency
EvaRegs.T1CON.bit.TMODE =0x01; // Timer in continuous up/down counting mode
EvaRegs.T1CNT=0x0000; // Timer start value

/* Handle the interrupt in some way */
```

It is assumed that a correct interrupt handler is loaded into the PIEVECT.
Further explanation of the interrupt handler is beyond the scope of this text. For further details such as bit patterns and so forth, please consult the relevant datasheets also refer to [9.34]

5.4 Modulation

5.4.1 Basic facts

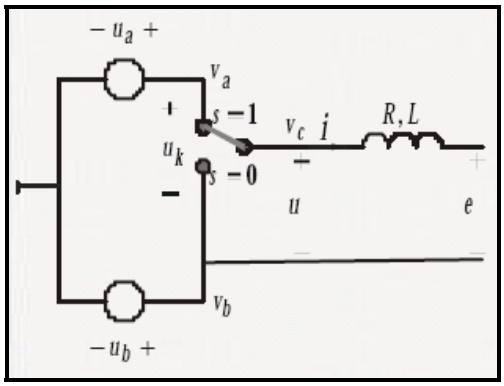

Figure 5.4-1: Switching principal [9.42]

In the most fundamental control case as described in Figure 5.4-1, only two states are possible, either s equals one or s equals zero. Clearly, the output follows the switch, and a very simple controller can thereby be implemented. It is easy to be misled, and jump to the conclusion that this is an adequate solution for a one-phase motor. This is however not the case! Certainly, a switch may toggle between different potentials in theory. In practice is this not possible due to the use of transistors[1]. In order to implement a controller that enables different current directions, two transistors are needed.

One can however study the use of a switched controller compared to a continuous one. Please note that equations cited in chapter 5 were not developed in the scope of this project, for further details refer to [9.1] , unless stated otherwise.

The output voltage *u* can be expressed as in Equation 5.4-1.

$$u = \begin{cases} u_a & s=1 \\ u_b & s=0 \end{cases}$$

Equation 5.4-1

To enhance the switch dependency of the output voltage one can use Equation 5.4-2 which clearly displays that.

$$u = s\cdot(u_a - u_b) = s\cdot u \;\Rightarrow u = \begin{cases} u & s=1 \\ 0 & s=0 \end{cases}$$

Equation 5.4-2

[1] In general is it possible to make an AC output from a transistor via biasing, this is however not possible as the voltage level is high.

As a result, the following equation will be adequate.

$$P = U \cdot I = \begin{cases} \begin{cases} U \cdot I & s = 1 \\ 0 & s = 0 \end{cases} & \textit{Switched controller} \\ U \cdot I & \textit{Continous controller} \end{cases}$$

Equation 5.4-3

Power will thereby only be consumed when the switch is active; while a continuous controller uses power constantly. This also means that the efficiency η significantly will improve due to the following equation.

$$\eta = \frac{P_{in}}{P_{out}} = \begin{cases} \begin{cases} 1 & s = 1 \\ \times & s = 0 \end{cases} & \textit{Switched controller} \\ \prec 1 & \textit{Continous controller} \end{cases}$$

Equation 5.4-4

5.4.2 Sinusoidal Modulation

Under the assumption that attached system is symmetrical, the following voltage reference levels will occur. Based on the power invariant transformation in Equation B.1-6, Equation 5.4-5 will be adequate, and thus following voltage reference levels will be present.

$$\begin{cases} u^*_a = \sqrt{\frac{2}{3}} \cdot u^*_\alpha \\ u^*_b = \frac{1}{\sqrt{2}} \cdot u^*_\beta - \frac{1}{\sqrt{6}} \cdot u^*_\alpha \\ u^*_c = -\frac{1}{\sqrt{2}} \cdot u^*_\beta - \frac{1}{\sqrt{6}} \cdot u^*_\alpha \end{cases}$$

Equation 5.4-5

As all reference values are sinusoidal with the same maximum amplitude as U_{dc} modulations index reaches 1 as the top values match the carrier wave, please refer to Equation 5.3-1 and Figure 5.4-2 also note the scale $V \cdot s^{-1}$.

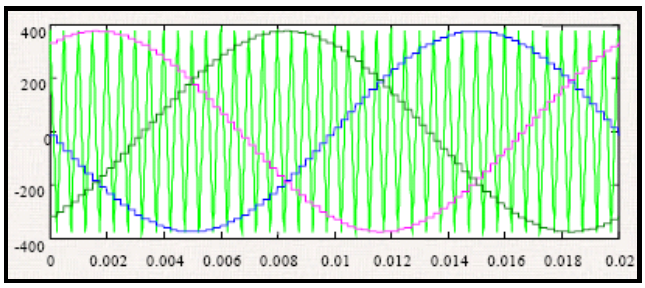

Figure 5.4-2: Sinusoidal Modulation [9.44]

The most critical aspect towards sinusoidal modulation is that m equals 1. As a direct consequence is it impossible to increase the voltage reference as the carrier wave cannot reach higher values.

5.4.3 Symmetrical Modulation

The fact that the modulation index equals 1 for sinusoidal modulation, leads to the need for a modified modulation technique. The symmetrical modulation method is an adapted sinusoidal added with a zero sequence u^*_z which can be described as in Equation 5.4-6.

$$\begin{cases} u*_a = u*_a - u^*_z \\ u*_b = u*_b - u^*_z \\ u*_c = u*_c - u^*_z \end{cases}$$

Equation 5.4-6

By choosing u^*_z wisely, the benefits are cleared, when comparing this method to sinusoidal modulation will be present. A u^*_z according to Equation 5.4-7 leads to a lower modulation index due to the fact that reference voltage will be lowered, note that U_{dc} is unchanged. Please refer to Figure 5.4-3 for visualization, note the scale $V \cdot s^{-1}$.

$$u^*_z = \frac{\max(u^*_a, u^*_b, u^*_c) + \min(u^*_a, u^*_b, u^*_c)}{2}$$

Equation 5.4-7

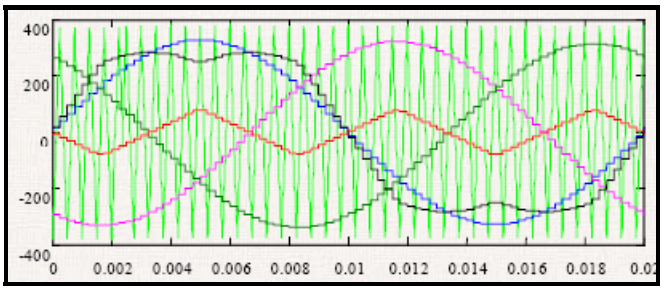

Figure 5.4-3: Symmetrical Modulation [9.45]

Reference voltages may hereby be increased if needed, compared to sinusoidal modulation.

5.4.4 Reduced Switching Modulation

The given modulation method uses same basic configuration as symmetrical modulation. Please refer to Equation 5.4-6, u^*_Z is however different.

$$u^*_z = -\min\left(\frac{U_{dc}}{2} - \max(u^*_a, u^*_b, u^*_c) - \frac{U_{dc}}{2} - \min(u^*_a, u^*_b, u^*_c)\right)$$

Equation 5.4-8

The u^*_z will lead to a modulation display according to Figure 5.4-4. Note that the top voltage value is clamped to U_{dc} the number of switches will be reduced by a third because of this. Only one phase is displayed. Note the scale $V \cdot s^{-1}$.

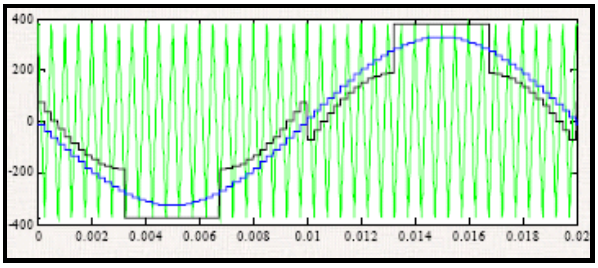

Figure 5.4-4: Reduced Switching Modulation [9.46]

The fact that *m* is less than one leads to the same conclusions as under symmetrical modulation.

5.5 Space Vector Control; theory and implementation

5.5.1 Basic facts

All controllers used in motor control work in similar ways. Despite different implementation techniques, the primary goals are the same; to control the output signal based on the input. Numerous techniques are at hand, but since vector control will mainly be used; only this will be discussed.

Figure 5.5-1 displays a three-phase system with 3 switches or 6 transistors depending on view. The fundamental goal in controlling the system is to toggle correct transistor based on current measurement levels within the system. As 6 transistors are present 8 combinations are possible, each phase potential can be either positive or negative, which gives 2^3 combinations in total.

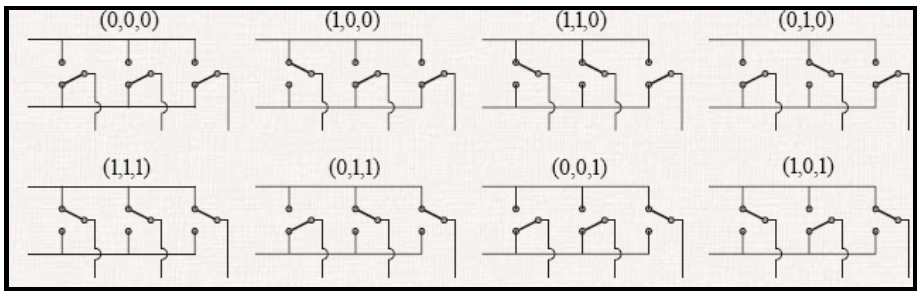

Figure 5.5-1: Possible switch combinations [9.47]

Each vector $\vec{u}_{60X}, x = 0...5$ represents a unique combination of switches, which gives 8 vectors in total, please refer to Figure 5.5-2. Note the two vectors [0, 0, 0] and [1, 1, 1], these vectors are defined as zero vectors because neither combination affects the output voltage. The inactivity is caused by the fact that no difference in potential will be present.

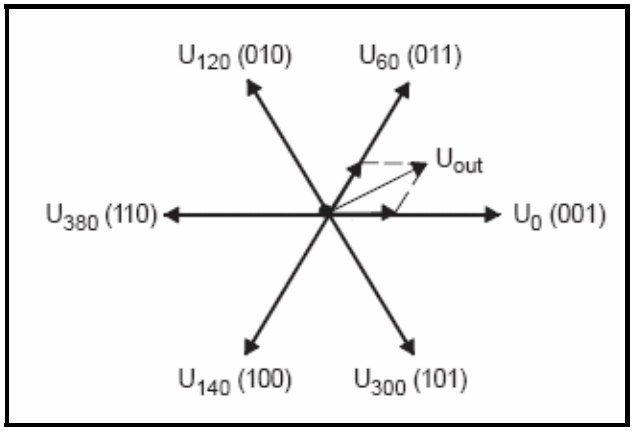

Figure 5.5-2: Space Vector visualization

Based on power invariant transformation, Equation B.1-6, Equation 5.5-1 will be an adequate representation of the different vectors. Clearly, U_{dc} still is the rectified voltage.

$$\begin{cases} \vec{u}(1,0,0) = \sqrt{\frac{2}{3}}U_{dc} = -\vec{u}(0,1,1) \\ \vec{u}(0,1,0) = \sqrt{\frac{2}{3}}U_{dc}e^{\frac{2\pi j}{3}} = -\vec{u}(1,0,1) \\ \vec{u}(0,0,1) = \sqrt{\frac{2}{3}}U_{dc}e^{\frac{4\pi j}{3}} = -\vec{u}(1,1,0) \\ \vec{u}(0,0,0) = 0 = \vec{u}(1,1,1) \end{cases}$$

Equation 5.5-1

The vector U_{out} is based on adjacent vectors; clearly, U_{out} is based on three vectors, two active vectors and one passive, zero vector. The combination is determined within the controller. The implemented regulator calculates a different operation time for each vector combination, which yields U_{out}.

Thus

$$U_{out} = U_x \cdot t_1 + U_{x+60}t_2 + U_z * (T - t_1 + t_2)$$

Equation 5.5-2

5.5.2 Vector Time Control

In order to enable desired U_{out}, time control is imperative. The control principal is based on the fact that different vectors need different operating times. By calculating two different switching times, U_{out} may be produced. Each switching time is calculated based on actual voltage levels in the [d, q]-frame.

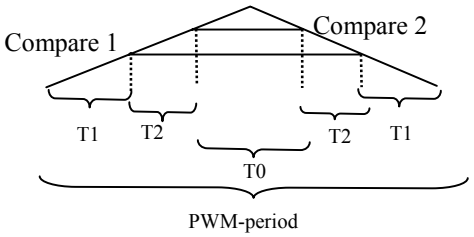

Figure 5.5-3: Visualization of switching period

During one switching period three different vectors are used; one start vector that is utilized until T1 is reached the next following vector is loaded into the corresponding register. When T2 is reached, a zero vector is called, based on the start vector [0,0,0] or [1,1,1] that is used. The pattern is inverted on the right-hand side. All time values are calculated continuously within the control algorithm.

5.6 DSP Implementation

The F2812 has a built-in support for generation of space vectors. This feature truly facilitates implementation within the algorithm. However, as a user, certain decision needs to be made in order to enable space vector handling. These are all stated below with more detailed explanations.

- Selection of Software/Hardware switching

This option is used to allow the user to use both software[1] and hardware[2] switching techniques. Depending on the technique, different settings will be used. The SW switching requires adequate vectors on every given switching time to perform while the HW only needs a start vector within every switching period.

Within every switch period, the following occurs regardless of technique.

- Adequate vector is loaded into DSP controller register

Based on technique this is either [0,0,0] in SW or any other vector for HW, [1,1,1] is not an acceptable start vector.

As the carrier wave intersects a predefined value i.e. compare 1 vector changes will take place. Following occurs in respectively SW and HW techniques

- SW: User must define next following vector to be used
- HW: The built-in support sets the new vector. This is the next following/prior vector based on the direction of motor rotation.

On the second intersection, i.e. compare 2, a zero vector will be loaded into the register. This is either done by the software or automatically. As the counter number decreases the same vectors will be used in reverse order. Note that 7 vectors are to be used in the SW, method meanwhile 5 are used in HW mode. The difference in vectors used is due to the usage/ending of zero vectors in SW mode.

In order to set the correct start vector the software must therefore be able to derive what sector the [d, q]-frame is located in. Thus an angle calculation or estimation is vital.

The built-in HW mode is used in the implemented algorithm due to its advantages.

Following pseudo code describes the usage.

```
/*---Prior Code ends here-----*/
//HW mode is selected
//Enable space vector
EvaRegs.ACTRA.D = XXX; //Start vector is selected
```

[1] Denoted SW from here on
[2] Denoted HW from here on

```
EvaRegs.TxCMP1 = yyy ; //Set compare value 1
EvaRegs.TxCMP2 = zzz ;//Set compare value 2
//-------Action block-----------//
```

A given start vector [0,1,1] will result in following output order.
[0,0,1] => [0,1,1] => [1,1,1] => [0,1,1] => [0,0,1]

Figure 5.6-1 visualise this. Each direction change is caused by an intersection between the carrier wave and compare X. Note the usage of zero vectors causing both channels to be simultaneously high.

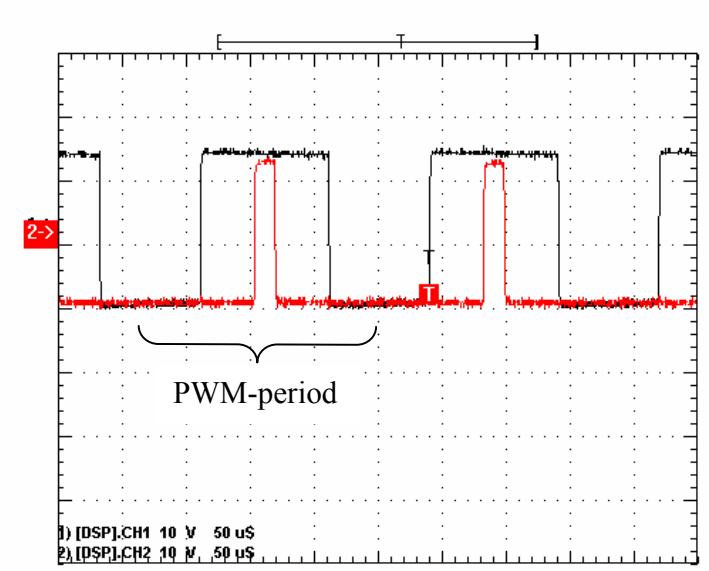

Figure 5.6-1 Vector changes during one period

The zero-vector is either [0,0,0] or [1,1,1]. The selection is based on the fact that only one bit is allowed to change within every step. As this is done within the DSP no respect needs to be taken. Yet in SW mode, the user must be aware of this as the software selects every vector.

6 Power in a generic three-phase system

6.1 General information

This is not imperative knowledge for the implemented system; however controlled current and voltage within the system controls the load power. Based on this, following section is included.

A generic three-phase load according to Figure 6.1-1 may be expressed according to Equation 6.1-1.

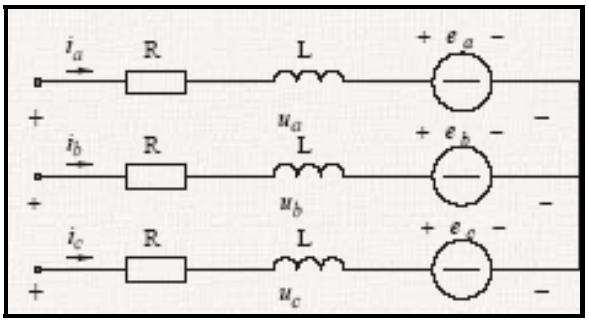

Figure 6.1-1: Generic 3-phase system [9.48]

By measurements of i_a, and i_b conclusions about the system can be made.
Power invariant transformation gives the following voltage levels:

$$\begin{cases} \sqrt{\frac{2}{3}}u_a = R\cdot i_a + L\cdot\frac{di_a}{dt} + j\cdot\omega\cdot L\cdot i_a + e_a \\ \sqrt{\frac{2}{3}}u_b = R\cdot i_b + L\cdot\frac{di_b}{dt} + j\cdot\omega\cdot L\cdot i_b + e_b \\ \sqrt{\frac{2}{3}}u_c = R\cdot i_c + L\cdot\frac{di_c}{dt} + j\cdot\omega\cdot L\cdot i_c + e_c \end{cases} \Leftrightarrow \vec{u}^{\alpha\beta} = R\cdot\vec{i}^{\alpha\beta} + L\cdot\frac{d\vec{i}^{\alpha\beta}}{dt} + j\cdot\omega\cdot L\cdot\vec{i}^{\alpha\beta} + \vec{e}^{\alpha\beta}$$

Equation 6.1-1

The active power p(t) can be expressed as:

$$p(t) = \mathrm{Re}\left\{\vec{u}^{\alpha\beta}\cdot\vec{i}^{\alpha\beta}\right\} = \mathrm{Re}\left\{\left(R\cdot\vec{i}^{\alpha\beta} + L\cdot\frac{d\vec{i}^{\alpha\beta}}{dt} + j\cdot\omega\cdot L\cdot\vec{i}^{\alpha\beta} + \vec{e}^{\alpha\beta}\right)\cdot\vec{i}^{\alpha\beta}\right\} =$$

$$\left(R\cdot i_d{}^2\right) + \left(R\cdot i_q{}^2\right) + L\left(\frac{di_d}{dt}i_d + \frac{di_q}{dt}i_q\right) + e_q\cdot i_q$$

Equation 6.1-2

The assumption that Equation 6.1-2 $\approx i_q\cdot|e|$, gives that $P(t)\propto i_q$ and thereby $T(t)\propto i_q$; because $T(t) = i_q\cdot\psi$ this is accurate under the assumption that i_d equals zero.

6.2 Current ripple

Equations from here on are to be found in [9.3] unless stated otherwise, please refer to it for details. As $\vec{u}$ in a generic circuit may be expressed as in Equation 6.1-1 the following simplified equation will be adequate:

$$\vec{u} = R\vec{i} + L\frac{d\vec{i}}{dt} + \vec{e} \approx \frac{d\vec{i}}{dt} = \frac{\vec{u} - \vec{e}}{L}$$

Equation 6.2-1

As $\vec{u}$ is a complex vector Equation 6.2-1 can be separated into real and imaginary parts according to Equation 6.2-2.

$$\frac{d\vec{i}}{dt} = \frac{\vec{u} - \vec{e}}{L} \Rightarrow \begin{cases} \frac{di_\alpha}{dt} = \frac{u_\alpha - e_\alpha}{L} \\ \frac{di_\beta}{dt} = \frac{u_\beta - e_\beta}{L} \end{cases} \approx \begin{cases} \frac{\Delta i_\alpha}{\Delta t} = \frac{u_\alpha - e_\alpha}{L} \\ \frac{\Delta i_\beta}{\Delta t} = \frac{u_\beta - e_\beta}{L} \end{cases} \Leftrightarrow \begin{cases} \Delta i_\alpha = \frac{u_\alpha - e_\alpha}{L}\Delta t \\ \Delta i_\beta = \frac{u_\beta - e_\beta}{L}\Delta t \end{cases}$$

Equation 6.2-2

By usage of coordinate transformation according to *Coordinate Transformation, pB-3,* Equation 6.2-3 leads to:

$$\begin{cases} \Delta i_d = \frac{1}{L}\left(\left(u_\alpha \cos\theta + u_\beta \sin\theta\right) - \left(e_\alpha \cos\theta + e_\beta \sin\theta\right)\right) \cdot \Delta t \\ \Delta i_q = \frac{1}{L}\left(\left(u_\beta \cos\theta - u_\alpha \sin\theta\right) - \left(e_\beta \cos\theta - e_\alpha \sin\theta\right)\right) \cdot \Delta t \end{cases}$$

Equation 6.2-3

According to Equation 6.1-2 only i_q affects active power in a given circuit, a calculation of Δi_q will thereby reveal variation in the output power. However, a ripple in i_d will result in a ripple in magnetic flow ψ, and must therefore be taken into consideration. Note that Δt represents the time each space vector is used.

7 Speed and Torque Controller

The control system is based on two/three different PI controllers. In a PMSM application two controllers are used, i.e. one inner and one outer controller loop. The inner loop is a current controller meanwhile the outer act as a torque controller. The control system may be extended with a third PI controller that thereby is used for the DC machine.

7.1 Current Controller

7.1.1 PI Controller

A general description of a PI controller follows. Both u_d and u_q are based on *Three-phase and Coordinate Transformation, pB-1*.

$$\begin{cases} u_d = L \cdot \dfrac{d}{dt} i_d + R i_d - \omega L i_q + e_d \\ u_q = L \cdot \dfrac{d}{dt} i_q - R i_q + \omega L i_d + e_q \end{cases}$$

Equation 7.1-1

Since a discrete controller should be implemented, some assumptions have to be made:

$$\begin{cases} e_{d\,(k)} = e_d\,(k-1) \\ i_{d\,,(k)} = i_d\,, ref\,(k-1) \\ i_{q\,,(k)} = i_q\,, ref\,(k-1) \end{cases}$$

Equation 7.1-2

Based on Equation 7.1-2, Equation 7.1-1 may be expressed as:

$$\begin{cases} u_{d\,,(k)} = L \cdot \dfrac{di_d}{dt} + R i_d - \omega L i_q + e_{d(k-1)} \\ u_{q,(k)} = L \cdot \dfrac{di_q}{dt} - R i_q + \omega L i_d + e_{q\,(k-1)} \end{cases}$$

Equation 7.1-3

To estimate $\dfrac{di_d}{dt}$ and $\dfrac{di_q}{dt}$ different options are at hand, as previous values are known, backwards Euler approximation is used, please refer to [9.4] for more details. As a result, the following statement is valid.

$$\begin{cases} \dfrac{di_d}{dt} = \dfrac{i_{d,(k)} - i_{d,(k-1)}}{T_s} \\ \dfrac{di_q}{dt} = \dfrac{i_{q,(k)} - i_{q,(k-1)}}{T_s} \end{cases} \Rightarrow \begin{cases} u_{d,(k)} = L \cdot \dfrac{i_{d,(k)} - i_{d,(k-1)}}{T_s} + Ri_d - \omega L i_q + e_{d(k-1)} \\ u_{q,(k)} = L \cdot \dfrac{i_{q,(k)} - i_{q,(k-1)}}{T_s} - Ri_q + \omega L i_d + e_{q(k-1)} \end{cases}$$

Equation 7.1-4

Finally, i_d, and i_q are assumed to fulfill Equation 7.1-5.

$$\begin{cases} i_d = \dfrac{i_{d,(k)} + i_{d,(k-1)}}{2} \\ i_q = \dfrac{i_{q,(k)} + i_{q,(k-1)}}{2} \end{cases}$$

Equation 7.1-5

This as a result leads to:

$$\begin{cases} u_{d,(k)} = L \cdot \dfrac{i_{d,(k)} - i_{d,(k-1)}}{T_s} + R\dfrac{i_{d,(k)} + i_{d,(k-1)}}{2} - \omega L \dfrac{i_{q,(k)} + i_{q,(k-1)}}{2} + e_{d(k-1)} \\ u_{q,(k)} = L \cdot \dfrac{i_{q,(k)} - i_{q,(k-1)}}{T_s} - R\dfrac{i_{q,(k)} + i_{q,(k-1)}}{2} + \omega L \dfrac{i_{d,(k)} + i_{d,(k-1)}}{2} + e_{q(k-1)} \end{cases}$$

Equation 7.1-6

Based on given assumptions:

$$\begin{cases} u_{d,(k)} = L \cdot \dfrac{i_{d,ref(k)} - i_{d,(k)}}{T_s} + R\dfrac{i_{d,ref(k)} + i_{d,(k)}}{2} - \omega L \dfrac{i_{q,ref(k)} + i_{q,(k)}}{2} + e_{d(k)} \\ u_{q,(k)} = L \cdot \dfrac{i_{q,ref(k)} - i_{q,(k)}}{T_s} - R\dfrac{i_{q,ref(k)} + i_{q,(k)}}{2} + \omega L \dfrac{i_{d,ref(k)} + i_{d,(k)}}{2} + e_{q(k)} \end{cases}$$

Equation 7.1-7

This may be interpreted as a P controller:

$$\begin{cases} u_{d,(k)} = \left(\dfrac{L}{Ts} + \dfrac{R}{2}\right)(i_{d,ref(k)} - i_{d,(k)}) + Ri_{d,(k)} - \omega L \dfrac{i_{q,ref(k)} + i_{q,(k)}}{2} + e_{d(k)} \\ u_{q,(k)} = \left(\dfrac{L}{Ts} + \dfrac{R}{2}\right)(i_{q,ref(k)} - i_{q,(k)}) - Ri_{q,(k)} + \omega L \dfrac{i_{d,ref(k)} + i_{d,(k)}}{2} + e_{q(k)} \end{cases}$$

Equation 7.1-8

By construing $i_{d,\,k}$ and $i_{q,\,k}$ according to Equation 7.1-9.

$$\begin{cases} i_{d\,,(k)} = \sum_{n=0}^{k-1} i_{d\,,ref\,(n)} - i_{d\,(n)} \\ i_{q\,,(k)} = \sum_{n=0}^{k-1} i_{q\,,ref\,(n)} - i_{q\,(n)} \end{cases}$$

Equation 7.1-9

One can express Equation 7.1-8 as Equation 7.1-10.

$$\begin{cases} u_{d\,,k} = \left(\frac{L}{Ts} + \frac{R}{2}\right)\left((i_{d\,,ref\,(k)} - i_{d(k)}) + \frac{1}{\left(\frac{L}{R \cdot T_s} + \frac{1}{2}\right)} \sum_{n=0}^{k-1} \left(i_{d\,,ref\,(n)} - i_{d\,}(n)\right) \right) - \frac{\omega L}{2} \cdot \left(i_{q\,,ref\,(k)} + i_{q\,,(k)}\right) + e_{d(k)} \\ u_{q,k} = \left(\frac{L}{Ts} + \frac{R}{2}\right)\left((i_{q\,,ref\,(k)} - i_{q(k)}) - \frac{1}{\left(\frac{L}{R \cdot T_s} + \frac{1}{2}\right)} \sum_{n=0}^{k-1} \left(i_{q\,,ref\,(n)} - i_{q\,}(n)\right) \right) + \frac{\omega L}{2} \left(i_{d\,,ref\,(k)} + i_{d\,,(k)}\right) + e_{q\,(k)} \end{cases}$$

Equation 7.1-10

Hereby the following parameters for a PI controller may be set:

$$K = \left(\frac{L}{Ts} + \frac{R}{2}\right) \quad T_i = \frac{1}{\left(\frac{L}{R \cdot T_s} + \frac{1}{2}\right)} \quad K_c = \frac{\omega L}{2}$$

Equation 7.1-11

Thus, the following PI controller will be used:

$$\begin{cases} u_{d\,(k)} = K\left((i_{d\,,ref\,(k)} - i_{d(k)}) + \frac{1}{T_i} \sum_{n=0}^{k-1} i_{d\,,ref\,(n)} - i_{d\,}(n) \right) - K_c \left(i_{q\,,ref\,(k)} + i_{q\,(k)}\right) + e_{d(k)} \\ u_{q(k)} = K\left((i_{q\,,ref\,(k)} - i_{q\,(k)}) - \frac{1}{T_i} \sum_{n=0}^{k-1} i_{q\,,ref\,(n)} - i_{q\,}(n) \right) + K_c \left(i_{d\,,ref\,(k)} + i_{d(k)}\right) + e_{q\,(k)} \end{cases}$$

Equation 7.1-12

Note that Equation 7.1-12 is a general expression; L_q and L_d replaces L in a PMSM with salient poles

7.2 Torque Controller

7.2.1 PI Controller

The outer PI controller generates a current reference value according to Equation 7.2-2.

$$T_{ref} = K \cdot \left((\omega_{ref} - \omega) + \frac{1}{Ti} \cdot \sum_{N} \left(\omega_{ref} - \omega_k \right) \right)$$

Equation 7.2-1

$$T_{ref} \Rightarrow Iq_{ref} \; thus \; Iq_{ref} = \frac{T_{ref}}{\psi}$$

Equation 7.2-2

Please refer to Figure 7.2-1 for a system layout.

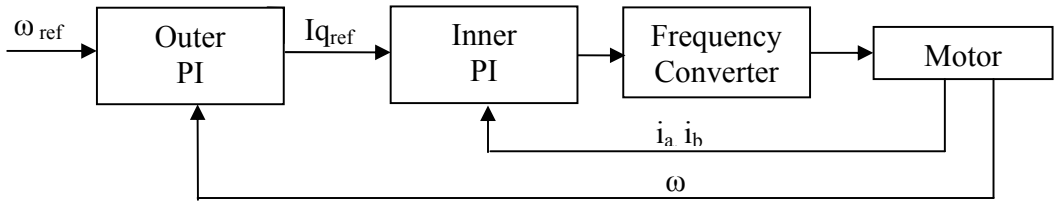

Figure 7.2-1: PMSM Controller layout

An expanded system that includes a DC machine uses a PI controller to derive the T_{ref} for the DC machine. The controller is therefore based either on T_{ref} or on a predefined speed, ω_{ref}. A combination of the systems is also possible

.

7.3 Program development

The following flowchart gives the overall idea behind the implemented program. The interrupt routine is triggered on TIMER underflow. That is every time the timer reaches zero.

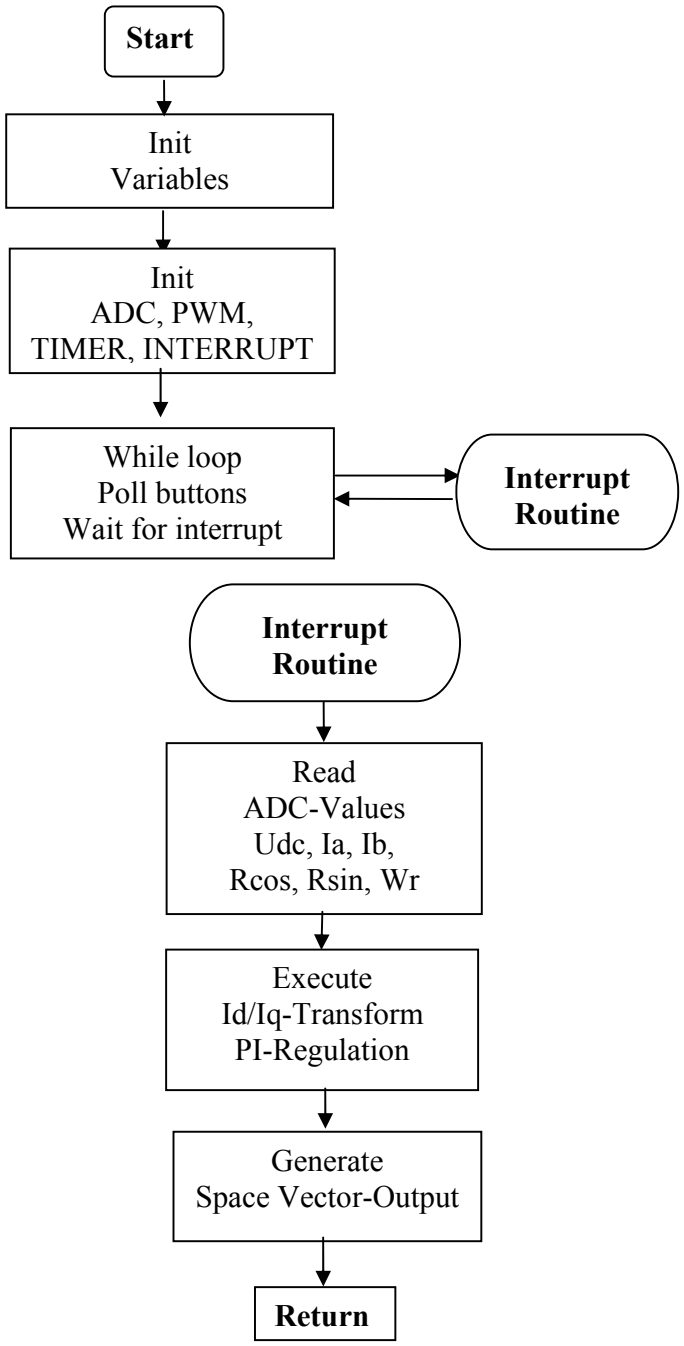

Figure 7.3-1: Controller flowchart

7.4 Code Composer Studio

Following pseudo code describes the controller implementation; please refer to *C-Code, pC-1* for the complete program.

```
Main(){
//Init_ADC();
//Init_PWM();
//Init_Buttons();
//Init_Diode();
While(1){
//Wait for Action
//Read ADC
// Motor_Control();
}
```

8 Interface card Settings and Layout

8.1 Laboratory setup

The used lab system consisted of two different motors, one PMSM and one DC machine with their shafts connected as one. Please refer to Figure 8.1-1

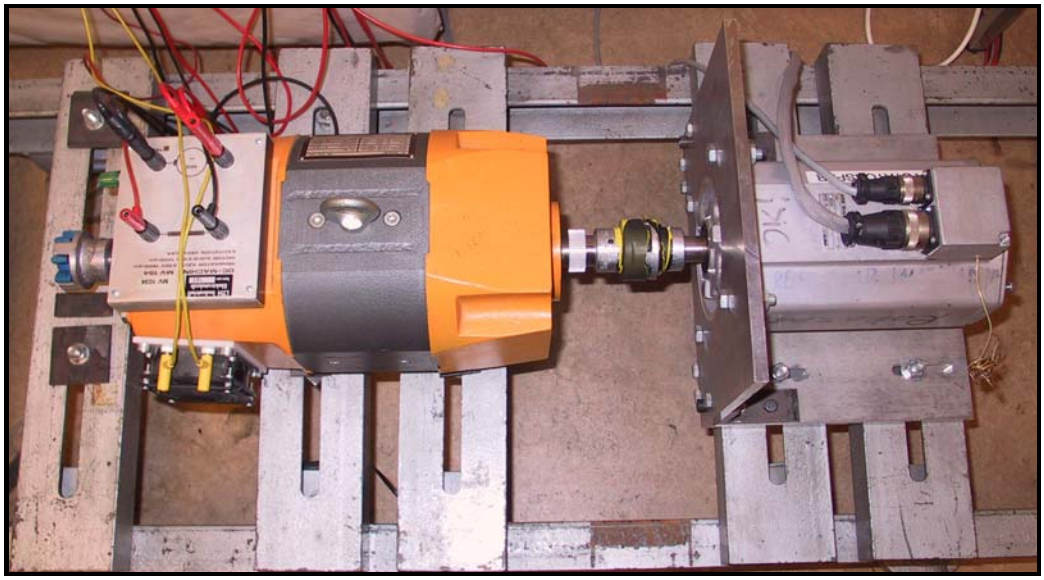

Figure 8.1-1: The laboratory setup

The total system is controlled via the developed card together with the frequency converter. Two external units are also used; these are a resolver unit combined with a measurement unit. See Figure 8.1-2 for details.

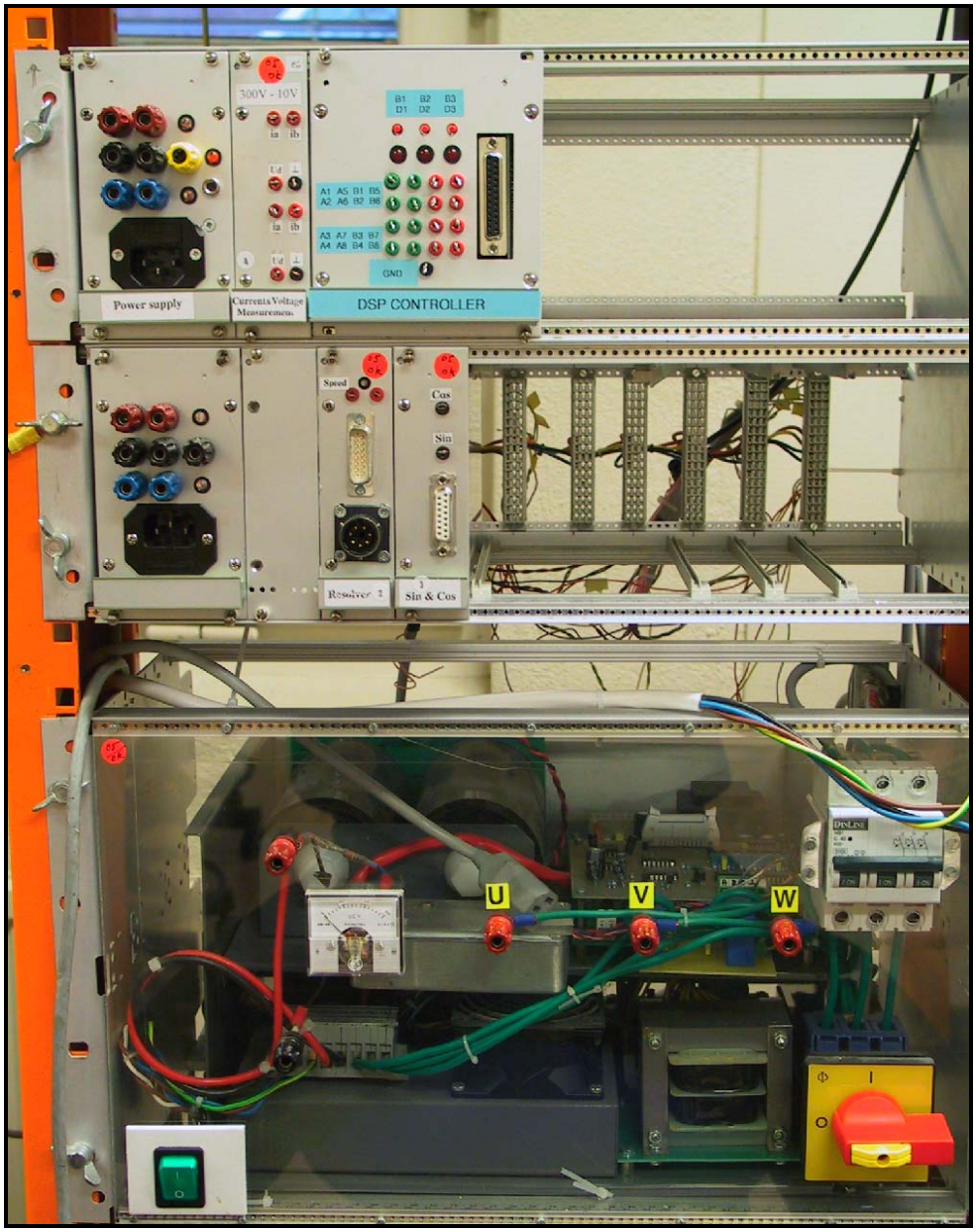

Figure 8.1-2: Complete laboratory setup excluding motors.

The following cards are included above, starting from the top.

1. Power Supply, Current and Voltage measurement card and DSP Controller.
2. Power Supply, Resolver card: Omega, Sinus and Cosine signals.
3. Frequency converter.

8.1.1 External Connections

The following signals are used in the lab setup.

- ω_r i.e. Electrical rotation speed
- U_{dc} i.e. Supply Voltage
- i_a i.e. Phase current a
- i_b i.e. Phase current b
- Sin(θ) i.e. Sinus for the motor angle
- Cos(θ) i.e. Cosine for the motor angle

All pins all placed in the same order as they are connected into the backplane, thus the following ADCx channels are used

ADA0 ⇔Wr
ADA1⇔ Udc
ADA2 ⇔ Ia
ADA3 ⇔ Ib
ADA4 ⇔Sin(θ)
ADA5 ⇔ Cos(θ)

All channels are adjusted to specified signal. Based on the signal, the gain/offset potentiometers were adjusted to give a voltage range [0.5, 2.5] on the input ADC pin on the DSP. By doing so, one get a safety margin for disturbing signals that may occur and can be fatal to the DSP.

WARNING: User is advised to read the following points before the system is installed.

- The channel ADA1 is adjusted to fit the current signals, i.e. no negative voltage occurs. The channel is hereby **not** $\pm 10V$ tolerant. An input level of –10V will undoubtedly cause fatal damage to the DSP.

- When a change in given, gain/offset is needed and damage may occur to the ADCx, in order to prevent this **removal** of the cable between the interface card and the DSP is recommended.

- All **unused** ADCx channels are set to be approximately $\pm 10V$ tolerant. Each channel needs therefore to be separately adjusted before usage.

8.2 Result/Conclusion

The outline of the development were based on following criterion

- Input measurement signals $\pm 10V$
- DSP input [0,3] V
- PWM signal output [-15,15] or [0,15] V
- 3 Control buttons placed on the front panel
- 3 Indicator light emitting diodes placed on the front panel
- 16 ADC-channels should be measurable on the front panel
- As a confirmation test of the interface circuit, at least one motor controller should be implemented in the DSP.

The final product fulfills all prior demands; the interface card is designed to be universal as all ADC and PWM channels may be utilized. Since all buttons/diodes are individually programmable, different operations may be used.

The lab system consists of two different motors; thereby different implementation techniques may be used. A controller can be based either on the PMSM, DC machine or on both combined. As the implemented controller's main task was to verify the interface transformations[1], only one motor was used. The implemented controller is based on two PI regulators i.e. one inner and one outer controller, according to prior specifications.

In order to measure omega a resolver unit was used, please refer to Figure 8.1-2. The output signal is said to be 80 rad/s at 1V. Unfortunately the resolver signal is very noise and therefore hard to measure accurately. A filter was also implemented on the input signal to stabilize the value. Since the presented solution not is enough to guarantee a correct level, the regulation of omega is obstructed. Two different reference values are used to verify the interface card.

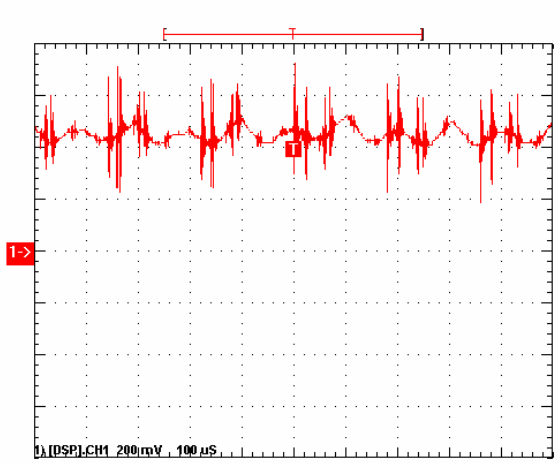

Figure 8.2-1: Reference value of Omega equals 40 rad/s

[1] From a project point of view that is

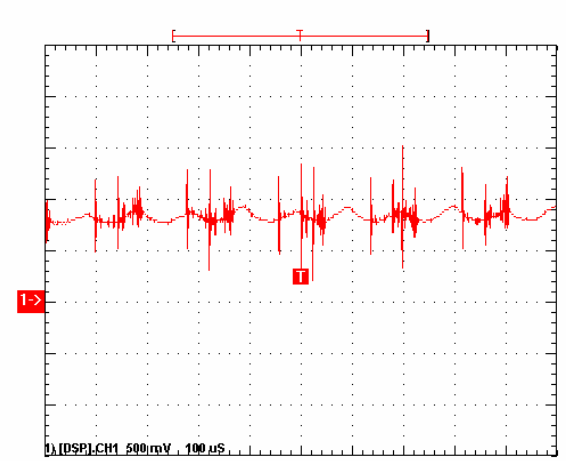

Figure 8.2-2 : Reference value of Omega equals 80 rad/s

The deviation is clearly shown in the figure, however the absolute value of this is impossible to determin due to the fact that Omega not is exactly 80 rad/s.

8.3 Improvements

The following improvements are suggested in further development.

- Make separate cards for PWM and ADC signals (to make more space between components, which implies easier fault detection).

- Remove wires on the Top-side of the card at the IC-sockets (to avoid faulty soldering).

- Wires from ADCxx should be placed in a common socket such as a HHEAD, contrary to the current design with individual wires.

- If possible, reduce power consumption and heat radiation, as a second option increase power handling capacity at critical points, i.e. use 0.5W resistors instead of 0.25W.

Many of the suggested improvements will be overruled when a professional manufactured interface card is used. This ensures better soldering and as a result of this more surface mounted components may be used.

9 References

9.1 Books

[9.1] Power electronic control, Mats Alaküla, pp.13-39, KFS, 2003

[9.2] Design with operational amplifiers and analog integrated circuits, Sergio Franco, pp 18-19, ISBN: 0-07-112173-0, 2002

[9.3] Power Electronics, Department of Industrial Electrical Engineering and Automation, Lund Institute of Technology, Per Karlsson pp15-43,

[9.4] Scientific Computing An Introduction Survey, Second Edition, Michael T Heath, p 398-399, ISBN: 0-07-112229-X

9.2 Datasheets components

[9.21] 7407
http://focus.ti.com/lit/ds/symlink/sn7407.pdf 7/19/2006

[9.22] TL074
http://focus.ti.com/lit/ds/symlink/tl074.pdf 7/19/2006

[9.23] LF347
http://www.national.com/ds/LF/LF147.pdf 7/13/2006

[9.24] LM311
http://www.national.com/ds/LM/LM111.pdf 7/13/2006

[9.25] EL2044
http://www.intersil.com/data/fn/fn7059.pdf 7/13/2006

[9.26] MAX 326
http://pdfserv.maxim-ic.com/en/ds/MAX326-MAX327.pdf 7/13/2006

[9.27] LT1230
http://www.linear.com/pc/downloadDocument.do?navId=
H0,C1,C1154,C1009,C1026,P1479,D3782 2006-10-16

9.3 Datasheets DSP

[9.31] Technical reference
http://c2000.spectrumdigital.com/ezf2812/docs/ezf2812_techref.pdf 7/13/2006

[9.32] Analog-to-Digital Conversion (ADC)
http://focus.ti.com/lit/ug/spru060d/spru060d.pdf 8/15/2006

[9.33] Event Manager (EV) Reference Guide
http://focus.ti.com/lit/ug/spru065d/spru065d.pdf 8/30/20068/

[9.34] System Control and Interrupts Reference Guide
http://focus.ti.com/lit/ug/spru078c/spru078c.pdf 8/15/2006

9.4 Figures

[9.41] http://c2000.spectrumdigital.com/ezf2812/docs/ezf2812_techref.pdf, p17, fig 2-3 and table 2, 8/29/2006

[9.42] Power electronic control, Mats Alaküla, p15, fig 3-2[1], KFS, 2003

[9.43] Power electronic control, Mats Alaküla, p27, fig 3-15[2], KFS, 2003

[9.44] Power electronic control, Mats Alaküla, p29, fig 3-19[2], KFS, 2003

[9.45] Power electronic control, Mats Alaküla, p31, fig 3-21[2] KFS, 2003

[9.46] Power electronic control, Mats Alaküla, p33, fig 3-23[2], KFS, 2003

[9.47] Power electronic control, Mats Alaküla, p27, fig 3-16[2], KFS, 2003

[9.48] Power electronic control, Mats Alaküla, p37, fig 4-3[2], KFS, 2003

[9.49] Power electronic control, Mats Alaküla, p38, fig 4-4[2], KFS, 2003

[1] Used figure is to be found in http://www.iea.lth.se/sed/L2.pdf 8/17/2006
[2] Used figures are to be found in http://www.iea.lth.se/sed/L5.pdf 8/17/2006

Appendix A Circuit characteristics

Channel 1 represents input voltage, while channel 2 represents circuit output in all presented plots.

A.1 LM311

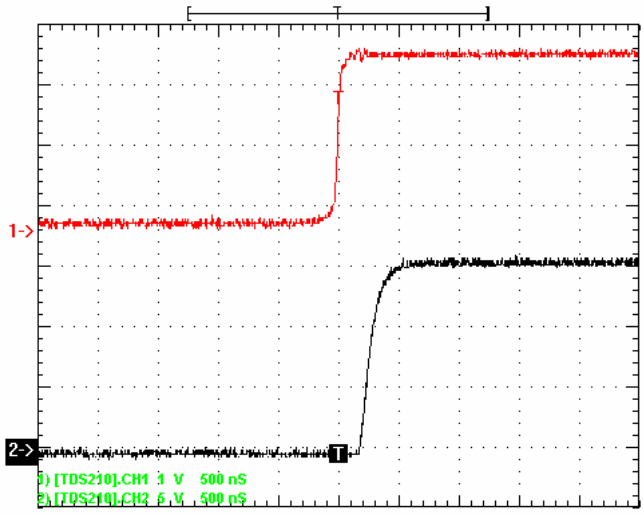

Figure A.1-1: Rise time for a LM311

As the threshold of the LM311 is reached the output will toggle, the output levels will thereby be+15V. A delay of approximately 250 ns can be detected as the signal toggles from low to high. Similar delay may be found on the opposite signal side, this is displayed in Figure A.1-2.

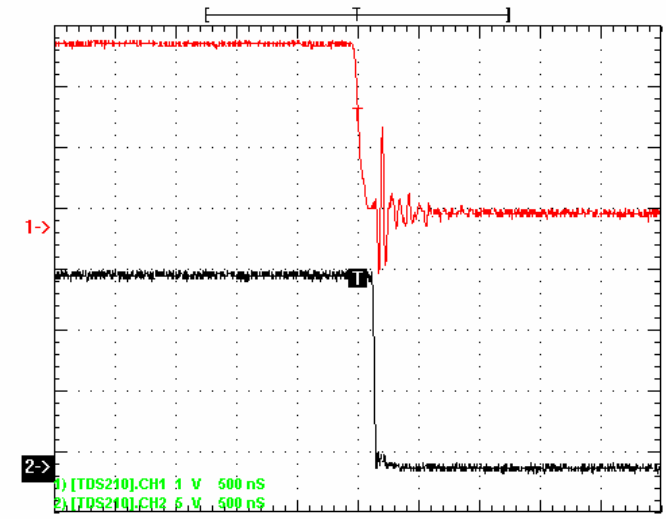

Figure A.1-2: Fall time for a LM311

A.2 LM311 with/without pull-up

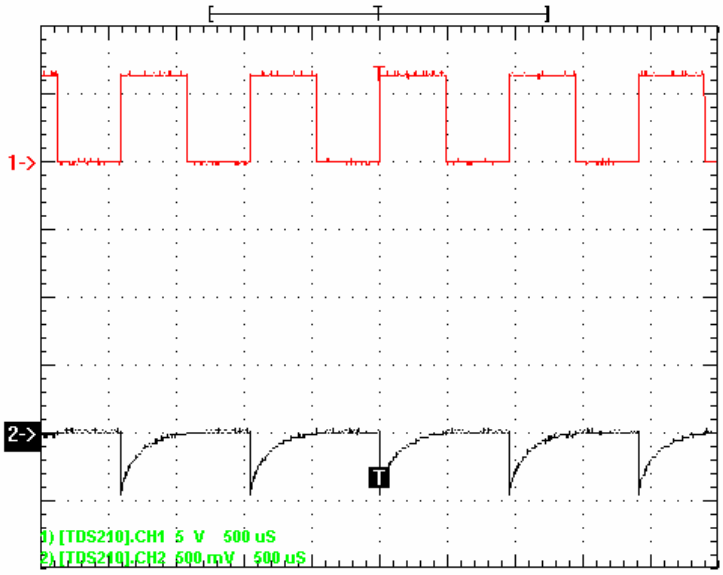

Figure A.2-1: A LM311 without pull-up

Studies have proven that a pull-up resistor at the output is vital to achieve adequate behavior. Without a pull-up resistor, output voltage level never rises above the zero level, regardless of supply voltage. The resistor used in the comparison holds a value of 2.2k Ω. This value is chosen as a tradeoff between different parameters. It is therefore possible to find a better value in some aspects, as only limited number of values has been tested better overall values may exist. Due to the internal design, only [0,15] V may be delivered at the output.

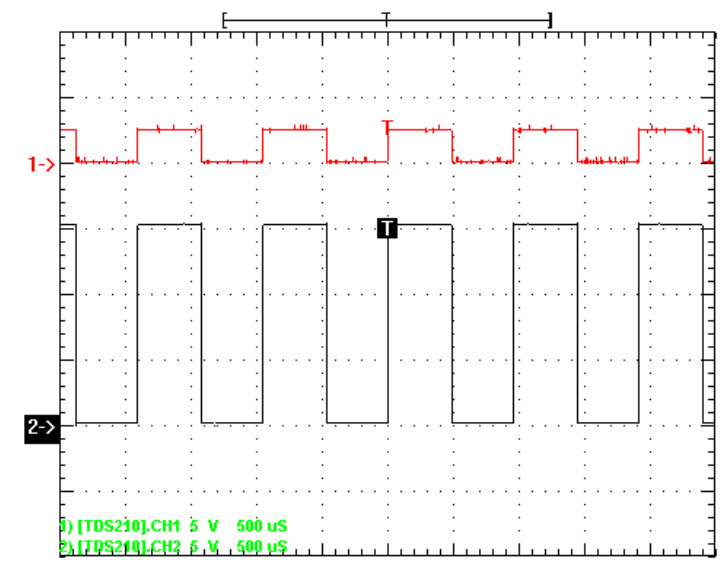

Figure A.2-2: A LM311 with pull-up

A.3 LM339

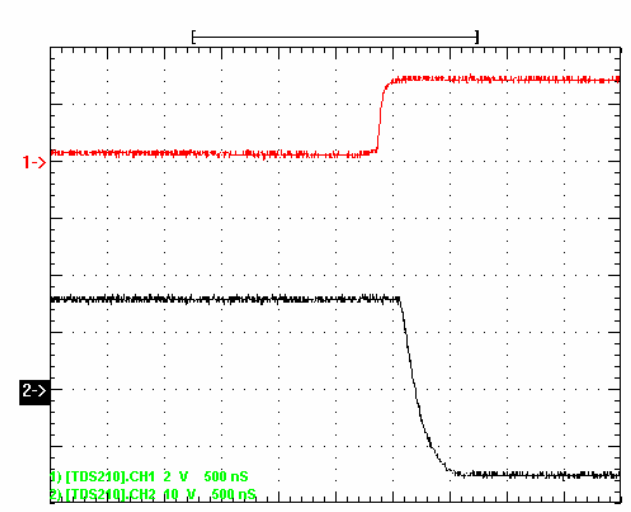

Figure A.3-1: Rise time for a LM339

Note the input voltage at [0. 3.3] V; the output levels are [-15,15] V. A delay is present in both Figure A.3-1 and Figure A.3-2.

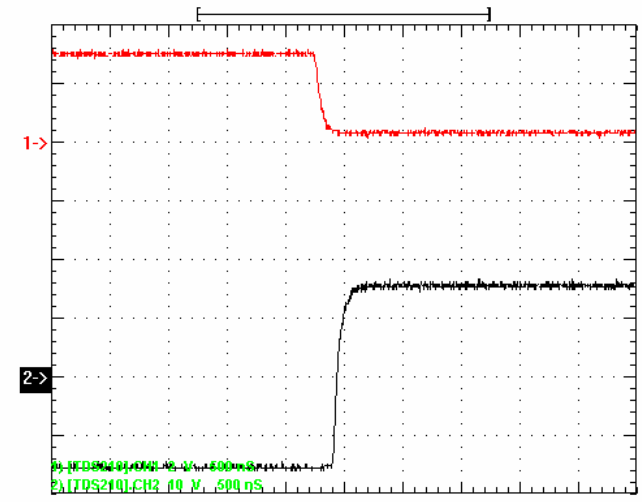

Figure A.3-2: Fall time for a LM339

A.4 LM339 with/without pull-up

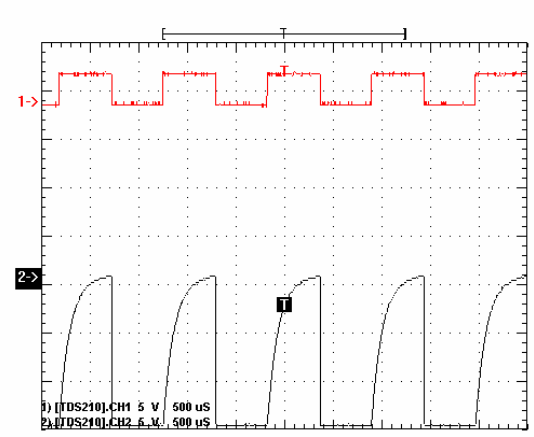

Figure A.4-1: A LM339 without pull-up

Same output characteristics are to be found for a LM339 with/without pull-up. Yet, differences in the design give a possible output of $\pm 15\,V$.

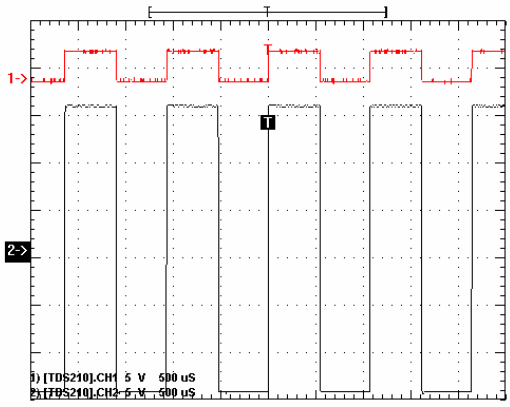

Figure A.4-2: A LM339 with pull-up

A.5 7407

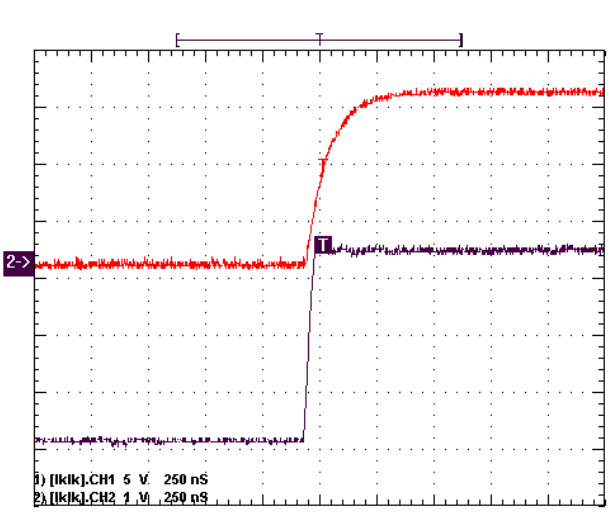

Figure A.5-1: Rise time for a HC4707

Note that the delay of approximately 25ns in Figure A.5-2.

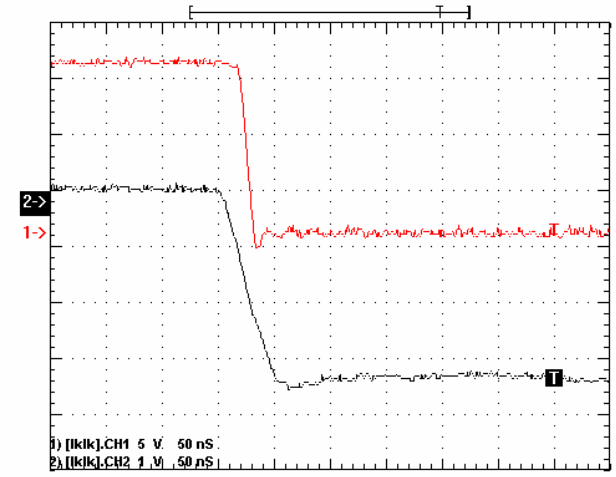

Figure A.5-2: Fall time for a HC4707

A.6 LF347

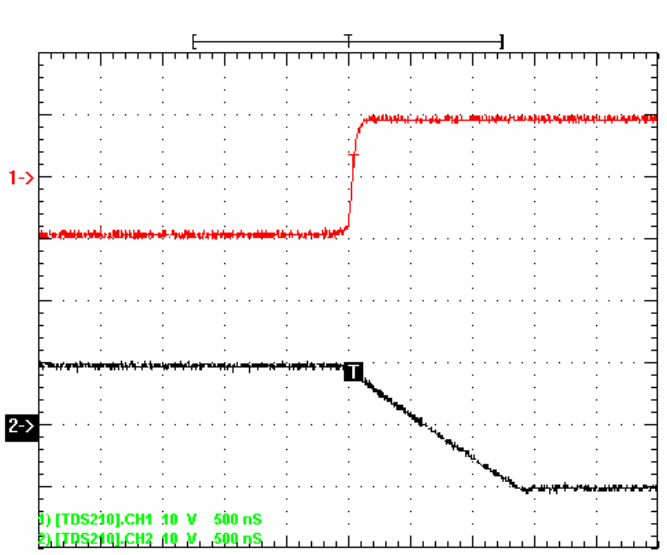

Figure A.6-1: Rise time for a LF347

No delay is present, either on rise nor fall time.

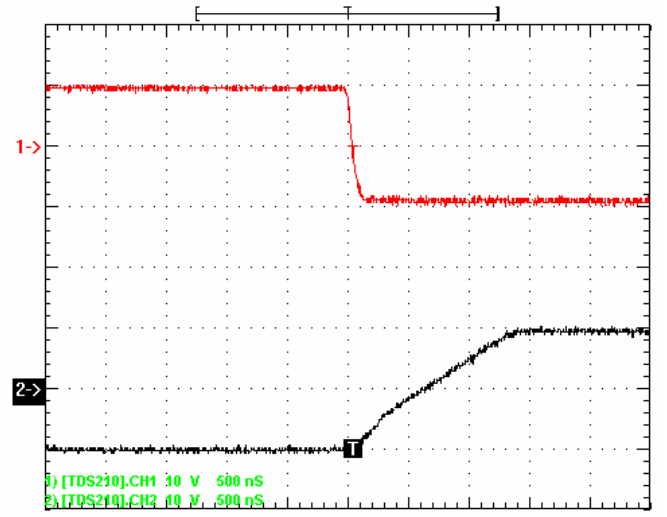

Figure A.6-2: Fall time for a LF347

A.7 TL074

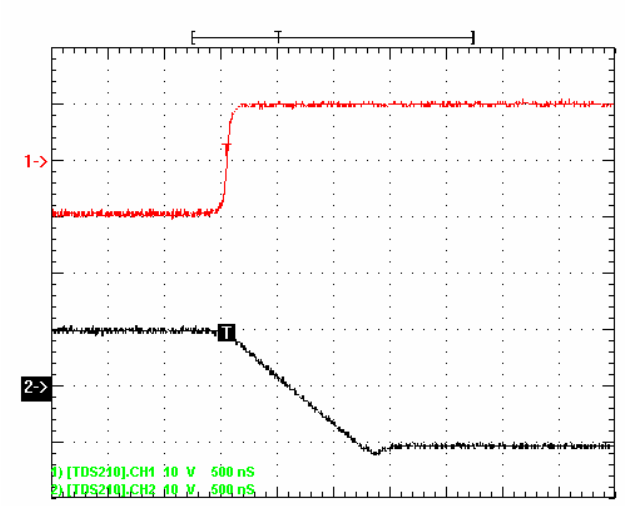

Figure A.7-1: Rise time for a TL074

No delay is present, the respectively rise and fall times are roughly the same.

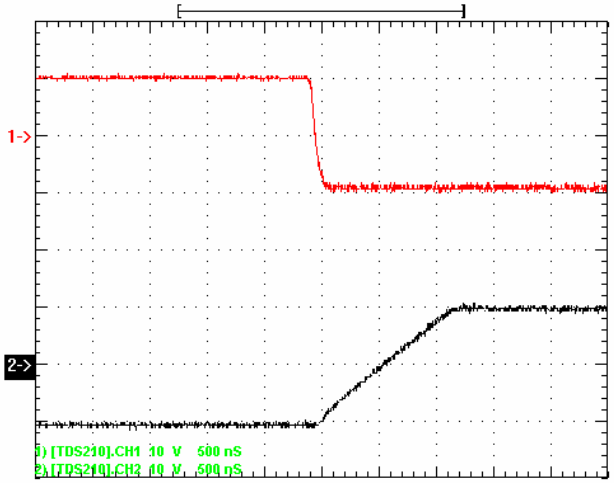

Figure A.7-2: Fall time for a TL074

A.8 EL2044 vs. LM347

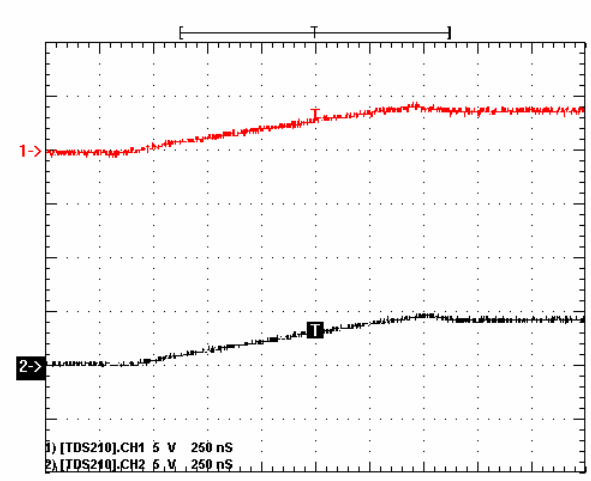

Figure A.8-1: An EL2044 vs. a LM347 working as a follower

Here, channel one represents the EL2044 while channel two represents the LM347.

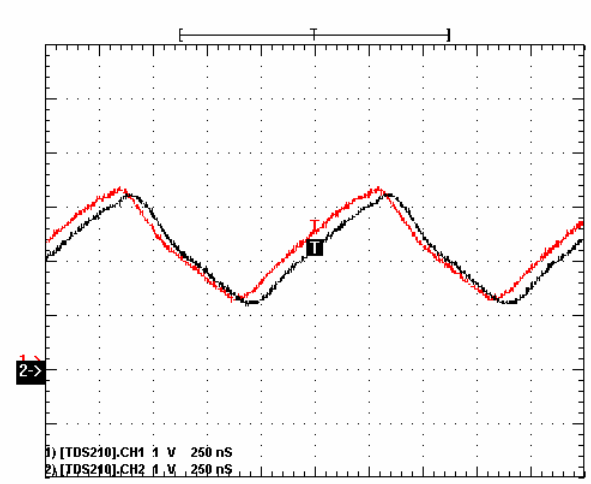

Figure A.8-2: An EL2044 vs. a LM347 at 0.9 MHz

A.9 LT1230

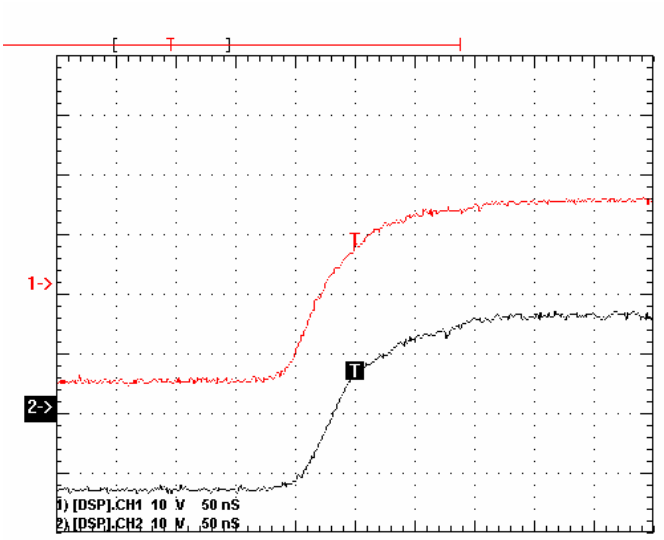

Figure A.9-1: Internal delay of LT1230CN

The internal fall time delay was significantly small i.e. not possible to measure.

A.10 Total Delay in PWM and ADC Circuit

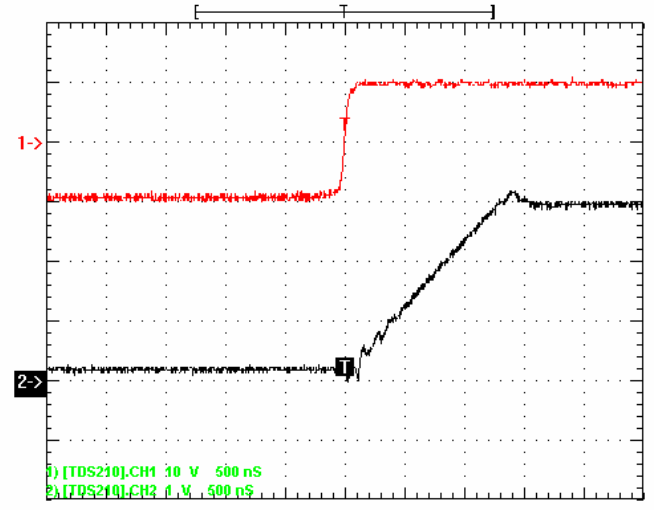

Figure A.10-1: Total delay in an ADC Circuit (Rise time)

No difference in rise nor fall times is present.

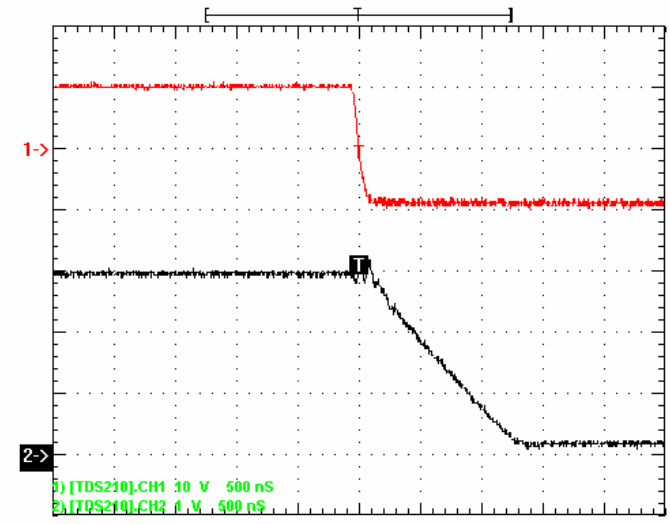

Figure A.10-2: Total delay in an ADC Circuit (Fall time)

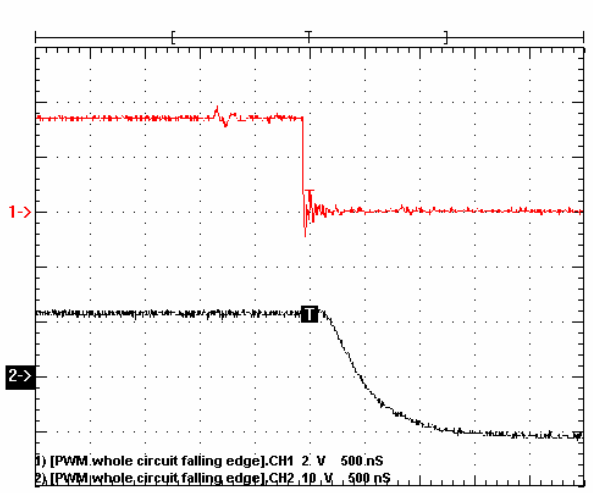

Figure A.10-3: The total delay in a PWM channel (Fall time)

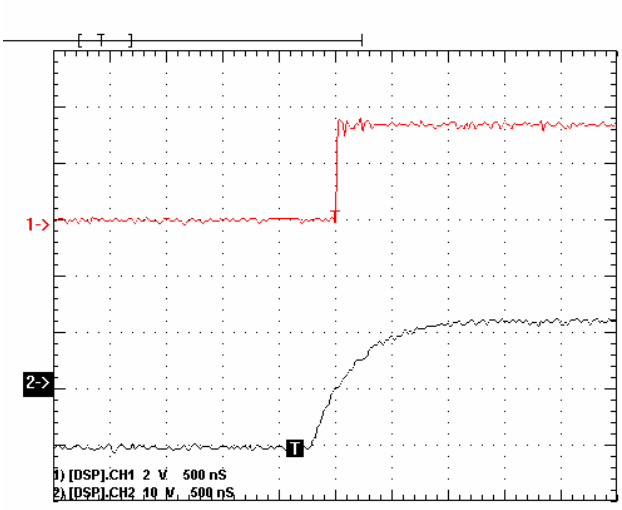

Figure A.10-4: The total delay in a PWM channel (Fall time)

Appendix B Three-phase and Coordinate Transformation

B.1 Power invariant

Equations from here on originate from [9.1] . In a given two-dimensional [α, β] system $\vec{u}$ may be expressed as in Equation B.1-1.

$$\vec{u} = u^* \vec{\alpha} + j \cdot u^*_{\beta}$$

Equation B.1-1

From a power point of view may $\vec{s}$ be expressed in the same system, the relation with a three-dimensional system is displayed:

$$\vec{s} = s_\alpha + j \cdot s_\beta = K\left[s_a + s_b \cdot e^{\frac{j \cdot 2 \cdot \pi}{3}} + s_c \cdot e^{\frac{j \cdot 4 \cdot \pi}{3}} \right]$$

Equation B.1-2

It is assumed that following relation concerning power p(t) is fulfilled:

$$p(t) = u_\alpha(t) \cdot i_\alpha(t) + u_\beta(t) \cdot i_\beta(t) = u_a(t) \cdot i_a(t) + u_b(t) \cdot i_b(t) + u_c(t) \cdot i_c(t)$$

Equation B.1-3

As a consequence:

$$u_\alpha = k \cdot \frac{3}{2} \cdot u_a \qquad i_\alpha = k \cdot \frac{3}{2} \cdot i_a$$

$$u_\beta = k \cdot \frac{\sqrt{3}}{2}(u_b - u_c) \qquad i_\beta = k \cdot \frac{\sqrt{3}}{2}(i_b - i_c)$$

Equation B.1-4

Further more is the system assumed to fulfill the following criteria i.e. symmetrical:

$$\left.\begin{array}{l} \mathrm{U_a + u_b + u_c = 0} \\ \mathrm{i_a + i_b + i_c = \ \ 0} \end{array}\right\} \Rightarrow p(t) = k^2 \cdot \frac{3}{2}\left[u_a(t) i_a(t) + u_b(t) i_b(t) + u_c(t) i_c(t)\right]$$

Equation B.1-5

This leads to $k = \sqrt{\frac{2}{3}}$ which as a direct consequence leads to:

$$\vec{s} = s_\alpha + j \cdot s_\beta = \sqrt{\frac{2}{3}}\left[s_a + s_b e^{\frac{j2\pi}{3}} + s_c e^{\frac{j4\pi}{3}} \right] = \sqrt{\frac{3}{2}} s_a + j \frac{1}{\sqrt{2}}(s_b - s_c)$$

Equation B.1-6

Thus:

$$\left.\begin{array}{lll} s_\alpha = \sqrt{\frac{3}{2}} s_a \quad s_\beta = \frac{1}{\sqrt{2}}(s_b - s_c) & & \\ s_a = \sqrt{\frac{2}{3}} s_\alpha \quad s_b = -\sqrt{\frac{1}{6}} s_\alpha + \sqrt{\frac{1}{2}} s_\beta & s_c = -\sqrt{\frac{1}{6}} s_\alpha - \sqrt{\frac{1}{2}} s_\beta & \end{array}\right\} \; 3 to 2 \; phase and vise versa$$

Equation B.1-7

B.2 Amplitude invariant

Depending on application, different transformations may be needed. The amplitude invariant transform stand as a complement to the power invariant one. This is an important technique. Even though this is not used in the project, a short presentation will follow:

$$K^1 = \frac{2}{3} \Rightarrow \; \vec{s} = s_\alpha + j \cdot s_\beta = \frac{2}{3}\left[s_a + s_b e^{\frac{j2\pi}{3}} + s_c e^{\frac{j4\pi}{3}} \right] = s_a + j\frac{1}{\sqrt{3}}(s_b - s_c)$$

Equation B.2-1

The value of K is based on the fact that the amplitude should be constant throughout the transformation.

Thus:

$$\left.\begin{array}{lll} s_\alpha = s_\alpha \quad s_\beta = \frac{1}{\sqrt{3}}(s_b - s_c) & & \\ s_a = s_\alpha \quad s_b = -\frac{1}{2} s_\alpha + \frac{\sqrt{3}}{2} s_\beta & s_c = -\frac{1}{2} s_\alpha - \frac{\sqrt{3}}{2} s_\beta & \end{array}\right\} \; 3 to 2 \; phase and vise versa$$

Equation B.2-2

p(t) can thereby be expressed as below:

$$p(t) = u_a(t) i_a(t) + u_b(t) i_b(t) + u_c(t) i_c(t) = \frac{3}{2}\left[u_\alpha(t) i_\alpha(t) + u_\beta(t) i_\beta(t) \right]$$

Equation B.2-3

B.3 Coordinate Transformation

In order to further facilitate calculations the [α,β]-frame needs to be represented in a new coordinate system. In this [d, q]-frame θ is eliminated as a parameter this is achieved by allowing the [d, q]-frame to rotate with the angle θ in the [α,β]-frame. The frame rotates hereby with the electrical speed of the measured motor.

An output voltage U may be represented as below in the both coordinate systems:

$$U^{\alpha\beta} = U_\alpha + j \cdot U_\beta = U^{dq} e^{-j\theta} = (U_d + j \cdot U_q)(\cos\theta - j \cdot \sin\theta) = (U_d \cos\theta + U_q \sin\theta) + j(U_d \cos\theta - U_q \sin\theta)$$

Equation B.3-1

$$\begin{cases} U_d = U_\alpha \cos\theta + U_\beta \sin\theta \\ U_q = U_\beta \cos\theta - U_\alpha \sin\theta \end{cases} = \begin{vmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{vmatrix} \begin{vmatrix} U_\alpha \\ U_\beta \end{vmatrix}$$

Equation B.3-2

As a consequence of this, the voltage U may be represented as in Figure B.3-1.

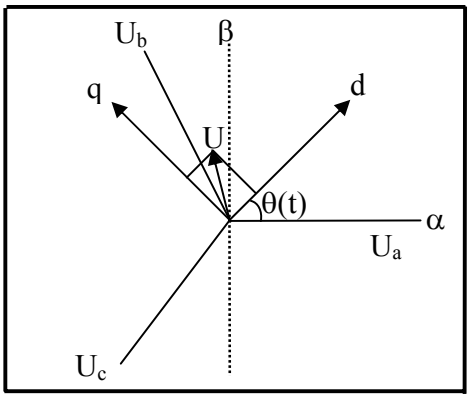

Figure B.3-1: Voltage U represented in the [d, q]-frame

As the coordinate system rotates with ω, angle dependency within the [d, q]-frame is eliminated.

The same transformations are applied to currents within the system.

Appendix C C-Code

C.1 C-Code

Note that no standard files are specified below.

```
#include <stdio.h>
#include <DSP281x_Device.h>
#include <DSP281x_Gpio>
#include <DSP281x_Examples.h>
#include <math.h>
#include "parameters.h"
int f=0;

void InitEVA(void);
interrupt void T2UFINT_ISR(void);


double Ialpha_Transform(double a,double b){
return sqrt(3.0/2)*a;
}

double Ibeta_Transform(double a,double b){
return (1.0/sqrt(2.0))*(a-b);
}

double Id_Transform(double alpha, double beta){
return alpha*cos(theta)+beta*sin(theta);
}

double Iq_Transform(double alpha, double beta){
return beta*cos(theta)-alpha*sin(theta);
}

double Ud_Transform(double Id,double Iq){
return Kd_In*(Id_ref-Id)+(1.0/tid_In)*((Id_ref-Id)+Id_ErrTot)-
Omega*Ld*0.5*(0.5*(Iq_ref+Iq));
}

double Uq_Transform(double Id,double Iq){
return Kq_In*(Iq_ref-Iq)+(1.0/tiq_In)*((Iq_ref-
Iq)+Iq_ErrTot)+Omega*0.5*Lq*(0.5*(Id_ref+Id))+Psi*Omega;
}

void Set_Outer_Controller(void){       //Speed Controller
K_Out=2*CHI*ALPHAS*J;    //Mec speed
```

```
ti_Out=(2*CHI)/ALPHAS;   //Mec speed
}

void Set_Inner_Controller(void){       //Current Controller
Kd_In=(Ld/Ts)+(R/2);
tid_In=R/Kd_In;

Kq_In=((Lq/Ts+R/2));
tiq_In=1/((Lq/(R*Ts)+1/2.0));
}

double Calc_T_Ref(void){
return K_Out*(OmegaRef-Omega);
}

double Calc_Iq_Ref(double T){
return T/Psi;
}

void main(){

   EALLOW;
   InitSysCtrl(); // Global variable for this example
   DINT;   // Step 3. Clear all interrupts and initialize PIE vector table, Disable CPU
interrupts
   InitPieCtrl();
   IER = 0x0000;// Disable CPU interrupts and clear all CPU interrupt flags:
   IFR = 0x0000;
   InitPieVectTable();

   PieVectTable.T1UFINT=&T1UFINT_ISR;
   PieCtrlRegs.PIEIER2.bit.INTx6=1; //enable T1 period interrupt
   IER |= M_INT1; // Enable CPU Interrupt 1
   IER |=M_INT2;
   EINT;          // Enable Global interrupt INTM
   ERTM;          // Enable Global realtime interrupt DBGM
  EvaRegs.EVAIMRA.bit.T1UFINT=1;
  EDIS;


//-------------------------------------------------INIT-----------------------------------------------
Init_PWM();
Init_Enable();
Init_Buttons();
Init_Diode();
```

```
InitEVA();
Init_ADC_Set();
Set_Outer_Controller(); //Speed Controller
Set_Inner_Controller(); //Current Controller


while(1){ //MAIN-loop

if (ButtonDown(1)) OmegaRef+=0.0010;  // Increase OmegaRef
if (ButtonDown(3)) OmegaRef-=0.0010;   // Decrease OmegaRef
}
}

void InitEVA(){

EvaRegs.T1PR = 0x230;           //Fswitch=5.5kHz,
                                //Fswitch = ClockSpeed/TimerPreScaler/Period
EvaRegs.T1CON.bit.TMODE=1;       // Timer mode
EvaRegs.T1CON.bit.TPS=2;         //timer prescaler

EvaRegs.ACTRA.all = 0x0AAA;      // Output polarities
EvaRegs.GPTCONA.bit.TCMPOE = 1;
EvaRegs.COMCONA.bit.CENABLE=1;
EvaRegs.COMCONA.bit.SVENABLE=1; //Space vector enable
EvaRegs.COMCONA.bit.CLD=0x01;
EvaRegs.COMCONA.bit.ACTRLD=0x01;
EvaRegs.COMCONA.bit.FCOMPOE=1;
EvaRegs.T1CON.bit.TENABLE=1;
EvaRegs.T1CON.bit.TECMPR=1;
EvaRegs.T1CNT = 0x0000;      // Timer1 counter
}

interrupt void T1UFINT_ISR(void){
//-------------------------------------Outer Loop-----------------------------------------------------
 T_Ref=Calc_T_Ref();
  Iq_ref=Calc_Iq_Ref(T_Ref);
//---------------------------------Current Control Algorithm,Inner LOOP-------------------------

for(f=0;f<9; f++){

//-----------------------------------ADC CONVERSIONS-----------------------------------------

UDc=AdcRegs.ADCRESULT0>>4;      //Supply voltage ADC A1
UDc=(UDc-610)/9.9;                    //Input adjustment

Ia=AdcRegs.ADCRESULT1>>4;       //Phase current A @ ADC A2, 25A @10V
```

```
Ia=(0.0126*Ia-25)*(25.0/15.9);          //Input adjustment

Ib=AdcRegs.ADCRESULT2>>4;       //Phase current B @ ADC A3, 25A @10V
Ib=(0.0118*Ib-25)*(25.0/15.5);          //Input adjustment

Rsin=AdcRegs.ADCRESULT3>>4;    //SIN angle on ADC A4
Rsin=(0.00458*Rsin-10.0)*(10/6.4);    //Input adjustment

if (Rsin>10.0) Rsin=10.0;       //protection
if (Rsin<-10.0) Rsin=-10.0;

Rcos=AdcRegs.ADCRESULT4>>4;           //COS angle on ADC A5
Rcos=(0.00477*Rcos-10.0)*(10/6.9); // Input adjustment

if (Rcos>10.0) Rcos=10.0;       //protection
if (Rcos<-10.0) Rcos=-10.0;

Res=AdcRegs.ADCRESULT5>>4;      //Resolver on ADC A6;
Res=Res*0.01+Res_old*0.99;
Res_old=Res;

Res=(Res*0.0047896-9.4793)*80;  // Rescale resolver input
if (Res<0) Res=0;

Omega=Res;
theta=atan(Rsin/Rcos); // Calc theta

//---------------------------------------Sector selection-----------------------------------------
if(Rcos<0.0 && Rsin>0.0){ //Second
theta=theta+PI;
}
if(Rcos<0.0 && Rsin<0.0){ //Third
theta+=PI;
}
if(Rcos>0.0 && Rsin<0.0){ //Forth
theta=2*PI+theta;
}

//----------------------------------Current Transformation-----------------------------------------

Ialpha=Ialpha_Transform(Ia,Ib);    // a,b transformed to alpha,beta-system
Ibeta=Ibeta_Transform(Ia,Ib);

Iq=Iq_Transform(Ialpha,Ibeta);      // alpha,beta transformed to d,q-system
Id=Id_Transform(Ialpha,Ibeta);
```

```
//-----------------------------------Reference voltage transformation-----------------------------
        UQ=0.1*Uq_Transform(Id,Iq);                  //Ud,Uq calculated
        if(abs(UQ)>abs(UDc)){
        if(UQ<=0.0){
        UQ=-(UDc);
        }

        else{
                UQ=(UDc);
   }
        }


        UD=0.1*Ud_Transform(Id,Iq);                  //Ud,Uq calculated
        if(abs(UD)>abs(UDc)){
        if(UD<=0.0){
        UD=-(UDc);
        }
        else{
                UD=(UDc);
   }
        }

//---------------------------------------Duty Cycle Calculation---------------------------------
        test1=sqrt(UQ/UDc*UQ/UDc+UD/UDc*UD/UDc);
        T1=TMRPWM*sin(theta)*sqrt(2.0)*test1;
        T2=TMRPWM*cos(theta+PI/6.0)*sqrt(2.0)*test1;

        T1=T1/TMRPWM;
        T2=T2/TMRPWM;

        if (T1<0.0) T1=-1*T1;
   if (T2<0.0) T2=-1*T2;

 //-------------------------------------------------------------------------------------------------

//-------------------------------------Output Selection-----------------------------------------

if( theta>=0.0 && theta<(PI/3)){
EvaRegs.ACTRA.bit.D=1;
}

if(theta>=(PI/3.0)&& theta<(2*PI/3.0)){
EvaRegs.ACTRA.bit.D=3;
}
```

```
if(theta>=(2*PI/3.0) && theta<PI){
EvaRegs.ACTRA.bit.D=2;
}

if(theta>=PI && theta<(4*PI/3.0)){
EvaRegs.ACTRA.bit.D=6;
}

if(theta>=(4*PI/3.0) && theta<(5*PI/3.0)){
EvaRegs.ACTRA.bit.D=4;
}

if(theta>=(5*PI/3.0)){
EvaRegs.ACTRA.bit.D=5;
}

} //------------------END OF INNER LOOP-----------------------------

// Update, anti-windup

 if(Id_ErrTot<20) Id_ErrTot+=0.2*(Id_ref-Id); //saving old error
if(Iq_ErrTot<20) Iq_ErrTot+=0.2*(Iq_ref-Iq);


EvaRegs.CMPR1 =(Uint16)(((0.5*T1))*0x230)*0.5;
EvaRegs.CMPR2 =(Uint16)(((0.5*T1)+(0.5*T2))*0x230)*0.5;

EALLOW;
EvaRegs.EVAIFRA.bit.T1UFINT=1; // reset flag
PieCtrlRegs.PIEACK.all=PIEACK_GROUP2; // Ack
EDIS;
}

 //-----------------------------------------------END OF OUTER LOOP--------------------------


**********************************BREAK*******************************
#include <stdio.h>
#include <DSP281x_Device.h>

void Init_ADC_Set(void){

#define ADC_MODCLK 0x3   // HSPCLK = SYSCLKOUT/2*ADC_MODCLK2 =
150/(2*3)        = 25MHz
EALLOW;
SysCtrlRegs.HISPCP.all = ADC_MODCLK;          // Specific clock setting
```

```
EDIS;

InitAdc();   //initialising ADC module

AdcRegs.ADCTRL1.bit.CONT_RUN = 1;
AdcRegs.ADCMAXCONV.all = 0x6;         // Setup 6 conv's on SEQ1
AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0x1; // Udc
AdcRegs.ADCCHSELSEQ1.bit.CONV01 = 0x2; // Ia
AdcRegs.ADCCHSELSEQ1.bit.CONV02 = 0x3; // Ib
AdcRegs.ADCCHSELSEQ1.bit.CONV03 = 0x4; // cos
AdcRegs.ADCCHSELSEQ2.bit.CONV04 = 0x5; // sin
AdcRegs.ADCCHSELSEQ2.bit.CONV05 = 0x0; // Resolver
AdcRegs.ADCTRL2.bit.SOC_SEQ1=1; //Start of conversion
}

**********BREAK**********
#include <stdio.h>
#include <std.h>
#include <DSP281x_Device.h>
#include <DSP281x_Gpio>

void Init_Buttons(void){
//------------ BUTTONS INIT ---------------------
EALLOW;
GpioMuxRegs.GPAMUX.bit.C1TRIP_GPIOA13=0; //Button B3
GpioMuxRegs.GPAMUX.bit.C2TRIP_GPIOA14=0; //Button B2
GpioMuxRegs.GPAMUX.bit.C3TRIP_GPIOA15=0; //Button B1

GpioMuxRegs.GPADIR.bit.GPIOA13=0; //Read
GpioMuxRegs.GPADIR.bit.GPIOA14=0;
GpioMuxRegs.GPADIR.bit.GPIOA15=0;

//Read BUTTONS ON or OFF:

GpioDataRegs.GPADAT.bit.GPIOA13=1; //Pull-up
GpioDataRegs.GPADAT.bit.GPIOA14=1;
GpioDataRegs.GPADAT.bit.GPIOA15=1;
EDIS;
}

Bool ButtonDown(int buttonNbr){
if (buttonNbr==1 && GpioDataRegs.GPADAT.bit.GPIOA13==0) return TRUE;
if (buttonNbr==2 && GpioDataRegs.GPADAT.bit.GPIOA14==0) return TRUE;
if (buttonNbr==3 && GpioDataRegs.GPADAT.bit.GPIOA15==0) return TRUE;
else return FALSE;
}
```

```
void Init_Diode(void){

EALLOW;
GpioMuxRegs.GPBMUX.bit.C4TRIP_GPIOB13=0;
GpioMuxRegs.GPBMUX.bit.C5TRIP_GPIOB14=0;
GpioMuxRegs.GPBMUX.bit.C6TRIP_GPIOB15=0;

GpioMuxRegs.GPBDIR.bit.GPIOB13=1; //Write
GpioMuxRegs.GPBDIR.bit.GPIOB14=1;
GpioMuxRegs.GPBDIR.bit.GPIOB15=1;

GpioDataRegs.GPBDAT.bit.GPIOB13=0; //0=ON, 1=OFF
GpioDataRegs.GPBDAT.bit.GPIOB14=0;
GpioDataRegs.GPBDAT.bit.GPIOB15=0;
EDIS;
}

void DiodOn(int diodNbr){
if (diodNbr==1) GpioDataRegs.GPBDAT.bit.GPIOB15=0;
if (diodNbr==2) GpioDataRegs.GPBDAT.bit.GPIOB14=0;
if (diodNbr==3) GpioDataRegs.GPBDAT.bit.GPIOB13=0;
}


void DiodOff(int diodNbr){
if (diodNbr==1) GpioDataRegs.GPBDAT.bit.GPIOB15=1;
if (diodNbr==2) GpioDataRegs.GPBDAT.bit.GPIOB14=1;
if (diodNbr==3) GpioDataRegs.GPBDAT.bit.GPIOB13=1;
}
**************************BREAK*************************************
#ifndef PARAMETERS_H
#define PARAMETERS_H
#endif
#define PI 3.14159265

//------------------------------START PMS MACHINE SETTINGS------------------------------
#define RHO 200*PI
#define CHI 1.0/sqrt(2.0)
#define J 0.002
#define R 0.5
double Ld=0.003;
double Lq=0.003;
#define Kcd_In Omega*Ld*0.5
#define Kcq_In Omega*Lq*0.5
//---------------------------------------------------------------------------
```

```
double test1=0;
double theta=0.0;

double Kd_In,tid_In;
double Kq_In,tiq_In;
double K_Out,ti_Out;

double WDELTA =2*100*PI;
double ALPHAS=2*100*PI;

double TMRPWM=0.0002; // Defines PWM period,5k
double T1;
double T2;

double th_Old=0.0;
double Psi=0.9;
//----------------------------------------- PMSM Current----------------------------------------
double Id,Iq,Id_ref,Iq_ref,Ialpha,Ibeta;
double Ia=0.0;
double Ib=0.0;
double Id_ErrTot=0;
double Iq_ErrTot=0;
double Ia_ErrTot=0;
double Id_ref=0.0;
double Iq_max=25.0;
//------------------------- PMSM Voltage---------------------------------
double Ua,Ub,Uc,Ud,Uq,Ualpha,Ubeta;
double UDc=0.0;
double UD=0.0;
double UQ=0.0;
//-------------------------------------PMSM Timer-----------------------------------------
double T1=0.0;
double t1=0.0;
double t2=0.0;
//----------------------END PMS-MACHINE SETTINGS-----------------------------------------

//-------------------------------GENERAL PARAMETERS---------------------------------
Uint16 Max=0;
Uint16 Min=4095;
double Omega=0;
double OmegaRef=60;
double T_ref=1.0; // Torque reference
double T_Ref=1.0;
double T=0.0;
```

```
double Tsw=(1.0/5000); //Swith rate
double Ts=(1.0/5000);// Sampling rate -Set by user
double Rsin,Rcos,Res;
double Res_old=0.0;

**********NO MORE**********
```

Appendix D Circuit Schematics and PCB layout

D.1 Schematics

D.1.1 ADC Circuit

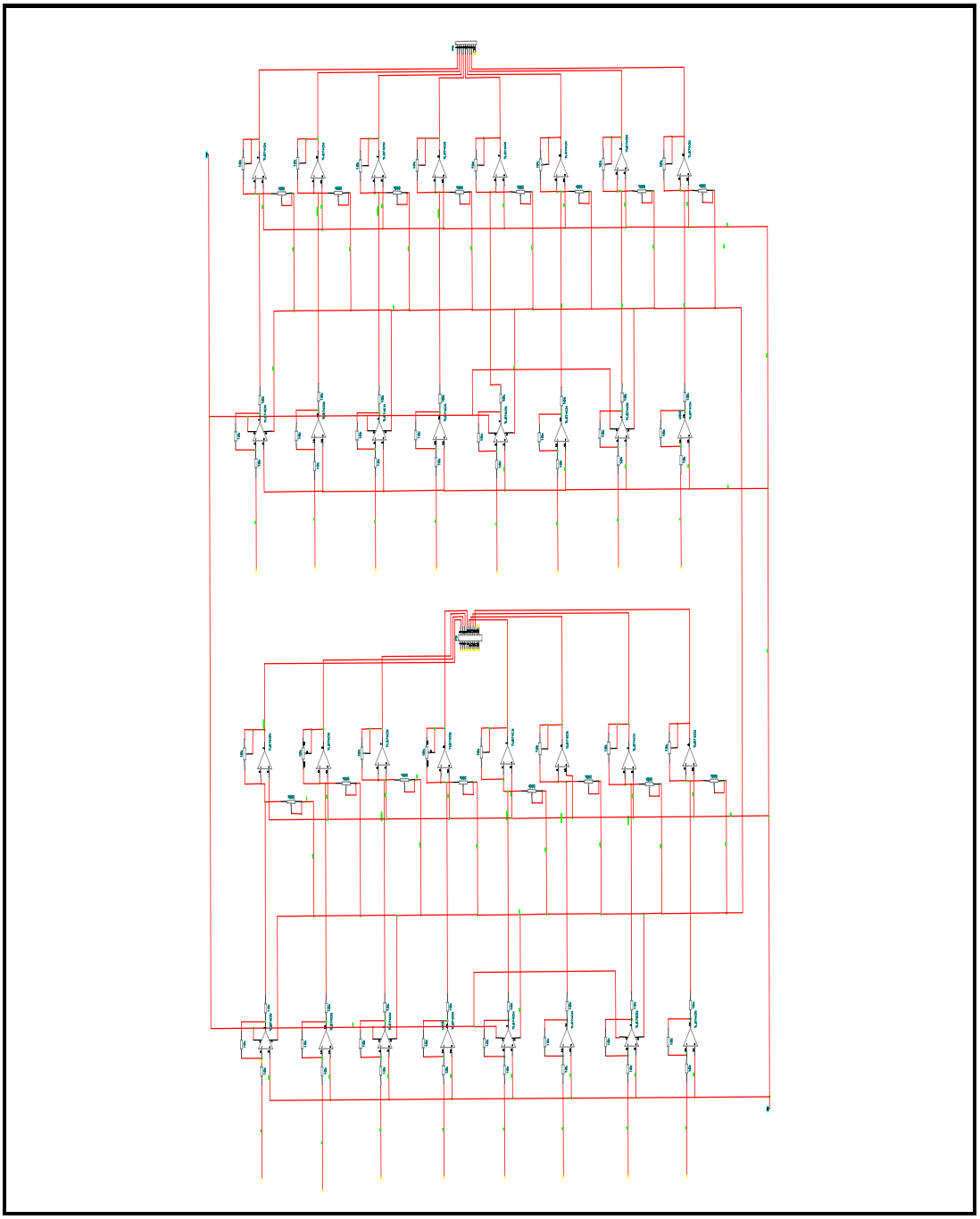

Figure D.1-1: ADC Circuit

D.1.2 PWM Circuit

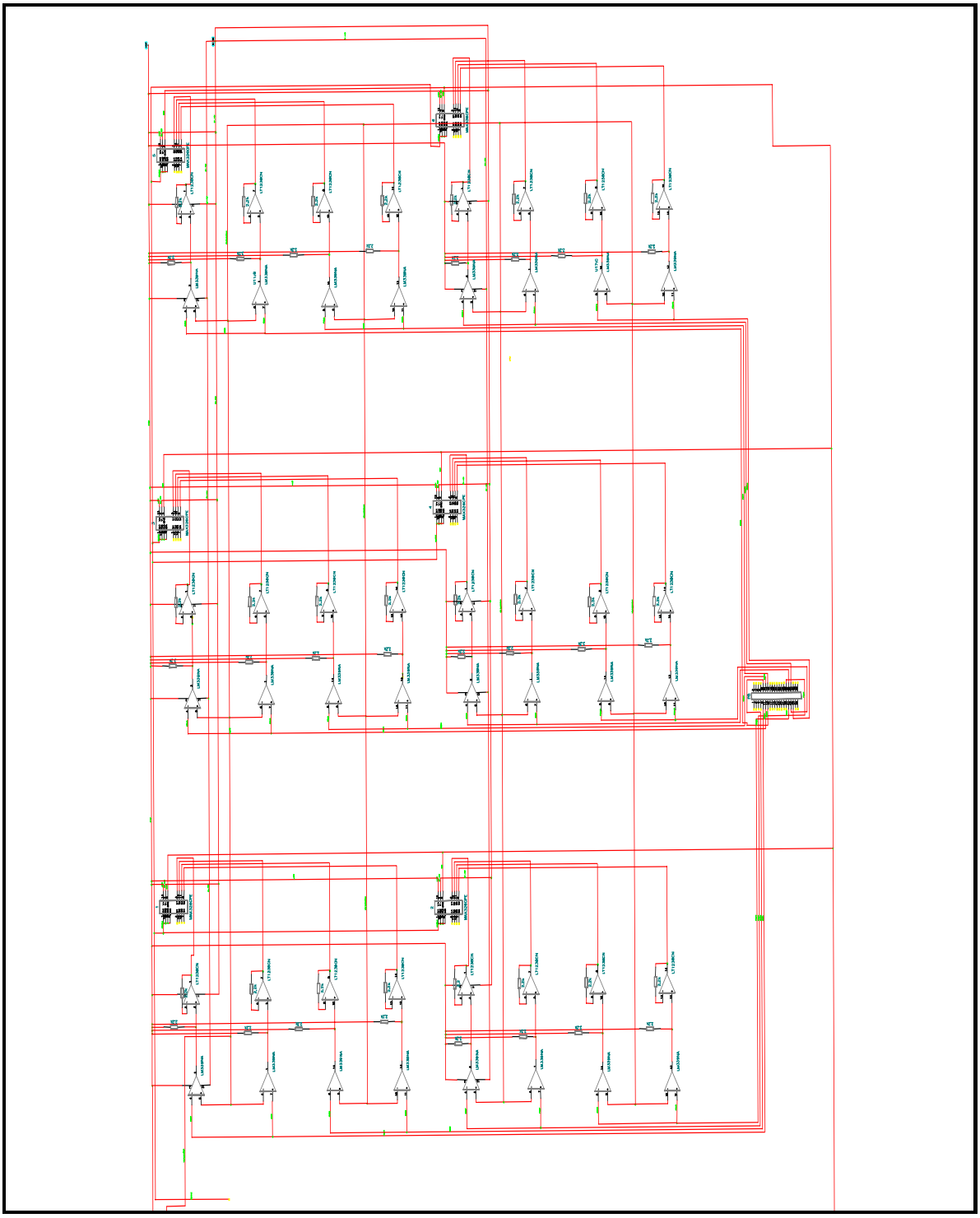

Figure D.1-2: PWM Circuit

D.2 Layout

D.2.1 Power Electronic Card

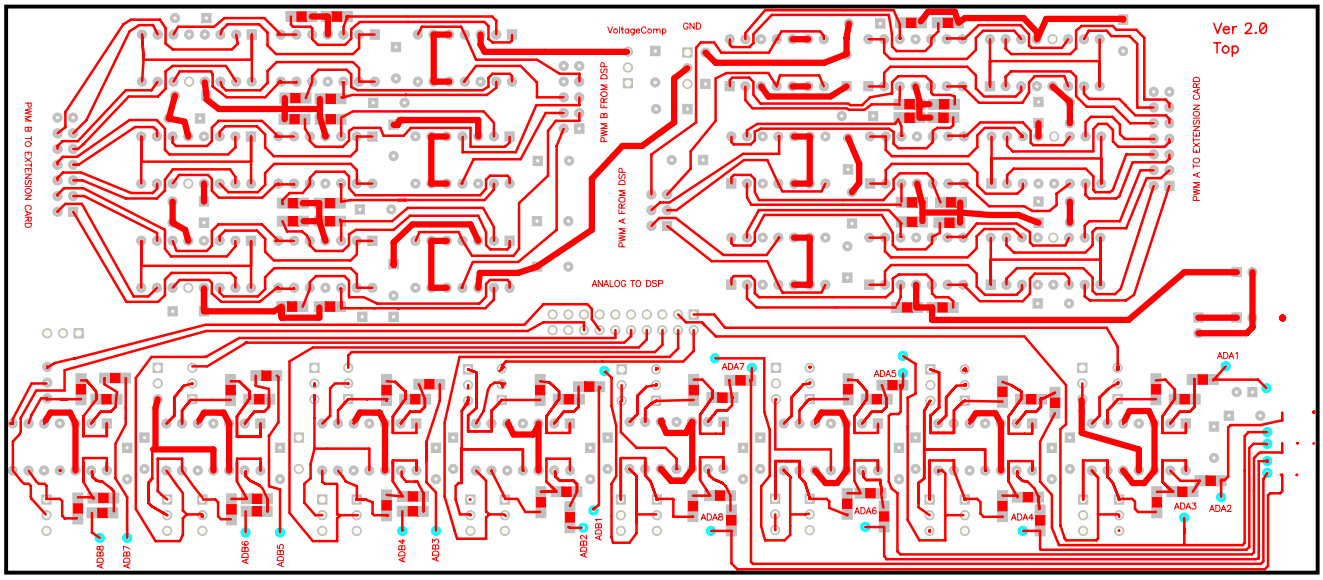

Figure D.2-1: Power Electronic Card, Top layer

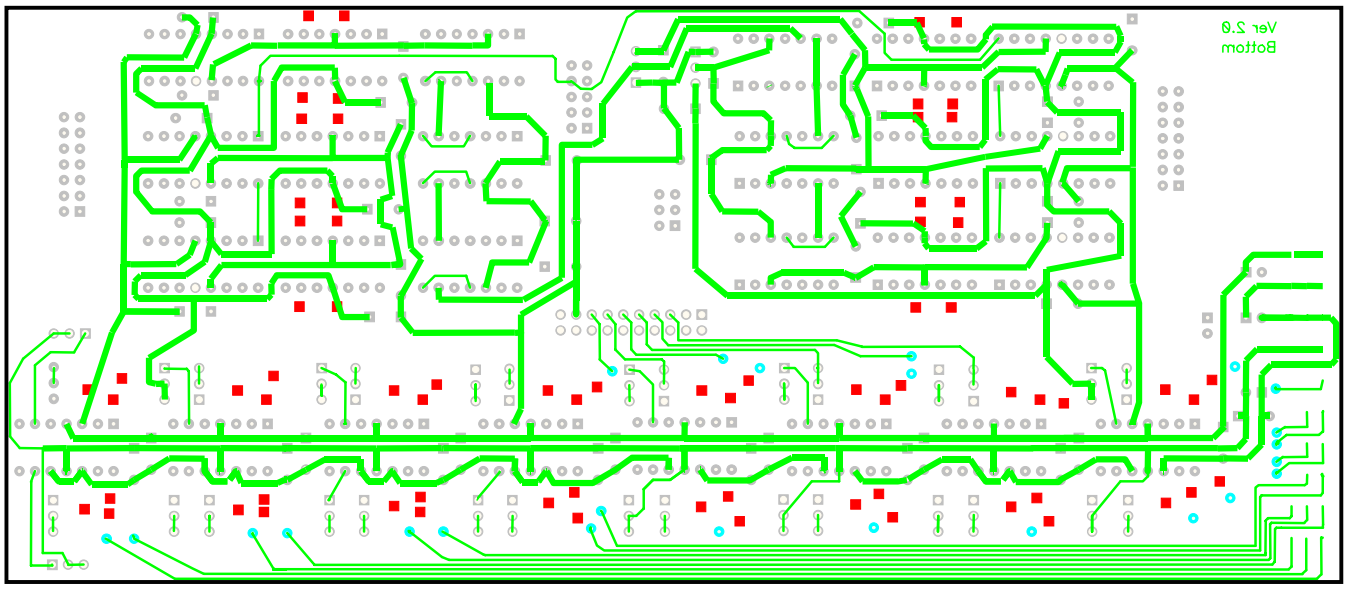

Figure D.2-2: Power Electronic Card, Bottom Layer

Please note that bottom layer is displayed as in P-Cad 2000 i.e. inverted.

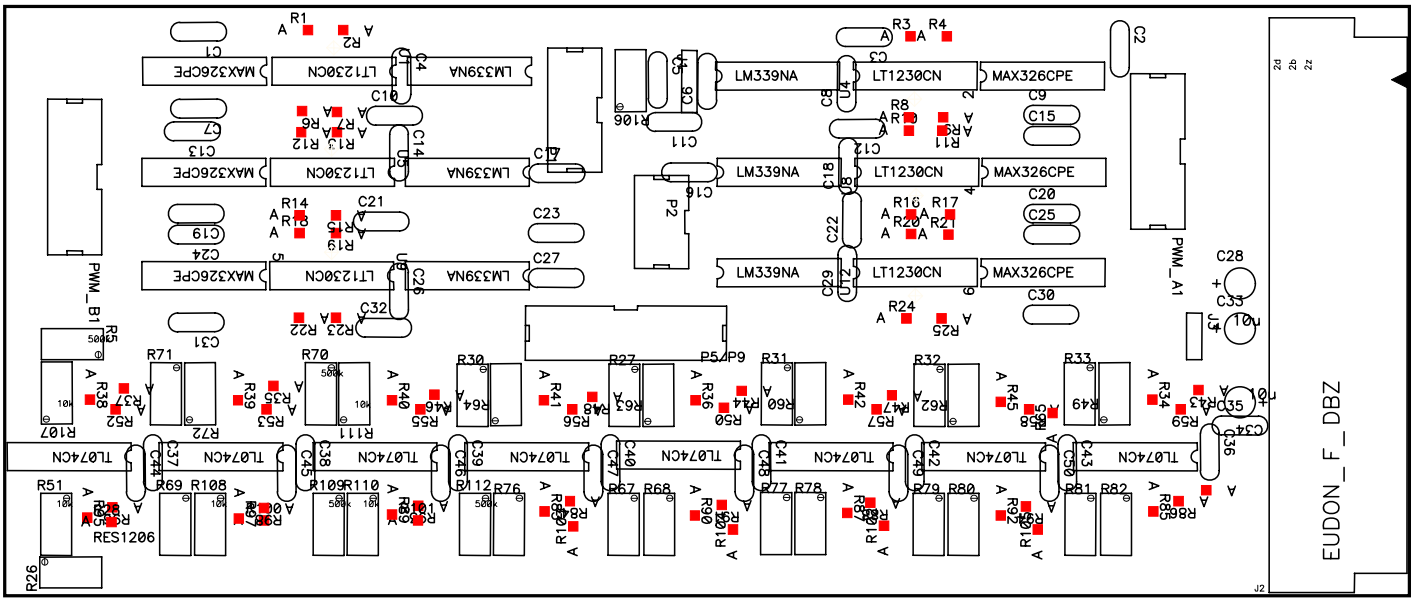

Figure D.2-3: Power Electronic Card, Top silk

Note that no components are to be found on the bottom layer.

D.2.2 Extension Card for EUDON Connector type F

Figure D.2-4 shows the layout of the circuit board for the extension card used for making PWM signals available on the back plane. Please note that bottom layer is displayed as in P-Cad 2000 i.e. inverted; No connections are to be found on the top layer. For the layout of the back plane EUDON connections, please refer to Table 2.

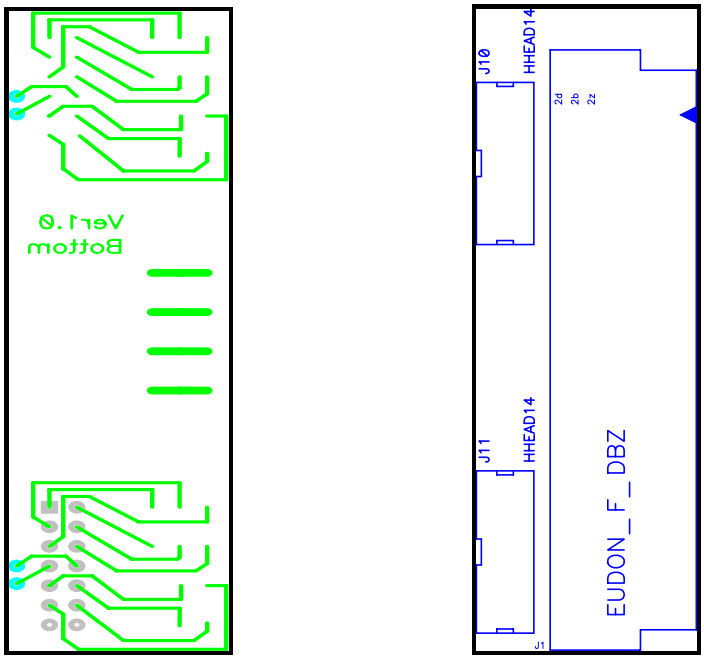

Figure D.2-4: Extension Card, Bottom layer (left) and Top silk (right)

D.3 Component list

Following components are to be found within the final system.

Count	Component	Value	Description	Notes	Component Nr
48	Resistor	10k	CC1206	0.25W	R34-R59, R65-R66, R83-R105
17	Resistor	10k	POT3T	POT6 4W	R5, R49, R51, R54, R61-R64, R68, R70, R72, R74, R76, R78, R80, R82
16	Resistor	500k	POT3T	POT6 4W	R26-R33, R67, R69, R71, R73, R75, R77, R79
48	Resistor	2.2k	CC1206	0.25W	R1-R4, R6-R25[1]
3	Capacitor	10µ		Tantal	C28, C33-C34
47	Capacitor	100n			C1—C27, C29-C32, C35-C50
6	Analog switch		MAX326CPE	DIP16	Refer to layout for details
6	Comparator		LM339NA	DIP14	Refer to layout for details
6	OP		LT1230CN	DIP14	Refer to layout for details
8	OP		TL074CN	DIP14	Refer to layout for details
2	HHEAD10				P1
2	HHEAD6				P2
4	HHEAD20				P5/P9
2	HHEAD14				PWM_A- PWM_B
1	HSIL3			SIL3	J1
2	EUDON		DIN/F/DBZ		J2
1	HSIL2			SIL2	J3

Table 1: Component list

[1] LT1230 feedback resistor not specified as the location is onboard the circuit.

Appendix E Internal and External connection layout

E.1 Front Connections

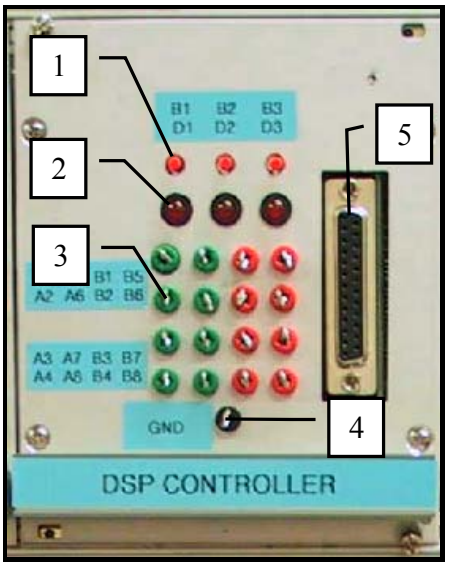

Figure E.1-1: Front view panel with numbered blocks

1 Three individually programmable buttons. See Figure E.1-2 for schematics.
2.Three individually programmable LEDs. See Figure E.1-2 for schematics.
3 16 measurement loops for ADC-channels arranged as in Figure E.1-3
4 Measurement loop with ground level.
5 25-pin DSUB from DSP with JTAG-interface to computer parallel port.

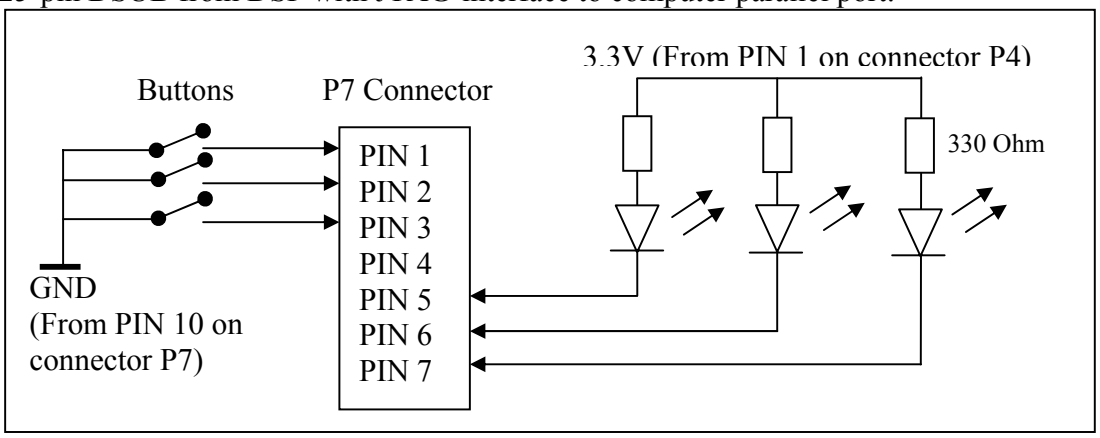

Figure E.1-2: Schematics of LEDs and Button connections

The 330 Ω series resistance is derived from: $R = \frac{U}{I} = \frac{3{,}3}{0.010} = 330\ \Omega$

Equation E.1-1

Block 3 Green loops			Block 4 Red loops	
ADC A0	ADC A4		ADC B0	ADC B4
ADC A1	ADC A5		ADC B1	ADC B5
ADC A2	ADC A6		ADC B2	ADC B6
ADC A3	ADC A7		ADC B3	ADC B7

Figure E.1-3: Arrangement of measurements loops on front panel

E.2 Rear Connections

	d	b	z	d	b	z
2	PWM 1A	PWM 2A	PWM 3A	X	X	X
4	PWM INV 1A	PWM INV 2A	PWM INV 3A	X	X	X
6	PWM 4A	PWM 5A	PWM 6A	X	X	X
8	PWM INV 4A	PWM INV 5A	PWM INV 6A	X	X	X
10	X	X	X	X	X	X
12	X	X	X	X	X	X
14	(+) 15V	(+) 15V	(+) 15V	(+)15V	(+)15V	(+)15V
16	(+) 5	(+) 5	(+) 5	(+)5	(+)5	(+)5
18	GND	GND	GND	GND	GND	GND
20	(-) 15V	(-) 15V	(-) 15V	(-)15V	(-)15V	(-)15V
22	X	X	X			ADCIN A0
24	X	X	X	ADCIN A1	ADCIN A2	ADCIN A3
26	PWM 1B	PWM 2B	PWM 3B	ADCIN A4	ADCIN A5	ADCIN A6
28	PWM INV 2B	PWM INV 2B	PWM INV 3B	ADCIN A7	ADCIN B0	ADCIN B1
30	PWM 4B	PWM 5B	PWM 6B	ADCIN B2	ADCIN B3	ADCIN B4
32	PWM INV 4B	PWM INV 5B	PWM INV 6B	ADCIN B5	ADCIN B6	ADCIN B7

Table 2: Extension Card (left), Power Electronic Card (right)

X= Pin not used in the current application. Note that Pins with the extension *INV* are to be used in CMOS mode.

E.3 Internal Connections

E.3.1 DSP

This chapter describes the connections that are used inside the DSP board to connect to the interface card.

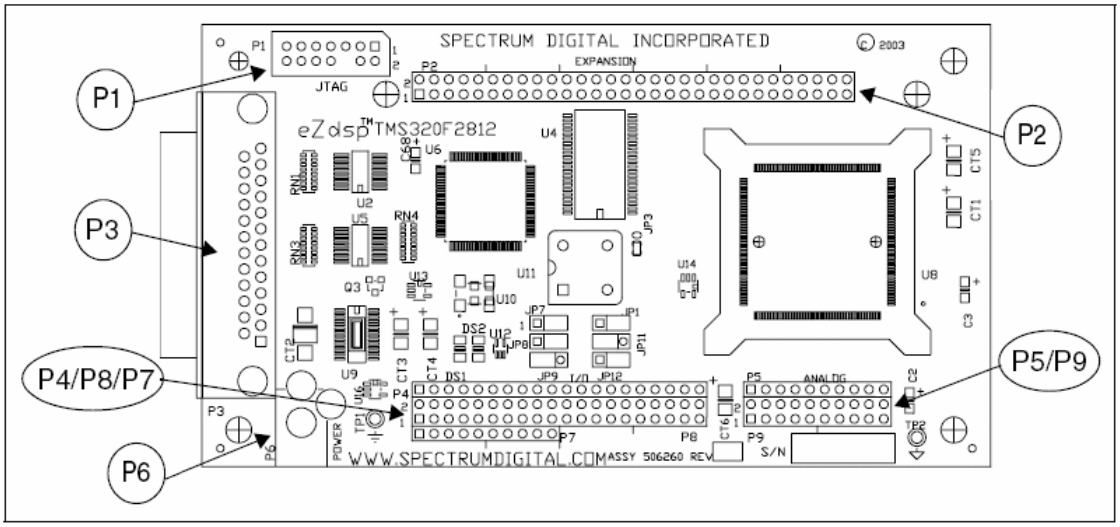

Figure E.3-1: PCB layout of the DSP development board [9.41]

P1. JTAG-interface (not used in this project).
P2. Expansion port (not used in this project).
P3. Parallel port /JTAG controller interface these are visible on front panel.
P4. I/0-interface (not used in this project).
P5. Analog interface. Connecting ADC B-channels. Pin configurations are shown in Table 3.
P6. Power connector (+5V).
P7. I/O-interface. Connecting front panel buttons and LEDs. Pin configurations are shown in Table 4.
P8. I/O-interface. Connecting PWM channels and ENABLE signal for PWM output. Pin configurations are shown in Table 5.
P9. Analog interface. Connecting ADC A-channels and VREFLO (set to analog ground) Pin configurations are shown in Table 6.

ADCIN B0	ADCIN B1	ADCIN B2	ADCIN B3	ADCIN B4	ADCIN B5	ADCIN B6	ADCIN B7	X	X
1	2	3	4	5	6	7	8	9	10

Table 3: Block diagram of the connector P5

BUTTON #1	BUTTON #2	BUTTON #3	X	LED #1	LED #2	LED #3	X	X	GND
1	2	3	4	5	6	7	8	9	10

Table 4: Block diagram of the connector P7

X	X	X	X	PWM A2	PWM A4	PWM A6	X	X	X	X	X	X	X	PWM B1	PWM B3	PWM B5	ENABLE	X	X
2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
X	X	X	X	PWM A1	PWM A3	PWM A5	X	X	X	X	X	X	X	X	PWM B2	PWM B4	PWM B6	X	X
1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39

Table 5: Block diagram of the connector P8

ADCIN A0	ADCIN A1	ADCIN A2	ADCIN A3	ADCIN A4	ADCIN A5	ADCIN A6	ADCIN A7	VREFLO	X
2	4	6	8	10	12	14	16	18	20
GND	GND	GND	GND	GND	GND	GND	GND	GND	GND
1	3	5	7	9	11	13	15	17	19

Table 6: Block diagram of the connector P9

X = Pin not used in the current application.

E.3.2 Layout of the interface card

This chapter describes the connection features on the interface card.

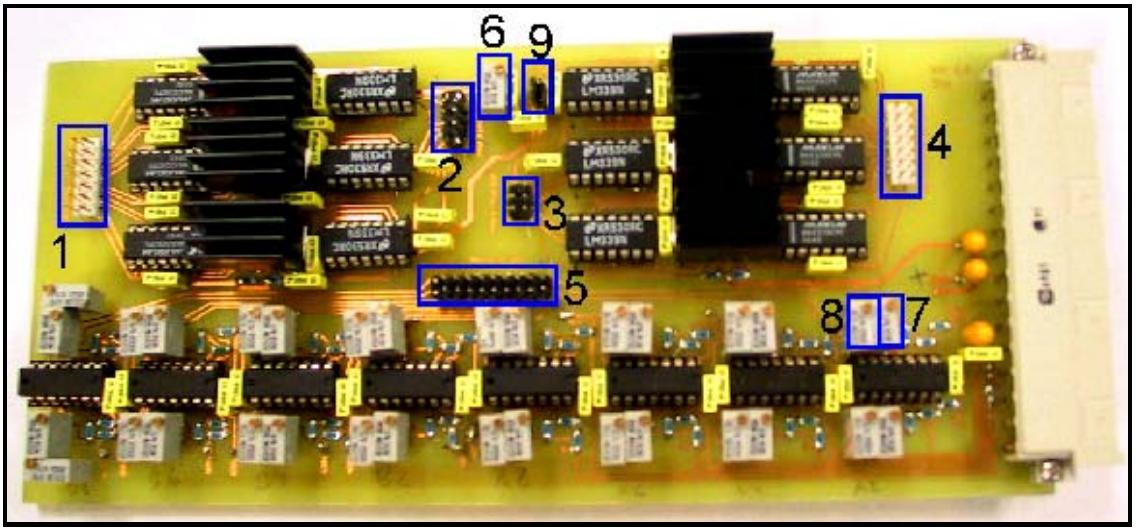

Figure E.3-2: Important spots on the interface card

Explanation of the numbered blocks in **Figure E.3-2**

1. Connector for PWM B-channel signal that leads to extension card on back plane. See Table 9 for pin descriptions.
2. PWM B-channel input from the DSP. See Table 7for pin descriptions.
3. PWM A-channel input from the DSP. See Table 7for pin descriptions.
4. Connector for PWM A-channel signal that leads to extension card on back plane. See Table 8for pin descriptions.
5. Connector for the ADC-channels to the DSP. See Table 10 for pin description.
6. Trim-pot (10 kΩ) for correcting voltage compare value (nominal 1.65v)
7. Trim-pot (10 kΩ) for trimming the ADC-channel signals (gain adjustments)
8. Trim-pot (500 kΩ) for trimming the ADC-channel signals (offset adjustments)
9. Switch for selecting between CMOS level (upper 2 pins shorted) or LAB level (lower 2 pins shorted) card seen as in Figure E.3-2.

PWM A2 10	**PWM A4** 12	**PWM A6** 14
PWM A1 9	**PWM A3** 11	**PWM A5** 13

PWM B1 30	**PWM B3** 32	**PWM B5** 34	**ENABLE** 36
X 29	**PWM B2** 31	**PWM B4** 33	**PWM B6** 35

Table 7: Block diagram of PWM A and PWM B connections from DSP

PWM 2A C_MOS 2	**PWM 4A C_MOS** 4	**PWM 6A C_MOS** 6	**PWM 8A C_MOS** 8	**PWM 10A C_MOS** 10	**PWM 12A C_MOS** 12	**X** 14
PWM 1A LAB 1	**PWM 3A LAB** 3	**PWM 5A LAB** 5	**PWM 7A LAB** 7	**PWM 9A LAB** 9	**PWM 11A LAB** 11	**X** 13

Table 8: Block diagram of the connector PWM A that leads to extension card

PWM 2B C_MOS 2	**PWM 4B C_MOS** 4	**PWM 6B C_MOS** 6	**PWM 8B C_MOS** 8	**PWM 10B C_MOS** 10	**PWM 12B C_MOS** 12	**X** 14
PWM 1B LAB 1	**PWM 3B LAB** 3	**PWM 5B LAB** 5	**PWM 7B LAB** 7	**PWM 9B LAB** 9	**PWM 11B LAB** 11	**X** 13

Table 9: Block diagram of the connector PWM B that leads to extension card

ADCIN B0 2	**ADCIN B1** 4	**ADCIN B2** 6	**ADCIN B3** 8	**ADCIN B4** 10	**ADCIN B5** 12	**ADCIN B6** 14	**ADCIN B7** 16	**X** 18	**X** 20
ADCIN A0 1	**ADCIN A1** 3	**ADCIN A2** 5	**ADCIN A3** 7	**ADCIN A4** 9	**ADCIN A5** 11	**ADCIN A6** 13	**ADCIN A7** 15	**VREFLO** 17	**X** 19

Table 10: Block diagram of the ADC connection.

Note that P5/P9 exists of both P5 and P9 excluded ground pins originating from P9.

X=Pin not used in the current application