EMG Signal treatment for Control of robotic prosthesis

Master thesis project submitted for the Master on Nanotechnologies for Integrated Systems on Grenoble INP, School of PHLEMA

Author: **Emmanouil Mandrakis**
Supervisor: **Dr. Davide Bucci**
Industry Supervisor: **Dr. Marc Pons Solé**

Grenoble 2018

Acknowledgements

I would first like to thank my thesis advisor, Professor Davide Bucci of PHELMA at Grenoble INP. Professor Bucci was always available whenever I ran into a trouble spot or had a question about my research or writing. He allowed this paper to be my own work, but steered me in the right direction whenever it was needed.

I would also like to thank my thesis supervisor, Dr. Marc Pons Sole, Senior Engineer at CSEM. He was always available to help me with any difficulties I faced, provided me advice and any equipment I needed to finish my work. Without his passionate participation and input, my master thesis project would not have been successfully conducted.

I would also like to thank the experts who were involved in the project and those who provided my solutions whenever I needed their expertise. Dr. Jean-Luc Nagel, was always available whenever I needed his advice, even though he was working on other projects.

I would also like to thank Dr. Stephane Emery for giving me the opportunity to work on his group of excellent engineers but most importantly for pushing my limits to become a better engineer.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Author

Emmanouil Mandrakis

ABSTRACT

The DeTOP (Dexterous Trans-radial Osseointegrated Prosthesis with neural control and sensory feedback) project is focused on the recovery of patients that have suffered amputation, providing a solution employing next generation technology on robotic arms, controlled naturally from the user. CSEM, one of the partners working for the DeTOP project, is developing the implant for the sensing of the EMG signals from the patient to control the robotic arm and also receive the sensing of the robotic arm, to give the user the sensation of touch. As the processing of the data is not taking place on the implant, the implant is a simple MPC node, receiving compressing and transmitting the EMG signals. For this master thesis project, characterization of the compression software that was available on the project and the development of an alternative solution is taking place and it is also demonstrated how the alternative solution is more efficient in means of power, memory and compression ratio.

Key-words:

EMG signals, Embedded Software, Power, Memory, Timing, Optimization, Digital Processing, Digital Compression, Huffman encoding, Prosthesis application, implantable, neural control, robotic arm, MPC node, communication, PCB, characterization

ABSTRACT (French)

Le projet DeTOP (prothèse ostero-intégrée trans-radiale avec contrôle neuronal et rétroaction sensorielle) se concentre sur la récupération des patients ayant subi une amputation, fournissant une solution utilisant la technologie de nouvelle génération sur des bras robotisés, contrôlée naturellement par l'utilisateur. Le CSEM, l'un des partenaires du projet DeTOP, développe l'implant pour la détection des signaux EMG du patient afin de contrôler le bras robotisé et de recevoir également la détection du bras robotique pour donner à l'utilisateur une sensation de toucher. Comme le traitement des données n'a pas lieu sur l'implant, l'implant est un simple nœud MPC, recevant la compression et la transmission des signaux EMG. Pour ce projet de thèse, la caractérisation du logiciel de compression disponible sur le projet et le développement d'une solution alternative est en cours et il est également démontré que la solution alternative est plus efficace en termes de puissance, de mémoire et de taux de compression.

Key-words:

EMG signals, Embedded Software, Power, Memory, Timing, Optimization, Digital Processing, Digital Compression, Huffman encoding, Prosthesis application, implantable, neural control, robotic arm, MPC node, communication, PCB, characterization

ABSTRACT (Italian)

Il progetto DeTOP (Destrezio trans-radiale osteointegrato con controllo neurale e feedback sensoriale) è incentrato sul recupero di pazienti che hanno sofferto di amputazione, fornendo una soluzione che impiega la tecnologia di nuova generazione sui bracci robotici, controllata naturalmente dall'utente. CSEM, uno dei partner che lavorano per il progetto DeTOP, sta sviluppando l'impianto per il rilevamento dei segnali EMG dal paziente per controllare il braccio robotico e anche ricevere il rilevamento del braccio robotico, per dare all'utente la sensazione del tatto. Poiché l'elaborazione dei dati non avviene sull'impianto, l'impianto è un semplice nodo MPC, che riceve la compressione e la trasmissione dei segnali EMG. Per questo progetto di tesi di laurea, la caratterizzazione del software di compressione che era disponibile sul progetto e lo sviluppo di una soluzione alternativa è in atto e viene anche dimostrato come la soluzione alternativa sia più efficiente in termini di potenza, memoria e rapporto di compressione.

Key-words:

EMG signals, Embedded Software, Power, Memory, Timing, Optimization, Digital Processing, Digital Compression, Huffman encoding, Prosthesis application, implantable, neural control, robotic arm, MPC node, communication, PCB, characterization

LIST OF ABBREVIATIONS

MPC node	Micro Processing and Communication node
DeTOP	Dexterous Trans-radial Osseointegrated Prosthesis with neural control and sensory feedback
MuAP	Muscle Action Potential
EMG	Electromyography
EEG	Electroencephalography
ECG	Electrocardiogram
ExG	Any of the EMG, ECG and EEG.
ADC	Analog to Digital Converter
SoC	System on a Chip
UART	Universal Asynchronous Receive Transmit
Fft	Fast Fourier Transform
PCB	Printed Circuit Board
GPIO	General Purpose Input Output
FIFO	First in First out, ordered stack of data.
RTL	Register Transfer Level

Table of Contents

Table of Figures

1. Introduction

1.1. DeTOP

DeTOP (Dexterous Trans-radial Osseointegrated Prosthesis with neural control and sensory feedback) is a European research project addressing the problem of recovery of hand function on patients after amputation. The goal is to develop a robotic arm with next generation technology for tansradial prosthesis, concerning the whole user experience. More specifically, the implanted electrodes will allow a bidirectional communication between the user and the prosthesis, giving the user the ability to recover both the control of the limp and also the sensing of touch from the robotic arm as it is illustrated on Figure 1. The electrodes will be implanted, in order to achieve more accurate recordings, not affected by the skin – electrode capacitance and also by the noise introduced by the EMG signals of neighboring muscle groups. [1]

The project is a partnership between ten partners. Scuola Superiore Sant'Anna, Lund University, Integrum AB, University of Gothenburg, University of Prensilia, Essex University and CSEM, where each of the partners specialize in a key aspect of the project.

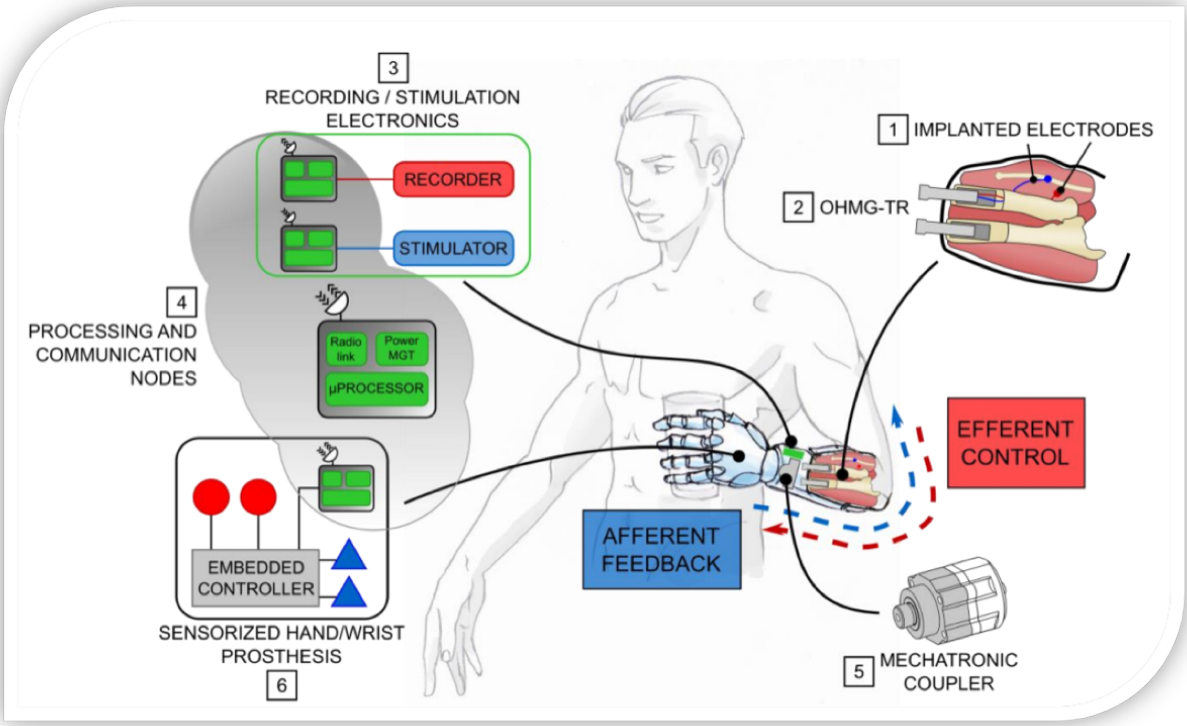

Figure 1 – Summary of DeTOP project.
www.detop-project.eu

The robotic hand will have a main control unit, which will process the signals received from the touch sensors at the edges of the fingers, in order to provide a sensory feedback to the user, giving him the ability to have the sense of touch. The main control unit will also receive EMG signals from the implanted MPC node, which will be classified, in order to be translated into the voluntary movements of the user for the robotic hand. The implanted MPC node, will be connected to implanted electrodes sensing the EMG signals generated from the remaining muscle groups. Each patient will be studied by a doctor to define the number of electrodes to be employed and also the positioning. For this reason, the algorithms employed both on the implanted controller and the main control unit, need to be trained. The communication between the implant and the main control unit of the robotic arm will be wireless, employing a low energy – high throughput wireless protocol developed by CSEM. The specific wireless protocol will be low energy when the user is idle and provide high throughput when movement is detected. The implanted MPC node will be powered by batteries and in order to minimize the need for maintenance, which will raise the need for surgery, making the

implant a low power device is a high priority, without sacrificing the quality of the process. The size of the transmitted information of course, is driving the power consumption.

1.2. CSEM

CSEM is a private, non-profit Swiss research and technology organization, focusing on generating value for a sustainable world.[1] CSEM operations aim on bringing new technologies on the industry of electronics. The company's industrial context is on the internet of things, energy and transportation, healthcare and industrial solutions among others.

The contribution of CSEM to the DeTOP project is the development of the MPC (Micro Processing and Communication) node. The MPC node, will have as an input the EMG signals from the implanted electrodes, on the analog part and, treat them on the analog front end, compress them with a lossless compression algorithm and transmit them wirelessly to the robotic arm. One big challenge, since it is an implanted device, is the energy consumption of the system. Even though a low power wireless protocol, developed in CSEM, is being used, minimizing the power by minimizing the size of the transmitted information is vital. The proposed structure of the MPC node that will be implanted is illustrated on Figure 2. An external memory, with the ability to store both data and instructions, an external clock for the calibration of the internal clock and a battery, are needed for the operation of the MPC node. The analog part has 20 differential channels. The digital part is an Icyflex2 core, a processor designed and developed in CSEM. IO (Input Output) pads, digital and analog, are available, for the communication of the device and also for the sensing. The IcyTRX is the low energy wireless protocol.

The operation of the device is studied for a wide temperature range, from 0-120°C, but for the specific application, since the device will be implanted, it will operate in the range of the human body temperature, which is 36.6 - 37°C. I proposed that the chip designed for the DeTOP project, can also be used for other applications. Wearable technologies, Internet of Things and other ExG applications make excellent candidate projects to employ this chip. The

[1] https://www.csem.ch/home

fact that it provides 20 differential analog channels, with a digital core having a processor running at 153Mhz, on low power, makes it an ideal SoC to be involved in many applications where sensors, processing and wireless are employed.

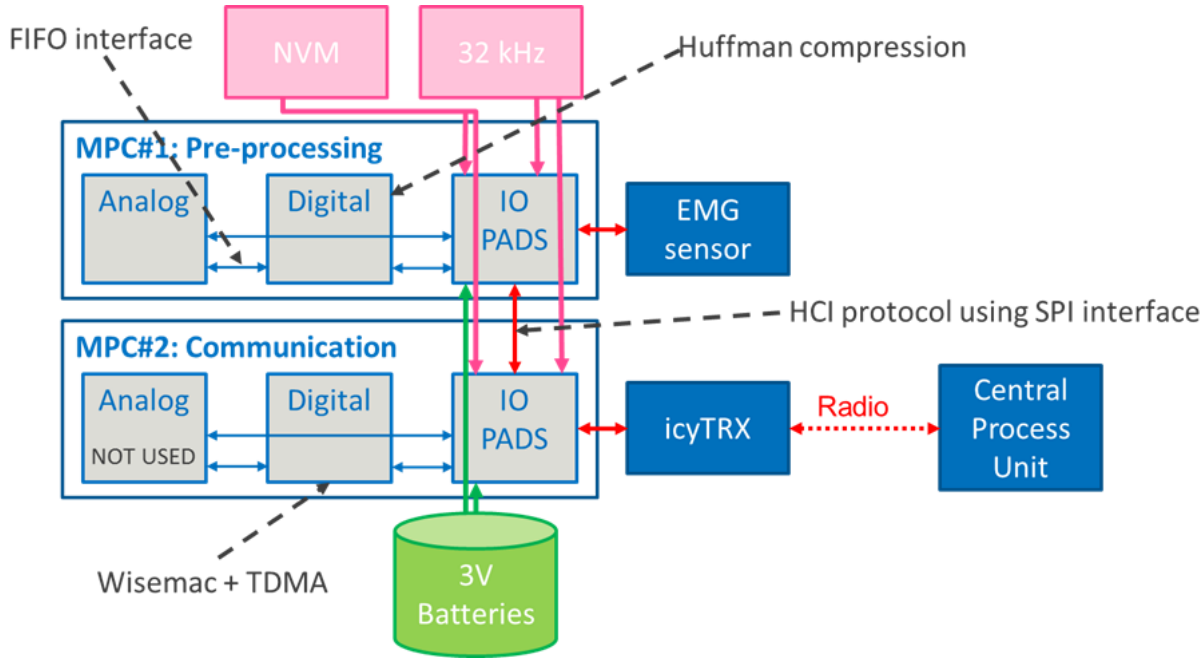

Figure 2 - Structure of the MPC node for the DeTOP project

1.3. State of the art

There have been many research projects on artificial limbs and neural interfaces but amputees continue to use technology developed over forty years ago. All products currently available on the market, use electrodes attached to the skin, and the surface electromyogram signals for the control. On this way, the error rate (how many times the action of the robotic limb does not follow the will of the user) and the response time are rising. On the DeTOP project, implanted electrodes are employed providing higher accuracy. Also, there is not a solution with natural response time (below 100ms) and error rate bellow 10%. Most of current solutions employ machine learning for the classification of the recorded EMG signals [2] [3] [4] [5], others are using Finite State machines [6]. Also, most of the state-of-the-art application are using two to six electrodes while the DeTOP chip can employ up to twenty electrodes. All of those challenges are the problems the DeTOP project aims to solve.

1.4. Gantt Diagram

A Gantt diagram is developed to illustrate the distribution of my work over time. The diagram is illustrated on Figure 3. At the beginning of my internship I studied the nature of EMG signals having the goal to emulate their behavior, in order to be used on a demonstrator. The solution I proposed is that the EMG signal generator can be a computer playing a sound file, where the EMG signal is transformed into a sound file. For this reason, I developed a python script which is normalizing EMG recordings and generating a sound file from that recordings. The sound file, when played on a speaker, pulses of voltage are vibrating a membrane attached on a magnet. My suggestion is that if the sound wires are fed on the analog inputs of the chip, the EMG signal generator will be emulated. Furthermore, understanding the particularities of the EMG signals, will help develop better software for the compression. The next step for which I devoted the period of 2 months, I studied how to compress information, focusing on the particularities of the EMG signals. The Huffman algorithm was found to be the most efficient for lossless compression according to bibliography. [7] On parallel I started evaluating the chip which we received at the beginning of March. I started with a training on how to write embedded software, compile it and load it through the UART, for this chip. Then I continued by testing the peripherals and the memory for errors before starting trying compression algorithms. After understanding the hardware of the chip, I made proposals for other applications where it could be used. Then, after a first version of the compression software was available, I started the evaluation of the chip, while executing this software. The results were disappointing, so I decided to write my own software instead of improving it which proved to be very time consuming. In the future my software will be used on a demonstrator after the Analog part of the chip will be characterized.

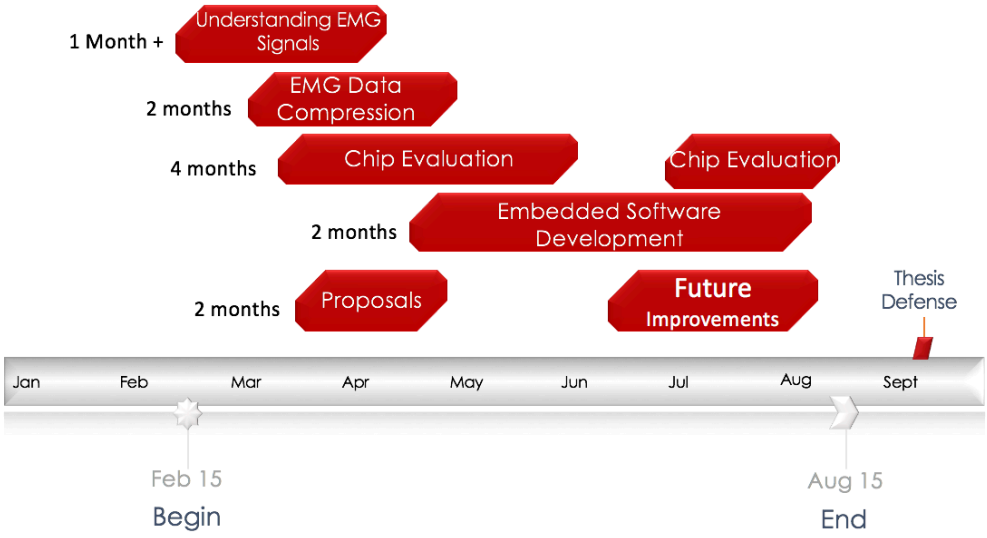

Figure 3 Gantt Diagram

2. Theoretical Background

2.1. EMG Signals

Electromyography is the study of muscle function based on the examination and analysis of the electrical signals that emanate from the muscles. The Electromyogram is the electrical manifestation of the contracting muscle, which can either be a voluntary or involuntary muscle contraction. The EMG signals are affected by many factors, such as the anatomical and physiological properties of the muscles and also the instrumentation and the process used to record EMG signals. Noise can be introduced both by the environment, the neighboring ExG signals and also the instrumentation used for the detection. The basic functional unit of the muscle contraction is a motor unit which is comprised of a single alpha motor neuron and all the fibers it communicates with. These muscle fibers contract when the action potential of the motor nerve reaches a depolarization threshold. The depolarization generates an electromagnetic field which is measured as a very small voltage, called EMG. [8]

The EMG signal is the algebraic summation of individual MuAPs (Muscle Action Potential). The motor unit action potential is the spatial and temporal summation of the individual muscle action potentials for all the fibers of a single motor unit. Hence the EMG signal is a composite of the two mechanisms used to increase muscle force, recruitment of additional motor units and a more rapid firing of the same motor units. [8] EMG signals are at the order of 10μV – 5mV. Sampling frequencies above half the sampling rate will be reconstructed as frequencies bellow half the sample rate. For the emulation, in order to avoid the complexities of detailed descriptions of the generation of individual muscle fiber action potentials, at the highest level, the EMG signal can be represented as a band limited Gaussian noise, where each individual MuAP is represented by a random Dirac delta function. [9] An example of an EMG signal is illustrated on Figure 4. The specific EMG signal is taken from the arm using surface electrodes, while the thumb and the index was contracting. The values are normalized to 1, and plotted with a python script I created, that is also generating a sound file. The sound file can be the analog input to the DeTOP chip for a demonstration.

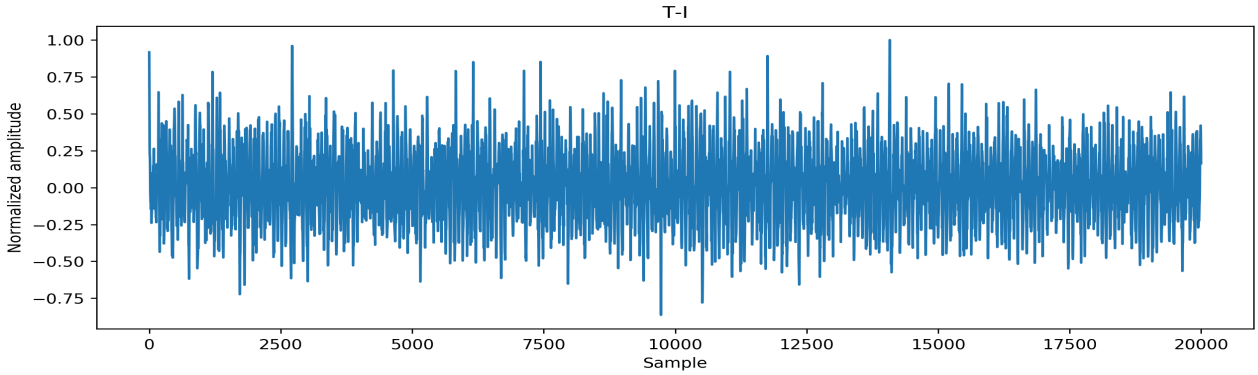

Figure 4: EMG signal of the arm muscles while the Thumb is contracting to touch the index. On the x axis one can observe the number of samples while on the y axis, it's the amplitude of the EMG signal. The amplitude is normalized to the value of 1, on a symmetric way, leaving the zero potential to be on the zero value and not from peak to peak. The data used are EMG recordings, processed and plotted by a python script.

2.2. Lossless Compression

The processing of the EMG signals in order to control the robotic arm will take place on a controller on the robotic arm. At the level of the MPC node that will be implanted, there is no information as to which of the signals are useful, hence the treatment of the data before the transmission needs to be lossless. In order to reduce the power consumption, the compression of the data before the transmission is vital. For this reason, a lossless compression algorithm is employed. The bibliography of Information Theory, present the Huffman compression algorithm, to be the most efficient compression algorithm (Highest compression ratio for a lossless compression algorithm). [7] Since the Huffman compression is statistical, this can be not true in some cases and even have information inflation instead of information compression. The dictionary employed for the compression can help avoid it.

In more detail, the Huffman compression, is creating an encoding of the characters of message, where the characters that appear more often have a code with a smaller memory footprint. In other words, each character has a variable length of bits that it is being represented and there is no other character whose representation begins with the representation of the other character. This avoids introducing a space between each character, since the length each time is unknown. The first step of the Huffman compression

algorithm is to measure the appearance frequency of each character of the alphabet. A binary tree is being created following the Huffman algorithm, which is also the dictionary for the decoding. Given an encoded character, following the bits that represent it to cross the tree, will lead to the decoded character. The sum of the length of each decoded character multiplied by the frequency of appearance of this character on a message, gives the entropy of the message. The entropy of the message is the biggest compression that can theoretically be achieved. The Huffman algorithm has the shortest expected length of the encoded message. The dictionary can be generated each time, where the message will begin with the dictionary, or it can be developed externally, given a big database of messages transmitted. The dictionary can also be constant, published both on the encoding and the decoding chip. The choice depends on the size of the message, where the dictionary can have a comparable size to that, making the compression pointless.

3. CSEM MPC

The first version of the MPC node for the DeTOP project is illustrated on Figure 5. The digital part is on the right and the analog part is on the left. The transistors of the analog part, can be seem to be much wider as it is expected, where the first stage, the Low Noise Amplifier, has the wider transistors since it focuses on low noise. The photo of the chip is taken before packaging, and the wire bonding of the chip is visible, which is connected to the pads of the packaging.

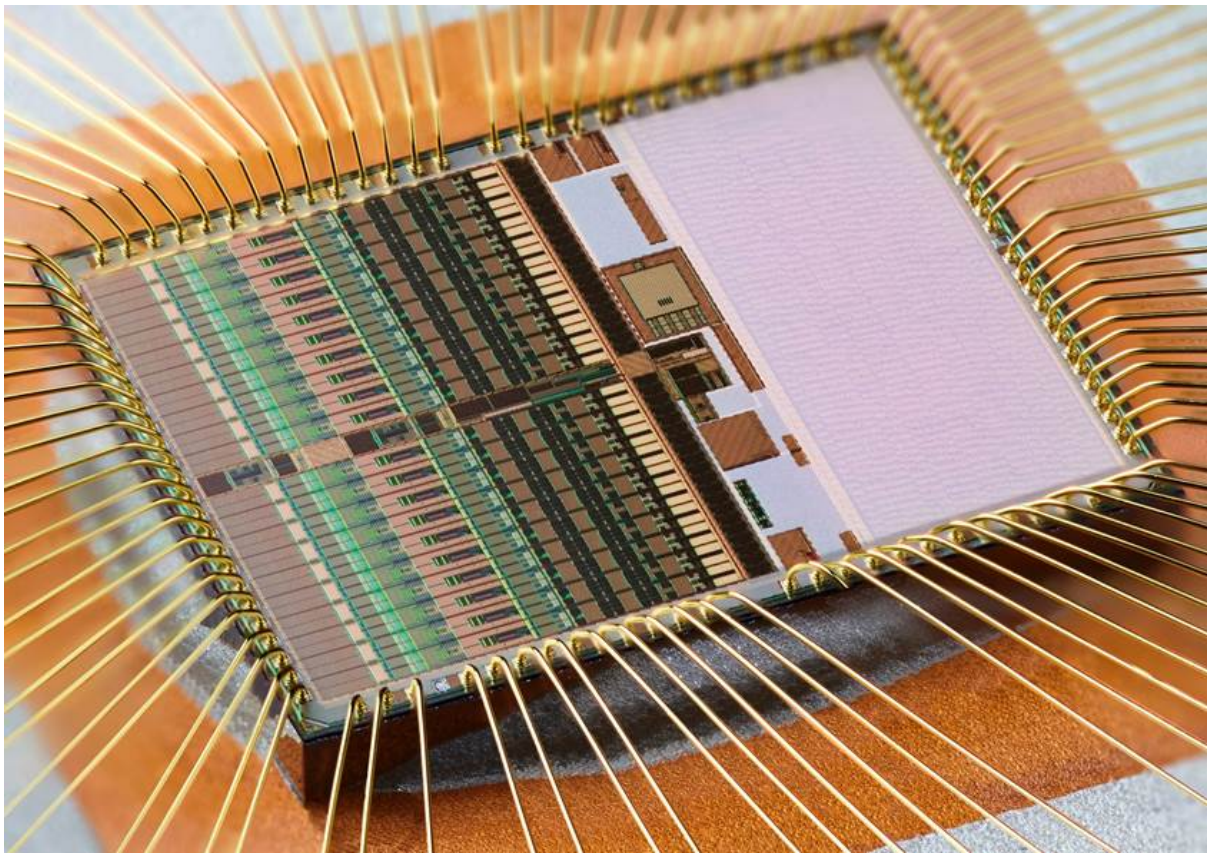

Figure 5 : Photo of the DeTOP chip, taken before packaging. On this photo the Digital part is on the right and the Analog part is one the left, where one can see that the analog transistors are much wider. 55nm transistor technology is employed, on the TSMC fab. The process of the chip used is TT.

3.1. Analog Part

The analog part of the MPC node, has 20 analog inputs. Those analog inputs are being treated by the analog front end, in order to enhance the useful information and reject the noise,

before the digital treatment. Each channel is differential (40 channels in total) to filter out spatial noise.

The first stage is a low noise amplifier, which is also a filter that cuts frequencies bellow 0.5 Hz. The amplification of the signal is 28dB. It is necessary because the EMG signal voltages can be too low.

The second stage is a programmable high pass filter. Its behavior is being defined by registers on the digital part of the chip. Their values be changed with the embedded software on the Icyflex2 core and provide an amplification gain from 6 to 18 dB. This amplifier is optimized for low power since the noise specification is met on the low noise amplifier.

The next stage is a low pass filter cutting frequencies below 8 KHz. The cut off frequency of the low pass filter can be also defined on the software and tune it from 125 Hz to 8kHz.
The next stage is the ADC (Analog to Digital Converter). Since the ADC is a single channel (differential), a multiplexer is needed to change the channel that is being sampled each time. The ADC sampling frequency is limiting the sampling frequency to be divided each time by the number of channels employed.

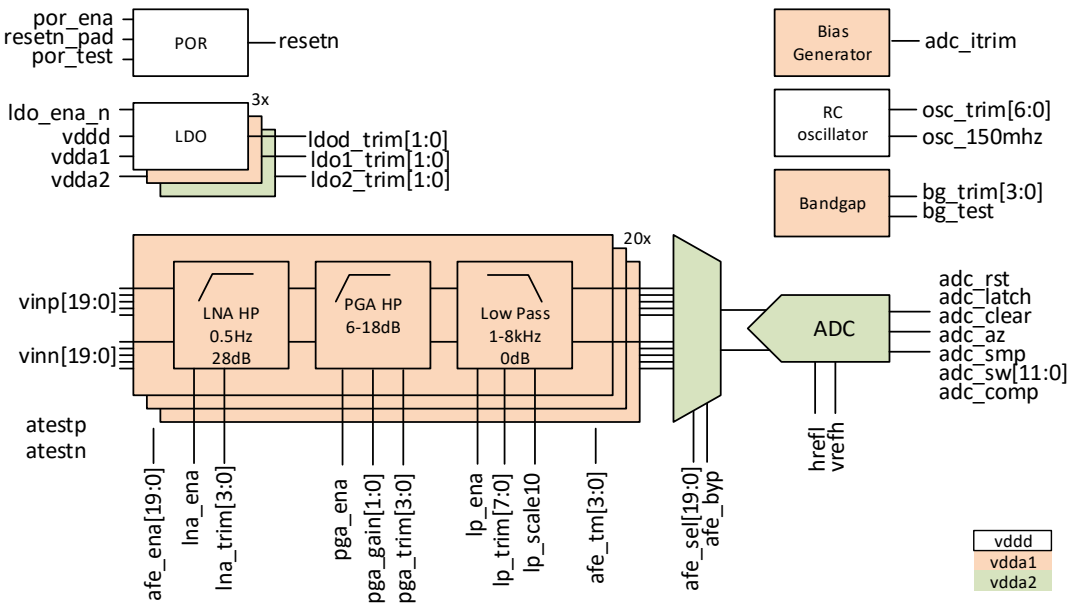

Figure 6: DeTOP Analog top-level block

3.2. Digital Part

The digital block of the DeTOP chip is an ICYflex2 processor with a memory of 256 kb. The clock can be calibrated to have a frequency up to 153 MHz and remain stable, with a low power consumption, as it is reported on the verification of the chip. A big percentage of the power consumption, around 46.52 %, when running a testing code, was coming from the memory part. This means that the power consumption can be improved, by using embedded software with limited access to the memory. The power breakdown can be seen on Figure 7, referring to the verification simulation running a testing software doing an fft on the Icyflex core.

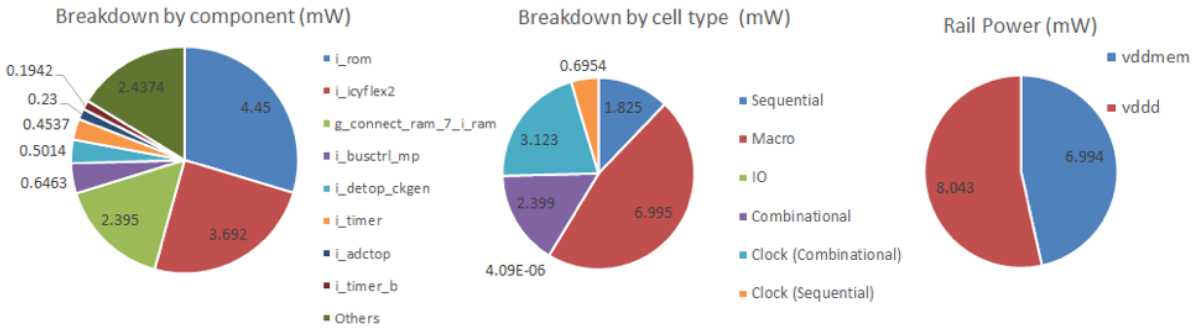

Figure 7: Power breakdown for fft code versus the components(left), the cell type (middle) and the rail (right)

The verification is taking into consideration only the Icyflex core and not the peripherals, hence the measurements on the lab will have higher values.

3.3. Testing equipment

The chip was tested on a testing PCB (Printed Circuit Board). Its top view is illustrated on Figure 8. The test board is providing access to all of the chip's pins, for the control of the peripherals, the IO (Input Output) and the power supplies, in order to test the board. The testing equipment needed in the lab for testing the chip, was a function generator, the model Aglilent 33250A, which was used to generate a clock signal, constant at 32kHz constant for the calibration of the internal clock. There is also the possibility to provide the clock

externally, in which case, a calibrating signal is not needed. For the power supply, 5V DC, from a Triple Power Supply HM8040-3 is being used. The PCB has voltage regulators to provide the voltage supplies needed for the operation of the chip. More specifically, a Vcc voltage, at 3.3 Volts is powering the Operation Amplifiers and the main supply of the chip. The Vddd voltage, is the power supply for the digital part, at 1.2 Volts, same with the Vddmem which is the supply for the memory part. This is connected to the supply of the Vddd, but when the power consumption of the memory needs to be measured, it is provided separately.

The software of the chip for testing is loaded and tested from a debugger, on a JTAG chip. A logic analyzer is connected to the GPIO ports on the board in order to evaluate the timing, using flags in the processes of the embedded software, and passing the fag values to the registers that control the GPIOs. A photo of the equipment during the testing can be seen on Figure 9.

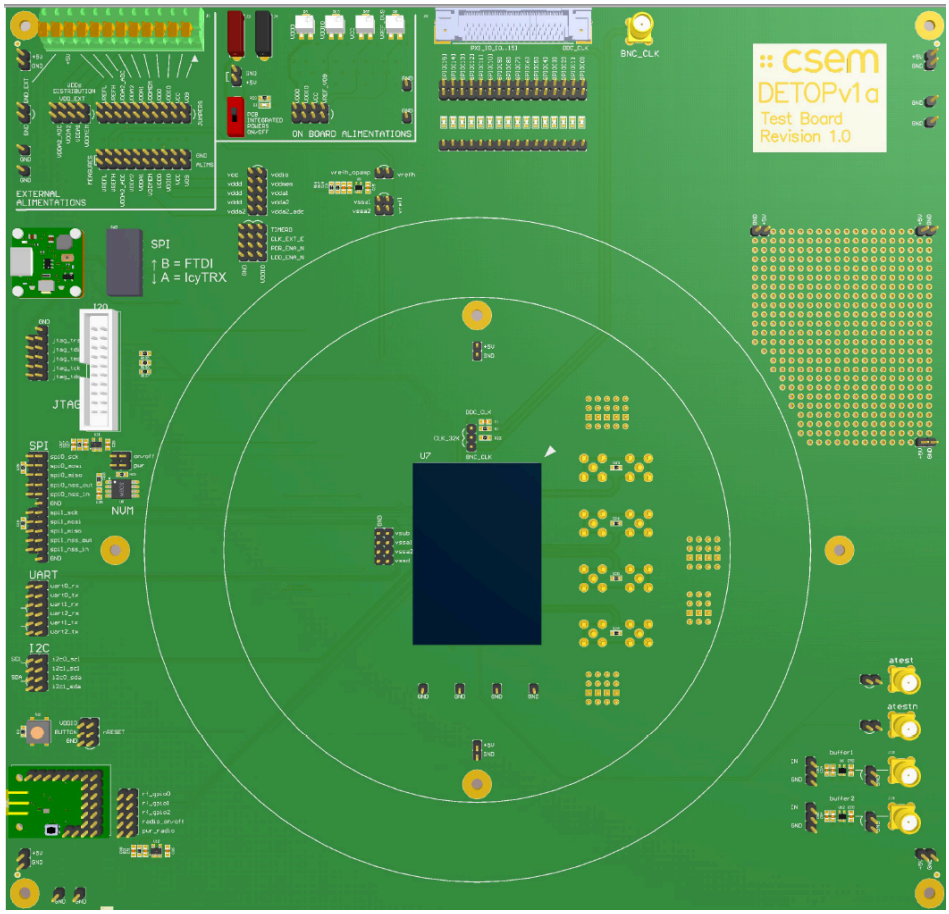

Figure 8 : Test board for the DeTOP Chip designed and assembled on CSEM. This board is providing easy access to all the peripherals and supplies of the DeTOP chip that are needed for the characterization and also has a wide area in the middle to force temperature on the chip, if it is needed to be characterized on a specific temperature regime.

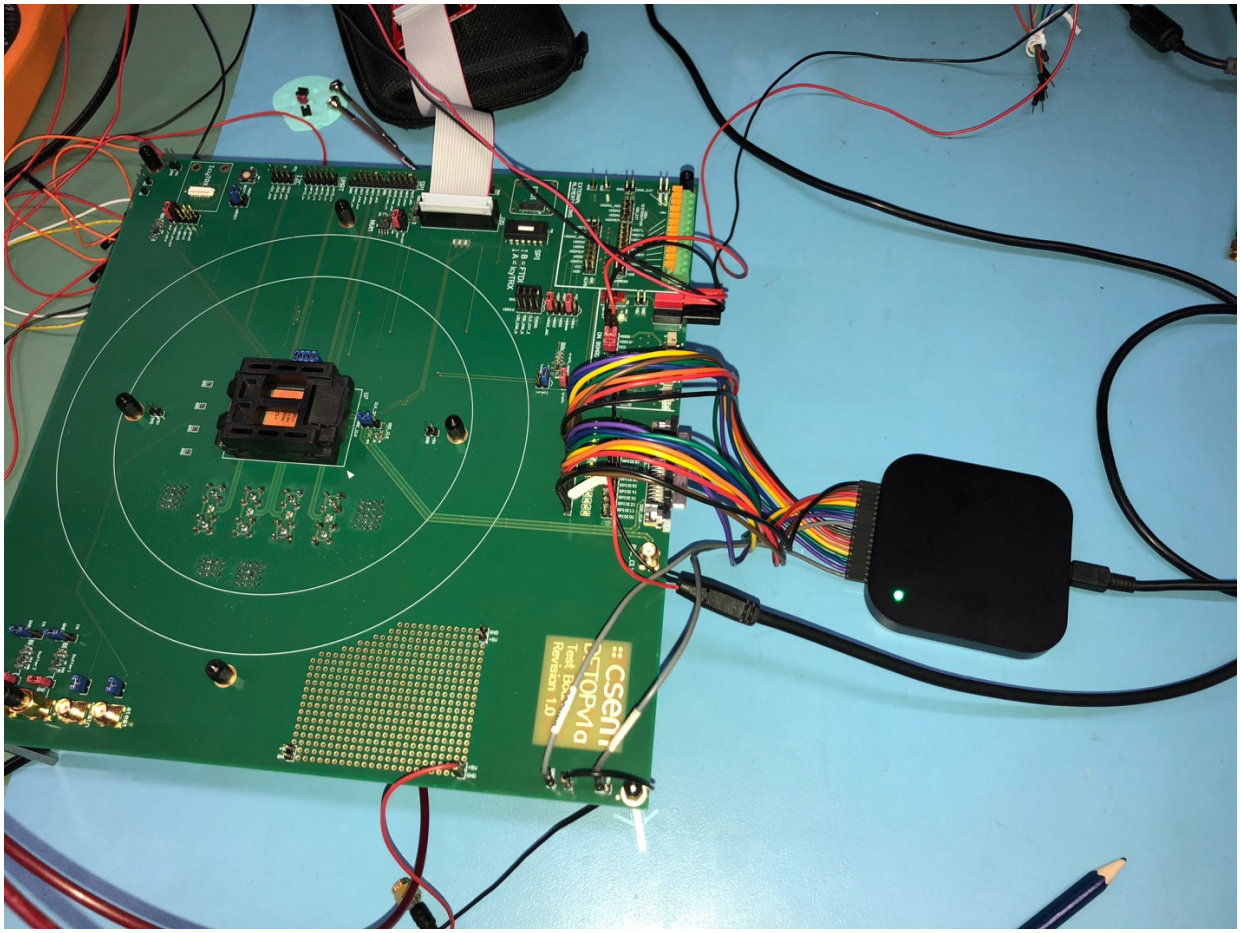

Figure 9: Testing environment. The DeTOP chip is mounded on the test board. The software is loaded using a JTAG board, and the GPIO (General Purpose Input Output) ports are connected with a logic analyzer, used to evaluate the timing and several software flags

4. Compression algorithm

For the compression of the data, the Huffman encoding is being used. That is because the Huffman algorithm is a lossless compression algorithm with the highest compression ratio. Theoretically, the compression ratio could not be bigger than the entropy of the data and the Huffman algorithm has a compression ratio slightly bigger than that. [7]

4.1. Algorithm

The Huffman encoding is providing a bit representation of each character, where the characters that appear more often are represented with shorter codes, than with their normal representation. For this reason, a statistical analysis, with the frequency of appearance of each character over the transmitted message needs to be conducted. On the current application I used a big dataset of EMG data recordings. Having all of the characters of the message and their frequency of appearance, the Huffman algorithm is providing the bit representation of each character. If the dictionary is saved on a 16-bit array though, all characters will have the same number of bits, which will not introduce any compression. For this reason, a second array is being generated having the number of bits in each character of the dictionary.

On the current application the output of the ADC is a number with a minimum value of 0, corresponding to -1.8 V and a maximum value of 4095, corresponding to 1.8V. The dataset used for the generation of the dictionary needs to be normalized accordingly before the generation of the dictionary. Samples representing a potential around 0V are the most common on the dataset, hence the values of the dictionary around 2000 need to have the smallest code. Referring to the EMG signals nature, while the muscle is inactive the potential is negative with a small value. Its value is not referred here, since it depends on the muscle group. For this reason, the most common value is between 1700 to 1900.

The embedded software is using a lookup table, where it transforms each value of the output of the ADC into the encoded value and packages it into a package with a size of 80 bytes, defined by the wireless transmission protocol. Encoding and packaging happen on parallel,

where there is a buffer filling with values of the input until it reaches the maximum size of the package, when it will be encoded. This is easily found by the bit-count array. Since the dictionary has 4096 values and the package is only 80 bytes (640 bits), the generation of the dictionary does not need to happen on the chip, hence a constant dictionary can be used. Both the encoder on the implant and the decoder on the central processing unit are referring to the same dictionary.

4.2. Evaluation of the first version

A preliminary version of the code needed to be embedded on the chip for compression, was available on the project. The first part of my work on CSEM was to evaluate this software and find ways to improve it. A lot of problems were found on the code, making it unsuitable for the DeTOP chip and also for the project's needs. The most important drawback of that code, was the dictionary it was using. The dictionary had the smallest codes on the extreme values, which appear quite rare on the EMG data and the biggest on the middle values which are the most common. The data are similar to a band limited white gaussian noise, making the values on the edges almost inexistent. Another problem was that a script to generate the dictionary was not provided, which is the core of the Huffman algorithm. More specifically, the Huffman algorithm is about the generation of the most efficient dictionary. The embedded software is a simple encoding software, using the Huffman dictionary as an encoding lookup array. On the first run of the code, the compression ratio was equal to -12.5%. An input message 128000 bits long was imputed on the chip and the output was 144000 bits long.

Another problem I faced with this software is loading it on the chip. The version we received was 625kB, when the RAM memory of the chip was 256 kB. When the first version of the software was developed, the platform for testing it was not available and a platform emulator without any memory constrains, on the software, only the specification. By removing some of the debugging routines found on the software that were not used, I resized it to fit on the platform, by making some modifications also on the memory mapping, that was used by the linker.

I also measured the power consumption of the chip, by measuring the current on the supplies, when running the software. The power consumption was excessive, comparing to the power consumption predicted by the simulation during the chip verification. With a voltage supply of 1.2 V, at room temperature, the power consumption is equal to 13.428mW. A big part of the power consumption is found to be on the memory rail, measured at 4.683mW. The timing of the software was not a problem, even at low frequencies. The reported power consumptions are at 153MHz operation of the chip.

In order to visualize the output of the software, I wrote software to employ the UART (Universal Asynchronous Receive Transmit) peripheral and a python script I was running on a laptop. In order to evaluate the timing, I was using a digital analyzer, sampling at 16MHz, from the GPIOs of the chip.

4.3. Second version algorithm and re-evaluation

In order to solve the problem of the power consumption, the first approach I used was to improve the software, by resizing the buffers of the software, and also removing or modifying functions that was less or not used at all. This was very time consuming, because the software was using many functions on different files, often serving no purpose, so I decided to write from scratch my own software, doing the simple encoding with a lookup table. When I started writing this software I had already done many tests on the platform, having understood how to write efficient code in order to improve it. In order to improve the compression ratio, I wrote a software which is not run on the platform and it will be presented on the next chapter. In order to reduce the power consumption, I focused on a solution with a limited use of memory. For this I employed less buffers. The ADC has a FIFO buffer, hence a copy of this on the software is not needed.

The total memory used by this software is 52kB, significantly smaller than the version that was initially available. This is also reflected on the power consumption, where the power consumed on the memory rail is equal to 1.430mW comparing to the 4.683mW that I measured running the other version of the software. The current on the memory rail, where the Voltage is the same, equal to 1.2 V, was measured on several frequencies of the chip from

38MHz up to 153 MHz. The upper limit is the nominal clock frequency of the chip and the bottom limit of 38MHz is the minimum clock frequency, where the compression is successful, given that the data are received with a sample rate of 120kS/s. Having the software successfully running on such a low frequency can be a great gain. If the MPC node will only run the compression algorithm, it translates to a much lower power consumption (total) and if other software will also be embedded on the chip, it means that both the memory and the processing load, has room for more software and higher complexity. On Figure 10 the current on the memory rail is plotted, in the referred frequency range, for the new and the old algorithm. It can be observed that on the old algorithm, the power consumption on the memory lane was increasing significantly with the frequency, which means that the execution has more memory accesses and less calculations comparing to the algorithm that I developed. In order to measure the current on the memory rail, I connected an external Voltage supply in series with an Ampere meter and the power supply of the memory on the board, the Vddmem port.

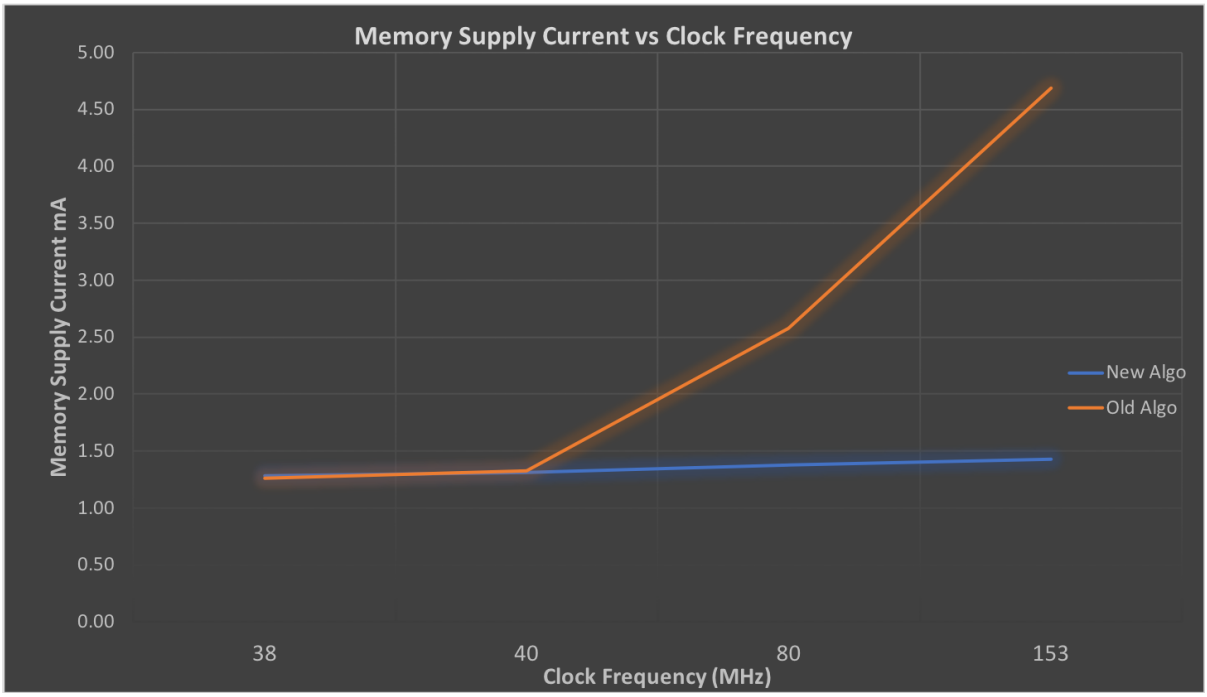

Figure 10: Current on the memory lane versus the Clock Frequency

The total current on the chip, including the memory, was also measured to compare the new software with the old one. The resulting current is plotted on Figure 11. In higher frequencies

the difference is becoming bigger on the power consumption. The leakage current is irrelevant of the software embedded on the chip. It was measured to be 0.594mA, with a voltage supply of 1.2V. The leakage current is a useful information in order to define the dynamic power supply of the chip.

The dynamic power supply is measured as µW/MHz. The calculation of the dynamic power supply is performed with Equation 1. The dynamic power consumption of the chip is plotted on Figure 12, for clock frequencies ranging from 38MHz up to 153MHz.

Equation 1: Calculation of the Dynamic Power Supply

$$\frac{(I_{vdad} - I_{leak}) \cdot V_{dad}}{f_{clk}}$$

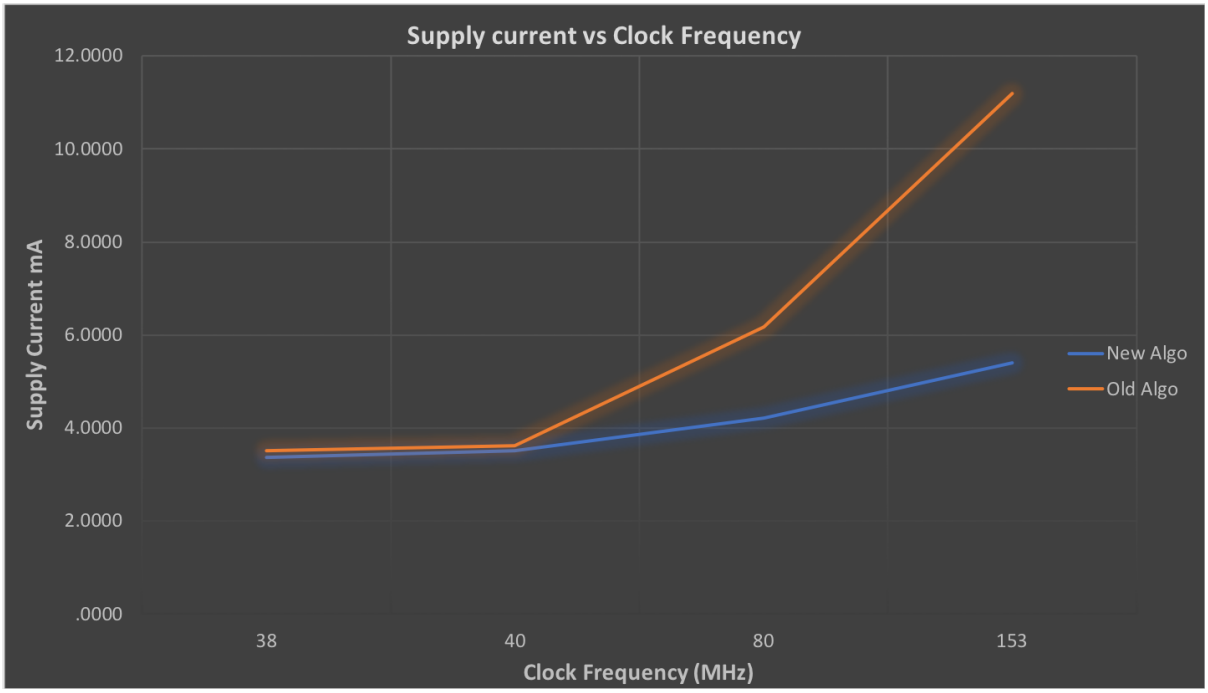

Figure 11: Current supply versus the clock frequency, running the first version of the software and the new algorithm.

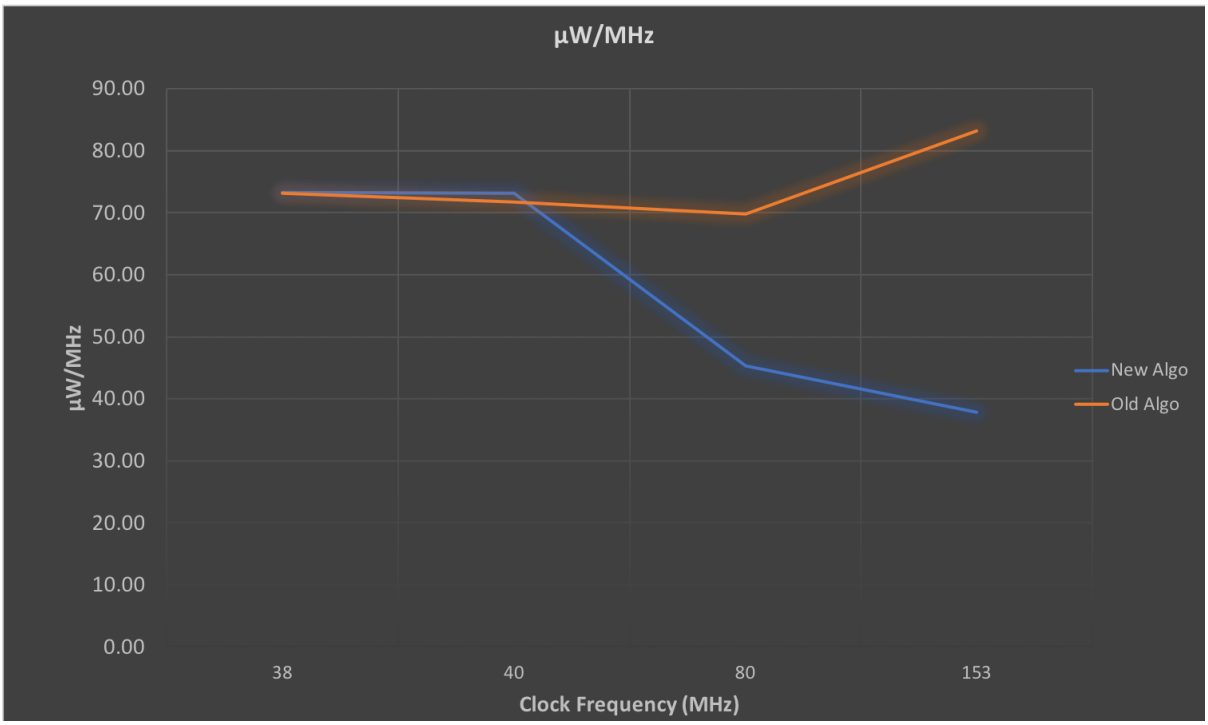

Figure 12: Dynamic Power consumption versus the Clock frequency

The choice of 38MHz as a minimum frequency was made after considering the ability of the software to compress the input faster than it is received. In order to observe if the processor is fast enough to achieve it, with various clock frequencies and find the minimal clock frequency, in order to minimize the power consumption, I introduced flags on the software, which were passed on the registers that control the GPIOs. The state of the GPIOs was observed with the use of the logical analyzer. The processor was interrupted every 3 ms and signals were compressed during this time. While the compression was happening, a flag was equal to zero and when the packages were ready to send the flag was becoming equal to 1. The width of the flag pulse equal to 1 needed to be minimized. The first channel of the logical analyzer is representing the pulses of the system interrupt and the third channel is representing the status of the calculation. The timing with the clock frequency at 153 MHz is illustrated on figure 13. The third channel has most of the time a value of 1, which means that most of the time the processor is idle. On Figure 14, Figure 15 and Figure 16, the timing constraint described above seems to be met. On Figure 17 on the other hand, with a frequency of 35MHz, the timing constrains are not met. The third channel is still zero when the next interruption arrives which means that the processor is still compressing data, and new data are arriving.

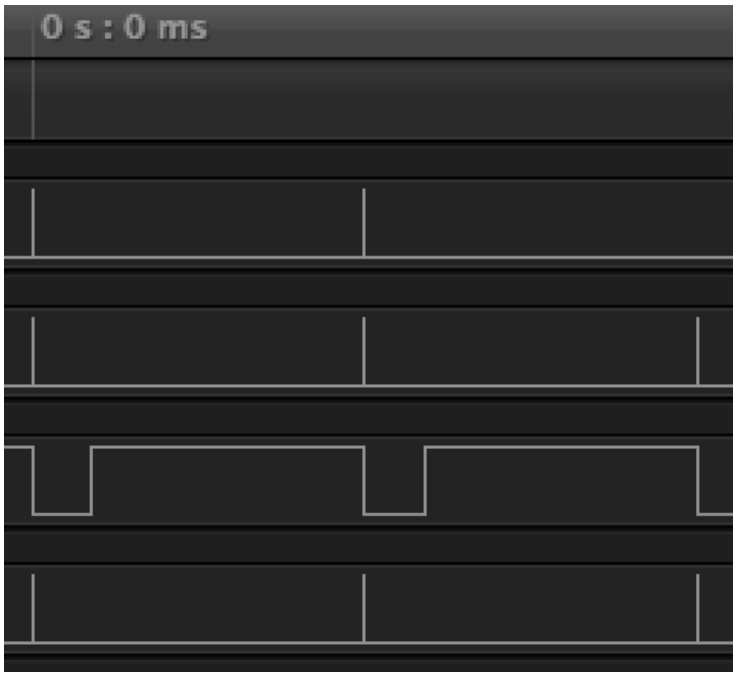

Figure 13: Timing of the software with a clock frequency of 153MHz

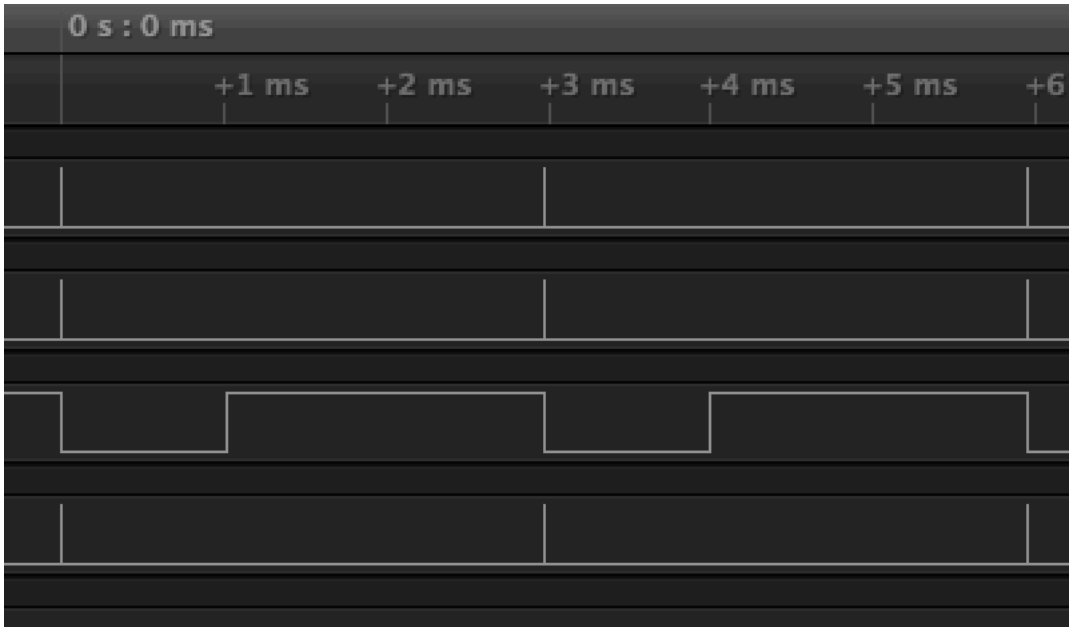

Figure 14: Timing of the software with a clock frequency of 80 MHz

Figure 15 : Timing of the software with a clock frequency of 40 MHz

Figure 16: Timing of the software with a clock frequency of 38 MHz

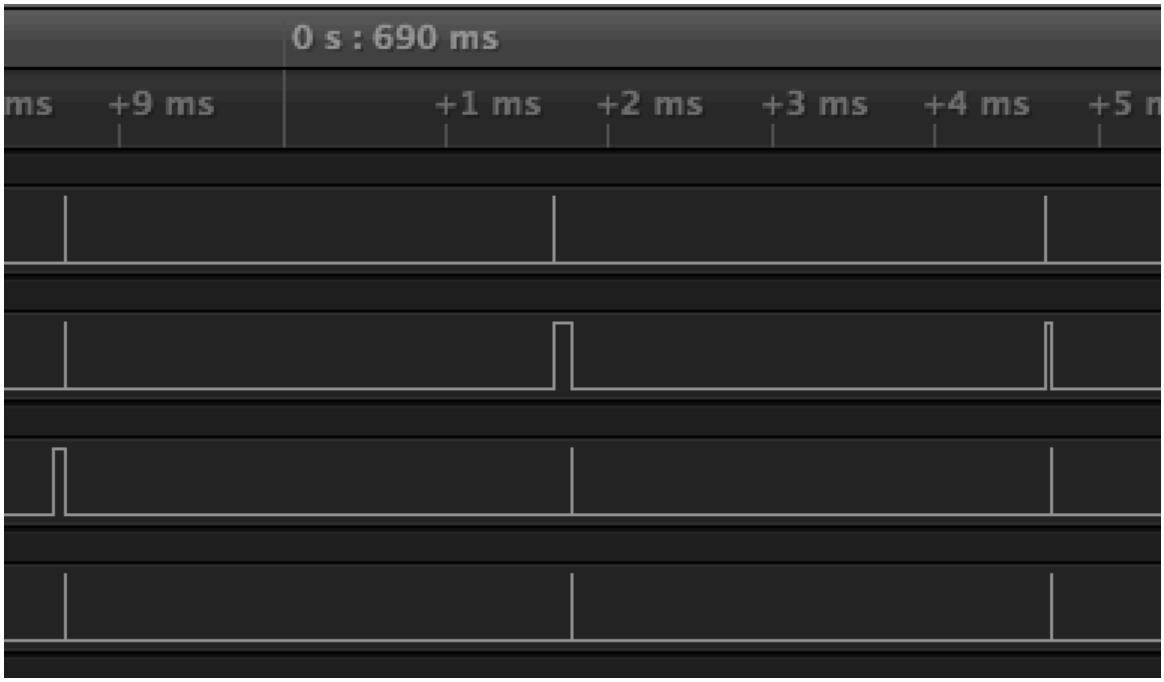

Figure 17: Timing of the software with a clock frequency of 25 MHz

The setup of the characterization employed the function generator and the DC power supply that was mentioned and also an amperemeter of Hewlett Packard, the model 34401A and a computer where the JTAG was connected, for loading the software and debugging it. For the debugging a version of the gdb debugger was used, that was developed for debugging the icyflex2 core. The instrumentation can be seen on Figure 18. All the software that was embedded on the SoC was written in C programming language.

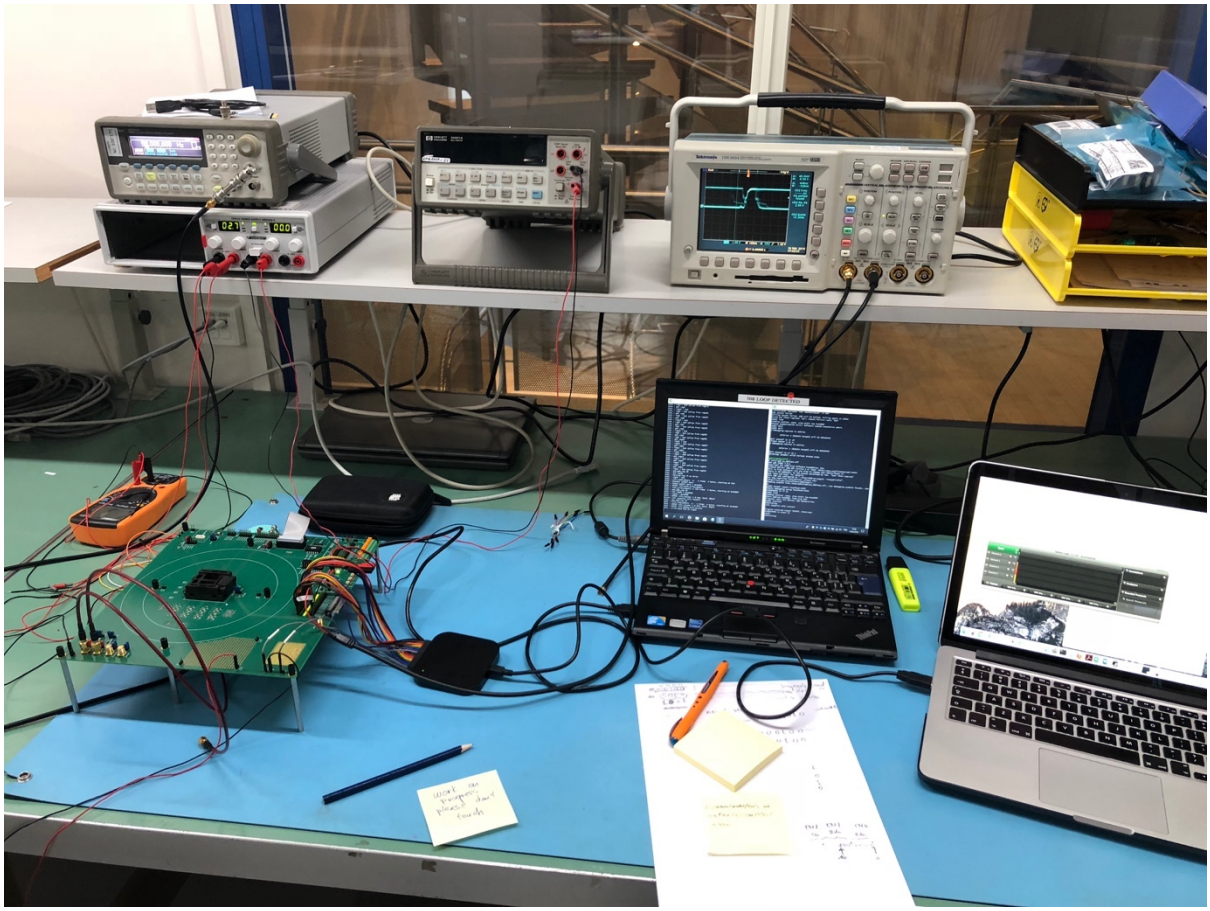

Figure 18: Instrumentation for the characterization of the DeTOP chip mounded on the first revision of the testing board.

4.4. Dictionary

The dictionary is being generated externally on a computer using a python script. Since it's not executed on the microcontroller, the power consumption and the timing is not an issue. The metric to evaluate a good dictionary is to achieve a high compression ratio. For this I wrote a software which is studying a big dataset, of EMG signals, normalizes it in the range of 0 to 4096 and then measures the frequency of appearance of each value. Since the maximum voltage that the ADC can detect is 1.8 Volts and the minimum voltage is -1.8 Volts, I also introduced a saturation on the normalization at these values. The compression ratio achieved is 45%, which is a very good compression ratio given that the dictionary is being generated externally and also that it's a dataset of 4096 values and not a small alphabet. Moreover, each patient may need a different dictionary, hence this part of the software provided is vital.

5. Minor problems

5.1. Analog Part

When I started working on the compression algorithm, the input could not be fed from the analog part, as it will happen on the final implementation of the DeTOP project. This is because the analog part was not characterized yet. It was being characterized during the execution of the software writing and the characterization of the algorithm. When I tried to use it, there was an overconsumption of current which is a problem that another group of engineers was trying to solve. In order for this problem to be solved, the chip was sent for FIB cuts, before redesigning the analog part. For my testing, I needed an input for the software. At the end, after trying several potential solutions, on the final characterization, I had the EMG data recordings loaded on the memory of the DeTOP chip. In order to simulate the delay of the ADC sample rate, which is slower than the processor, I introduced system interruptions with a timer defined delay, to make the samples available as fast as with the ADC.

5.2. Communication

In order to print a message on a screen from the DeTOP chip, for example the encoded message, the compression ratio calculation, or the total success of the software, I needed a UART chip to connect to the computer with which I was programming the chip, a script on the computer employing the UART and also software employing the UART peripheral of the chip. A more challenging task was writing information on the chip. This was needed because at first, I had written a python script that was employing the UART protocol to emulate the behavior of the analog part of the chip which was not yet available. The UART though has a limited baud rate of 1Mbit/s, making it too slow to achieve 120kS/s. For this reason, the EMG data were loaded on the memory of the chip for the compression algorithm characterization.

5.3. Access to the registers

Each of the blocks of the SoC, need calibration and this is a very time-consuming procedure. In order to calibrate those blocks, the programmer needs to access the SoC's registers on the code. The registers are listed on a library, created for the DeTOP project. This library is generated automatically, using the RTL design of the project. On each of those registers, each bit is a flag that enables a procedure, or a multibit value that calibrates a block. For example, for the calibration of the processors' clock, the first register (ANACTRL_0) should take a value which depends on the calibration clock's frequency and the target frequency. In each software I was running on the chip, this calibration needed to be performed at the beginning of the software. On a different software, where I was using a block of the AFE, I needed to access a register to enable the AFE, another to enable the block I was using, another to disable the block that I was not using and also to bypass it. For this reason, I created an easy to use on a high-level library, which is performing these tasks. Before compiling the software, with preprocessor macro commands, the test that is employed was defined, and a header file is generated, before compiling the software from the Makefile. All the user needed to do was to include this header file on the project.

6. Conclusion

6.1. Achievements

During the completion of the master thesis, the goal was to characterize for the timing, the power consumption, the compression ratio and the memory of the chip used by the compression algorithm that was initially available on the project. The next goal was to propose and perform improvements on this software. Even though I faced challenges, I describe on the previous chapter, the software was characterized, and the parts of the software that seemed to underperform were found in order to be improved. More specifically, the dictionary was producing a data inflation instead of a data compression, while there was not a script to regenerate a dictionary based on different values. The software was very power consuming especially on the memory rail. Also, the size of the software on the ram of the chip was on the limit to fit. This was translated on two problems. Another software could not fit, so the chip could not perform any other procedures. Also, the power consumption when running the software was very high.

A python script, studying a data set of EMG data, generating a new dictionary was created which solved the problem of the data inflation, achieving a data compression around 45% (up to 50% but this depends on the input) This will also be a useful tool in the future to customize the dictionary to the patient's needs.

The new version of the embedded software is also developed, different from the one that was initially available, introducing big gains in memory and power consumption. The size in the RAM memory of the new software is 19.5% of the size of the initial software. Moreover, power consumption of the chip was reduced, by 51.7% on the maximum frequency. Since this is a chip to be implanted, the power consumption reduction is a very important achievement.

6.2. Future Work

The DeTOP project is not finished yet, there are more milestones to be achieved. The second version of the DeTOP chip will be manufactured soon, which will employ also the analog part for the sensing of the EMG signals from the implanted electrodes.

Also, patients are currently being interviewed and evaluated by a group of doctors, in order to make the first application on a real amputee. This will provide information and recordings of EMG data from implanted electrodes. The recordings we are currently using are taken from patients using surface electrodes. The whole system will be customized to the user, depending on the remaining muscle tissue, the number of electrodes that will be implanted and the specific muscle groups where the electrodes will be attached. Depending on the size of the muscle group, the potential of the EMG signal will be different, and a modification of the PGA stage will be necessary to achieve a readable and high-resolution output from the analog part.

Works Cited

[1] "Detop Project," 2016. [Online]. Available: http://www.detop-project.eu/summary-and-figures/. [Accessed 02 08 2018].

[2] S. A. Raurale, "Acquisition and Processing Real-time EMG signals for Prosthesis Active Hand Movements".

[3] B. Karlik, "Machine Learning Algorithms for Characterization of EMG Signals," 2014.

[4] S. K. M. T. a. G. D. Rami N Khushaba, "Toward Improved Control of Prosthetic Fingers Using Surface Electromyogram (EMG) Signals," 2012.

[5] Z. P. N. S. L. N. Prinz R., "Fearture Extraction Through wavelet De-Noising of Surface EMG Signals for the Purpose of Mouse Click Emulation," *IEEE,* 2006.

[6] J. a. A. O. A. Carlos G. Pinheiro, "The simulation of Click and double-click through EMG signals," *IEEE,* 2012.

[7] J. M. C. Thomas M. Cover, Elements of Information Theory, Wiley, 2006.

[8] M. L. Systems, EMG Analysis & EMG Graphing Software.

[9] R. M. a. D. Farina, Surface Electromyography, IEEE Press, 2016.