Industrial Electrical Engineering and Automation

CODEN:LUTEDX/(TEIE-5398)/1-45(2017)

Digital Power for Battery Chargers - Comparison and Implementation

Johan Tillmar

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

Digital power for battery chargers - comparison and implementation

Johan Tillmar

LUNDS UNIVERSITET
Lunds Tekniska Högskola

2017

Acknowledgment

Many thanks to everybody at Micropower for having me and providing a lovely summer for the third time in a row. Special thanks to Magnus, Tim and Christer for helping me and answering all my questions.

Abstract

In this project the concept of digital power was examined. Though not a new phenomena, the concept was new to Micropower. The company develops a wide array of battery chargers and wanted to include this concept in future generations of their products. The reasoning behind the decision is the possibilities of smoother development, better error handling and more communication.

As a first step different microcontrollers were examined to see which would fit the requirements of the company best. When an appropriate controller was chosen a development kit was ordered to confirm that the functionality was possible to implement. After that, an adapter card was manufactured to fit the digital power controller card in one of Micropowers chargers.

This worked well and the digital power solution provided the same functionality as before but now with more possible future improvements.

Sammanfattning

I det här projektet har digital styrning av kraftelektronik undersökts. Även om det inte är ett nytt fenomen generellt så är det nytt för Micropower. Micropower utvecklar batteriladdare till en rad olika applikationer och vill inkludera detta konceptet i framtida generationer av sina produkter. Anledningen bakom beslutet är smidigare utveckling, bättre felhantering och mer kommunikation.

Som ett första steg studerades olika microcontrollers för att se vilken som passade företagets krav bäst. När en lämplig controller hade valts införskaffades ett utvecklingskit för att bekräfta att funktionaliten som eftersöktes kunde implementeras. Efter det tillverkades ett adapter kort som anpassats så att utvecklingskortet kunde användas för en av Micropowers laddare.

Detta fungerade bra och den digitala styrningen gav samma funktionalitet som den analoga, men nu med bättre möjligheter att lägga till framtida förbättringar.

Acronyms

ADC - Analog to Digital Converter
CAN - Controller Area Network
CSG - Comparator and Slope Generator
HRPWM - High Resolution Pulse Width Modulation
MCU - MicroController Unit
PCC - Peak Current Control
PSFB - Phase Shift Full Bridge
PWM - Pulse Width Modulation

Contents

List of Figures

List of Tables

1 Introduction

The project was done together with Micropower and in it different microcontrollers (MCU) was compared to see which would work best for implementation in the company's battery chargers.

1.1 About the company

Micropower sells, develops and produces battery chargers, both conventional and high frequency chargers. The company started 1984 in Växjö and is today market leading in battery chargers for forklift truck applications in northern Europe. Around 30 engineers are focused on research and development in both Växjö, Stockholm and Salo, Finland.[7]

1.2 Background

Today Micropower uses analog control of a phase-shift full bridge which makes development of new models a hassle. Therefore a move to digital control of the power electronics is desired to make development easier and faster.

Digital power is the move to digital algorithms when it comes to the control and feedback in power supply. This gives more versatility and ease of use when developing new forms of power supply. In [8] the authors describe the impact of using digital control in power electronics and concludes that it would be very beneficial. The reasons of this being the previously mentioned design flow improvement but also a reduction of passive component, and with that a reduction to sensitivity for process and temperature variations. Also, a contribution to better power efficiency with multi-mode operation and adaptive gate-drive timing.

The paper also talks about the possibility to establish a bus system to exchange information with supervising systems but also other charger modules in the charger. A setup like that can have precision current limiting and current sharing but also simplify production by having modules that can be combined in different ways to fit the needs of the customers.

Having digital control will also open up the possibility to control other parts of the charger from the MCU, like the fan. Other parts of the charger can also be controlled from the MCU, like introducing active rectification with a boost converter.

1.3 Description

The project involved comparing different suppliers models of real time compatible micro controllers. The MCUs was compared with regards to PWM outputs, RAM and ROM etc. to see which one best suited the needs of the charger. The MCUs was also judged based on if there were good opportunities for testing and development.

The next step after finding a good MCU was to test it first independently and then, if time allowed, replace the analog controller with the digital one in an existing charger.

An example of a MCU that might be used is one of the ones listed in [9], which is produced by Texas Instruments.

1.4 Delimitations

Early in the project when the MCU to be tested was chosen a delimitation was made. This delimitation specified that the first goal was now to get the development kit for the MCU up and running. After that the basic functionality of a battery charger would be implemented. Namely, peak current control(PCC) in an inner loop, slope compensation in an outer loop, the ability to change various parameters and basic communication.

1.5 The project and expected results

This project will after completion have contributed to the students continued education in electrical engineering. It will also have given practical experience in how to work when implementing theory in a product. For Micropower, the project will be a start of a move from an old analog to a new digital solution in the control of the power electronics. This will add to their competitiveness and ease when developing.

The project will run alongside the studies of the student from beginning of February until August when the project will be presented.

1.6 Resources

At the start of the project the student will be able to conduct most investigation based in Lund with occasional meetings at Micropower in Växjö. When the project enters the phase of testing and implementation the student will spend more time in Växjö where place will be made available. All equipment will be supplied by Micropower.

1.7 People involved

People involved from Micropower	People involved from Lunds Tekniska Högskola
Magnus Pihl magnus.pihl@micropower.se Tim Ringborg tim.ringborg@micropower.se	Examinator: Johan Björnstedt johan.bjornstedt@iea.lth.se Main supervisor: Gunnar Lindstedt gunnar.lindstedt@iea.lth.se Assistant supervisor: Mats Alaküla mats.alakula@volvo.com

2 Current setup

Today Micropower uses analog control the central part of their chargers, a phase-shift full bridge converter. This makes development of new models a hassle. Therefore a move to digital control of the power electronics is desired to make development easier and faster. Below, the present setup and the advantages of digital control is described in more detail.

2.1 Power supply

In according to traditional switch mode power supply layout, Micropowers charger takes an AC input from the grid. This input is either 1-phase or 3-phase and is rectified using a diode rectifier built for either 1- or 3-phase. The voltage output from the rectifier creates the DC-link that feeds the phase-shift full bridge converter. The converter is then switched to generate a high frequency ac-current that requires a much smaller transformer than if traditional 50 Hz AC input had been used. After being transferred to the secondary side, the current is rectified again to create an appropriate voltage level for the battery being charged.

2.2 Phase-shift full bridge converter

Below the central part of the battery charger, e.g the converter used today is described and how it can operate.

2.2.1 Topology

A phase-shifted full bridge converter can be used for both step down conversion and to provide isolation in high power applications. The converter consists of four transistors (MOSFET or IGBT) in a full bridge configuration on the primary side of the transformer. On the secondary side there can either be rectifier diodes or switching transistors. If transistors are used synchronous rectification can be performed which is preferable is the converter is operating at low voltage. With this topology Zero Voltage Switching can be achieved to lower switching losses and further improve efficiency. The topology can be seen in Figure 2.1 below.[1]

Basic operation

The bridge on the primary side consists of four switches, Q_A - Q_D, where Q_A and Q_B is in one leg and Q_C and Q_D is in the other. Q_A and Q_B are switched at 50% duty cycle, i.e 180° phase shifted in regards to each other. The same is true for the other leg with Q_C and Q_D. The phase difference between the two legs is what decides how much the diagonal switches, Q_A-Q_D and Q_B-Q_C, overlaps and therefore transfers energy to the transformer. The waveforms can be seen in Figure 2.2 below.

2.3 Different type of control

The converter can be run in different modes that are described below.

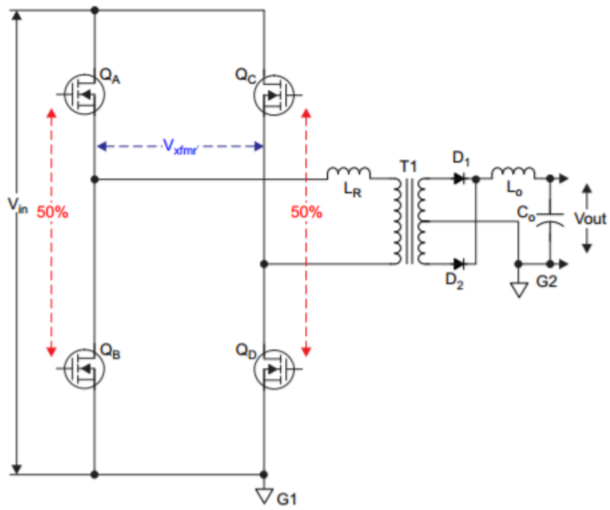

Figure 2.1: Phase-shift full bridge[1]

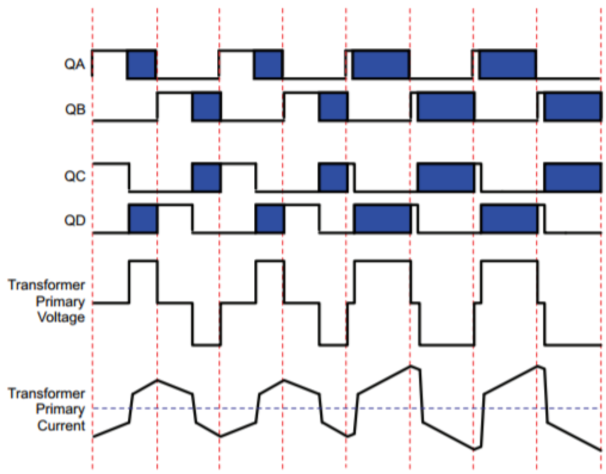

Figure 2.2: Waveforms for phase-shifted full bridge[1]

VMC - Voltage Mode Control
This control measure the average output of the converter and then changes the phase shift of the bridge to match the output to the reference.

PCMC - Peak Current Mode Control
This type of control is what Micropower uses today. The control works by having an external comparator that compares the current to a set value and changes the PWM accordingly. It has many benefits, among them are voltage feed forward, limiting the current cycle by cycle and flux balancing.

3 Digital Power Control

Digital power is the move to digital algorithms when it comes to the control and feedback in power supply, instead of using analog controllers. This gives more versatility and ease of use when developing new forms of power supply. Using digital power control also gives a reduction of passive component, and with that a reduced sensitivity for process and temperature variations. Additionally, a contribution to better power efficiency can be given by multi-mode operation and adaptive gate-drive timing. [8]

3.1 Requirements of good digital control

According to [8] certain requirements exist on the controller in order for optimal performance of the digital power application. Cited in the article is fast processing from the microcontroller core in order to make the digital application comparable to the analog one. If the power conversion application is at relatively low power the complexity and pricing of the implementation cannot be too high. What is also needed is fast A/D converters in order for the control algorithms developed to function as intended.

For fast transient measurements such as over-voltage protection fast analog comparators might be necessary. It would also be beneficial if the output could directly control the modulation scheme instead of having to use processing time to handle an interrupt.

3.2 Communication

The advantage of utilizing digital power in a power conversion application is the ability to communicate more with the controller. The MCUs investigated in this report feature a number of different possible protocols such as UART, SPI and I^2C. For this project the preferred protocol will however be CAN.

3.3 Slope compensation

The control scheme used in Micropowers chargers is peak current control. A problem that might occur when this is used is that when the duty cycle exceeds 50% the system might become unstable and give rise to sub-oscillations. This is demonstrated by an example in Figure 3.1 below.

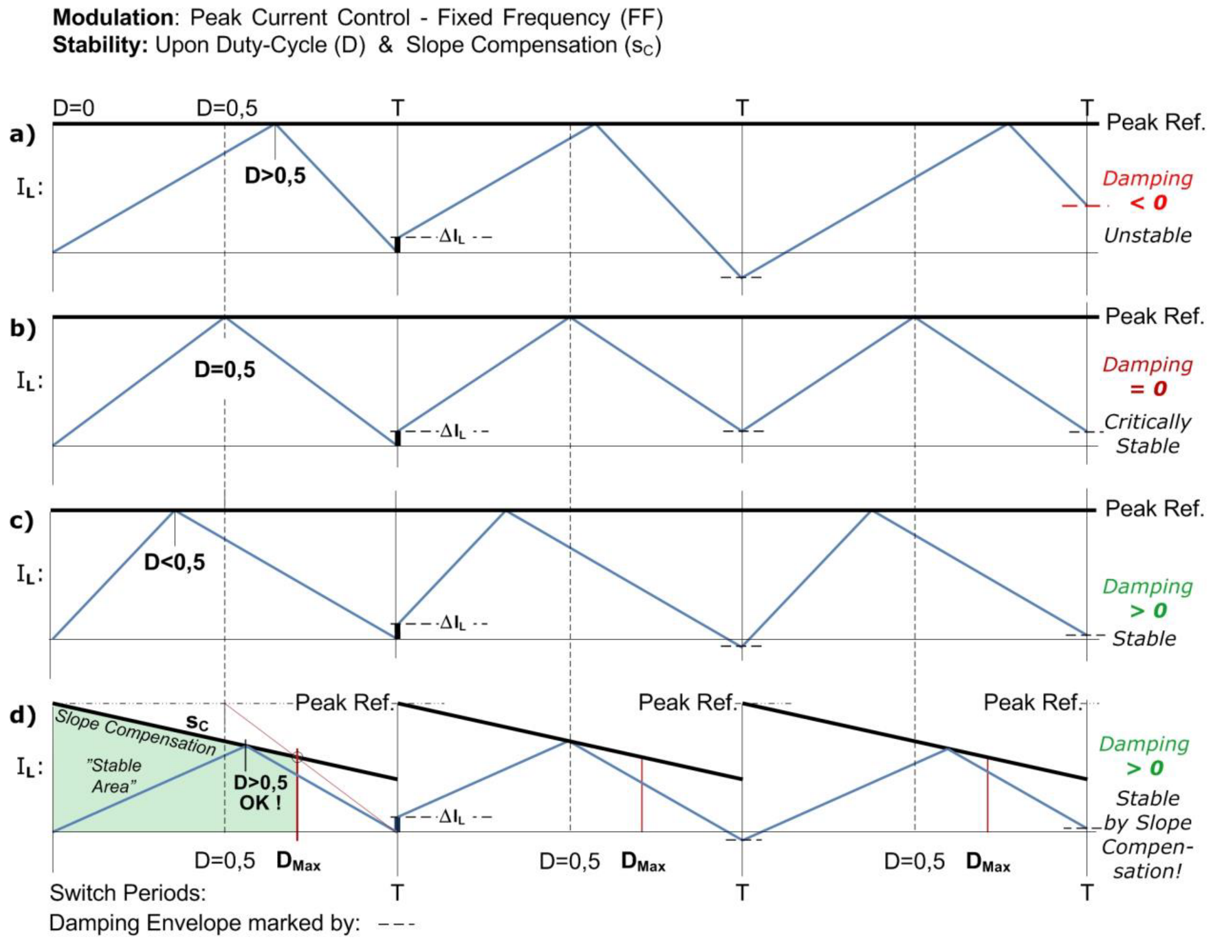

Figure 3.1: Demonstration of the oscillations that will occur when duty cycle exceeds 50%[2]

In the first example of Figure 3.1 the peak current reference is set to a value that causes the duty cycle to exceed 50% the current will .... Figure 3.1 also shows in the last example that when the peak current reference is a slope, the system will be stable even when the duty cycle is above 50%.

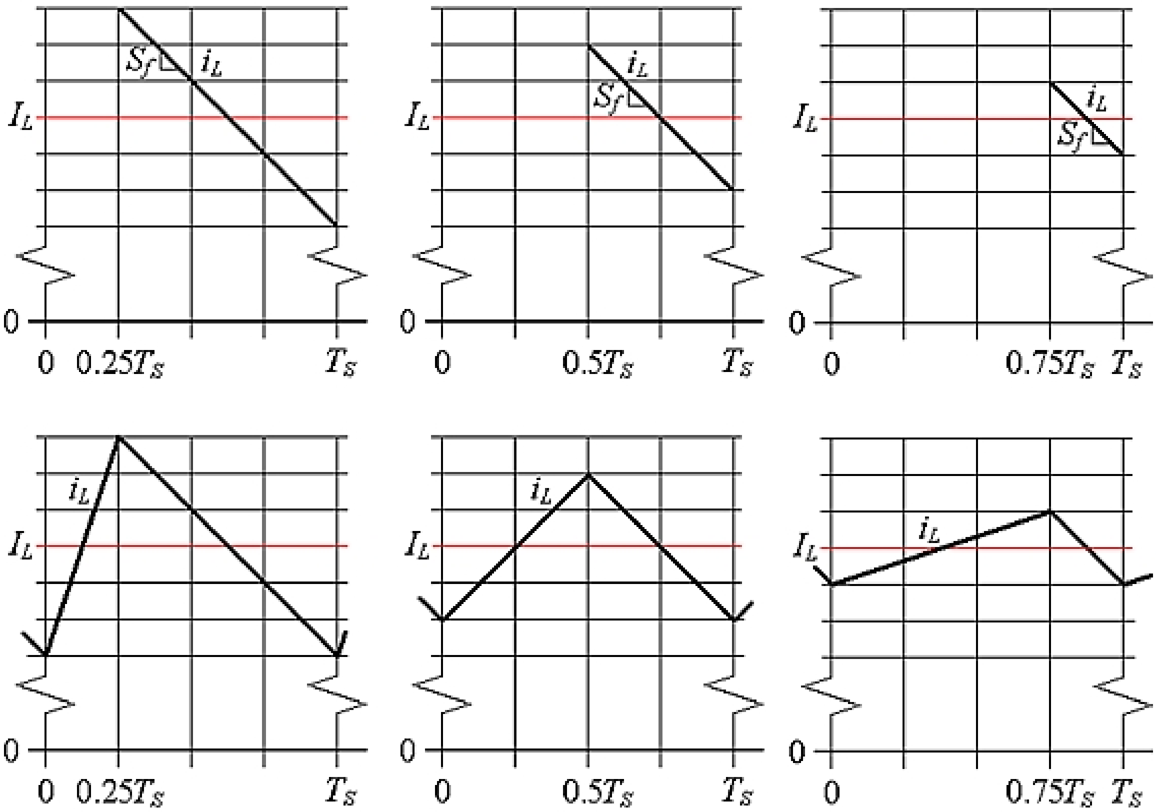

Figure 3.2: Slopes with different duty cycles but same average current[3]

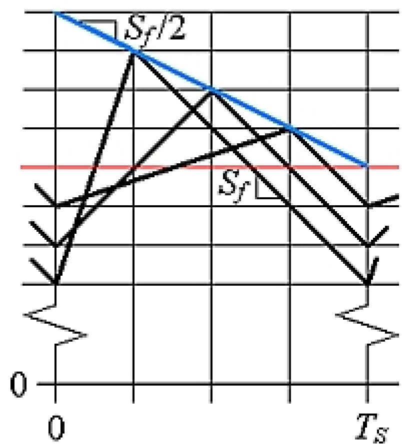

Figure 3.3: Proof by demonstration for the correct slope of the compensation[3]

To show what the optimal slope used in the compensation should be Figures 3.2 and 3.3 is used. In the first figure it shows how the same average current is achieved with different duty cycles. These current slopes is combined in Figure 3.3 and it shows that the apex of all the slopes form a slope with half the decline as the current slope. This means that the slope compensation should be chosen as

$$S_c = S_f/2 = \frac{-V_{out}}{2 * L}$$

where V_{out} is the output voltage of the converter and L is the main inductance.

3.4 Pulse Width Modulation

To control voltage levels in the power supply Pulse Width Modulation(PWM) is used. This is the process of switching some transistor configuration on and off to create a lower voltage level than the input. This is demonstrated in Figure 3.4 where a pulse generated by a switch that is controlled with PWM. In the figure, D.T means the on time of the transistor and the voltage out will be the average of this pulse. The output voltage can be calculated with D as

$$V_{out} = y_{max} * \frac{D.T}{T}$$

[4]

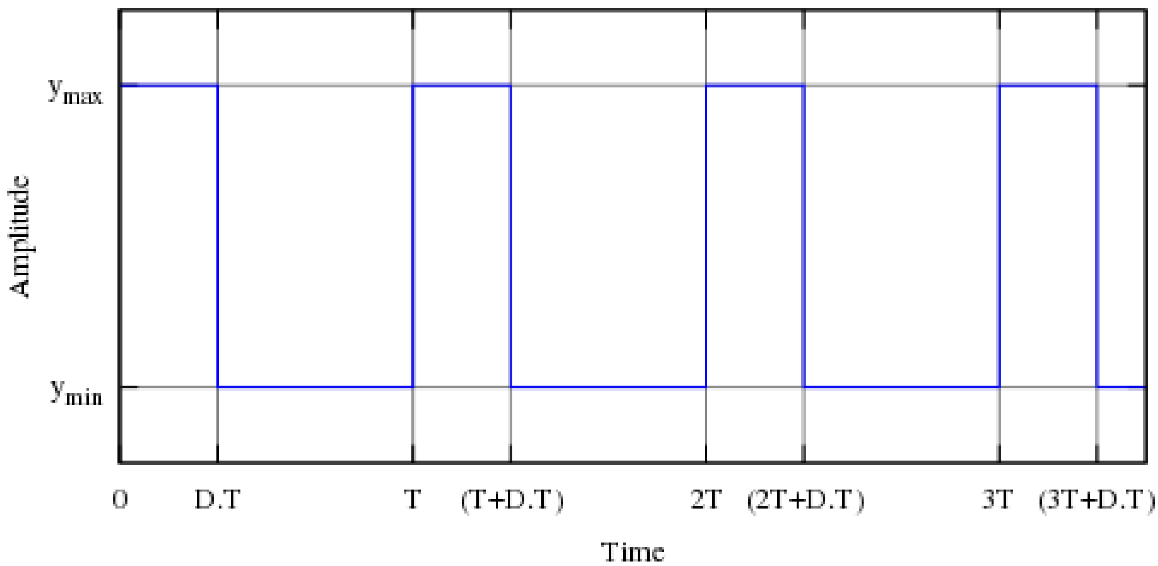

Figure 3.4: A pulse created with PWM[4]

This is how PWM is done in general, for the application in this project a fixed duty cycle of 50% for each leg of the bridge is used. The legs are then phase shifted in regards to each other as described in the previous section.

3.5 Blanking

Another parameter that needs to be taken into account when doing peak current control is blanking time. It is the time after the transistor/s have been turned on that the comparator will ignore measurement of the current. This is because after turn-on the current will be rushing for a short while until the diode is no longer in reverse recovery mode. If there was no blanking time, the comparator might measure a current higher than reference and turn off the transistors. This would cause the current and voltage drop which would be bad.

3.6 Dead time

The time to wait between switching on and off the transistor in each leg is referred to as the dead time. This is parameter that has to be tuned depending on the charger configuration. Different power levels will have different capacitor, and the dead time has to changed accordingly.

4 Microcontroller

To see what microcontroller is best suited for the needs of Micropower a market evaluation of what controllers are available will be conducted. Different suppliers models will be compared and the best one will be chosen based on the specifications.

4.1 Description

A microcontroller is described as a small computer on a single integrated circuit. It has one or several CPU:s and controllable input/output peripherals. It can also be described as a microcontroller unit or MCU which will be how it is referred to from now on. The different functions available for MCUs which will be relevant for this project is described below.

4.1.1 Timers / Pulse width modulation

To control the power electronics in the battery charger pulse width modulation, PWM, is used. The modulation is created using counters which counts pulses from the clock of the circuit. To achieve correct frequency of the PWM or measure a specific time using a timer the clock frequency can be scaled down using a prescaler which will tell the counter unit not to count all pulses.

The output from the PWM will then be determined by comparing the value of the counter to a register that has been set previously. Depending on what mode the PWM should operate in it can either set the output low/high when the compare matches or use two counter register and set high when the first one matches and low when the second one matches.

4.1.2 Communication - measurement

To communicate with the analog world the MCU can use Analog to Digital Converter (ADC) or Digital to Analog Converter (DAC).

The ADC senses an analog voltage and compares that to the supply voltage of the MCU. Then, depending on the resolution of the ADC, a number between 0 and 2^n-1 is returned, where n is the number of bits used in the converter. This number is then used to scale the measured voltage with the help of the supply voltage.

4.1.3 Communication - CAN

MCUs can usually implement an array of communications protocol but for this project the most interesting one will be CAN. CAN or Controller area network is a vehicle bus standard that does not need a host to function. It allows devices to communicate using a message based protocol.[10] The microcontroller will provide two pins as input and output and these will in turn be connected to a CAN-driver. The driver will transform the signals to high and low levels.

4.1.4 CPU

The heart of the MCU is the Central Processing Unit (CPU). It performs the instructions stored in the memory and can, in certain models, do calculations with the help of a Floating Point Unit (FPU) according to the IEEE745 standard. The CPU also handles interrupts.

4.1.5 Memory

To store the instructions used for the application MCUs typically use flash memory. Instead of ROM memory the content can be overwritten while still retaining it while the device is turned off.

To store program variables the Random Access Memory (RAM) is used. In microcontroller applications this is typically represented by Static RAM(SRAM). This type of configuration supports fast access and does not require periodic refreshment but is a complex circuit and therefore hard and expensive to create large quantities of.

4.2 Specification

The initial specification for the MCU are listed below. These demands are the minimum requirements but the MCU should also have enough ports and modules to allow for further development.

- 5 ADC
- 4 PWM outputs for H-bridge
- High speed comparator for control scheme
- Floating point unit for control calculation
- CAN communication
- Some digital inputs for over-voltage protection

4.3 Suppliers

4.3.1 Texas Instruments

Texas Instruments provides real time controllers in their C2000 series. Their development kit for a phase-shifted full bridge uses the cheapest version in [11] but with some modifications. The alternatives from Texas instruments are the following, where both of which are from the C200 piccolo series which are good for broad closed loop applications.

TMS320F2802x

Processing	40-60MHz, 32-bit
Memory	Up to 64kB flash, up to 12kB RAM
PWM	8 channels of HRPWM
ADC	12-bit with 4.6 MSPS capacity
Comparator	2 analog w/ 10 bit ADC
Connectivity	UART, SPI & I^2C

Table 4.1: Specifications for 2802x MCU

TMS320F2803x

Processing	60MHz, 32-bit + independent CLA with FPU
Memory	Up to 128kB flash, up to 20kB RAM
PWM	14 channels of HRPWM
ADC	12-bit with 4.6 MSPS capacity
Comparator	3 analog w/ 10 bit ADC
Connectivity	CAN, UART, SPI & I^2C

Table 4.2: Specifications for 2803x MCU

4.3.2 ST

The choice from ST is part of their F3 series of 32 bit microcontrollers. The series has been designed with cost reduction and simplification of application design. ST states that the series have fast processing core with good analog peripherals.

STM32F334

Processing	72MHz, 32-bit ARM Cortex M4 with FPU
Memory	Up to 64kB flash, up to 16kB RAM
PWM	10 channels
ADC	2 x 12-bit with 5 MSPS capacity
Comparator	3 with 26ns response
Connectivity	CAN& SPI

Table 4.3: Specifications for STM32F334 MCU

4.3.3 Infineon

Infineon is a German producer of semiconductor solutions. The choice from them is from the XMC4000 series which is based on the ARM Cortex-M4 core and are recommended for applications in digital power conversion, motor control and IO applications. The line suites the needs of this project and the XMC4200 has enough performance, fulfills the specification and is reasonably priced. The specific configuration is listed below.

XMC4200

Processing	80Mhz ARM Cortex-M4 with FPU
Memory	Up to 256kB flahs, up to 40kB RAM
PWM	4 channels High Resolution PWM
ADC	4x12-bit
Comparator	Included
Connectivity	SPI, I^2C, UART and 2 CAN Nodes

Table 4.4: Specifications for XMC4200

4.3.4 Renesas

Renesas offer a line of more exclusive microcontrollers. These are faster and has more memory but comes at a much higher cost.

RX62G

Processing	100 MHz with RC CPU core
Memory	Up to 256kB flahs, up to 16kB RAM
PWM	4 channels PWM
ADC	8x12-bit, 12x10bit
Comparator	Included
Connectivity	SPI, I^2C, UART and 1 CAN Node

Table 4.5: Specifications for RX62G

4.3.5 NXP

NXP has a Kinetic line which is optimal for motor control or digital power applications. The KV1x is one of the entry versions in that series.

KV1x

4.4 Comparison

When all considered manufactures products had been found that matched the initial specification they were compared to see which would be best suited for Micropowers charger solution. The criteria that the comparison was made based on is listed below.

Processing	75 MHz with ARM Cortex-M0 CPU
Memory	Up to 128kB flahs, up to 16kB RAM
PWM	2x6 and 4x2 timers for high-accuracy PWM
ADC	2x16bit
Comparator	2 with 6-bit DAC
Connectivity	SPI, I^2C, UART and FlexCAN

Table 4.6: Specifications for KV1x

The MCU:

1. Has features and performance that fulfills specifications
2. Has a development kit available for easy start-up
3. Is reasonably priced
4. Has long time left on product lifecycle
5. Has good compatibility with existing development platform

The first criterion is already fulfilled by default so next the microcontrollers was compared based on the availability of a good development kit. The best development kit was from Texas instrument which was a complete phaseshifted full bridge converter based on the less powerful MCU from Texas, Table 4.1. Some of the other MCUs had PSFB converters but they were often based on other MCUs so the development kits were only for digital power.

Regarding the price of the MCU, the alternative from Renesas, Table 4.5, had a very high price even for the cheapest versions so this alternative was disregarded. Initial pricing on the other alternatives in the comparison were fairly even with one of the alternatives from Texas, Table 4.2, being a bit more expensive.

The choice from ST was excluded because compared to the rest of the lineup seemed weaker and more adapted to motor control.

The third criterion regarding lifecycle was fulfilled by all, the manufactures are all established on the market and will hopefully keep the MCUs alive for many years.

By this time the search was narrowed down to the MCUs from Texas and Infineon. The reason for this was that they both offered greater features that the competition, good development kits and all at a reasonable price. Further pricing information was now brought in with the help of the purchasing department at Micropower. The indicated prices can be seen in Table 4.7 below. Prices were brought in for different variations of the chosen MCUs.

Infineon	
XMC4200F64F256BAXQMA1	2,4 €
XMC4400F64F512BAXQMA1	3,4 €
Texas Instruments	
TMS320F28035PAGT	4,6 €
TMS320F28075PZPT	10,4 €

Table 4.7: Prices for considered MCUs

The development environment differed between the two candidates. Texas Instrument had their own environment to program the MCU while the Infineon MCU could be programmed from the same environment already used at the company. This is a huge benefit since there won't be anyone that has to learn a new environment. Infineon does provide an environment of their own called Dave, it is not necessary to use but can be helpful while starting up development. The environment provides modules called APPs that each represent a function in the MCU. This makes it easy to configure different functions of the MCU to get started.

Based on this the MCU chosen was the XMC4200 from Infineon.

5 Development kit and DAVE

To get started with development of the digital power application a development kit was acquired from Infineon. To actually develop the application, and write necessary code, a tool from Infineon called DAVE was used. Descriptions of these are found below.

5.1 Description of devkit

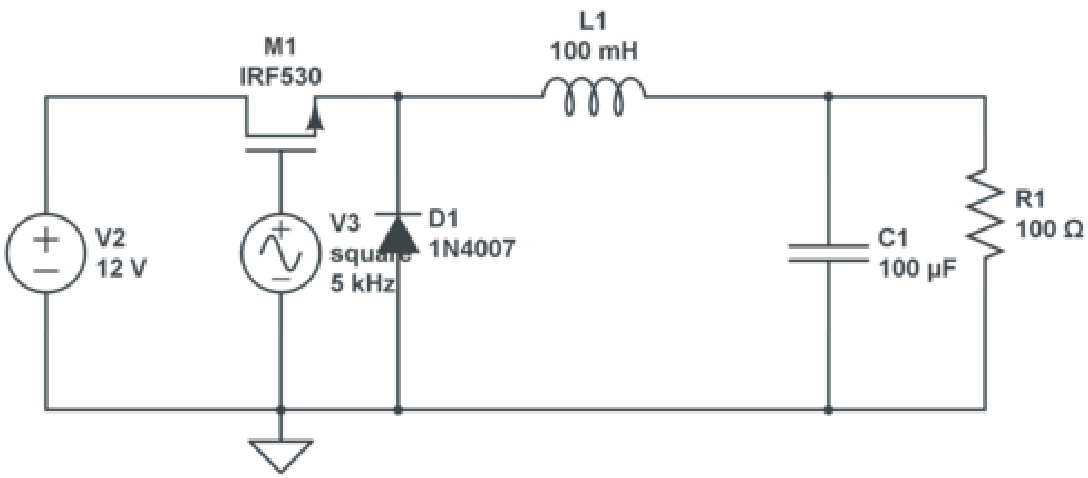

Figure 5.1: Buck converter [5]

The best suited development kit that was available for the XMC4200 was a digital power exploration kit[6]. This came with a converter board and a control card board that contained the MCU. On the converter board was a synchronous Buck converter. This is a slightly modified version of the regular Buck converter that can be seen in Figure 5.2. The regular buck converter works by conducting input voltage a certain percentage of the time with the help of a transistor. The modulation signal sent to the transistor is usually a PWM signal but it also works with a regular square wave as in the figure. If PWM is used, the output voltage is the same percentage of the input as is used in the PWM. For a fixed square wave the output voltage is half of the input. When the transistor is not conducting the current on the output side is free-wheeling. In Figure 5.2 the current goes through the diode. In a synchronous Buck converter a transistor is used instead. This is done because the conducting voltage drop of a transistor is lower than that of a diode, which is good when operating voltages are lower.

The operating range of this converter is an input of 12V and an output of 3,3V. This is fine for developing and testing purposes but will change when implemented in the finished product. The converter used will also be different, but the power supply solution in the battery charger can be simplified as a Buck converter, so the setup in the conversion board is good.

The controller card also has a second XMC4200 that works as a debugger. With the help of this the program written for the MCU can be debugged through Infineons DAVE.

To get an idea of how the converter looks and how the control was done, see Figure 5.2.

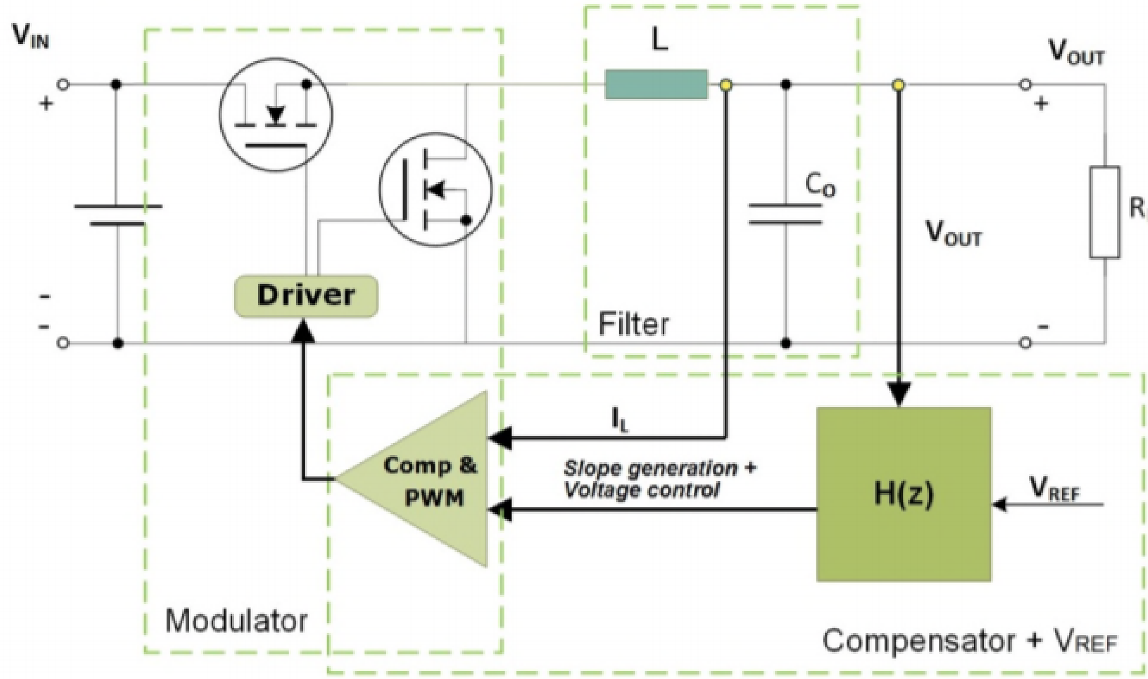

Figure 5.2: Synchronous buck converter with peak current control[6]

The kit also contains pins that allow an oscilloscope to display the PWM waveforms and switches that allow variable load to test performance.

5.2 Description of DAVE

DAVE is Infineons development platform for microcontrollers [12]. It can be used to flash and reprogram microcontrollers and also debug existing software. To make development easier there are modules called APPs in the program that makes it easy to enable and configure different functions in the MCU. Some of the APPs are described below. When the APPs are configured code can be generated based on the configurations made in the GUI. The GUI can be seen in Figure 5.3. The generated code is then included in the `main.c` function as follows. `DAVE_Init()` is then called to start and initiate all the modules configure with the APPs.

```
#include <DAVE.h>
DAVE_STATUS_t status;
status = DAVE_Init(); /* Initialization of DAVE APPs */

if (status == DAVE_STATUS_FAILURE) {
      XMC_DEBUG("DAVE␣APPs␣initialization␣failed\n");
      while (1U) {}
}

while(1U){//Main loop}
```

5.2.1 HRPWM

The HRPWM APP will activate the High Resolution Pulse Width Modulation and be able to configure different event and the frequency at which it should switch. The events available are among others, compare match and period match. These can be used, for instance, to trigger an AD conversion.

In the APP, incoming events can be configured to, among other things, reset the PWM cycle. This will be the case when peak mode current control is used.

5.2.2 CSG

CSG is short for Comparator and Slope Generator. Activating this APP will start a comparator that takes input from an ADC and a preset register. The APP can be configured so that events occur when the measured signal is greater than the specified. This is used in peak current control where the measured current is compared to the register and if the measured signal is greater, an event will tell the PWM to stop its current cycle.

It is also in this APP that the important slope compensation is configured.

5.2.3 VADC

An ADC APP, Analog to Digital Converter APP, will allow measurements from the real world in up to eight channels. The measurement can be triggered by other events and can create events when measurement is complete.

In the ADC APP there is also a possibility to limit certain currents and voltages, and create events when they exceed specified limits.

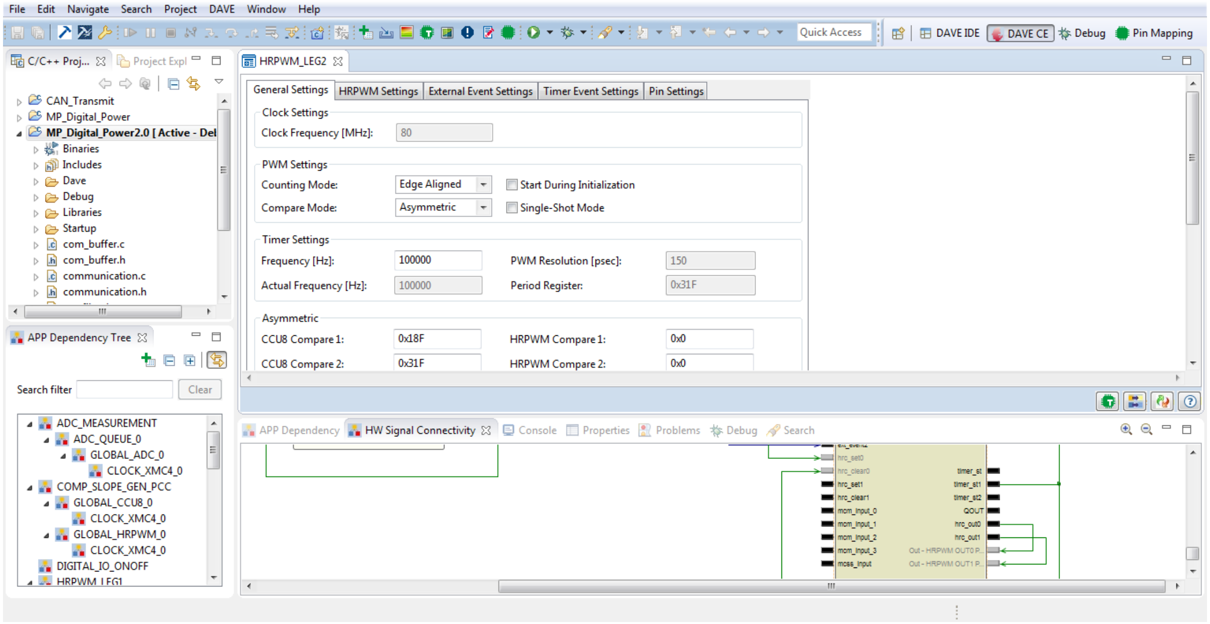

Figure 5.3: The DAVE GUI

6 Implementation

6.1 First delimitation - Development kit up and running

According to the delimitation made in the beginning of the project the first step would be to get a development kit for digital power up and running. Further, a similar setup to what exist in Micropowers chargers today would be implemented. In this section this the process of that implementation is described.

6.1.1 Design of control

For the first design test of the digital power solution, peak current control was used. This control scheme is used today in Micropowers chargers, so it was appropriate that the reference design should have that as well. Peak current control(PCC) consists of two loops, one inner loop that is on cycle-to-cycle basis and one slower that compares apparent value to reference value. A reference structure is show in Figure 6.1. The fast inner loop will reset the PWM signal when the current measured reaches a set reference. This reference is adjusted according to the slope added through slope compensation, see earlier chapter. PCC has many benefits, among them are voltage feed forward, limiting the current cycle by cycle and flux balancing. Some disadvantages is that the duty cycle is unpredictable and too high values might lead to oscillating behavior that has been previously discussed. [2]

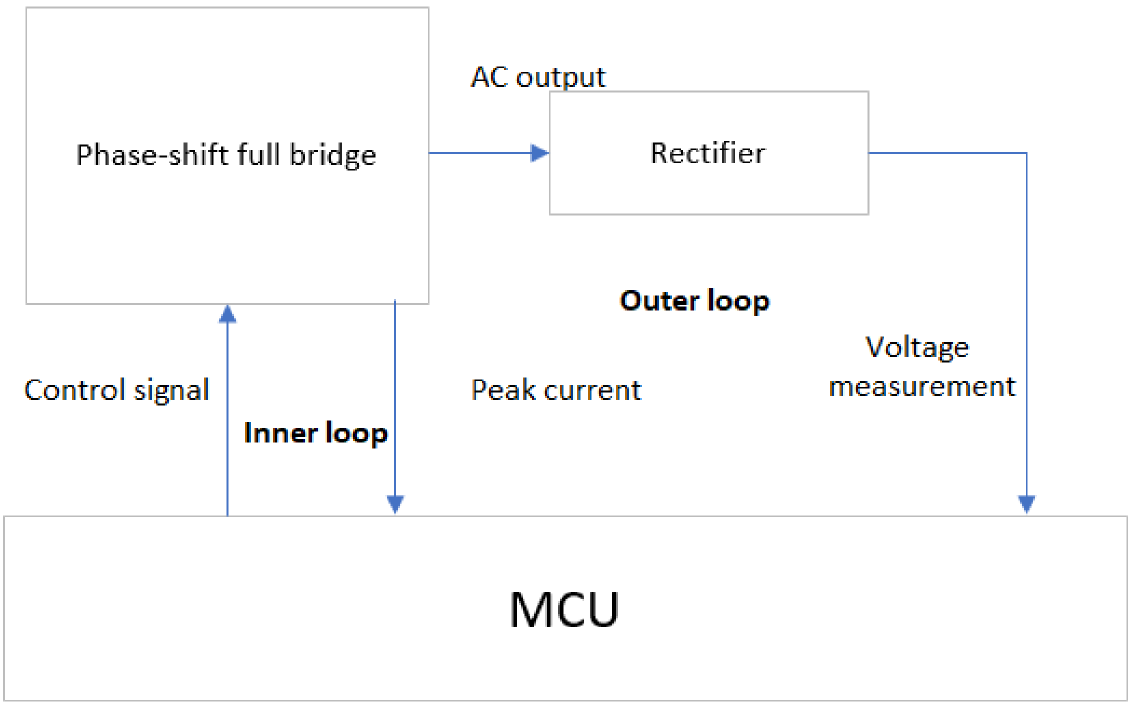

Figure 6.1: Loop structure of first implementation

The inner loop is realized by the CSG APP in DAVE. This has built in slope compensation and analog comparator that compares values from the fast AD converter. When the current matches the reference current peak an event is triggered that resets the HRPWM that will only turn on when the next cycle begins. In the first design the frequency of HRPWM, also the frequency of the inner loop, was set to 200kHz.

A slower outer loop was created that would handle the voltage control. This loop was executed with a frequency of 5kHz and every cycle the reference voltage was compared to an average of the output voltage. The average was given by a digital moving average filter over the last 16 samples. A new peak current reference was then computed with a simple PI-controller and the new reference was changed in the CSG APP.

In the first example studied there was a 2-pole-2-zero digital filter that was implemented for the control of the converter. This was replaced by a PI controller instead. This was because the digital filters coefficient were specific for one reference and one frequency. A PI controller felt more adaptable and intuitive. In the first listing the control signal is calculated and in the second the integral part is updated. This is according to realtime systems practice, the control signal should be updated as fast as possible to reduce the delay between measurement and actuation. Some calculations can therefore be delayed since they are not needed for the current control signal.

```
__STATIC_INLINE void calculate_control(PI_COEFF_t* ptr,
                              uint16_t vFb){
      ptr->e =(float) (ptr->ref - vFb);
      float acc = ptr->K * ptr->e + ptr->I;
      ptr->v = acc;

      acc = MIN( acc , ptr->max );
      acc = MAX( acc , -ptr->max );
      if ( acc < ptr->min) acc = ptr->min;

      ptr->u = acc;

      ptr->out = (uint32_t) ptr->u;
}
```

```
__STATIC_INLINE void update_states(PI_COEFF_t* ptr){
      ptr->I = ptr->I + ptr->E_COEFF*ptr->e + ptr->V_COEFF*(
         ↪ ptr->u-ptr->v);
}
```

6.1.2 Rest of system

The measurement signals was sampled with the same frequency as the switching, that is 200 kHz. After the message was read from the ADC it was added to a moving average filter to eliminate spikes in the measurement. The code for this can be seen in a listing later in the section.

Communication over the UART interface was first established to be able to change the reference and see important measurements. This was implemented in a slow loop outside the control loops that would send messages to the computer with a frequency of 3 Hz. This would also check incoming messages that would change the reference.

This was implemented with a message buffer that would be filled when measurements came in. The loop would then check if there were available messages and send them. The messages used a simple protocol that gave a message identity and a length so that the computer program knew what was sent and the size of the message. Functions were created to help with creating messages.

Slope compensation was already implemented in the hardware, but a method to change the slope adaptively was implemented. As described previously the slope of the compensation should be equal to half of the slope of the output current. The slope compensation is decided by two registers, one that decides how many pulses to swallow in order to determine how often a counter should be incremented. The second register decides how much the counter is incremented every time a pulse comes through. Calculations were made in the code to find the slope and then setting the registers to appropriate values.

```
float slope = ((float) vout) / (2*L*100000);

float dy = slope * CPU_PERIOD / MAGIC_NUMBER;

int gain = 1;
float puls = 0;
if (slope > 2000) {
	gain = 8;
	pulses = 0;
} else if (slope < 4) {
	gain = 1;
	pulses = 63;
} else {
	int i;
	for (i = 0; i < 4; ++i) {
		puls = 64 - dy * 64 / ((float)gain);
		if (puls < 64 && puls > 0) {
			break;
		}
		gain *= 2;
	}
}
```

The slope was calculated using the code in the listing above. The first line calculates the slope in mV/μs, `vout` is the output voltage given in mV from the AD-converter and L is the size of the primary inductance. In order to set correct values of the registers previously mentioned the slope over just one period of the CPU is used. This is what the second line is used for. Unfortunately, the second line also contains a magic number defined as 3.572. This is because when a value for the slope is entered in the DAVE gui, for the slope compensation component, the values for the pulse swallow and gain register give a different slope when calculated by hand. Fortunately, when different values of the slope were tested, a linear equation between calculated and entered value arose. The coefficient of this equation is the magic number.

The next part of the code finds appropriate pulse swallow value by trying different gains in ascending order.

After this snippet is executed, built in functions are used to set the registers to new values.

To be able to get the average of different measurements a moving average filter was implemented. This was realized by a circular buffer of a size that was a multiple of 2 so that byte shifting could be used instead of division. The code for the filter is listed below.

```
typedef struct moving_average_filter{
	size_t size;
	uint8_t filter_index;
	uint32_t sum;
	uint16_t filter[16];
} moving_average_filter_t;

__STATIC_INLINE void init_filter(moving_average_filter_t* f){
	int i;
	for(i = 0; i < f->size; ++i){
		f->filter[i] = 0;
	}
}

void add_to_filter(moving_average_filter_t* f, uint16_t v){
	f->sum -= f->filter[f->filter_index];
	f->filter[f->filter_index++] = v;
	if (f->filter_index >= f->size){
		f->filter_index = 0;
	}
	f->sum += v;
}

uint16_t get_average(moving_average_filter_t* filter){
	return filter->sum >> 4;
}
```

In Figure 6.2 below, the hardware connections for the digital power project in DAVE can be seen.

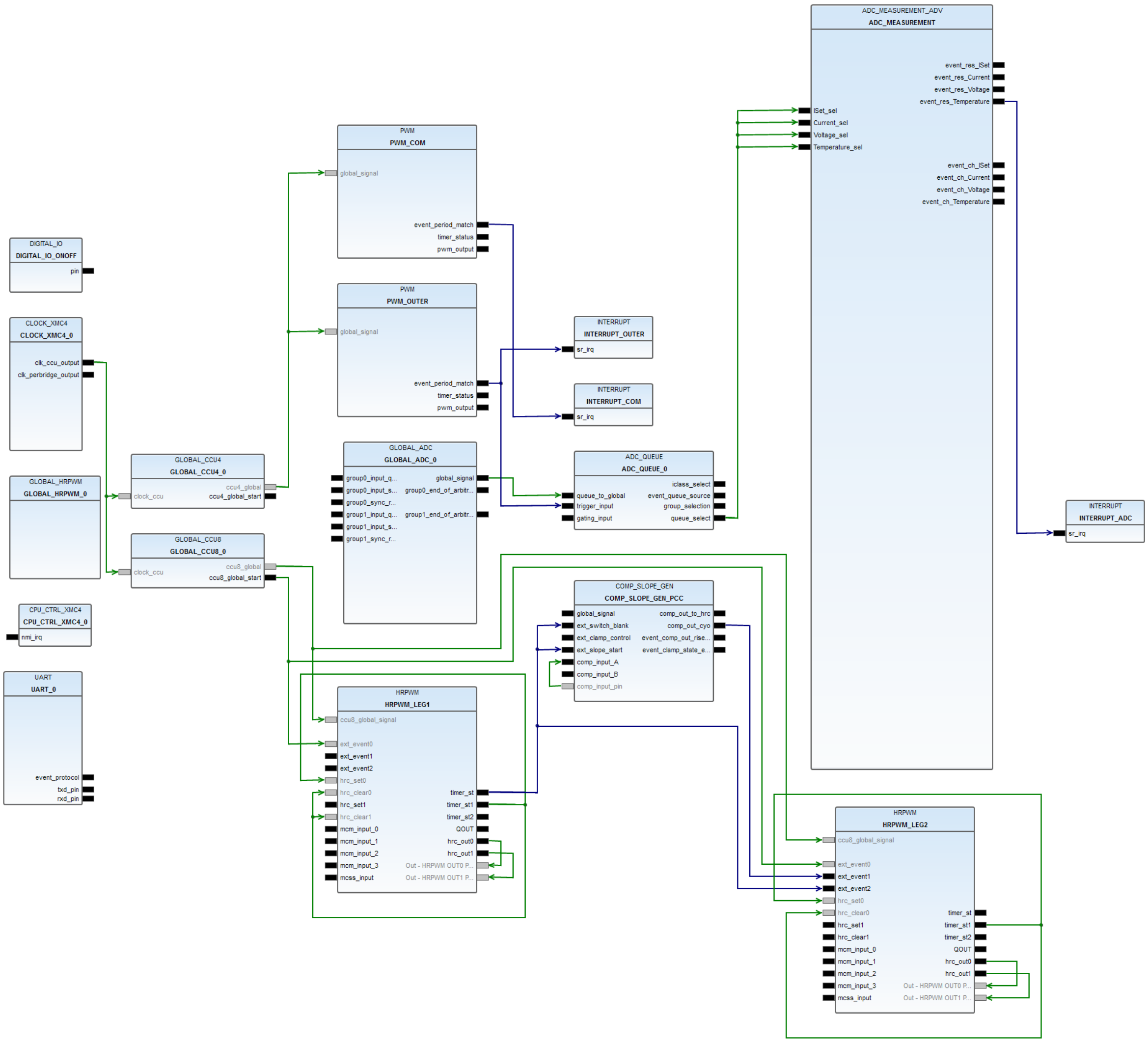

Figure 6.2: Hardware connections for digital power project

6.2 Pingvin implementation

In this section follows the implementation to test the digital power solution, that has so far only been implemented on a development kit, on one of Micropowers battery chargers (codenamed Pingvin). In order to do this the circuit board that contains the analog control of the charger was redesigned in order to fit the XMC4200 control card. This design was made with the following specifications that are required to control the battery charger.

PSFB control	4 PWM output
Power on	1 digital input
AD measurements	3 ADC inputs:
	Output voltage
	Output current
	Input current
Peak current control	1 fast comparator input
Options	1 digital input
Current reference through potentiometer	1 ADC input

Table 6.1: Requirements for MCU for Pingvin implementation

The first priority was to achieve a solid DC current control which requires the following.

- Control of bridge legs in PSFB
- Peak current control
- Controller for DC current
- Slope compensation
- Adaptive Deadtime control

Compared to the development kit, the PSFB requires two HRPWM modules to control the two legs of the bridge. The left leg in the bridge is considered the leading leg, so one of the modules is set to 50% duty cycle at the chosen frequency with the direct output connected to the upper transistor and the inverted output connected to the lower. The deadtime for both rising and falling steps can be chosen appropriately with the charger configuration.

The other leg of the converter bridge is also controlled with a module set to 50% duty cycle with the same frequency. But the status of this module can be changed from the peak current control scheme. This is done by external override of the status of the second HRPWM module. Two external inputs are configured for the module, one that triggers when the status should be overridden and the other one decides the level. The current coming out of the bridge is coming into the CSG app and is there compared to the slope generated. When the measured current reaches the slope, the module outputs a high signal which triggers the second HRPWM module. The level the status should be set to is the same as the level of the first HRPWM module because no power is transmitted from primary to secondary side when both HRPWM module have the same status.

When proper control of the bridge legs was implemented it was checked that increasing current peaks would give increasing DC current out. After that a simple PI controller was created to instead control the DC current. The ADC module was implemented so that the DC current could be measured.

Slope compensation was not implemented but could be done in a similar fashion to how it was done for the development kit.

Another thing that was not implemented was adaptive deadtime control. This is a method that changes the deadtime between turn-on of the transistors in each leg. This is needed because different configurations of the capacitors in the circuit requires different deadtimes.

7 Results

7.1 First experimental results

The development kit acquired from Infineon was easy to start up. On the website for the development program Dave, there were a lot of example projects that could be studied. There was also a few projects that were specifically designed for the development kit. One of these used Peak Current Control of the buck converter so this was chosen as a first test. It worked splendidly, all parts of it including the slope compensation. The PI-controller created could, with some tuning of the parameters, handle changes in reference quite good.

In order to test the implemented adaptive slope compensation a duty cycle of more than 50% was necessary. The reference was not able to be set higher than 4 V because of the AD-converters limit, so instead the supply voltage had to be reduced. This was more tricky than first expected because the gate driver needed the full 12 V from the original supply to drive the transistors. A cut was made in the board and a wire was soldered in to supply the gate driver while the original supply could be reduced to about 5 V which gave a duty cycle of >50%. With this setup it could be confirmed that the slope compensation worked.

The control scheme used was easy to implement because of the HRPWM module in the MCU that contains all the features to do the control almost without software involvement. The fast inner loop used only hardware connections and can therefore be very fast. The outer loop uses software to calculate the peak current reference and the slope compensation. This setup work well and there was plenty of time left for the processor to do communication, measurements and more in the future.

The communication over UART could be performed at the same time as everything else, even though the switching frequency was very high.

The measurements taken with the AD-converter was sent to the computer connected with UART. The values displayed were compared to measurements in the oscilloscope and matched well.

7.2 Pingvin result

The implementation of digital power on Micropowers charger was successful. The microcontroller could control the legs of the PSFB with the PCC scheme and the current output of the charger could be controlled with a simple PI controller.

The adapter card created, with the help of the engineers at Micropower, in order to be able to use the digital power control card worked reasonably well but some improvements can be made for the next iteration. An image of the adapter card can be seen in Figure 7.1.

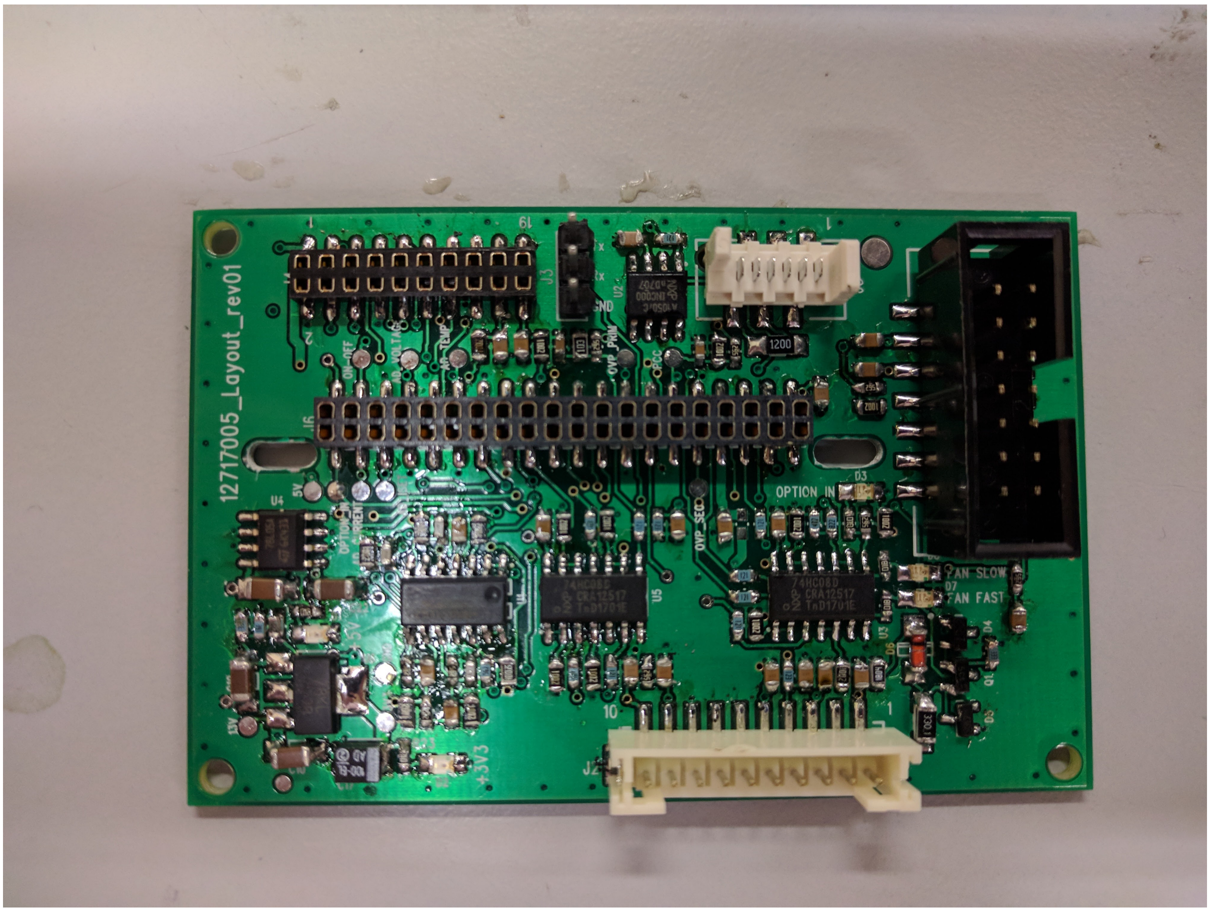

Figure 7.1: Adapter card created to be able to use digital power control card with Micropower charger

In Figure 7.2 below the plugged in adapter card can be seen when connected to the charger. In the figure the connections to the rest of the charger is shown. The secondary side connector delivers a scaled down version of output voltage and current signals for the ADC. The primary side connector delivers the current signal for PCC but also takes the control signals for the gate driver that drives the transistors in the bridge. The connector to supervising system is connected to a black box that can control on/off signal and the reference level through a potentiometer. This black box is used to simulate other, more advanced supervising system.

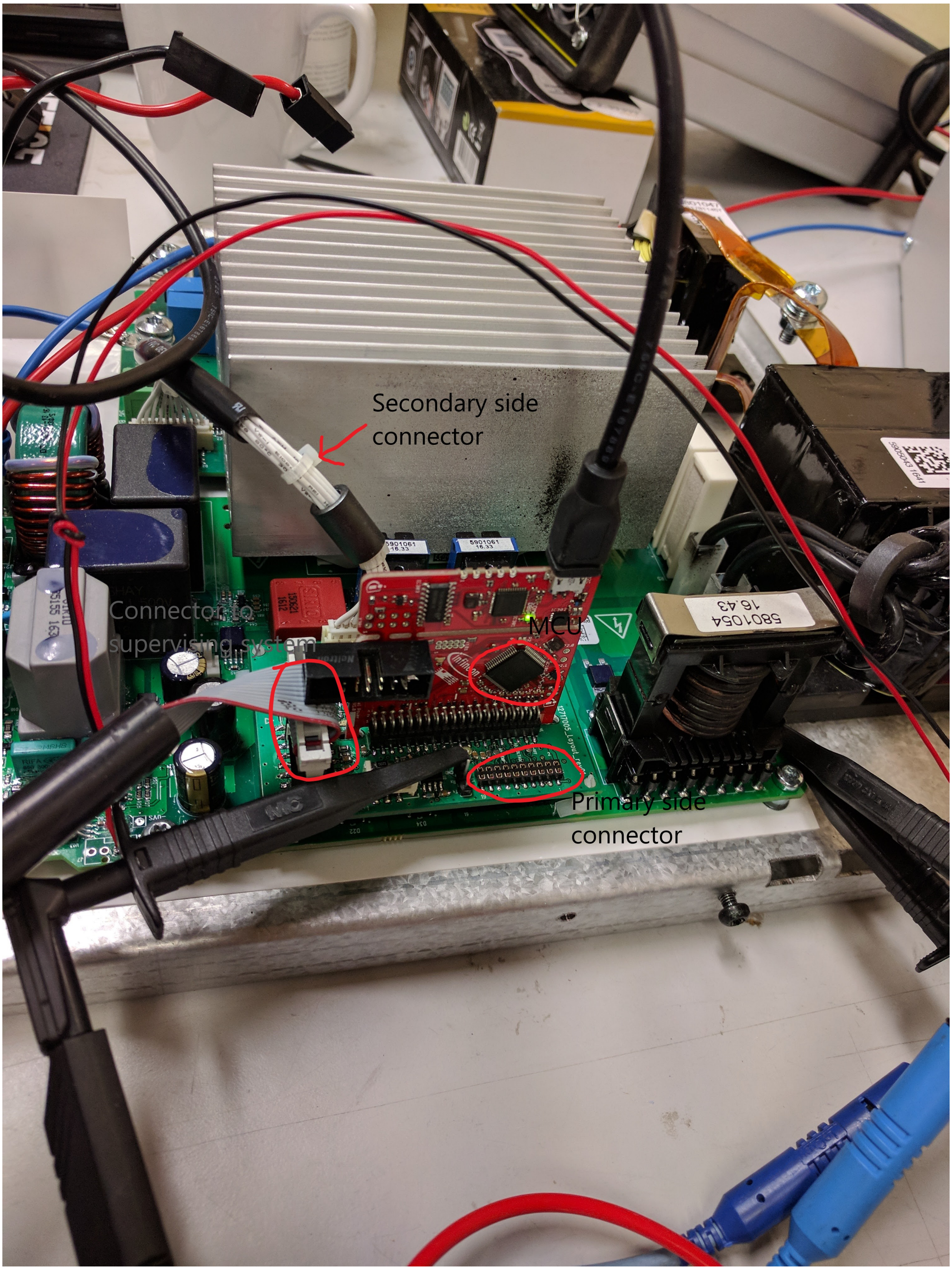

Figure 7.2: Adapter card and control card plugged into Micropower charger

The waveform generated by the HRPWM modules when the PCC scheme was used can be seen in Figure 7.3. The yellow and red lines represent the signals to the left leading leg and the green and blue lines are from the right leg. Power is transferred to the secondary side of the transformer when either red and blue or yellow and green are both high.

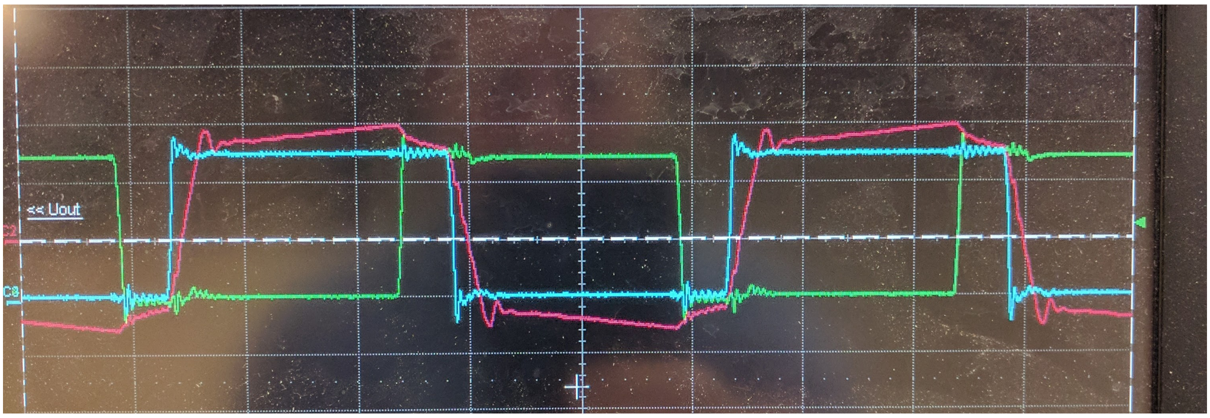

Figure 7.3: PWM waveform created by the PCC scheme

Since the figure is photographed of an oscilloscope, the scales are not so clear. For the voltage signals each square is 20 V and for the red current signal each square is 2 A.

8 Discussion

Below are comments on how the implementation went on the development kit. It is also discussed what challenges arose and future improvements when digital control was implemented on one of Micropowers charges.

8.1 First delimitation

The process of getting the development kit up and running went surprisingly easy. No real issues were encountered so development went smoothly. A lot of help could be found in the examples of digital power in the development environment. The examples went as far as to get the PCC scheme working for one HRPWM module but other things was also implemented to get a better current control, such as slope compensation. Some things were tricky to get working but since the experimental board had such low power it was safe and easy to do lots of testing and experimenting.

The only problem encountered was mentioned in the result, that the gate driver for the transistor required a higher supply than was required to test the slope compensation.

8.2 Challenges with implementing in battery charger

When the digital power control was instead implemented on one of Micropowers battery chargers more challenges arose. This was expected since it is such a complicated system compared to the experimental board.

The first challenge was to get a proper control of the bridge legs. Because only one module was used in the development kit the complexity was very low. Now with two modules the task was a bit harder. To achieve the desired result a more complex scheme was required than what had previously been used. The documentation had to be scoured to finally find a functionality called external override that allowed for the status of the timer to be overridden, this status decides the output from the module. With this functionality no software had to be included but instead the modules are configured to talk to each other when the current exceeds the limit.

When the control scheme was implemented a problem arose that has not yet been solved. The problem occurs when the ADC is sampled and the value is studied in the debugger for the program it differs from when a physical measurement is taken on the board. However, the difference seem to be static with a factor of 1.25 between the two so the reason behind this just has to be found.

8.3 Improvements

Regarding improvements of the project, they can be split in two categories. One of which is future improvements that includes more advanced functionality and integration with existing system. This is described in the next section. The other improvements are things that regard mistakes and those that were not implemented due to lack of time.

The mistakes made in the final stage of the project were isolated to the construction of the adapter card. They are not very serious and can easily be fixed in the next iteration.

Firstly, the on/off signal coming in from the black box should be pulled high instead of low due to the construction in the box itself. Secondly, the 3,3V supply for the adapter card could come from the control card itself instead of a separate IC circuit. Thirdly, the specification for the control signals to the switches was mixed up so the A and B signal should be swapped. And lastly, the on/off signal should be connected to another port that supports pin interrupt. As it is now, the value of on/off have to be polled each control iteration.

The features that were not implemented were proper slope compensation and adaptive deadtime control. The charger still works without these but they would have made the control even better if implemented.

9 Future work

This project has provided Micropower with a strong foundation to start from when it will finally be implemented. The goal of the company is fully implement digital control of their power electronics but a first stage will probably be to add a MCU to the present setup and use it for communication with supervising systems. From this point the digital control can be added later.

In addition to the inputs and outputs that were required for the Pingvin implementation, listed in Table 6.1, some more are wanted for future improvements. Listed below are requirements for the MCU if more functionality is to be added to the charger module.

- 4 additional AD inputs for measurement of
  - RMS current
  - Another temperature
  - Two levels of supply voltage
- 2 digital inputs for a tacometer
- 2 PWM outputs for synchronous rectification
- CAN communication, 2 extra ports required (TX/RX)

These extra pins required for the functionality listed above are available for the microcontroller. The synchronous rectifier can be control from one of the two remaining HRPWM modules, the AD measurements can be taken from the second ADC module which has a lot of channels available and digital inputs are plentiful for the tacometer.

In Figure 9.1 below is an account on how the digital power software is to be implemented. It also shows which connections there are to the outside world. All of the features listed in the figure is now implemented today but will be a reference for future implementation.

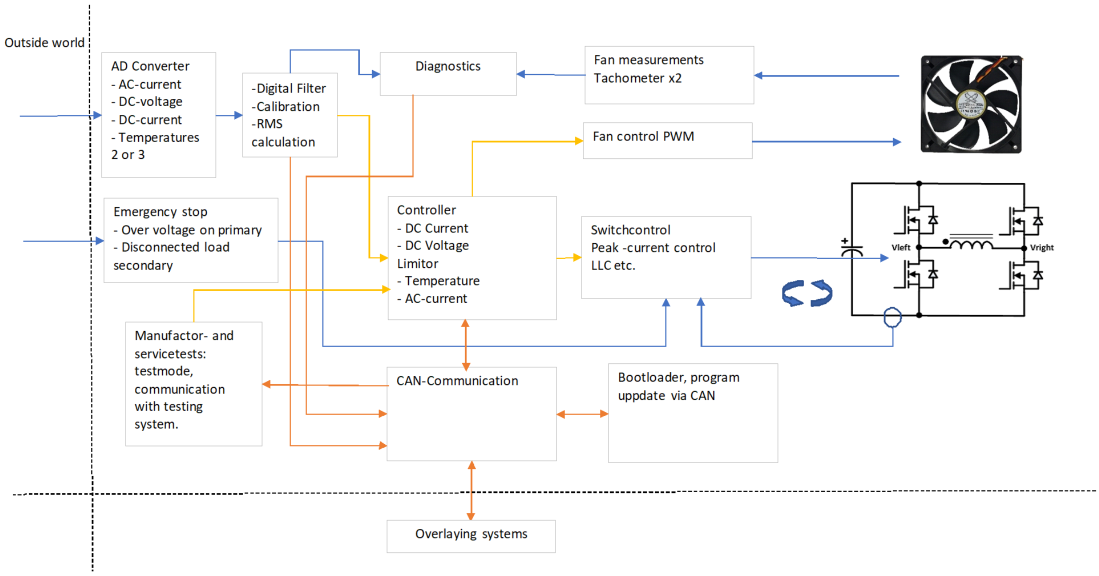

Figure 9.1: Sketch of digital power application and how it is connected

The ambition in the future is to create a controller that can take specified references for output voltage, output current and temperature and control so that output does not exceed what was specified in the reference. Today, supervising control is calculated by overlaying systems but the ambition is that with digital power, all control can be done on board and only reference needs to be sent from supervising system. This communication will then be implemented with a CAN bus.

Another ambition is that a power electronic module will be built as specified from the sketch above with one MCU and set of switches. Multiple modules can then be combined to provide higher output power. Modules can also cooperate to deliver higher efficiency when low power is needed.

A References

[1] H. Nene, "Apllication report: PSFB Control Using C2000 Microcontrollers." http://www.ti.com/lit/an/sprabr1/sprabr1.pdf, 2013. [Online; accessed 15-Febuary-2017].

[2] *XMC4000/1000 - Introduction to Digital Power Conversion*, Jan. 2015. https://www.infineon.com/dgdl/Infineon-XMC4000_XMC1000_+Introduction_to_Digital_Power_Conversion-AN-v01_00-EN.pdf?fileId=5546d4624b0b249c014b497c0c33089e.

[3] S. Franco, "Slope Compensation in PCMC DC-DC converters." http://www.edn.com/electronics-blogs/analog-bytes/4439029/Slope-Compensation-in-PCMC-DC-DC-Converters, 2015. [Online; accessed 4-July-2017].

[4] "Pulse width modulation." https://en.wikipedia.org/wiki/Pulse-width_modulation, 2017. [Online; accessed 4-July-2017].

[5] "Buck converter image." https://www.circuitlab.com/circuit/9am85v/buck-converter/, 2017. [Online; accessed 11-October-2017].

[6] "Synchronous buck converter with xmc digital power explorer kit." https://www.infineon.com/dgdl/Infineon-AP32319_Synchronous_Buck_converter_with_XMC_Digital_Power_Explorer_Kit-AN-v01_00-EN.pdf?fileId=5546d462557e6e8901559b9168515eec, 2017. [Online; accessed 8-August-2017].

[7] "Micropowers website." http://micropower-group.com/about/about/, 2017. [Online; accessed 16-Febuary-2017].

[8] Maksimovic, Zane, and Erickson, "Impact of digital control in power electronics," in *2004 Proceedings of the 16th International Symposium on Power Semiconductor Devices and ICs*, pp. 13–22, May 2004.

[9] *TMS320F28055 | Piccolo Microcontroller*, Jan. 26 2017. http://www.ti.com/product/tms320F28055.

[10] "Can bus." https://en.wikipedia.org/wiki/CAN_bus, 2017. [Online; accessed 4-July-2017].

[11] "C2000™ Piccolo™ 32-bit microcontrollers." http://www.ti.com/lsds/ti/microcontrollers_16-bit_32-bit/c2000_performance/real-time_control/f2802x_f2803x_f2806x/overview.page", year = 2017, note = "[Online; accessed 16-Febuary-2017].

[12] "Dave (version 4) - development platform for xmc microcontrollers." https://www.infineon.com/cms/en/product/microcontroller/32-bit-industrial-microcontroller-based-on-arm-registered-cortex-regi
dave-version-4-free-development-platform-for-code-generation/

channel.html?channel=db3a30433580b37101359f8ee6963814, 2017. [Online; accessed 8-August-2017].