RIGA TECHNICAL UNIVERSITY

K. Ozols

Asynchronous data acquisition of electroencephalogram signals

DOCTORAL THESIS

The doctoral thesis was carried out at the
Institute of Electronics and Computer Science

2017

RIGA TECHNICAL UNIVERSITY
Faculty of Electronics and Telecommunications
Institute of Radio Electronics

Kaspars OZOLS
Student of the Doctoral study program ”Electronics”

ASYNCHRONOUS DATA ACQUISITION OF ELECTROENCEPHALOGRAM SIGNALS

Doctoral Thesis

Academic supervisor
Dr.sc.comp., senior researcher
M.GREITĀNS

Riga 2017

ABSTRACT

Nowadays, brain computer interface (BCI) systems, which are based on electroencephalogram (EEG) signals, are becoming more accessible and convenient, and allows to control ”by thoughts”, for example, wheelchair, robotic prosthesis or even a car. Since wireless BCI systems use batteries, efficient energy management is crucial for increased operation time. One part of this system is analog-to-digital converter (ADC), where energy consumption can be significantly reduced. Even further, by using appropriate ADC it is also possible to reduce the amount of information to be transmitted, thus greatly reducing the energy consumption of a transmitter.

In order to choose the most appropriate ADC, first a literature review on EEG signals and BCI systems is carried out, to define requirements for selection of ADC. Then, based on these requirements, an in depth analysis of synchronous and asynchronous ADCs is performed, in order to identify their advantages and disadvantaged as well as suitability for BCI applications.

Asynchronous Sigma-Delta modulator (ASDM) ADC is selected for further in depth analysis. It shows that for wide dynamic range signals (e.g. EEG), high switching activity of ASDM circuit appears when the input signal amplitude is low, causing increased power consumption of a wireless BCI.

To improve efficiency, a new method, called Amplitude Adaptive Asynchronous Sigma-Delta modulator (AA-ASDM), is presented and described in detail, including description of signal encoding and fast and real-time reconstruction. In order to verify and assess the proposed method in practice, various simulations, modelings and physical implementations are carried out, including development of one complete wireless BCI system. The experimental research results show that by using AA-ASDM for asynchronous EEG signal acquisition, it is possible to reduce the switching activity by up to 68.85% and thus proportionally the power consumption of a wireless transmitter. Finally, at the end of this work, a summary and conclusion is given.

This thesis is the result of the research carried out at the Institute of Electronics and Computer Science within the framework of the National Research Programme „Cyber-physical systems, ontologies and biophotonics for safe&smart city and society” project No. 4.: “Development of technologies for secure and reliable smart-city” and European Social Fund’s (ESF) supported project “R&D Center for Smart Sensors and Networked Embedded Systems”.

The work consists of 174 pages, 72 figures, 183 sources of literature and 12 appendixes.

ANOTĀCIJA

Mūsdienās, uz elektroencefalogrammas (EEG) signāliem balstītas domu jūtīga saskarnes (BCI) kļūst arvien ērtākas un pieejamākās, ļaujot ar "domu spēku" kontrolēt, piemēram, ratiņkrēslu, robotisku protēzi vai pat auto. Bezvadu BCI par enerģijas avotu parasti tiek izmantotas baterijas, tāpēc, lai paildzinātu sistēmas darbības laiku, ir būtiski veidot sistēmu no tādām komponentēm, kas patērē maz enerģiju. Viena no komponentēm, kur enerģijas patēriņu var būtiski samazināt, ir analogais-ciparu pārveidotājs (ADC). Vēl vairāk, lietojot atbiltošu ADC, ir iespējams arī samazināt pārraidāmo datu apjomu, tādā veidā samazinot arī raidītāja enerģijas patēriņu.

Lai izvēlētos visatbiltošāko ADC, vispirms, darbā tiek veikta literatūras analīze par EEG signāliem un BCI kopumā, lai nodefinētu prasības ADC izvēlei. Balstoties uz šīm prasībām, tālāk darbā tiek veikta sinhrono un asinhrono ADC analīze, ar mērķi identificēt to priekšrocības un trūkumus, kā arī to piemērotību BCI pielietojumiem.

Asinhronais Sigma-Delta modulators (ASDM) tiek izvēlēts kā atbilstošākais ADC priekš BCI, tāpēc par to tiek veikta padziļinātāka analīze. Šī analīze parāda, ka priekš plaša dinamiskā diapazona signāliem (piem., EEG), ASDM ķēdes slēgšanās aktivitāte pie zemām ieejas signāla apmlitūdām ir liela, tādā veidā radot paaugstinātu BCI sistēmas enerģijas patēriņu.

Lai uzlabotu ASDM effektivitāti, darbā tiek piedāvāts jauns risinājums - Amplitūdas Adaptīvs Asinhronais Sigma-Delta modulators (AA-ASDM), kurš darbā tiek detalizēti aprakstīts gan no teorētiskās puses, gan prakstiskās. Lai praksē pārbaudītu un novērtētu piedāvāto AA-ASDM, darbā tiek veikta virkne simulāciju un modelēšana, kā arī ir izstrādāta fiziska iekārta kā daļa no darbā radītās bezvadu BCI sistēmas. Eksperimentālie rezultāti parāda, ka izmantojot AA-ASDM priekš EEG signālu kodēšanas, ir iespējams iegūt par 68.85% mazāku ķēdes aktivitāti, kā lietojot standarta ASDM. Tas rezultējas arī ar proporcionālu energījas patēriņa samazinājumu raidītājā. Darba beigās tiek dots īss paveiktā kopsavilkums, kā arī izdarīti secinājumi.

Promocijas darbs ir izstrādāts Elektronikas un datorzinātņu institūtā Valsts pētījumu programmas "Kiberfizikālās sistēmas, ontoloģijas un biofotonika drošai&viedai pilsētai un sabiedrībai" projekta Nr.4. „Tehnoloģijas drošai un uzticamai gudrajai pilsētai" un ESF projekta Nr.2009/0219/1 DP/1.1.1.2.0/09/APIA/VIAA/020 "Viedo sensoru un tīklotu iegulto sistēmu pētījumu un attīstības centrs" ietvaros.

Darbā ir 174.lpp., 72 attēli, 183 izmantotie literatūras avoti un 12 pielikumi.

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to those who have contributed to this thesis and supported me in one way or the other during this amazing journey.

First, I would like to express my special appreciation and thanks to my supervisor Senior Researcher Dr.sc.comp. Modris Greitans, you have been a tremendous mentor for me. I would like to thank you for believing in me and giving me the opportunity to work in the Institute of Electronics and Computer Science. I would like to thank you for your continuous support, inspiration and encouragement to my research and for allowing me to grow as a researcher. Your help and advice on both research as well as on my career have been priceless. I could not have imagined having a better supervisor and mentor.

I would like to express my sincere gratitude to my advisor Researcher Dr.sc.ing. Rolands Shavelis, for the continuous support, patience, motivation, and immense knowledge. His guidance and constant feedback has helped me in all the time of research and writing this thesis.

My special heartfelt thanks to my family. Words cannot express how grateful I am to my mother Marite Ozola, father Raimonds Ozols and sister Ilze Ozola for all of the sacrifices that you've made on my behalf. Thank you for always believing in me and encouraging me to follow my dreams.

Special thanks to my beloved girlfriend Sintija Fengane for helping and supporting me in whatever way she could during this challenging period. She has been a constant source of strength and inspiration.

I thank my colleagues for the stimulating discussions, support and for all the fun we have had during this period. Thank you for energizing me to strive towards my goal, it was great sharing laboratory with all of you.

And finally, last but by no means least, a big "Thank you!" goes to all my closest friends who helped, supported, inspired, motivated and encouraged me to make this thesis happen.

Thank you!

Kaspars Ozols

Riga, 2017

TABLE OF CONTENTS

List of figures

List of tables

ABBREVIATIONS

AA-ASDM - *Amplitude Adaptive Asynchronous Sigma-Delta Modulator*

AA-ASDM1 - *AA-ASDM with additional envelope encoding*

AA-ASDM2 - *AA-ASDM without additional envelope encoding*

ADC - *Analog-to-Digital Converter*

AP - *Action Potential*

ASDM - *Asynchronous Sigma-Delta Modulator*

ATS460 - *Waveform Digitizer*

Avg. - *Average*

BSS - *Blind Source Separation*

BCI - *Brain Computer Interface*

BMI - *Brain Machine Interface*

CMOS - *Complementary Metal Oxide Semiconductor*

CMRR - *Common Mode Rejection Ratio*

CNS - *Central Nervous System*

CTR - *Counter*

DAC - *Digital to Analog Converter*

DC - *Direct Current*

DDS - *Direct Digital Synthesis*

DNI - *Direct Neural Interface*

DRL - *Driven Right Leg*

DSP - *Digital Signal Processing*

ECG - *Electrocardiogram*

EEG - *Electroencephalogram*

EMG - *Electromyogram*

EMI - *Electromagnetic Interference*

ENOB - *Effective Number Of Bits*

EOG - *Electrooculogram*

FFT - *Fast Fourier Transform*

GND - *Ground*

GUI - *Graphical User Interface*

HDR - *High Dynamic Range*

ICA - *Independent Component Analysis*

IF - *Intermediate Frequency*

ISM - *The industrial, scientific and medical radio band*

LC - *Resonant Circuit consisting of an inductor and a capacitor*

LC ADC - *Level Crossing ADC*

MMI - *Mind-Machine Interface*

MSB - *Most Significant Bit*

OOK - *On Off Keying*

PC - *Personal Computer*

PCB - *Printed Circuit Board*

PLL - *Phase Locked Loop*

PNS - *Peripheral Nervous System*

SAR - *Successive Approximation Register*

SMD - *Surface-Mounted Device*

SNR - *Signal to Noise Ratio*

SNDR - *Signal to Noise plus Distortion Ratio*

SoC - *System on Chip*

SoD - *Send on Delta*

SRG - *Shift Register*

SWR - *Sine Wave Crossing*

S&H - *Sample and Hold*

TDC - *Time to Digital Converter*

TEM - *Time Encoding Machine*

UWB - *Ultra Wide-band*

VCC - *Power Supply Voltage*

VCO - *Voltage Controlled Oscillator*

XNOR - *Exclusive Neither Disjunction*

NOMENCLATURE

$\mathbf{a}$ - vector

$\mathbf{a}^T$ - transposed vector

b - ASDM/AA-ASDM trigger parameter

C, c - maximum value of the modulus of a signal

$c(t)$, $\tilde{o}(t)$ - signal time-varying envelope function

$\mathbf{D}$ - diagonal matrix

E - energy of the signal

$e(t)$ - error signal

F_{max} - maximum frequency

$\mathbf{G}$ - matrix

$\mathbf{G}^{-1}$ - inverse matrix

$\mathbf{G}^{+}$ - pseudo-inverse matrix

$g(t)$ - signal reconstruction base function

$\hat{g}(t)$ - periodic approximation of $g(t)$

$h(t)$ - signal reconstruction base function/impulse response of a filter

N_{ASDM} - number of switching time instants

P - signal power

$\sup_{k\in Z}(x_k)$ - least number y that is greater than or equal to all numbers x_k, $k \in Z$

T - period/Nyquist step

t - continuous time

t_n, τ_n - discrete time

t_k - ASDM/AA-ASDM trigger switching time instants

$w(t)$ - window function

$x(t)$ - analog signal

$\bar{x}(t)$ - mean value

$\hat{x}(t)$ - reconstructed signal

$\hat{x}_{L,M,K}(t)$ - reconstructed signal from intervals

$y(t)$ - output of the integrator

Z - set of all integers

$z(t)$ - ASDM/AA-ASDM output trigger signal

α, β - constant

Δ - quantization step

δ - ASDM/AA-ASDM trigger parameter

$\delta_{k,n}$ - Kronecker symbol

Θ - length of a signal

$\theta(t)$ - function, which describes the rising and falling edge of the window function

κ - ASDM/AA-ASDM integrator parameter

ξ - constant

$\varsigma(t)$ - integral signal representation

τ_{max} - maximum distance between two consecutive trigger switching time instants

τ_{min} - minimum distance between two consecutive trigger switching time instants

Ω - cut-off cyclic frequency

ω - cyclic frequency

INTRODUCTION

Almost everything we encounter in our daily lives - sound, light, temperature, pressure, smell, flavor, etc. are in analog form. On the other hand, nowadays, almost all information is stored and processed in digital form. In order to fill this gap between the real world and the digital world, various analog to digital converters (ADCs) are used.

One very rapidly growing sector, which demands good quality ADCs is neuroscience. Human brain has more than 100 billion neurons which conduct electrical impulses (signals) and are the core components of the nervous system. The nervous system, which includes the brain and spinal cord, which together comprise the central nervous system (CNS) and the ganglia of the peripheral nervous system (PNS) is responsible for sending, receiving, and processing nerve impulses throughout the body. All the organs and muscles inside your body rely upon these nerve impulses to function. Basically, the nervous system controls everything that human do, see, hear, smell, feel, think, remember, dream, etc. by conducting electrical impulses. In fact, brain neuronal activity generates electrical currents, which in turn generate electrical field potentials which can be measured by using special electrodes, located on the scalp in certain places. [1] The measured signals, called electroencephalogram (EEG) signals, are in analog form and must be digitized (by ADC) to enable easier and much faster signal storage, analysis, processing and research. Since EEG signals contain information about the brain neuronal functions and neurophysiological properties, nowadays, by using advanced signal processing techniques [2], it is possible to understand different processes in the human body. Even further, by using advanced signal processing techniques in modern Brain Computer Interface (BCI) systems [3], [4], [5], it is possible to control ”by thoughts”, for example, wheelchair, robotic prosthesis, computer, or even a car [6], [7], [8], therefore it is reasonable to believe that BCI is a future technology and it is very important to develop such a field.

In general, BCI is a communication pathway between the brain and an external device. Typical BCI system consists of four main parts: electrodes, differential amplifiers, ADC(s) and data processing / visualization device (i.e. PC). But, nowadays, in order to make it more accessible and convenient to use, a modern BCI systems also include wireless data transmission [9], [10], [11] and a battery as a power source. Despite the fact that development of such systems very increasingly becomes the subject of research [12], [13], [14], still there exist various problems, weaknesses and limitations.

In order to prolong the life of battery and thus operation time of wireless BCI system, management of energy consumption is a crucial factor. As mentioned before, one very important part of BCI system is ADC, where energy consumption can be significantly reduced. Even further, by using appropriate ADC it is also possible to reduce the amount of information to be transmitted, thus greatly reducing the energy consumption of a transmitter as well. But, excluding low energy consumption, ADC must also ensure proper sampling rate and resolution for good signal quality as well as small physical size for convenient and lightweight on-head device implementations.

Two types of ADCs can be differentiated: a) synchronous ADCs, for example, Flash ADC, Digital Ramp ADC, Successive Approximation ADC, Tracking ADC, Sigma-Delta ADC and Slope (integrating) ADC [15], [16], [17]; and b) asynchronous ADCs, for example, Zero Crossing ADC, Sine Wave Crossing ADC, Level Crossing ADC, Send-on-Delta ADC, Peak Sampling ADC and Asynchronous Sigma-Delta Modulator [18], [19], [20]. Each of these topologies have its own advantages and disadvantages depending on the end-use application.

For BCI application, it is shown that asynchronous designs, instead of synchronous (which are used in almost all available BCI systems), in ADCs, exhibit better properties such as lower energy consumption, immunity to metastable behavior, modular design, low complicity, exclusion of electromagnetic interference (EMI) and absence of clock jitter [21], [22].

Since EEG signals can be classified as a wide dynamic range signals, non-uniform sampling method called Asynchronous Sigma-Delta Modulator (ASDM) has a great potential to improve energy efficiency in BCI system [23], while maintaining other quality requirements. ASDM is a Time Encoding Machine (TEM) which transforms the amplitude information of the signal into time information or time codes without the amplitude quantization error that exists in the clocked converters. The method replaces high precision analog amplitude quantizer with 1bit comparator. This reduces the analog circuit complexity. The number of transitions can be controlled by adjusting step size that is hysteresis of the comparator. This in turn reduces the switching activity and the dynamic power consumption. [24] Latest implementations show that it is possible to create standard ASDM with power consumption not exceeding 7.5nW [25].

However, due to wide dynamic range that EEG signals have, a high switching activity of ASDM circuit appears when the input signal amplitude is low, causing increased power consumption of a wireless BCI system. Considering the above-mentioned, the main aim of this thesis is to develop an improved method for signal encoding based on ASDM, which allows to reduce the power consumption of the wireless BCI system, while maintaining the desired signal quality. In order to reach the aim, following tasks have been defined:

- to carry out literature review and analysis on EEG signals, including neural activities, action potentials, brain rhythms, EEG signal properties and measurement techniques;
- based on EEG signal and BCI system properties, to define requirements for ADCs;
- based on defined requirements, to carry out literature review and analysis on synchronous and asynchronous ADCs and select the most appropriate ADC;
- to carry out an in depth analysis and research on ASDM, and identify the points of improvement;
- to carry out an in depth research on these improvements;
- to develop, test and assess the proposed methods.

Accordingly, the author has defined three theses:

1. In comparison to ASDM, the decrease of the number of switchings per second of the Amplitude Adaptive Asynchronous Sigma-Delta modulator (AA-ASDM) output trigger is determined by the mean value and the maximum value of the modulus of the signal and the mean value of its estimated envelope function.
2. For electroencephalogram (EEG) signals, the AA-ASDM with the proposed envelope function has up to 68.85% less average switchings per second of the output trigger, compared to ASDM, while maintaining a 22-bit resolution of AA-ASDM conversion.
3. In comparison to ASDM, by using AA-ASDM it is possible to reduce energy consumption of an event-driven data transmitter by 50% depending on the circuit parameters.

Based on the defined thesis and set tasks, the Doctoral thesis is divided into five sections. In Section 1, a literature review on neural activities, action potentials, brain rhythms, EEG signal properties and measurement techniques is carried out. Also, in this section, an in depth analysis of synchronous and asynchronous analog-to-digital converters is performed, in order to identify their advantages and disadvantaged as well as suitability for BCI applications. Based on this analysis, one the most suitable ADC for BCI is selected for in depth analysis in Section 2.

This section is dedicated to research on ASDM, particularly on signal encoding and decoding fundamental principles. Also, the main advantages and disadvantages are identified and analyzed. Knowing the main drawbacks of ASDM, in Section 3 the proposed new method, called AA-ASDM is presented. This section describes the main theoretical principles of AA-ASDM signal encoding and reconstruction, as well as fast and real-time reconstruction. In order to verify and assess this theory in practice, in Section 4 various simulations, modeling and physical implementations are carried out for both ASDM (as a reference design) and AA-ASDM. Finally, at the end of this work, in Section 5, conclusion with suggestions for future research is given. In appendices several mathematical derivations and *Matlab* codes related to ASDM and AA-ASDM are given.

This Doctoral thesis is based on the papers **[26]**, **[27]**, **[28]** **[29]** by the author of this Thesis and paper co-authors Rolands Shavelis and Modris Greitans. Here and further in the text, the references with authors own contribution are highlighted in bold un underlined.

1. BACKGROUND AND RELATED WORK

The main purpose of this section is to describe previous research in the field of analog-to-digital converts (ADC) and choose one the most appropriate for electroencephalogram (EEG) encoding and brain computer interface (BCI) application as whole. In order to achieve this aim, first of all it is necessary to understand the nature of EEG signals. For example, is it random? Is it linear? Is it stationary? What are typical amplitudes and frequencies? etc. For this reason, the Section 1.1 is dedicated to understanding how our brain works, what are the properties of EEG signals, how these signals can be measured and how BCI systems work. Based on the results achieved in Section 1.1, the most suitable for EEG signal encoding, synchronous (Section 1.2.1) and asynchronous (Section 1.2.2) ADC's will be chosen and analyzed against criteria set out in the Section 1.2. After the analysis, performed in Section 1.2.1 and Section 1.2.2, the most appropriate/promising ADC for brain computer interface (BCI) application will be selected for further in-depth analysis in Section 2. At the end of this section a summary and conclusions are given.

1.1 Electroencephalogram Signals

A brain-computer interface (BCI), sometimes called a mind-machine interface (MMI), direct neural interface (DNI), or brain–machine interface (BMI), is a direct communication pathway between the brain and an external device [30]. In order to establish this communication, first it is necessary to measure brain neural activity. The brain neural activity can be measured by the electroencephalogram (EEG) and the measured signals are called EEG signals. In order to understand main properties of EEG signals, which further will be used to choose the most appropriate ADC, in this section a brief overview on neural activities, action potentials, EEG generation, brain rhythms, EEG measurement techniques and description of typical BCI will be given.

1.1.1 Neural Activities and Action Potentials

Human brain has more than 100 billion neurons (or nerve cells) which conduct electrical impulses (signals) and are the core components of the nervous system [1]. The nervous system, which includes the brain and spinal cord, which together comprise the central nervous system (CNS) and the ganglia of the peripheral nervous system (PNS) is responsible for sending, receiving, and processing nerve impulses throughout the body [1]. In general, the CNS consists of two types of cells: glial and nerve cells, which are located between neurons. As it is shown in Fig. 1.1, each nerve cell consists of cell body (with nucleus), axons and dendrites. Nerve cells sense changes in the environment, transmit information to other neurons (via axon terminal) and respond to sensations. A nerve cell body has a single nucleus and contains most of the nerve cell metabolism [2]. The axon, which can be several meters long, is specialized in information (electrical impulses) transfer in the nervous system. In order to transmit the information faster, axons are insulated by Schwann cells. [1] [2] [31]

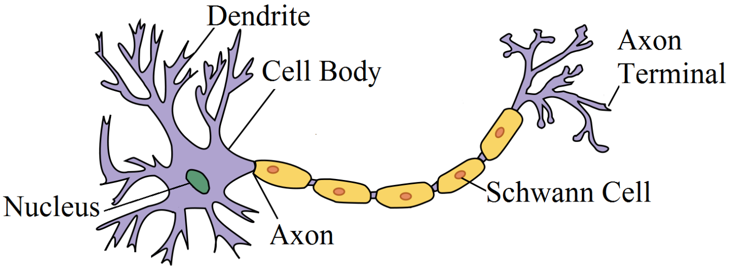


Figure 1.1: Structure of a nerve cell [32]

Dendrites are connected to either the axons or dendrites of other cells and receive electrical impulses from other nerves or relay the signals to other nerves [2]. In the human brain, through dendrite connections, each nerve is connected to approximately 10,000 other nerves [2].

Generally, most of the activities in CNS are related to the synaptic currents, which are transmitted between the synapses of axons and dendrites in order to pass the information. The information transmitted by a nerve over distances in the nervous system is called an action potential (AP). Action potentials can be initiated by different types of stimuli, for example, sensory nerves respond to stimuli such as sound, light, pressure, electricity, touch, smell, etc. On the other hand, the CNS nerves are stimulated mostly by chemical activity at synapses. All the organs and muscles inside humans' body rely upon these stimuli currents (nerve impulses) to function. Basically, the nervous system controls everything a human do, see, hear, smell, feel, think, remember, dream, etc. by conducting electrical impulses. [1] [2] [31]

Action potentials are caused by an exchange of Na+ (sodium) and K+ (potassium) ions across the neuron membrane and can only occur when opposite charges exist on two sides of a cell membrane (intra- and extracellular space). In short, AP is simply an electric current, which is created by sudden reversal in charge, that travels down an axon of a nerve cell. [2] [31]

As it is shown in Fig.1.2, when a nerve cell is resting (not transmitting impulses) the membrane potential is $-60mV$. At this state Na+ and K+ ions move their concentration gradients through their membrane channels to opposite sides of membrane, thereby ensuring that the number of positive charges on the outside is higher than on the inside. When nerve cell is sending a signal (impulse), the membrane potential depolarizes (becomes more positive (rising phase)), producing a spike. At this phase, additional Na+ channels open and the Na+ influx drives the interior of the cell membrane up to approximately $+25mV$. After the peak of the spike, Na+ channels close and the K+ channels open therefore membrane potential re-polarizes and becomes more negative than the resting potential. After hyper-polarization phase, which prevents the neuron from receiving another stimulus and thus ensuring that the signal is proceeding in one direction, membrane potential returns to normal - resting level. Usually, for human, the AP of most nerves last between 1 and 10 milliseconds and amplitude ranges between $-60mV$ and $25mV$. [2] [31] [33] [34]

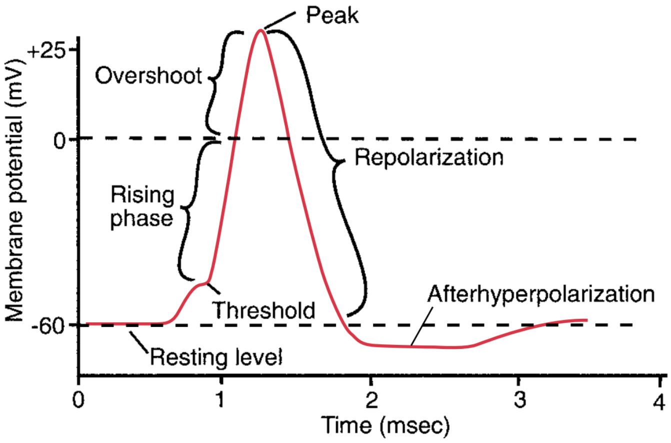


Figure 1.2: The phases of an action potential [33]

An inflow of anions into the nerve cell and overflow of cations from the nerve cell causes a potential change along the nerve cell membrane. Primary trans-membranous currents generate secondary ional currents along the cell membranes in the intra- and extracellular space. The portion of these currents that flow through the extracellular space is directly responsible for the generation of field potentials. These field potentials, which are usually with less than 100 Hz frequency, are called EEGs and can be measured also by non-invasive methods from scalp. Basically, an EEG signal measurement is a measurement of currents that flow during synaptic excitations of the dendrites of neurons in the cerebral cortex. The measured signal is a sum of the large number of nerve cells (neurons) potentials, since only large number of neurons can generate enough potential to be recorded from the scalp electrodes. [1], [2] [35]

1.1.2 Brain Rhythms

According to [36] the cerebral cortex can be divided into 52 Brodmann's "areas" (Fig. 1.3) which designate functional regions in the cortex. For example, there is an area for the "motor" cortex, which control our muscles, an area for "visual" cortex, which we see with, areas for somatosensory cortex, and so on. [36]

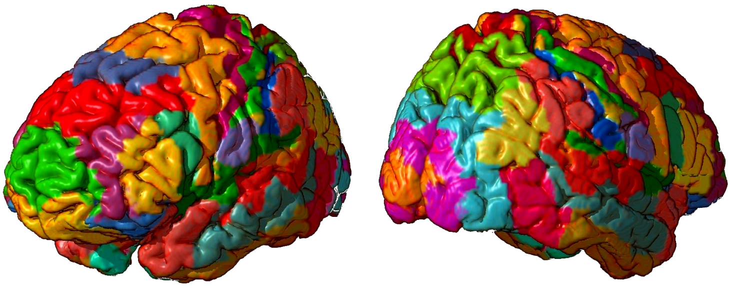


Figure 1.3: 3D representation of Brodmann areas [37]

By taking into account Brodmann's areas, when measuring and analysing neural activities in the brain, it is possible to understand different processes in our body. Even further, it's possible to diagnose many neurological disorders and other abnormalities in the human body, such as, epilepsy, seizures, sleep disorders, physiology, and many more. [2]

In order to be able to analyze EEG signals, it's necessary to understand main EEG signal properties. Our brain primarily operates in five brainwave states or brain rhythms, which can be divided into bands by frequency. Brain rhythms range from being wide awake and active, the beta brainwave state, to the calm, focused alpha state, and the creative, deeply meditative theta state, down to low wave, deep, dreamless, restorative sleep referred to as the delta brainwave state. Figure 1.4 shows the typical brain rhythms with their typical frequency ranges and amplitude levels. [38] [39] [40]

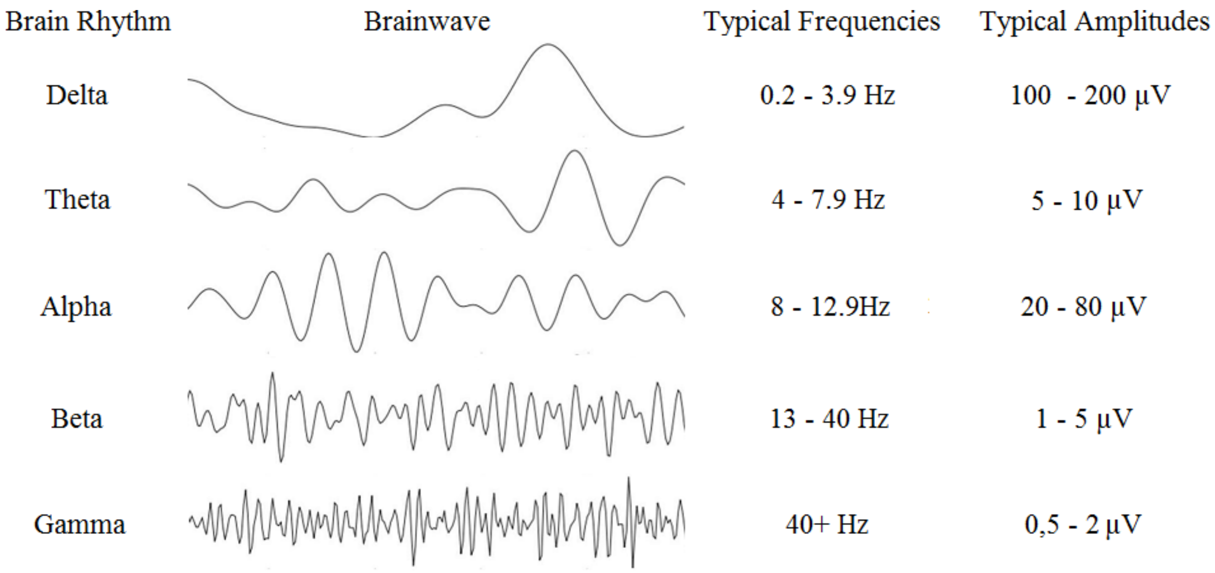


Figure 1.4: Brain Rhythms (adopted from [39])

Delta waves. The typical frequency range is from 0.2 – 3.9 Hz, while typical amplitudes, measured from the scalp, are from 100 - 200μV. Delta is the slowest band of brainwaves with the highest amplitude and are primarily associated with deep, dreamless sleep and may be present in the waking state. It is easy to confuse delta waves with artifact signals from neck and jaw muscles. Nevertheless, by using simple signal processing methods it is easy to distinguish delta waves from responses caused by excessive movement. [41] [42] [43]

Theta waves. The typical frequency range is from 4 – 7.9 Hz, while typical amplitudes are from 5 - 10μV. Theta waves are associated with light sleep, creative inspiration, extreme relaxation, deep meditation or even hypnotherapy and self-hypnosis. High theta wave activity in an awake adult are abnormal and are caused by various pathological problems. [41] [43] [44]

Alpha waves. The typical frequency range is from 8 – 12.9 Hz, while typical amplitudes are from 20 - 80μV. Alpha waves are associated with tranquility, light meditation and relaxation without any attention or concentration. Alpha activity has also been associated to the ability to recall memories, reductions in stress and anxiety and lessened discomfort and pain. Most of the subjects produce more alpha waves when their eyes are closed. Usually, alpha waves appears as a sinusoidal or round shaped signal. A peak of alpha waves can regularly be seen in the beta waves in frequencies up to 20 Hz. Also, quite often an alpha wave response is seen at 75 Hz. [41] [43] [45]

Beta waves. The typical frequency range is from 13 – 40 Hz, while typical amplitudes are from 1 - 5μV. Beta waves are the usual waking rhythm associated with mental activity, active thinking, active attention, concentration, solving concrete problems, etc. High-level beta waves may occur when a human is in a state pf panic. A central beta rhythm can be blocked by tactile stimulation or motor activity. Many people lack sufficient beta activity, which can cause mental or emotional disorders such as depression or insomnia. By stimulating beta activity it is possible to improve emotional stability, attentiveness and concentration [46]. [41] [43]

Gamma waves. The typical frequency range is from 40 - 100 Hz, while typical amplitudes are from 0.5 - 2μV. Gamma waves are associated with language and memory processing, formation of ideas and various types of learning [47]. Gamma waves are thought to represent binding of different populations of neurons together into a network for the purpose of carrying out a certain cognitive or motor function [48]. [43]

Besides primary brain waves delta, theta, alpha, beta and gamma, there are other brain waveforms, like rho, mu, phi, kappa, tau, chi and lambda. [41]

All these brain rhythms together are forming the electroencephalogram signal (Fig. 1.5).

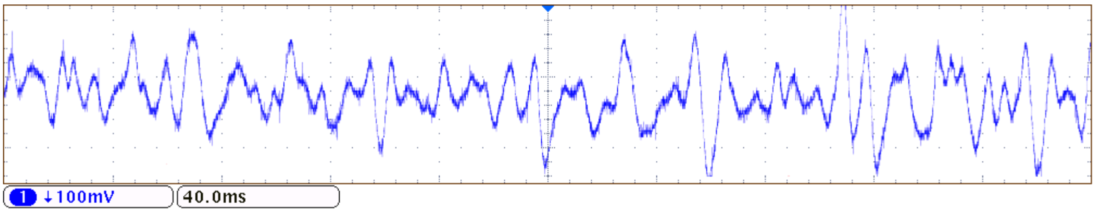


Figure 1.5: Electroencephalogram (EEG) signal

In general, EEG signals are the projection of neural activities that are attenuated by leptomeninges, cerebrospinal fluid, dura matter, bone, galea, and the scalp. Cartographic discharges show amplitudes of 0.5–1.5 mV and up to several millivolts for spikes. However, on the scalp the amplitudes commonly lie within range of 0.5–200 μV. [41]

1.1.3 EEG Measurement

In neuroscience, electroencephalography is a very often used measurement technique for brain electrical potential measurement. Using special electrodes, located on the scalp in certain places, it is possible to get information about the brain neuronal functions and neurophysiological properties.

EEG measurement and recording systems

Usually multi-channel electroencephalography systems consist of four main parts: electrodes, differential amplifiers (one for each channel), analog-to-digital converter (ADC) and data processing / visualization device (i.e. PC). But, nowadays, a modern multi-channel BCI systems also include wireless data transmission (see Fig. 1.6) [9], [10], [11]. Depending on the design, some systems are supplemented by other parts, for example, hardware filters, preprocessing units, etc.

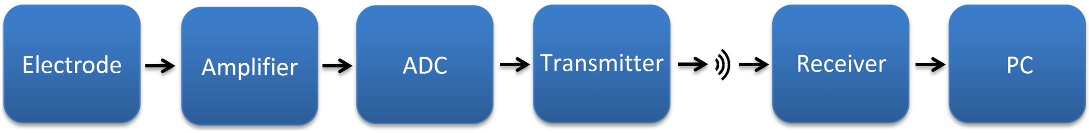


Figure 1.6: Block diagram of generalized EEG measurement and recording system

The EEG recording electrodes are crucial for high quality data acquisition. Currently, for EEG recording systems, different types of electrodes are used, such as, disposable, reusable silver/gold disc electrodes, headbands and electrode caps, saline-based electrodes, needle electrodes, etc. Non-invasive scalp electrodes usually are divided into two groups: wet and dry electrodes, where due to better contact with the skin, wet electrodes present higher signal quality. For multi-channel EEG systems with a large number of electrodes, one of the most popular scalp electrodes are AgCl disks. Since high impedance (more than $5k\Omega$) between the cortex and the electrodes can lead to distortions, modern EEG recording systems are often equipped with impedance measurement sensors. [2] [49]

Since the amplitude of the EEG signal, measured from scalp, are only few μV, amplifiers are an important part of the modern EEG measurement systems. In order to avoid distortions, these amplifiers must meet certain requirements. First of all, they must have high input impedance (at least $10M\Omega$), so they provide minimal loading of the signal being measured. Besides that, the input circuit must provide protection, since any current appearing across the amplifier input terminals produced by the amplifier is capable of affecting the signal being measured. Also, the amplifiers should have isolation and protection circuitry, so the current through the electrode circuit can be kept at safe level and any artifact generated by such current can be minimized. Furthermore, the output impedance of the amplifier must be low with respect to lead impedance and it must operate in particular frequency range, thus ensuring optimal signal to noise ratio (SNR). Modern EEG systems use differential amplifiers, where signals are obtained from bipolar electrodes. In this case, the differential amplifier must have high common-mode-rejection ratio (CMRR) (at least $120dB$), in order to minimize interference due to the common-mode signal. [49], [50]

In order to make EEG signal processing and analysis more effective, different ADCs are used to digitize analog EEG signal. Since EEG signals can be classified as High Dynamic Range (HDR) signals (see Section 1.4), ADC must have appropriate resolution not to lose important components of the signal. By knowing typical amplitudes of the measured EEG signal from previous section (0.5–200 μV), it is possible to calculate desired resolution of the ADC.

In the case shown in Figure 1.7, the maximum peak to peak amplitude of the EEG signal is $2A$, but minimum peak to peak amplitude: $2a$.

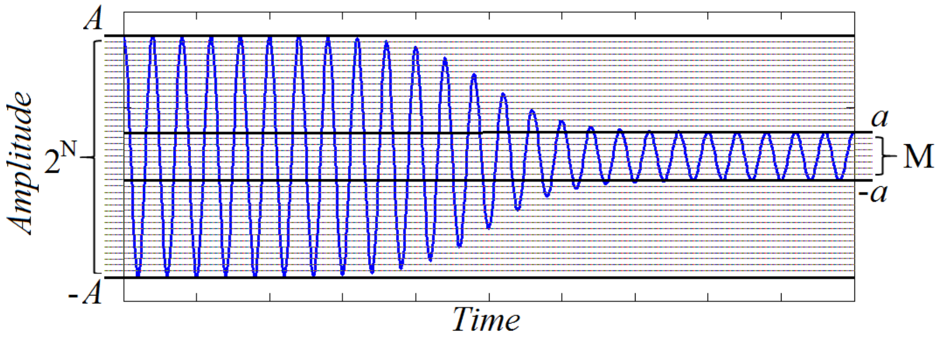


Figure 1.7: Calculating ADC resolution depending on the signal dynamic range; $2A$ - maximum peak to peak amplitude, $2a$ - minimum peak to peak amplitude, M - quantization levels

By assuming that it is necessary to have M quantization levels between a and $-a$ we can write a following expression:

$$\Delta_2 = \frac{2a}{M}, \tag{1.1}$$

where $a > 0$ and Δ_2 is the quantization step.

In order to determine the desired resolution, i.e. how many bits (N) are necessary for ADC to encode EEG signal without loosing important components, we can write an equation for N-bit ADC, where dynamic range is $2A$:

$$\Delta_1 = \frac{2A}{2^N}, \tag{1.2}$$

where Δ_1 is the quantization step.
In this case, it is necessary to ensure following inequality:

$$\Delta_1 \leq \Delta_2 \tag{1.3}$$

From (1.3) it follows:

$$\frac{2A}{2^N} \leq \frac{2a}{M} \quad \longrightarrow \quad N = \frac{lnM \cdot \frac{A}{a}}{ln2} \tag{1.4}$$

By knowing typical EEG signal amplitudes and assuming that $M = 10$ is a sufficient condition, it is possible to calculate minimum number of bits, which are necessary to encode the EEG signal:

$$N = \frac{ln10 \cdot \frac{400}{1}}{ln2} = 11.9658 \approx 12\, bits. \tag{1.5}$$

Since EEG signal bandwidth in most cases is limited to approximately 200 Hz (see Section 1.4), there is no need to have ADCs with high sampling rate capabilities.

When EEG signal is digitized by ADC, it must be transmitted to signal reconstruction and processing unit, which usually is some kind of embedded system or PC. Usually bluetooth or wi-fi communication is used for wireless data transmission.

Currently, in modern BCI systems, one of the biggest challenges is to reduce the power consumption in data acquisition part, i.e., to develop ultra low power amplifiers, ADCs and transmitters, in order to prolong the life of the battery and thus operation time of a wireless BCI system. State of the art low power, low noise amplifiers or their cascade with at least 80dB gain, which is necessary for EEG signal amplification, consume starting from 1,37nW [51], [52], [53], [54]. But low power ADCs, which can ensure 12-bit resolution, consume starting from 40nW [55], [56], [57]. As expected, the most power consuming part of the data acquisition system is transmitter. The power consumption of conventional transmitters varies between 3mW and 60mW [58], [59], [60] [61], but ongoing research shows, that it is possible to create a transmitter which consumes few hundreds of nW [62], [63] or even hundreds of pW if an Ultra Wide Band (UWB) technologies are used to transmit 100 events per second [64], [64], [65]. Although power consumption of UWB based transmitters is extremely low, the reception and reconstruction of the signal is very complex and usually inaccurate. Also, the generation of different shape UWB pulses is very complicated, thus limiting the number of channels which can be used for BCI system.

Electrode Positioning

The International Federation of Societies for Electroencephalography and Clinical Neurophysiology has recommended the 10-20 system, which is the most commonly used method for electrode placement on scalp for EEG tests and experiments (see Fig. 1.8). [41]

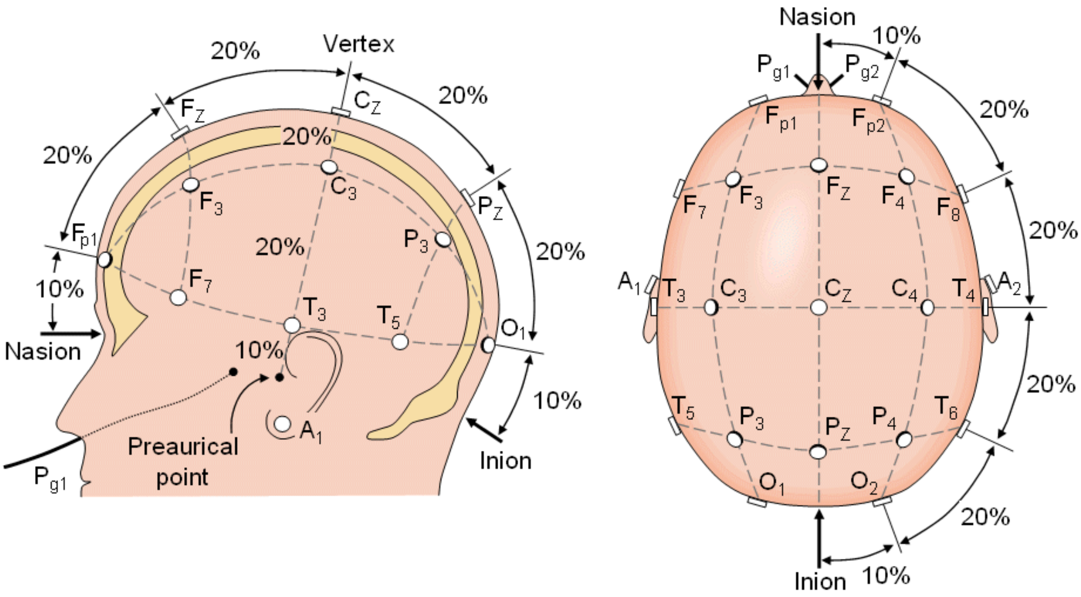


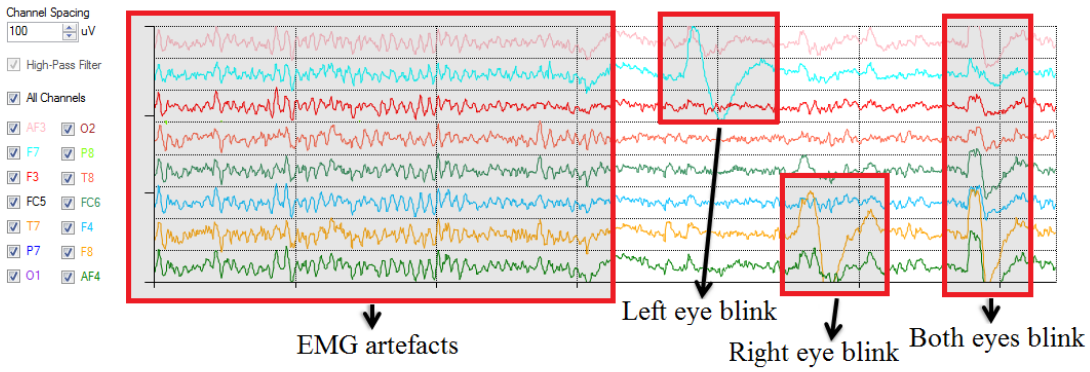
Figure 1.8: The international 10-20 system of electrode placement [66]

The 10-20 system, is based on the ratio (in %) between electrode location and underlying area of cerebral cortex, where 10% and 20% refer to the actual distances between adjacent electrodes of the total skull's front–back or right–left distance (see Figure 1.8). In order to identify the lobe and the hemisphere location, each of these positions has a designation, which consists of a letter and a number. In this case, the letter F stands for frontal lobe, T - temporal lobe, C - central lobe, P - parietal lobe, and O - occipital lobe. The numbers show in which hemisphere particular electrode is located, if even number - on the right hemisphere, if odd number - on the left hemisphere. In addition, also earlobe reference electrodes with expressions A1 and A2 are used in 10-20 system. Even though the original 10-20 system contains only 21 electrode, modern systems, for example ”10–5 system”, use up to 256 electrodes. [67]

EEG Signals and their Artifacts

EEG is complicated, continuous, non-stationary, non-linear and noisy signal, which is also affected by different artifacts. [41] [68] The main artifacts can be divided into biological and environmental/system artifacts.

The biological or internal artifacts include eye-induced artifacts or EOG (i.e. eye blinks, eye movements, extra-ocular muscle activity), body movement-related artifacts, EMG (muscle activation) induced artifacts, ECG (cardiac) and pulsation artifacts, glossokinetic artifacts and others. [41] [48] An example of EEG signals, which are affected by biological artifacts is shown in Figure 1.9. In order to remove biological artifacts different methods, such as independent component analysis (ICA) or blind source separation (BSS), are used. [69] [70] [71] The basic idea behind such methods is to ”remix” only those components that would result in ”clean” EEG signal by zeroing the weight of unwanted components. [48]



Channel Spacing 100 uV

☑ High-Pass Filter

☑ All Channels

Figure 1.9: EEG signal affected by biological artifacts

The environmental/system artifacts may result from 50/60 Hz power supply interference, an electrode pops from patient movement, impedance fluctuation, electrical noise from the electronic components, cable defects, unbalanced impedance of the electrodes, etc. [41] [48] In order to remove environmental/system artifacts and exclude any distortion of the EEG signals, different filters are designed. Highpass filters (cut-off frequency less than 0.5 Hz) are used to remove the low frequency components, for example, breathing. On the other hand, lowpass filters (cut-off frequency ≈50–70 Hz) are used to mitigate high-frequency noise. In addition, also notch filters with a null frequency of 50 Hz are used, in order to mitigate noise, coming from the strong 50 Hz power supply. [41]

1.2 Analogue-to-Digital Converters

Despite the fact that development of Brain Computer Interfaces (BCI) has increasingly become the subject of research [12], [13], [14], still there exist various problems, weaknesses and limitations. Since the recent trend is developing such systems more convenient in use [72], [73], most of them are equipped with wireless data transmission and accordingly with a battery as a power source. In order to prolong the life of the battery and thus operation time of wireless system, management of energy consumption is a crucial factor. One very important part of all BCI systems is analog-to-digital converter (ADC) (see Section 1.1.3), where energy consumption can be significantly reduced. In addition, by choosing appropriate ADC, it is also possible to reduce the power consumption of the transmitter, by reducing the amount of data to be transmitted. Therefore the main aim of this section is to choose and analyse both synchronous and asynchronous ADC's against requirements/criteria, which are set out based on the EEG signal properties (see Section 1.1) and BCI system as whole.

In order to be able to choose and analyse different ADCs, first of all the requirements/criteria, against which all ADCs will be examined, must be set out. Based on EEG signal properties and the current situation in the field of BCI's (see Section 1.1), four main criteria have been set out:

Energy Efficiency. In modern wireless BCI systems, energy efficiency is a crucial parameter, since lifetime of the battery is strongly related to the whole system's power consumption. The less power BCI system consumes, the longer it will operate. In this case, the most energy efficient ADCs will have a greater advantage to be selected.

Encoding Complexity. In order to develop a miniature and convenient BCI system, it is important to reduce both, size and weight. This can be achieved by reducing complexity of the overall BCI system. Since ADC is part of the whole system, it is also important to reduce its complexity (and therefore costs as well). This can be achieved by using fewer elements at the encoding side. ADC occupying the smallest area on the silicon die will have greater advantage to be selected. In this case, all ADCs are compared evenly by finding the smallest, most energy efficient ADC, which is implemented by using $180nm$ CMOS process technology and fits to the Resolution and Sampling Rate requirements (see below).

Resolution. In general, the resolution defines the smallest analog input signal voltage change that can be measured by the ADC. In practice, the resolution of the ADC is limited by the signal-to-noise ratio (SNR). If there is too much noise present in the analog input, it will not be possible to reconstruct accurately the signal beyond a certain number of bits of resolution, the Effective Number Of Bits (ENOB). ENOB defines ADC's dynamic performance at the specific input frequency and sampling rate. An error of an ideal ADC consists only of quantization noise. As the frequency of the input signal increases, the overall noise also increases, thereby reducing the ENOB. [74] In this case, the ENOB can be calculated as [74]:

$$ENOB = \frac{SNR - 1.76}{6.02} \tag{1.6}$$

Since EEG signals can be classified as a high dynamic range (HDR) signals it is important to have such ADC, which supports conversion of HDR signals. In this case, as shown in Section 1.1.3, ADC must ensure at least $ENOB \geq 12$ bit resolution.

Sampling Rate. As shown in Section 1.1.2 and Section 1.1.3, EEG signal maximum frequency does not exceed 200 Hz. Therefore, the parameter of sampling rate is not very demanding for ADC and can be ensured by almost all ADCs. In this case, ADC must ensure at least 400 S/s sampling rate.

Based on these requirements, in the next two sub-sections (Section 1.2.1 and Section 1.2.2), both synchronous and asynchronous ADC techniques will be analysed with the associated trade offs of each topology. After the analysis the most appropriate/promising ADC for BCI application will be chosen for further in depth analysis and research.

1.2.1 Synchronous Analogue-to-Digital Conversion

Currently, the most popular architectures of analogue-to-digital converters (ADCs) are Flash ADC (direct conversion), Pipeline ADC, Digital Ramp ADC, Tracking ADC, Successive Approximation (SAR) ADC and Sigma-Delta ($\Sigma\Delta$) ADC. Since all of these ADCs are driven by a global clock, where ADC states are changed only on rising or falling edges of the clock pulses, such ADCs are called synchronous ADCs. In this section, an overview of mentioned ADCs as well as analysis in relation to set out requirements (energy efficiency, encoding complexity, resolution and sampling frequency) is given.

1.2.1.1 Flash Analogue-to-Digital Converter

Flash ADC, also called parallel ADC, is the fastest type of ADCs. [75] Figure 1.10 shows the block diagram of Flash ADC. A N-bit Flash ADC consists of 2^N resistors and $2^N - 1$

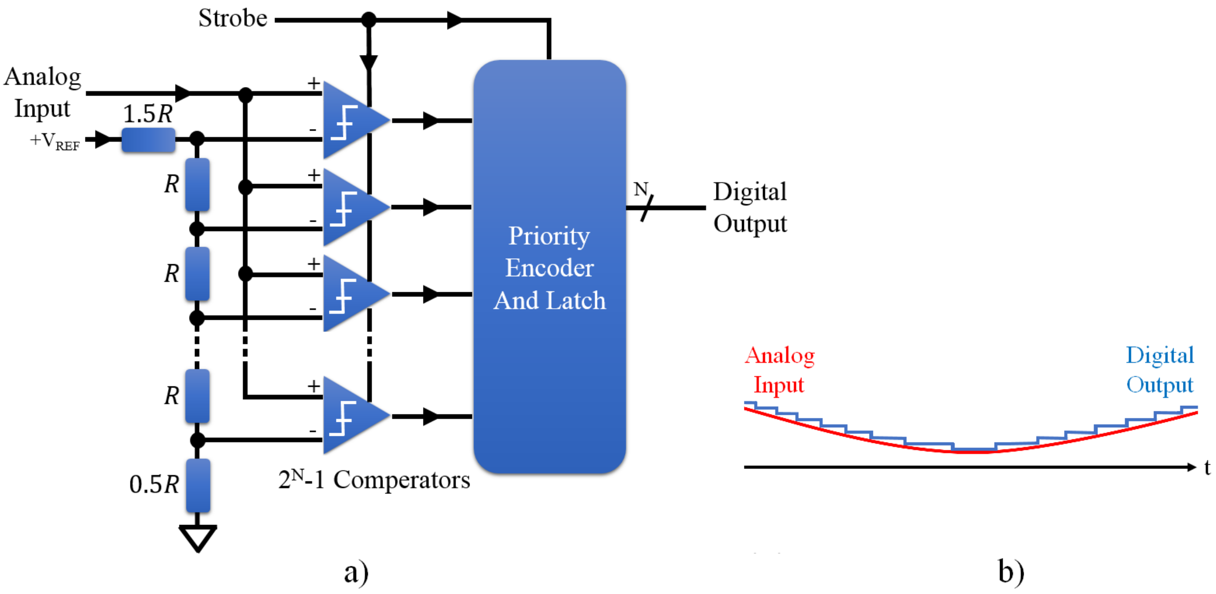


Figure 1.10: Flash ADC: a) block diagram; b) analog input signal (red color) and digital output signal (blue color)

comparators, each one comparing the input signal to reference voltage. When analog input signal voltage exceeds the reference voltage ($+V_{REF}$) at each comparator, the comparator outputs will sequentially saturate to a high state. It means that for a given input voltage, all the comparators below a certain point will have their input voltage larger than their reference voltage and a ”1” logic output, and all the comparators above that point will have input voltage lower than the reference voltage and a ”0” logic output. Since the input signal is applied to all the comparators at once, the output is delayed by one comparator delay from the input, and the encoder N-bit output by a few gate delays, so the conversion process is very fast. The outputs of comparators are connected to the priority encoder circuit, which produces an N-bit binary output, based on the highest-order active input, ignoring all other active inputs. [75] [76]

Energy Efficiency. Energy consumption is always a big consideration in flash ADCs, especially at resolutions above 8-bits. The latest research results show, that it is possible to create Flash ADC with power consumption 127 μW (180 nm technology) [77] or even 72 μW (45 nm technology) [78], however, resolution will be no more than 4-bits. If resolution is increased to

8-bits or more, the energy consumption will increase either, as it is shown in [79], where 8-bit Flash ADC consumes 690 μW (130 nm technology).

Encoding Complexity. As shown in Figure 1.10, Flash ADC architecture uses large numbers of resistors (2^N) and comparators ($2^N - 1$). It means, if there is a need for 8-bit ($N = 8$) Flash ADC, it will consist of $2^8 = 256$ resistors and $2^8 - 1 = 255$ comparators. Thereby there is a high energy dissipation due to large number of comparators and therefore relatively large (and therefore expensive) chip sizes, thus making Flash ADC encoding complex. [75] The most suitable Flash ADC, with 90,8mW power consumption, 8-bit resolution and 2 GS/s sampling rate will occupy 0.253mm^2 of a silicon die. [80]

Resolution. In practice, relatively high complexity hybrid Flash ADCs (e.g. Flash + SAR ADC) are available with up to 14 bits of resolution [81], [82], but standard Flash ADCs usually have only 3-8 bits. [75]

Sampling Rate. The maximum sampling rate is the main strength of Flash ADCs. It can be as high as 20 GS/s for low resolution (4-bit), high power consumption (15.5 mW) Flash ADCs [83] and 1.42 GS/s for 8-bit, 0.54 mW Flash ADC [84].

1.2.1.2 Pipeline Analogue-to-Digital Converter

Figure 1.11 shows the block diagram of Pipeline ADC, also called multi-step ADC. Pipeline

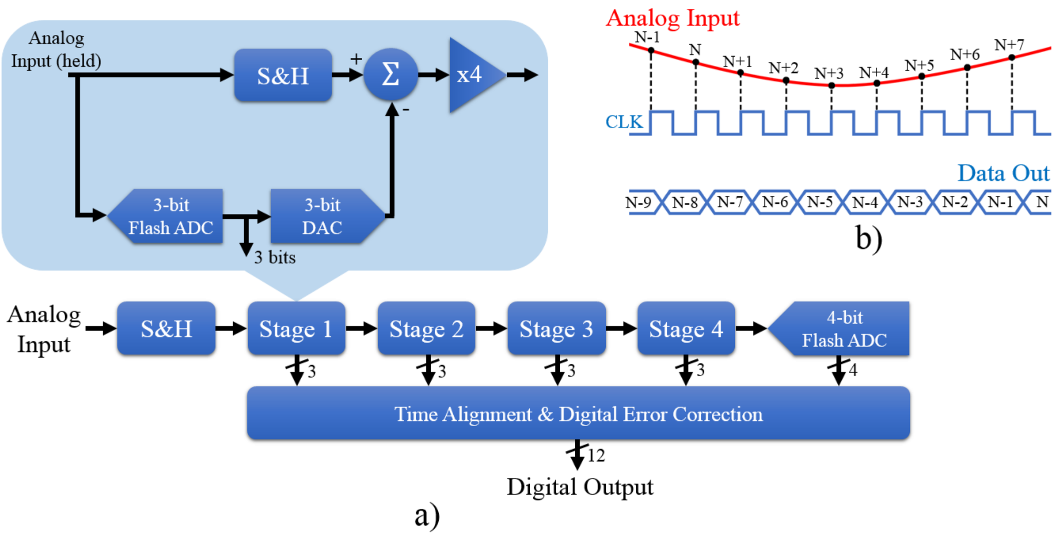


Figure 1.11: Pipeline ADC: a) block diagram; b) analog input signal (red color) and digital output data (blue color)

ADC's operation is very similar to Two-Step Flash ADC [85]. First, the analog input signal value is captured and held steady ("S&H" block), while the flash ADC in "Stage 1" quantizes it to 3 bits (see Figure 1.11).

The 3-bit Flash ADC output is then fed to a 3-bit digital-to-analogue converter (DAC) (accurate to 12 bits), and the analog output is subtracted from the analog input. The output of adder (residue) is then gained up by a factor of 4 and fed to subsequent stages. As soon as any of stages has performed its task, it can quantize the next input sample without waiting for the residue of a specific sample to reach the end of the pipeline, to complete its quantization. Therefore, the throughput of the pipeline ADC is similar to Flash ADC, but its latency is higher. The more stages there are, the higher the latency will be. Since the bits from each of the stages are determined at different time instants, they are time-aligned with shift registers before being fed to the digital-error-correction logic, where by estimating and correcting inter stage gain it is possible to enhance the resolution.[85] [86]

The main advantage of Pipeline ADC is high sampling rate (from 1 MS/s for high resolution 16-bit ADCs up to 100+ MS/s for lower resolution 8-bit ADCs) with high bandwidth, therefore Pipeline ADC covers a wide range of applications. On the other hand, parallelism increases not only throughput, but also power consumption and latency. [87]

Energy Efficiency. Pipeline ADC architecture offers one of the best performances in terms of speed and power. In such an architecture, the main power dissipation components in each of the Pipeline stages are operational amplifiers in Flash ADC and DAC (see Figure 1.11), which are consuming 65-80% of the total power. The power consumption from other components, such as clock generators, biasing circuits and digital circuitry, usually is much smaller and almost independent from the number of bits/stage. [88] There is a trade-off between the speed, resolution and power. The latest low power implementations show that it is possible to create a Pipeline ADC with power consumption as low as 1-10 mW [89], [90], [91], [92] with 50-200 MS/s and 10-bit resolution. But for preferred 12-bit resolution, low power Pipeline ADCs will consume 20-60 mW [93], [94], [95]

Encoding Complexity. As shown in Figure 1.11, Pipeline ADC has very complex architecture due to multiple "stages", where each "stage" consists of several elements, such as comparators, amplifiers, etc. If the number of stages is reduced, there will be more bits per stage and thereby more comparators, which increase the overall power consumption as well as occupied silicon die area. On the other hand, if the amount of amplifiers is lower (saves energy), accuracy is higher and latency smaller. But, if there is a large number of stages and thereby fewer bits per stage, there is a higher amount of amplifiers (which increase the overall power consumption), lower accuracy and makes larger latency. On the other hand, fewer comparators allow to reduce power consumption and complexity of the circuit significantly. Due to this trade-off, the least complex designs usually involve 3-4 bits per stage. [96] The most suitable Pipeline ADC, with $30mW$ power consumption, 12-bit resolution and 25 MS/s sampling rate will occupy $0.86mm^2$ of a silicon die. [97]

Resolution. In order to simplify the Pipeline ADC's layout and implementation, usually all ”stages” are designed with the same resolution, thereby it is possible to use the same reference voltage for all quantizers and DACs. Still, different design trade-offs may use different resolutions for each stage. Then, normally, the first stage is with a higher resolution. For high speed designs, usually lower per-stage resolution is used, but for low-power designs higher per-stage resolutions are used. [85] In practice, Pipeline ADCs have higher resolution then Flash ADC, which is up to 16-bits [98], while maintaining high sampling frequency.

Sampling Rate. In Comparison to Flash ADCs, Pipeline ADCs are almost as fast as Flash ADCs, but with much higher resolution. It can be as high as 5 GS/s for 8-bit, 150 mW Pipeline ADC [99] and 200 MS/s for 12-bit, 11.5 mW Pipeline ADC [93], [94].

1.2.1.3 Digital Ramp Analogue-to-Digital Converter

Digital Ramp ADC, also called as a stairstep-ramp ADC, has one of the simplest architectures of all ADCs. Figure 1.12 shows the block diagram of Digital Ramp ADC. In Digital Ramp ADC, the output of a free-running binary counter (CTR) is connected to the input of a Digital to Analogue Converter (DAC). At the beginning of the conversion, the CTR is set to zero and comparator's output is ”high”. Then, with each clock pulse, the CTR incrementally counts up until the output voltage of the DAC is higher than the analogue input voltage. At this moment of time, the comparator's output will switch to ”low” state and CTR will stop counting and reset to zero on the next clock pulse (see Fig. 1.12 b)). Also, at this moment, the shift register (SRG) will “load” the binary output of CTR, thus updating the circuit's digital output (see Fig. 1.12 c). As it can be seen in Fig. 1.12 b) and c), due to time which is necessary for ramp to rise, for

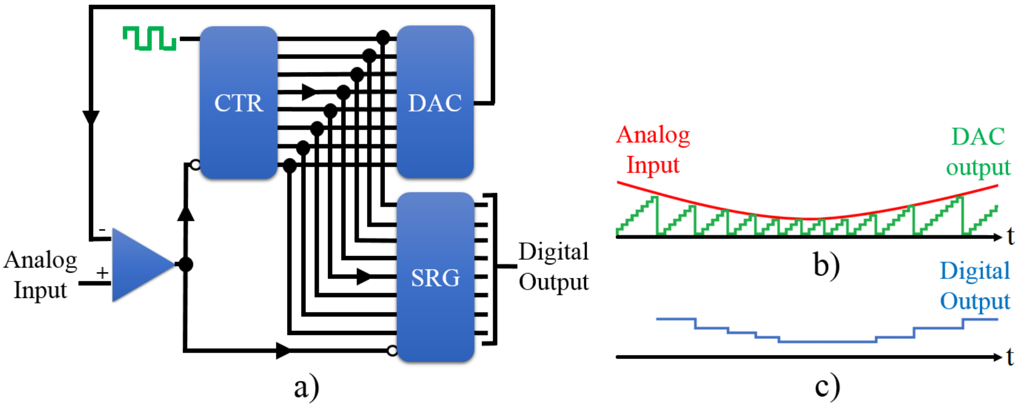


Figure 1.12: Digital Ramp ADC: a) block diagram; b) analog input (red color), DAC output (green color); and c) digital output (blue color)

higher input voltages it takes longer time to convert, therefore sampling rate is not fixed, which is unacceptable in many applications. [100]

Energy Efficiency. Even though Digital Ramp ADC has a simple architecture and thereby a potential to be very energy efficient, due to variations in sampling frequency and thereby low application range, there are very few research groups working in this field. As a result, the achieved performance isn't as high as it could be. Although Digital Ramp ADC requires only one comparison per clock cycle, the latest research results show that 12-bit Digital Ramp ADC will consume up to $3mW$ at average sampling rate of ≈1,28 MS/s [101], [102]. For lower resolution, it is possible to achieve power consumption of μW [103].

Encoding Complexity. Since Digital Ramp ADC usually does not need any calibration circuit and regardless of the resolution it has only one comparator, the Digital Ramp ADC architecture can be considered as a low complexity or even very low in most simple cases. [104] The most suitable Digital Ramp ADC, with $3.8mW$ power consumption, 12-bit resolution and 0.746 MS/s sampling rate will occupy ≈$0.017mm^2$ of a silicon die. [102]

Resolution. Current state-of-the-art Digital Ramp ADCs achieve up to 12 bit resolution and sampling rate up to 2,5 MS/s. [101], [102] The accuracy of the Digital Ramp ADC depends on the ramp generator (or the DAC) as well as the oscillator. [75]

Sampling Rate. Since Digital Ramp ADC's circuit needs to count from zero level all the way from the beginning of each count cycle, the sampling rate is not only relatively slow, but also depends on how high the input signal voltage is, i.e., for low input voltages, the sampling rate is higher, but for higher input voltages, it takes longer time to convert, which means sampling rate isn't fixed. [100] For low resolution applications (4-6 bits), Digital Ramp ADC can achieve up to 250 MS/s with power consumtion of $1mW$ [104], but for desired 12 bits, up to 2,5 MS/s ($24mW$) [105].

It is possible to avoid using a DAC by replacing it with an analog ramping circuit (integrator). Such ADC is called Slope, or integrating ADC. In this case, the reference voltage for comparator is a smooth sawtooth waveform rather than a “stairstep”. Nevertheless, Slope ADC has all the disadvantages of the digital ramp ADC, with the added drawback of calibration drift. The only advantage of this circuit is that it avoids the use of a DAC, which reduces circuit complexity. There are also dual-slope, triple-slope and even quad-slope ADCs which allow to increase the resolution and sampling rate at the cost of added complexity. [75], [100]

1.2.1.4 Tracking Analogue-to-Digital Converter

Tracking ADC is another variation of the counter-DAC-based converter (see Section 1.2.1.3 and Section 1.2.1.5). As shown in Fig. 1.13, instead of using regular ”up” counter, Tracking ADC uses an ”up/down” counter in order to drive the DAC. The ”up/down” counter is driven by the output of the comparator. When analog input signal exceeds the DAC output, the counter counts up, on the other hand, when the DAC output exceeds the analog input, the counter counts down, thus tracking the analog input signal value (see Fig. 1.13 b)). Since the counter never has to reset, the conversion speed is higher then in other ”counting” ADCs. Also, there is no need

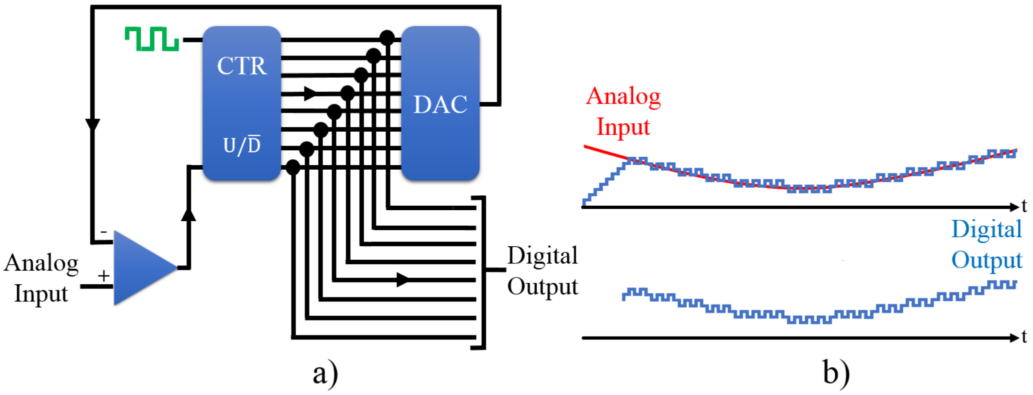


Figure 1.13: Tracking ADC: a) block diagram; b) analog input (red color) and digital output (blue color)

for a shift register and the binary output is updated with every clock. Tracking ADCs are not very common, but their advantages are valuable in synchro/resolver to digital converters, where Tracking ADCs are most often used. [75], [76], [100]

Energy Efficiency. Similarly to Digital Ramp ADC, Tracking ADC has a small application range and therefore there are very few research groups working in this field. Research results shows that it is possible to achieve 84μW power consumption for 6-bit, 50MS/s Tracking ADC and 0.49mW for 6-bit, 130MS/s Tracking ADCs [106], [107]. For higher resolution applications, the power consumption of Tracking ADC is also higher, for example, 12-bit ADC will consume from 50mW [108].

Encoding Complexity. In the most simple case, the complexity of Tracking ADC is low, however the binary output is never stable. The stability issue can be solved by a using specific shift register which increases the overall circuit complexity. Also, if the resolution is increased, the complexity will increase as well [107], [109]. Therefore, the Tracking ADC can be characterized as medium complexity. [100] The most suitable Tracking ADC, with 50mW power consumption, 12-bit resolution and 500 S/s sampling rate will occupy $\approx$0.35mm^2 of a silicon die. [109], [108]

Resolution. Currently Tracking ADCs can achieve up to 12 bits of resolution, with sampling rate up to 0.5 kS/s [108]. The resolution of Tracking ADCs is relatively low and does not exceed 12-bits. [75]

Sampling Rate. From Fig. 1.13 b), it can be seen that if the analog input signal changes slowly, the counter will be able to track it, but if there will be a sudden change in the signal, it will take thousands of clock cycles before the output is correct again. Therefore, Tracking ADCs can respond quickly to slowly changing signals, but slowly to rapidly changing signals. [75] Current research results show that for low resolution applications (up to 6-bits) it is possible to achieve up to 130 MS/s, but for desired 12-bits, only 0.5 kS/S, which is enough for EEG signal sampling. [108]

By replacing the ”up/down” counter with SAR logic, we obtain a Successive Approximation ADC (see next Section 1.2.1.5).

1.2.1.5 Successive Approximation Register Analogue-to-Digital Converter

Successive Approximation Register (SAR) ADC architecture is very similar to Digital Ramp ADC (see Fig. 1.12) and Tracking ADC (see Fig. 1.13), the only difference is a counter circuit (Successive Approximations Register), which allows to address the main disadvantages of Digital Ramp ADC (see Fig. 1.14). Figure 1.14 shows the block diagram of SAR ADC. In this case, instead of incrementally counting up in binary sequence, SAR counts by trying all values of bits starting with the most-significant bit (MSB) and finishing at the least significant bit (LSB). During the count process, when the input voltage is compared against the output of an N-bit DAC, the output of the comparator (high or low) is monitored by a register to see if the binary count is greater than or less than the analog signal input, and adjusts the bit values accordingly (see Fig. 1.14 b)). As each bit is determined, the output of the SAR corresponds to the value of analog input signal and form the basis of the SAR ADC serial output. It should also be noted that the overall SAR ADC accuracy and linearity is determined primarily by the internal DAC. [76] [75]

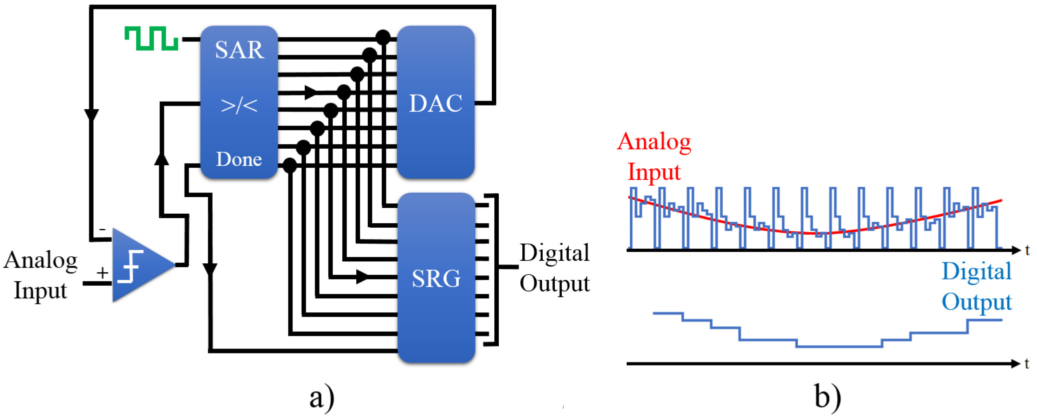


Figure 1.14: Successive Approximation Register ADC: a) block diagram; b) analog input (red color), DAC output (green color); and c) digital output (blue color)

SAR ADC is usually used for low power, medium to high resolution applications with sample rates up to 200 MS/s. The main advantages of SAR ADC are low latency, high accuracy, low power and simple architecture. Compared to Digital Ramp ADC, SAR ADC converges on the input signal much faster and it has a fixed sampling rate. Due to all mentioned SAR ADC advantages, it is one of the most popular ADCs used for bio-potential measurements. SAR ADCs are available in a wide variety of power consumptions, sampling rates, resolutions, input and output options, costs and package styles. [75]

Energy Efficiency. SAR ADCs are one of the least power consuming ADCs from all ADCs, especially, when resolution is between 4 and 10 bits, but sampling rate a few KS/s. The latest research results show that it is possible to create a SAR ADC with power consumption of impressive $717pW$ for 8-bits and 1 KS/s [110] and $3nW$ for 10-bits, 1 KS/s [111]. For higher resolution application the power consumption is also much higher, but still relatively low. For example, desired 12-bit SAR ADC will consume starting from $100nW$ (1 KS/s) [55], but 18-bit SAR ADC will consume already $30.52mW$ (5MS/s) [112].

Encoding Complexity. Although the SAR ADCs consist of one SAR, one DAC, one comparator and a logic control unit, their analog design is relatively intensive and more complex than Digital Ramp ADC, therefore it can be characterized as a medium complexity. [113] The most suitable SAR ADC, with $100nW$ power consumption, 12-bit resolution and 20 KS/s sampling rate will occupy $0.3mm^2$ of a silicon die. [55]

Resolution. Recent technological developments allow to extend the resolution of SAR ADCs up to 24 bits [114], [115], but usually SAR ADCs are in the range of 8-18 bits [116]-[117].

Sampling Rate. Recent SAR ADC design improvements have extended the sampling rate to GS/s region, but this is valid only for low resolution SAR ADCs. The most extreme case is 36 GS/s, 6-bit, 110mW SAR ADC [118], but for desired 12-bit resolution, the sampling rate is up to 36 MS/s, with power consumption of nW [55].

1.2.1.6 Sigma-Delta Analogue-to-Digital Converter

As shown in Fig. 1.15 a), in Sigma-Delta ADC, also called an oversampling converter, the analog input signal summed with the output of DAC is fed into an integrator, producing a slope, which corresponds to input magnitude. Then, in the N-bit ADC, the output of the integrator is compared to ground potential, producing high or low output, depending on whether the integrator output is positive or negative. Afterwards, the output of comparator is latched through a D-type flip-flop, which is clocked at high frequency, and the output of latch, through DAC and subtraction from the input, is fed back to the integrator, to drive it in the direction of a 0 volt output. At the end, the digital filter removes the noise, but decimator reduces the output

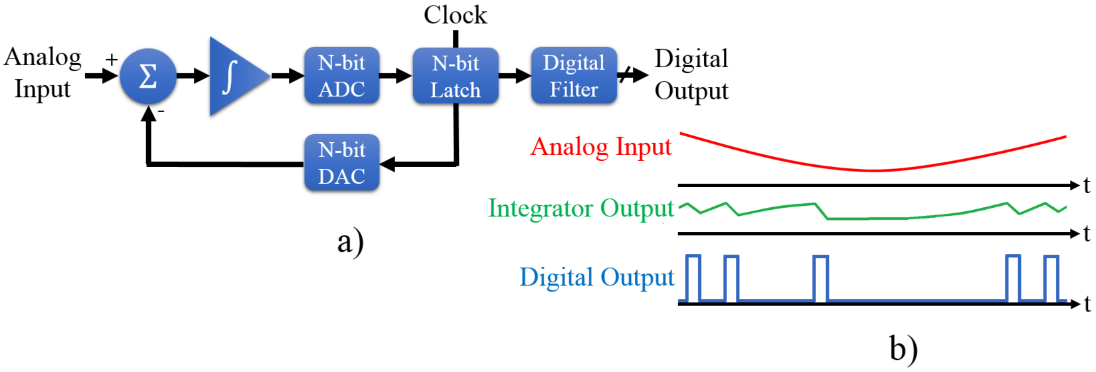


Figure 1.15: Sigma Delta ADC: a) block diagram; b) analog input (red color), integrator output (green color) and digital output (blue color)

data rate back to the Nyquist rate. The digital output of Sigma-Delta ADC is a serial bit stream. [75] [100]

Sigma-Delta ADCs are usually used for low speed, high resolution applications. Due to their advantages - high resolution and stability, low noise, power consumption and cost, Sigma Delta ADCs just as SAR ADCs are one of the most popular ADCs used for bio-potential measurement applications.

Energy Efficiency. Sigma-Delta ADCs can be considered very energy efficient, especially, when sampling rate is few KS/s or less. The latest research results show that it is possible to create a Sigma-Delta ADC with power consumption of $125nW$ for 8-bits and 64 S/s [119], but for higher resolution application the power consumption is also higher. For example, for desired 12-bits, Sigma-Delta ADC will consume starting from $600nW$ (0.5 KS/s) [120], but practical implementations show $150\mu W$ and more [121].

Encoding Complexity. As shown in Fig. 1.15, in the most simple case, Sigma-Delta ADC includes integrator, comparator, 1-bit DAC, which is in a feedback and a clock unit that provides proper timing for the modulator and digital filter. In this case, the complexity can be characterized as low, but, since multi-bit quantizers are usually used to increase the resolution, it also greatly increases the overall complexity and size of the circuit. Therefore, the complexity of Sigma-Delta ADC can be characterized as medium/high. [113] The most suitable Sigma-Delta ADC, with $13.3\mu W$ power consumption, 12-bit resolution and 512 S/s sampling rate will occupy $0.51mm^2$ of a silicon die. [57]

Resolution. Current state-of-the-art Sigma-Delta ADCs achieve up to impressive 32 bit resolution with sampling rate up to 38 KS/s [122], [123], but in this case, the ADC will consume from 20-100mW. However, there are also available Sigma-Delta ADCs with lower resolution, sampling rate and power consumption [120], [121].

Sampling Rate. The high resolution of the Sigma-Delta ADC comes at the expense of sampling rate. The fastest Sigma-Delta ADCs can achieve up to 640 MS/s [124], but energy consumption will be $\approx$500mW. Since EEG encoding demands only 200 Hz sampling frequency, there are various Sigma-Delta modulators which offer very low sampling rates (125-500 S/s) and low power consumption [121], [125], [126].

1.2.2 Asynchronous Analogue-to-Digital Conversion

Although some of synchronous ADC architectures (see Section 1.2.1) fit the defined requirements and could be used for EEG and BCI applications, most of them exhibit poor properties in terms of electromagnetic interference (EMI), complexity of circuit, sensitivity against power supply voltage, temperature and development process parameter variations, delays, etc. [21], [22].

An alternative to synchronous ADCs is asynchronous ADCs, which due to its great properties often are used for encoding of non-stationary signals. [14], [23] Since EEG signals can be classified as a wide dynamic range signals, non-uniform sampling methods have a great potential to improve energy efficiency, reduce complexity of the encoding and avoid unnecessary EMI in brain computer interfaces (BCI) [23].

The most popular non-uniform sampling methods found in literature are Zero crossing ADC, Sine-wave crossing ADC, Level crossing ADC, Send-on-Delta ADC, Asynchronous Sigma-Delta modulator and Peak Sampling ADC.

Therefore, in this section, an overview of mentioned ADCs as well as analysis in relation to set out requirements (energy efficiency, encoding complexity, resolution and sampling frequency) is given.

1.2.2.1 Zero Crossing Analogue-to-Digital Converter

Reference crossing techniques are one type of asynchronous encoding, where Zero Crossing ADC is the simplest of all. Figure 1.16 shows the block diagram of Zero Crossing ADC. In general, it detects the transition of an analog input signal waveform from positive to negative and negative to positive. When such transition occurs, there is a change in the output of comparator that coincides with the zero voltage condition. The output is a square wave. [127]

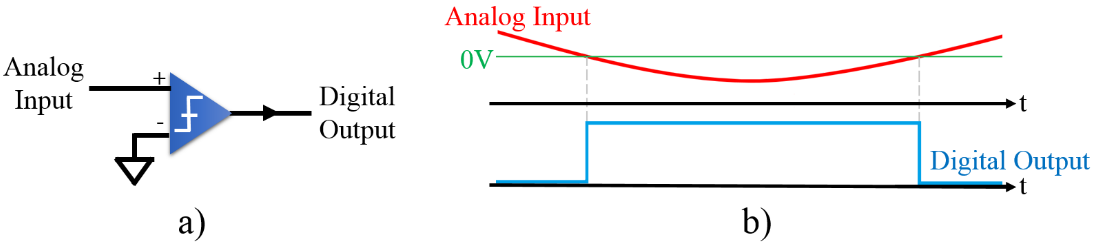


Figure 1.16: Zero-Crossing ADC a) block diagram; b) analog input (red color), zero voltage reference (green color) and digital output (blue color)

Although Zero Crossing ADC's block diagram appears to be very simple, in fact it is quite a challenge to create a Zero Crossing ADC for high frequency applications. For instance, if high accuracy is needed, already 1 kHz signal starts to present a real challenge for the comparator. Essentially, the comparator must have low input offset, high slew rate and speed. In practice, most of the comparators do not switch at exactly 0 V, and usually have asymmetrical dead band. Only with several modifications it is possible to obtain true 0V detection, which increase the complexity of the circuit. Also, supply voltage is quite critical factor for the comparator, the higher it is, the further the output voltage has to swing. In addition, high speed comparators will consume more power. [127], [128]

Even though Zero Crossing ADC has very low complexity and thus the energy consumption, it can't be used for EEG signal encoding, since one level crossings are not enough to ensure Nyquist step, i.e., the distance between two consecutive level crossings (time instants) will exceed the Nyquist step and thus the signal will not be recoverable. [129] Therefore Zero Crossing ADC will not be analyzed with regards to Energy Efficiency, Encoding Complexity, Resolution and Sampling Rate requirements.

1.2.2.2 Sine Wave Crossing Analogue-to-Digital Converter

Since it is not possible to reconstruct EEG signal from zero crossings (see Section 1.2.2.1), choosing an appropriate reference signal(s) (function(s)) is vital for effective implementation of ADC. One such reference signal is sinusoidal function, which can be easily generated, stabilized and used for input signal reconstruction. [19]

As shown in Fig. 1.17, in the most simple case, Sine Wave Crossing (SWC) ADC consist of a sine wave generator and a comparator, where analog input signal is compared with a sine wave reference signal. Digital output is ”1”, when the analog input signal exceeds the reference signal and ”0” when it does not. It should be noted that sine-wave amplitude must be higher than amplitude of the input signal. By knowing the reference signal and time instants of the digital output rising and falling edges, it is possible to reconstruct the original input signal. Since the comparator is a core element of the SWC ADC, it also determines performance of the SWC ADC. Also, the quality of the generated sine wave can effect the SWC ADC performance [19], [130]

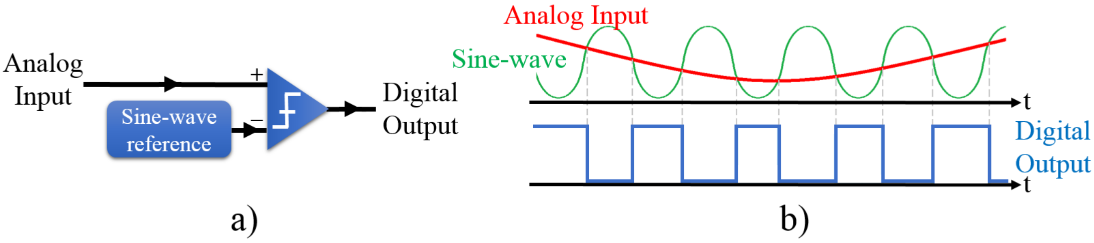


Figure 1.17: Sine Wave Crossing ADC a) block diagram; b) analog input (red color), sine wave reference signal (green color) and digital output (blue color)

Energy Efficiency. At the time of writing this thesis, exact calculations of SWC ADC energy consumption are not found in the literature. Therefore, a rough estimation of power consumption of a simple SWC ADC was performed, by analyzing SWC ADC’s circuit in Fig. 1.17 a).

Sine-wave signal can be generated by dynamic programming of DAC, Phase-Locked-Loop (PLL)-based technique [131], Direct Digital Synthesis (DDS) chips [132] or by simple circuit comprising two operational amplifiers (e.g., dual OpAmp chip LPV542, which consumes $\approx 882nW$) and few passive elements. The comparator, according to [19] must ensure a response time of at least $1\mu S$, in order to ensure 12-bit resolution. The most energy efficient comparator, which meets this requirement is TLV3201 [133], which consumes $\approx 108\mu W$. So, in the simplest case, with the mentioned components, it is assumed that SWC ADC would consume ≈110 μW.

Encoding Complexity. As the SWC ADC consists of two operational amplifiers, one comparator and few passive elements, it can be characterized as very low complexity circuit, even though physical implementation can not be found in the literature.

Resolution.The resolution mostly depends on the duration of the time slots, within which the comparator compares the values of input signal and reference signal. As shown in [19], in order to ensure necessary 12-bit, the comparator must ensure a response time of at least $1 \mu S$. [19]

Sampling Rate. The maximum sampling rate is highly dependent on sine-wave reference signal frequency. In the available literature it can be found that with several modifications, the average sampling rate can be achieved as high as 60 MS/s. [19]

1.2.2.3 Level Crossing Analogue-to-Digital Converter

As shown in Section 1.2.2.1, it is not possible to reconstruct EEG signal from zero reference signal level crossings. In order to be able to reconstruct it, several threshold (quantization) levels can be used. Figure 1.18 shows the block diagram of Level Crossing ADC. Such Level Crossing (LC) ADC approach usually consists of a comparator, up/down counter, n-bit DAC (which defines quantization levels), timer and control logic. In LC ADC, samples are generated only when the analog input signal crosses the predefined quantization levels, while the time in

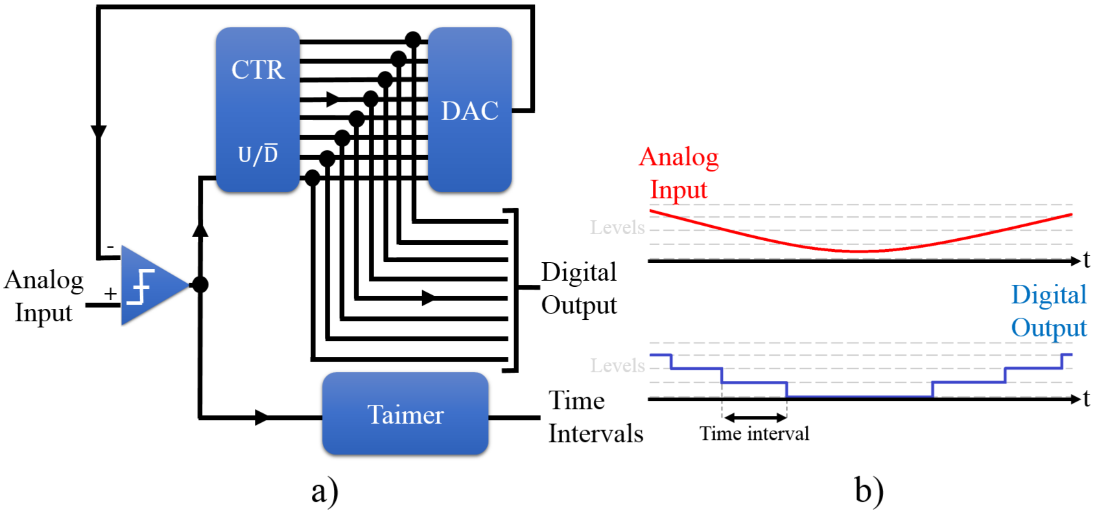


Figure 1.18: Level Crossing ADC a) block diagram; b) analog input (red color), quantization levels (gray color) and digital output (blue color)

between two consecutive samples is measured by a timer.The conversion results are composed of digital codes for the voltage magnitude and the time intervals of the samples. [134]

The main advantages of LC ADC are: low sampling rate (in some cases even below Nyquist [135]), especially for low frequency and low amplitude signals (i.e. EEG), and thus lower power consumption; low quantization noise floor (signal to noise plus distortion ratio (SNDR) exceeds the theoretical limit of conventional systems with the same resolution in amplitude); low EMI emission (due to absence of clock); and other (see [134], [136]).

Energy Efficiency. In LC ADC, the DAC and the comparator consume most of the power. The latest research results show that it is possible to create a LC ADC with power consumption of $582nW$ for 6-bits and 3.3kHz input bandwidth [134], $3\mu W$ for 8-bits and 20kHz input bandwidth [137], but for desired 12-bits, LC ADC will consume starting from $1.7mW$ with 144kHz input bandwidth [138], [139].

Encoding Complexity. As the LC ADC architecture is not driven by any global clock, but only by the analog input itself, it allows to reduce not only power consumption, but also complexity, die area, and electromagnetic emissions. [138] Therefore, the complexity of the LC ADC circuit can be characterized as low/medium. The most suitable LC ADC, with $1.7mW$ power consumption, 12-bit resolution and 144kHz (equivalent to ≈288kS/s) input bandwidth sampling rate will occupy $\approx 0.3mm^2$ of a silicon die. [138]

Resolution. The resolution in LC ADC mostly depends on the DAC and the comparator. Current state-of-the-art low energy LC ADCs achieve up to 12-bits with 144kHz input bandwidth. [138]

Sampling Rate. For 12-bits, the input bandwidth up to 10MHz (or ≈20 MS/s sampling rate, if roughly converted) can be achieved [138], [139].

In order to increase the noise resistance and reduce the number of samples, it is possible to use a LC ADC modification, called Send-on-Delta (SoD) ADC. In SoD ADC, the sampling is triggered if the signal deviates by predefined step value ”delta” from the value referred to the most recent sample. Thus, the sample is not triggered until the analog input signal remains within a certain interval of delta. [140], [141], [142] On the other hand, fewer samples mean more complicated signal reconstruction.

1.2.2.4 Asynchronous Sigma-Delta Modulator

Asynchronous Sigma-Delta Modulator (ASDM) is a Time Encoding Machine (TEM) which transforms the amplitude information of the signal into time information or time codes without the quantization error that exists in the clocked converters. Figure 1.19 shows the block diagram of ASDM. The ASDM consists of an integrator, a non-inverting Schmitt and a negative-feedback, where κ, δ and b are circuit parameters. [143]

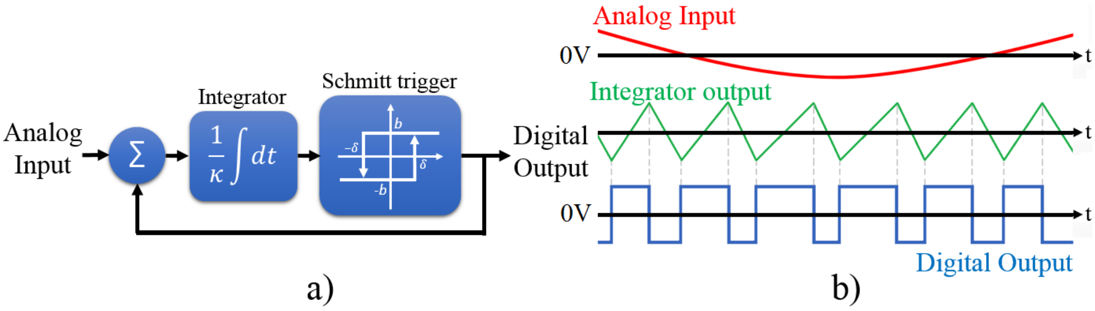


Figure 1.19: Asynchronous Sigma-Delta modulator a) block diagram; b) analog input (red color), integrator output (green color) and digital output (blue color)

Energy Efficiency. ASDM is one of the less power consuming ADCs. The latest research results show that it is possible to create an ASDM with power consumption of only $7.5nW$ for 8-bits and 250Hz input signal bandwidth [25], [144]. Knowing that ASDM frequency doubling increases the SNR by 1-bit (see (2.15) and (2.16)), in order to achieve desired 12-bits, it is necessary to increase ASDM frequency 16 times. As shown in [145], if ASDM frequency is increased 20 times, the overall ASDM power consumption will increase only by 25%, which means in [25] case, the proposed circuit for 12-bits will consume not more than 9,375nW.

Encoding Complexity. In ASDM, the complexity of the system is moved from the encoding part to the decoding part, therefore the modulator itself is very simple and can be implemented with very few elements. Therefore its complexity can be characterized as very low. For example, as shown in [146], ASDM circuit with 12-bit resolution can be implemented in $0.026mm^2$ area of a silicon die.

Resolution. ASDM usually is used for low resolution applications, but latest research results show, that it is possible to achieve up to 13-bit resolution with 2MHz input bandwidth [146]. The resolution highly depends on precision of the trigger (jitter, slew rate) as well as on time-to-digital (TDC) converter.

Sampling Rate. The fastest ASDMs can encode signals up to 12MHz (equivalent to ≈24 MS/s) [147], but then power consumption will be high. Since this requirement demands only 400 S/s sampling rate, there are various ASDMs which offer very low sampling rates (60-500 S/s) with lower power consumption [148], [25], [149].

1.2.2.5 Peak Sampling Analogue-to-Digital Converter

Peak Sampling ADC, also called Min-Max sampling ADC, captures a sample every time an analog input signal reaches its local maximum or minimum value (see Fig. 1.20). The main

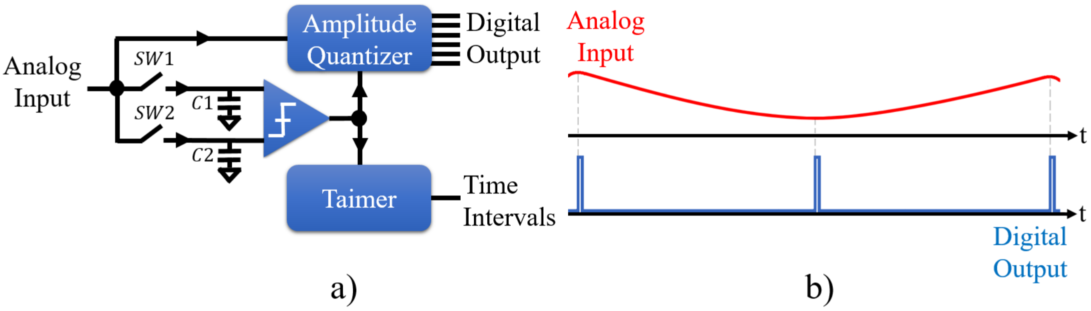


Figure 1.20: Peak Sampling ADC a) block diagram; b) analog input (red color) and digital output (blue color)

Peak sampling ADC advantage is that sampling density depends only on the input signal (in LC ADC case, it also depends on the placement of the levels). On the other hand, compared to uniform sampling or LC ADC, Peak Sampling ADC has lower signal to noise (SNR) ratio. Also, both amplitude and time instants must be quantized, therefore the number of bits per sample is larger. [150], [151], [152]

At the time of writing this thesis, only one physical implementation of Peak Sampling ADC has been presented [152], therefore the estimation of Peak Sampling ADC is very limited.

Energy Efficiency. In Peak Sampling ADC, the power consumption highly depends on the activity of input signal. For lower ratio of peaks, Peak Sampling ADC is more energy efficient. As shown in [152], it is possible to reduce SAR ADC power consumption by up to 18%, if the ratio of peaks does not exceed 10% or up to 9%, if ratio of peaks does not exceed 20%. Even though particular implementation consumes ≈15μW, it is clear, that for 7.5nW SAR ADC (see

1.2.1.5) it would be also possible to reduce the energy consumption for $\approx$ 18%, resulting in $\approx 6nW$ power consumption. [152]

Encoding Complexity. Since the overall architecture includes Peak Sampling circuit as well as SAR ADC circuit, based on previous consideration about SAR ADC, the overall circuit complexity can be characterized as medium/high. [152] The most suitable Peak Sampling ADC, with $15\mu W$ power consumption, 8-bit resolution and 1.25 MS/s sampling rate will occupy $0.23mm^2$ of a silicon die. [152]

Resolution. In [152] implementation, an 8-bit SAR ADC is used as a part of the system. But, since the resolution depends on the selected SAR ADC, by choosing appropriate SAR ADC for Peak Sampling ADC, theoretically up to 24-bit resolution can be ensured.

Sampling Rate. As shown in [152], the maximum conversion rate of Peak sampling ADC is $\approx$1.25 MS/s, which is mainly limited by the internal DAC delay in the conversion circuitry. But, if SAR ADC with higher sampling rate (e.g. 36 GS/s) is used (see 1.2.1.5), the proposed circuit in [152], theoretically could ensure 10% of this sampling rate, i.e. $\approx$3.6 GS/s.

1.3 Summary and Conclusions

In Section 1.1, the origin, properties and measurement techniques of an electroencephalogram (EEG) signals were described and analyzed. Human brain has more than 100 billion neurons which conduct electrical impulses and are the core components of the nervous system. These electrical activities can be measured by non-invasive methods from scalp by using electrodes, and the measured signals are called EEG signals. Since human's brains primarily operates in five brain rhythms: Delta (0.2-3.9Hz), Theta (4-7.9Hz), Alpha (8-12.9Hz), Beta (13-40 Hz) and Gamma (40+ Hz), all these components together are forming the EEG signal. Typically the amplitude of the measured EEG signal from scalp is between 0.5 and $200\mu V$.

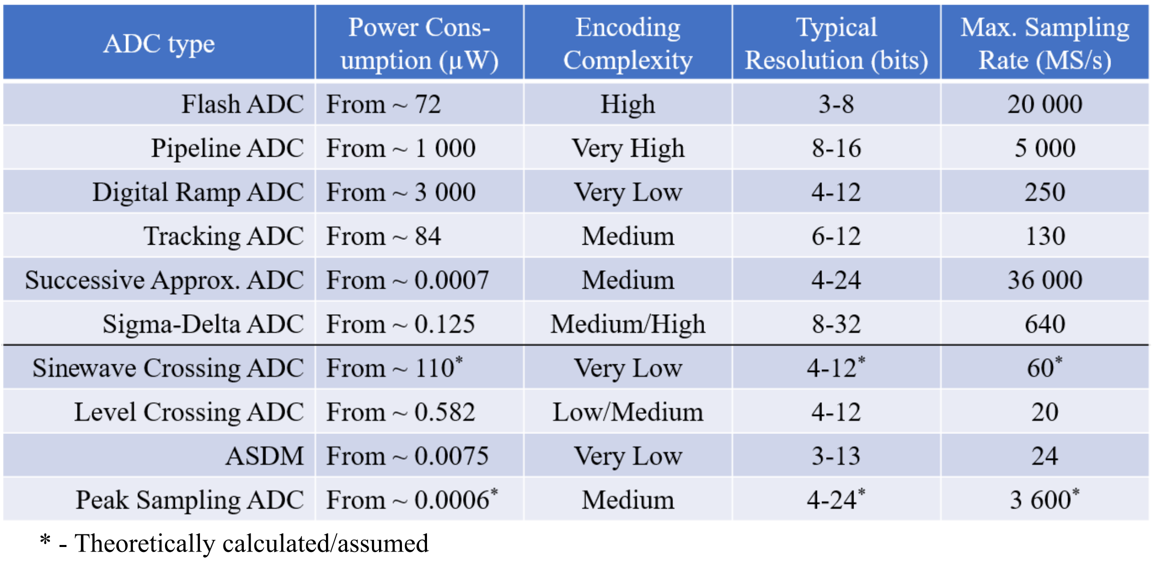
By using EEG signals, it is possible to create so called Brain Computer Interface (BCI), which can be used to control other electrical or electro-mechanical devices by ”thoughts”. If simplified, usually modern BCI consists of an electrode (as a sensor), an amplifier, an analog-to-digital converter (ADC), a transmitter, a receiver and a computer (for signal reconstruction and processing).

Since a recent trend is to develop wireless BCI devices, management of energy consumption becomes a crucial factor. As shown before, one very important part of all BCI systems is ADC, where power consumption can be significantly reduced.

In order to select the most suitable ADC for BCI, in Section 1.2, ten different synchronous and asynchronous ADCs were described and analyzed against requirements/criteria, which were set out based on the EEG signal properties and the current situation in the field of the BCI systems. These four main criteria are: Energy Efficiency (the less consumption, the better), Encoding Complexity (the smaller area of silicon die occupied, the better), Resolution (at least 12-bits) and Sampling Rate (at least 400 S/s).

The overall capabilities of all ADCs, which were analyzed against the criteria in Section 1.2.1 and Section 1.2.2, are summarized in Table 1.1, where for better visual comparison ”Max. Sampling Rate” column for asynchronous ADCs is obtained by converting maximum input bandwidth accordingly 1Hz≈2S/s. For example, for Flash ADC, the least power consuming Flash ADC circuit found in the literature consumes $72\mu W$, but then its resolution is only 4-bits, but sampling rate 4.2 GS/s. Still, the table shows that it is possible to create Flash ADC with higher resolution (e.g. 8-bits), but then other circuit parameters are affected, such as power consumption and maximum sampling rate, and vice versa, if there is a Flash ADC circuit with very high sampling rate (e.g. 20 GS/s), it will not be able to ensure $72\mu W$ power consumption as well as 8-bit resolution.

Table 1.1: Overall capabilities of different ADCs

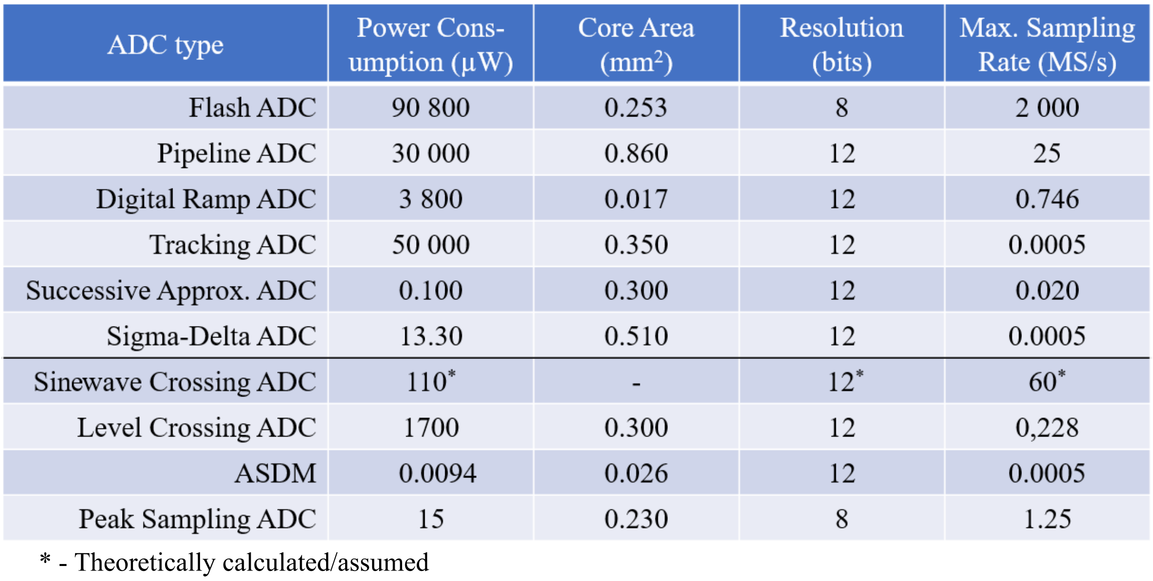


ADC type	Power Consumption (µW)	Encoding Complexity	Typical Resolution (bits)	Max. Sampling Rate (MS/s)
Flash ADC	From ~ 72	High	3-8	20 000
Pipeline ADC	From ~ 1 000	Very High	8-16	5 000
Digital Ramp ADC	From ~ 3 000	Very Low	4-12	250
Tracking ADC	From ~ 84	Medium	6-12	130
Successive Approx. ADC	From ~ 0.0007	Medium	4-24	36 000
Sigma-Delta ADC	From ~ 0.125	Medium/High	8-32	640
Sinewave Crossing ADC	From ~ 110*	Very Low	4-12*	60*
Level Crossing ADC	From ~ 0.582	Low/Medium	4-12	20
ASDM	From ~ 0.0075	Very Low	3-13	24
Peak Sampling ADC	From ~ 0.0006*	Medium	4-24*	3 600*

* - Theoretically calculated/assumed

As shown above, there are many trade-offs for each of the ADCs. In order to select the most suitable ADC for EEG signal encoding and BCI as whole, from each of the analyzed ADC types, the most suitable (against criteria) ADC was selected for further comparison. The final comparison is shown in Table 1.2.

Table 1.2: Comparison of different types of ADCs, where in each type the most suitable ADC for EEG/BCI application is selected, based on the set out criteria



ADC type	Power Consumption (µW)	Core Area (mm^2)	Resolution (bits)	Max. Sampling Rate (MS/s)
Flash ADC	90 800	0.253	8	2 000
Pipeline ADC	30 000	0.860	12	25
Digital Ramp ADC	3 800	0.017	12	0.746
Tracking ADC	50 000	0.350	12	0.0005
Successive Approx. ADC	0.100	0.300	12	0.020
Sigma-Delta ADC	13.30	0.510	12	0.0005
Sinewave Crossing ADC	110*	-	12*	60*
Level Crossing ADC	1700	0.300	12	0,228
ASDM	0.0094	0.026	12	0.0005
Peak Sampling ADC	15	0.230	8	1.25

* - Theoretically calculated/assumed

From analysis in Section 1.2, and overview in Table 1.2, it is possible to conclude that:

Flash ADC is not suitable for EEG signal encoding, since it does not meet the required resolution (12-bits). Also, the power consumption (98 000 μW) as well as complexity of the circuit (0.253mm^2) is high. In some special cases, mentioned in Section 1.2.1.1, the resolution can be increased, but then, the power consumption will be even higher. The complexity and power consumption can be slightly reduced by using two-step Flash ADC, but then there is an increase in latency. Flash ADC is designed for low resolution and high sampling rate applications, and for BCI applications there is no need for GS/s sampling rate, since the maximum EEG signal frequency is 200Hz;

Although **Pipeline ADC** meets the requirements of resolution and sampling rate, the power consumption is high (30 000 μW) and the complexity of the circuit is very high (0.860mm^2). Pipeline ADC could be used for EEG signal encoding, but it is clear that it is not the best option for BCI, where energy efficiency and small size play crucial role;

Digital Ramp ADC can be implemented in a very small dimensions on the silicon die (0.017mm^2), which makes it very attractive for BCI applications. On the other hand, although it can ensure the necessary resolution and sampling rate, the power consumption of the circuit is high (3 800 μW), which makes it less attractive for BCI applications;

Tracking ADC is able to ensure the necessary resolution and sampling rate, but the power consumption (50 000 μW) as well as complexity of the circuit (0.350mm^2) is too high for BCI applications;

SAR ADC is one of the best choices for EEG signal encoding and BCI applications as whole, since it can ensure not only the necessary resolution and sampling rate, but also very low power consumption (0.100 μW), despite the fact that complexity of the circuit is relatively high (0.300mm^2).

Sigma Delta ADC as SAR ADC is very attractive for EEG signal encoding and BCI application as whole. It offers high resolution (up to 32-bits) and low power consumption (13.3 μW for 12-bits) and proper sampling rate. The only disadvantage is the complexity of the circuit, which occupies more space on silicon die (0.510mm^2);

Zero ADC can not be used for EEG signal encoding, since it can't meet the required resolution, even though it has very simple architecture and potentially very low power consumption;

Sinewave ADC could be used for EEG signal encoding, due to its low complexity circuit and potentially low power consumption, but at the time of writing this thesis, no physical implementations could be found, therefore it was not possible to estimate its true performance;

Level Crossing ADC can be used for EEG signal encoding and BCI application as whole, since it offers simple circuit and low power consumption (from 582nW). But this is valid only for low resolution applications. For desired 12-bits, both the complexity of the circuit (0.300mm^2) as well as power consumption (9 μW) increase and make it less attractive for BCI applications;

ASDM exhibit excellent properties for EEG signal encoding and BCI application as whole. It offers very low power consumption (from 9.4nW) and complexity of the circuit (0.026mm^2), while preserving necessary resolution and sampling rate. The only ASDM disadvantage is more complex and resource demanding signal reconstruction;

Peak Sampling ADC, basically, is an energy efficient modification of SAR ADC and potentially could be used for EEG signal encoding. But, since at the time of writing this thesis, there was only one physical implementation of Peak Sampling ADC (with only 8-bits), it was not possible to estimate its true performance, even though theoretically, if better SAR ADC would be used in this circuit, it could offer very attractive and similar to SAR ADC properties.

By taking into account analysis and conclusions made above, SAR ADC and ASDM exhibit the most suitable properties for EEG signal Encoding and BCI application as whole. In this case, by considering parameters of both ADCs in Table 1.2 and advantages of asynchronous over synchronous designs, such as [21], [22], [153]:

- lower energy consumption;
- lower electromagnetic emissions (asynchronous circuits do not emit radiation at the clock frequency and harmonic frequencies);
- lower sensitivity against power supply voltage, temperature and development process parameter variations;
- lower delays (by avoiding the wait until the next clock edge, asynchronous circuits exhibit better average-case performance rather than worst-case performance);
- lower complexity (easier and cheaper to realize);
- better modularity and interlinking between individual circuit units;
- higher potential for energy efficiency improvement, if used for wide dynamic range signals (as EEG signals);
- absence of overheads and problems associated with distributing clock signals,

an ASDM method is chosen for further in depth analysis, research and improvements.

Although ASDM has many advantages, the use of asynchronous systems in real life applications are limited due to incompatibility with classical (synchronous) systems, which results in fewer research studies in this area, thus slowing down its development. Therefore, the author of this work believes that it is very important to understand the true potential of asynchronous systems and will dedicate himself to carry out the research in the field of ASDM.

2. ASYNCHRONOUS SIGMA-DELTA MODULATOR

From the analysis carried out in Section 1.2 and conclusions made in Section 1.3, it is clear that Asynchronous Sigma-Delta modulator (ASDM) can be used for electroencephalogram (EEG) signal encoding and has a huge potential to reduce energy consumption and thus prolongs the operation time of the battery in wireless Brain Computer Interface (BCI) systems. Therefore, the main purpose of this section is to carry out an in depth analysis of ASDM to understand its main advantages and disadvantages. At the end of this section the summary and conclusions are given.

2.1 Signal Encoding

Asynchronous Sigma-Delta modulator (ASDM) belongs to Time Encoding Machine (TEM) class, since amplitude information is converted into time information or time sequence. An example of basic zero crossing TEM is depicted in Figure 2.1. In this example, the difference $x(t){-}f(t)$ is passed through a comparator where $f(t)$ is a periodic ramp function. Since the comparator switches at $t = t_k$ when $x(t_k) = f(t_k)$ holds, the time sequence t_k, which corresponds to the comparator output $z(t)$ represents the input signal $x(t)$. The function $f(t)$ can also be chosen to be constant $f(t) = 0$ [128] or sinusoid $f(t) = A_0 sin(2\pi f_0 t)$ [154], where $A_0 \geq |x(t)|$ and f_0 exceeds the maximum frequency of the input signal $x(t)$. [19] [143] [155]

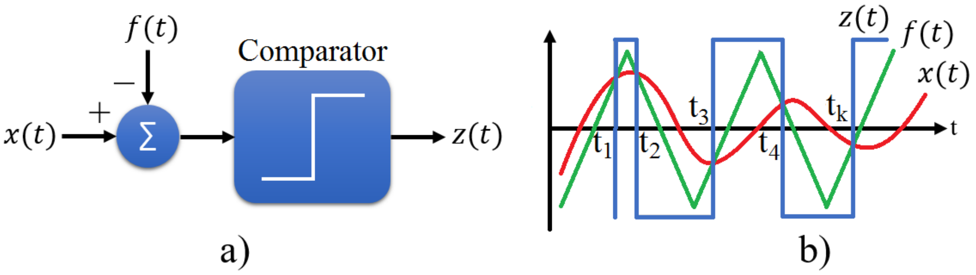


Figure 2.1: a) Time Encoding Machine (TEM) block diagram b) TEM input signal - $x(t)$, known function - $f(t)$ and comparator output - $z(t)$ [143]

ASDM is slightly more complicated then the simplest TEM and the block diagram of it is shown in Figure 2.2. It consists of an integrator with parameter κ, a non-inverting Schmitt trigger with parameters δ and b and a negative feedback. The values of these parameters determine the average switching rate of the trigger. [143] [156]

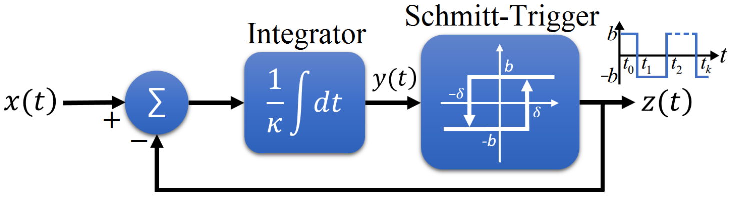


Figure 2.2: Asynchronous sigma-delta modulator (ASDM) block diagram [143] **[26]**

The input signal $x(t)$ of ASDM is bounded in amplitude as [156]:

$$|x(t)| \leq c < b \tag{2.1}$$

Since trigger output $z(t)$ has either b or $-b$ value, the integrator input is either $x(t)+b$ or $x(t)-b$. From (2.1) it follows the integrator output $y(t)$ is strictly increasing or decreasing function for $t \in [t_k, t_{k+1}]$, and

$$y(t_k) = (-1)^k \delta \tag{2.2}$$

The relationship between the binary output $z(t)$ and the input signal $x(t)$ of the ASDM for $t_{k+1} > t_k$, and integers $k \in Z$, is given by the integral equation [157]

$$\int_{t_k}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - b(t_{k+1} - t_k)] \tag{2.3}$$

Due to (2.1) the distances between consecutive triggering points t_k and t_{k+1} are bounded [158]

$$\tau_{min} = \frac{2\kappa\delta}{b+c} \leq t_{k+1} - t_k \leq \frac{2\kappa\delta}{b-c} = \tau_{max} \tag{2.4}$$

If the input $x(t)$ of the ASDM is bounded as (2.1), the output of the integrator at time $t_{k+1} > t_k$ is [157]

$$y(t_{k+1}) = y(t_k) + \frac{1}{k}\int_{t_k}^{t_{k+1}} [x(t) - z(t)]dt \tag{2.5}$$

If the state of the Schmitt trigger is $(-b, -\delta)$ at $t = t_k$, ($y(t_k) = -\delta$ and $z(t_k) = -b$), at some time $t_{k+1} > t_k$ we have that [157]

$$\delta = -\delta + \frac{1}{k}\int_{t_k}^{t_{k+1}} [x(t) + b)]dt = -\delta + \frac{1}{k}\int_{t_k}^{t_{k+1}} x(t)dt + b(t_{k+1} - t_k) \tag{2.6}$$

Right after t_{k+1}, the trigger switches to a (b, δ) state so that for some time $t_{k+2} > t_{k+1}$ [157]

$$-\delta = \delta + \frac{1}{k}\int_{t_{k+1}}^{t_{k+2}} [x(t) + b)]dt = \delta + \frac{1}{k}\int_{t_{k+1}}^{t_{k+2}} x(t)dt - b(t_{k+2} - t_{k+1}) \tag{2.7}$$

which when added with (2.6) gives

$$\int_{t_k}^{t_{k+2}} x(t)dt = (t_{k+2} - 2t_{k+1} + t_k) \cdot \kappa \cdot b = (\chi_k - \psi_k) \cdot \kappa \cdot b, \tag{2.8}$$

where ψ_k and χ_k are defined below. [157]

The output of the ASDM consists of a sequence of binary rectangular pulses (see Fig. 2.3), where duration of two consecutive pulses can be defined as $T_k = \psi_k + \chi_k$. If ψ_k is the duration of the pulse of amplitude 1 and χ_k the duration of the other pulse of amplitude -1, for $x(t)$, $t_k \leq t \leq t_{k+2}$, the duty-cycle $\frac{\psi_k}{T_k}$ can be expressed from (2.8):

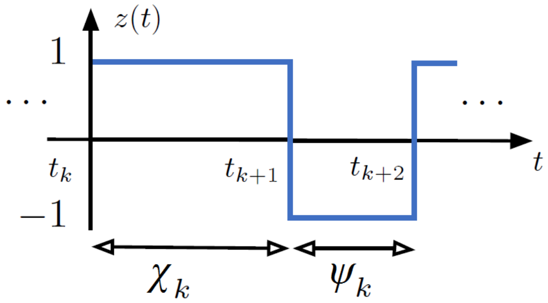


Figure 2.3: The output of ASDM [157]

$$0 < \frac{\psi_k}{T_k} = \frac{1 + \bar{x}(t)|_{t\in[t_k,t_{k+2}]}}{2} < 1, \tag{2.9}$$

where $\bar{x}(t)|_{t\in[t_k,t_{k+2}]} = \frac{1}{T_k}\int_{t_k}^{t_{k+2}} x(t)dt$ and the values of κ and b are chosen to be $\kappa = b = 1$.

If the input signal $x(t) = 0$ for all times ($-\infty < t < \infty$), then $\psi_k = \chi_k$ or $\psi_k/T_k = 0.5$. But if the input signal $x(t) = A$, where $|A| < 1$ and $t_k \leq t \leq t_{k+2}$, then [157]

$$\psi_k = \frac{(1 + A)T_k}{2} \tag{2.10}$$

and

$$\chi_k = T_k - \psi_k. \tag{2.11}$$

If the input signal $x(t)$ is not constant in a time segment, the duty-cycle is not clearly defined with respect to the amplitude. [157]

From the above it follows that the local average of the input signal $\bar{x}(t)|_{t\in[t_k,t_{k+2}]}$ can be obtained from the ψ_k and χ_k in the duty cycle modulation, i.e., [157]

$$\bar{x}(t)|_{t\in[t_k,t_{k+2}]} = \frac{\psi_k - \chi_k}{\psi_k + \chi_k} \tag{2.12}$$

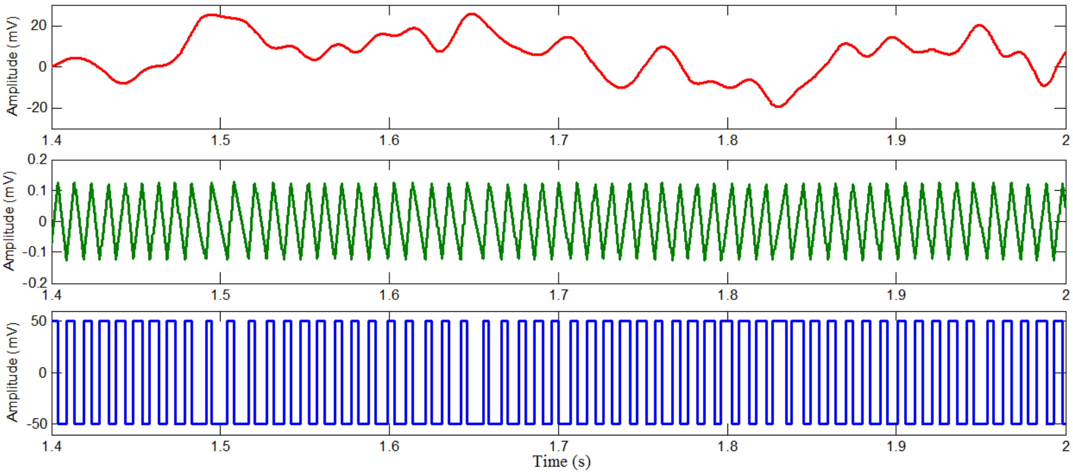


Figure 2.4: Operation of ASDM. EEG signal (red line) and corresponding ASDM integrator output $y(t)$ (green line) and ASDM trigger output $z(t)$ (blue line)

Figure 2.4 shows the operation of ASDM on an Electroencephalogram signal.

In order to calculate a signal to noise ratio (SNR) of the ASDM, first, the signal and noise power must be determined. The power of a sinusoidal signal is given by: [159]

$$P_s = \frac{1}{2}A_0^2, \tag{2.13}$$

where A_0 is the amplitude (voltage) of the input signal. The total noise power within bandwidth $2F_{max}$ is given by: [159]

$$P_N = \frac{8}{3}(1 - \frac{A_0^2}{2})\frac{1}{\bar{T}}T_{res}^2 2F_{max}, \tag{2.14}$$

where $\bar{T}$ is a constant mean value of T_k, but T_{res} denotes the resolution of the Time to Digital Converter (TDC), which is a device for recognizing events and providing a digital representation of the time they occurred. Assuming that $A_0 = 1$, the signal to noise ratio of ASDM can be calculated as follows: [159]

$$SNR = 10log_{10}\left(\frac{P_s}{P_N}\right) = 10log_{10}\left(\frac{3}{8}\frac{\bar{T}}{2T_{res}^2 F_{max}}\right) = 10log_{10}\frac{3}{8}\frac{\bar{T}}{2F_{max}} + 20log_{10}f_{res}, \tag{2.15}$$

where $f_{res} = \frac{1}{T_{res}}$.

By knowing SNR, it is possible to calculate Effective Number Of Bits (ENOB) of an ideal ASDM: [160]

$$ENOB = \frac{SNR - 1,76}{6,02}. \tag{2.16}$$

From (2.15) and (2.16) it follows that by doubling the frequency f_{res} of TDC, SNR and ENOB of ASDM increase by 6 dB and 1 bit, respectively.

2.2 Signal Recovery

In order to reconstruct the original signal encoded by ASDM, in this subsection, typical ASDM signal recovery algorithm as well as fast and real-time signal recovery algorithms are described.

The classical (uniform) sampling theorem by Claude Shannon states that: If a function contains no frequencies higher than F_{max}, it is completely determined by giving its ordinates at a series of points spaced $T = 1/(2F_{max})$ seconds apart [161]. It means that samples $x(t_n)$, which are taken at time instants $t_n = nT$, where $n \in Z$, fully represent the signal. The theorem is supplemented by signal recovery equation: [155]

$$x(t) = \sum_{n=-\infty}^{\infty} x(nT)\text{sinc}(\pi t/T - n\pi), \tag{2.17}$$

where

$$\text{sinc}(t) = \frac{\sin(t)}{t}. \tag{2.18}$$

The signal recovery equation (2.17) can not be directly used to reconstruct the signal sampled by ASDM, since the samples $x(nT)$ are not given, therefore in this section signal recovery algorithms for ASDM encoded signals are described.

2.2.1 Signal Recovery from ASDM Output Time Sequence

The signal, which is encoded by ASDM, can be perfectly recovered from the time sequence t_k (obtain from ASDM output signal $z(t)$) if the maximum distance does not exceed the Nyquist step:

$$\tau_{max} = \frac{2\kappa\delta}{b-c} \leq \frac{1}{2F_{max}}, \tag{2.19}$$

where F_{max} is maximum frequency of the input signal $x(t)$ [156]. By assuming that the signal $x(t)$ can be represented as

$$x(t) = \sum_{n \in Z} a_n g(t - \tau_n), \tag{2.20}$$

where

$$g(t) = \frac{\sin\Omega t}{\pi t} = \frac{\Omega}{\pi}\text{sinc}(\Omega t) \tag{2.21}$$

is the impulse response of an ideal low pass filter with cutoff frequency Ω, $\tau_n = \frac{t_k + t_{k+1}}{2}$ and a_n are the coefficients to be estimated, then from (2.3) and (2.20) follow that the coefficient values can be expressed as [156]

$$\mathbf{a} = \mathbf{G}^{+}\mathbf{q}, \tag{2.22}$$

where $\mathbf{G}^{+}$ is the pseudoinverse matrix of $\mathbf{G}$ and elements of vector $\mathbf{q}$ and matrix $\mathbf{G}$ are [156]

$$q_k = (-1)^k(2k\delta - b(t_{k+1} - t_k)) \tag{2.23}$$

$$G_{kn} = \int_{t_k}^{t_{k+1}} g(t - \tau_n)dt \tag{2.24}$$

As graphically shown in Figure 2.5, by finding coefficient values $\mathbf{a}$ (black bars), which are multiplied by sinc functions $g(t - \tau_n)$ (2.21) and summed together, it is possible to reconstruct the original signal $x(t)$ (red line).

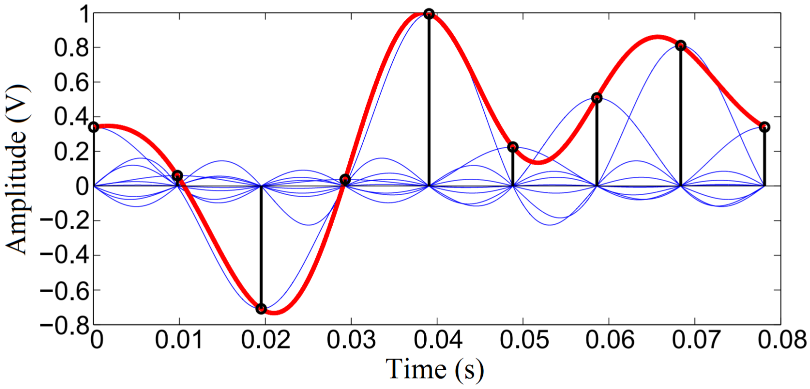


Figure 2.5: Visualization of signal recovery. Coefficients $\mathbf{a}$ (black bars), sinc functions $g(t - \tau_n)$ (blue lines), reconstructed EEG signal (red line)

Unfortunately, as shown in the author's paper **[27]**, by using this reconstruction method, signal recovery is time and resource consuming thereby must be improved in order to reconstruct the original signal in real time. In the next section (Section 2.2.2) an algorithm for faster signal recovery is described.

2.2.2 Fast Signal Recovery

As outlined in Section 2.2.1, the typical approach for signal recovery in not efficient enough for the real-time signal reconstruction. This is due to time consuming calculation of the coefficients a_n according to (2.22) since:

- numerical calculation of elements $g_{k,n}$ from matrix **G** (see (2.24)) requires to assign the function $g(t)$ with sufficiently fine step;
- matrix **G** pseudoinversion (see (2.22)) is carried out. [162]

In order to increase the speed of signal reconstruction, instead of finding the coefficients a_n, which correspond to signal representation (2.20), it is more efficient to find the coefficients d_n, which correspond to integral signal representation [162]

$$\int_{-\infty}^{t} x(u)du = \sum_{n\in Z} d_n g(t-\tau_n) \tag{2.25}$$

From (2.25) it follows [162]:

$$\int_{t_k}^{t_{k+1}} x(u)du = \sum_{n\in Z} d_n(g(t_{k+1}-t_n) - g(t_k - t_n)). \tag{2.26}$$

As the right side of (2.26) equals to q_k (see (2.3) and (2.23)), the equation (2.26) in matrix form can be written as

$$\mathbf{q} = \mathbf{PVd}, \tag{2.27}$$

where matrix **P** and **V** elements are

$$P_{kn} = \delta_{k+1,n} - \delta_{k,n} \tag{2.28}$$

$$V_{kn} = g(t_k - t_n), \tag{2.29}$$

where $\delta_{k,n}$ represents the Kronecker symbol: $\delta_{k,n} = 1$, if $k = n$, and $\delta_{k,n} = 0$, if $k \neq n$ [162]. The unknown coefficients d_n are obtained from (2.27) as:

$$\mathbf{d} = \mathbf{V}^{+}\mathbf{P}^{-1}\mathbf{q}, \tag{2.30}$$

where the elements of inverse matrix $\mathbf{P}^{-1}$ at row k and column n is -1, if $k \leq n$ and 0, if $k > n$. [162] In comparison to (2.22), in this case, the matrix **V** is required for estimation of the coefficients, and since the elements of **V** are found according to (2.29), the calculation of matrix **V** is precise and significantly faster. **[28]**, [162]

The next step to increase the speed of signal reconstruction, is to fasten matrix **V** pseudoinversion. This is achieved by using the periodic approximation of $g(t)$ **[28]**, [162]:

$$\hat{g}(t) = \xi \sum_{m=-M}^{M} e^{jm\frac{2\pi F_{max}}{M}t}, \tag{2.31}$$

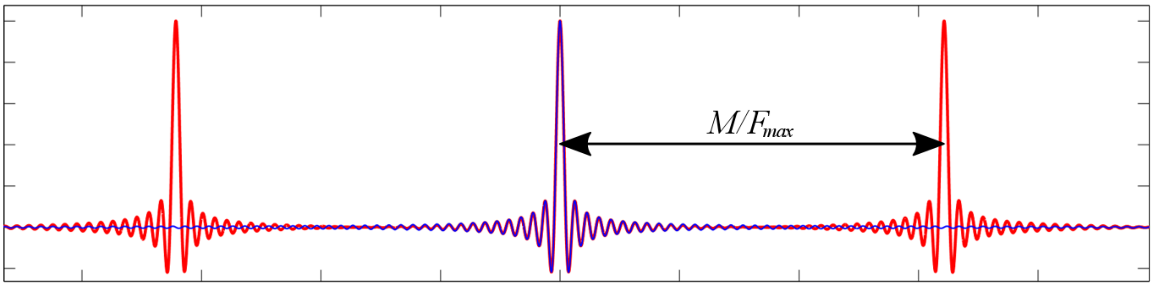


Figure 2.6: functions $g(t)$ (blue line) and $\hat{g}(t)$ (red line)

which more closely approximates $g(t)$, as M tends to infinity. The coefficient $\xi = 2F_{max}/(2M+1)$ and the value M/F_{max} determine the repetition period (see Fig. 2.6). [162]

In result, the expression (2.27) becomes:

$$\mathbf{q} = \xi \mathbf{P}\hat{\mathbf{V}}\mathbf{d}, \tag{2.32}$$

where matrix $\hat{\mathbf{V}}$ elements are:

$$\hat{V}_{kn} = \sum_{m=-M}^{M} e^{jm\frac{2\pi F_{max}}{M}(t_k - t_n)}. \tag{2.33}$$

Multiplying both (2.32) sides by $\mathbf{P}^{-1}$ and expressing $\hat{\mathbf{V}} = \mathbf{R}^H\mathbf{Y}$, gives

$$\mathbf{P}^{-1}\mathbf{q} = \xi \mathbf{R}^H \mathbf{Y}\mathbf{d}, \tag{2.34}$$

where $(.)^H$ is a designation of conjugated transposed matrix, but matrix **R** and **Y** elements are

$$R_{mk} = e^{-jm\frac{2\pi F_{max}}{M}t_k} \tag{2.35}$$

$$Y_{mn} = e^{-jm\frac{2\pi F_{max}}{M}t_n} \tag{2.36}$$

By denoting diagonal matrix as $\mathbf{D} = diag(t_{k+1} - t_k)$, $k \in Z$, and multiplying (2.34) both sides by **RD**, give [28], [162]

$$\mathbf{R}\mathbf{D}\mathbf{P}^{-1}\mathbf{q} = \xi \mathbf{R}\mathbf{D}\mathbf{R}^H\mathbf{Y}\mathbf{d} \tag{2.37}$$

Equivalently with

$$\mathbf{\Phi} = \xi \mathbf{R}\mathbf{D}\mathbf{R}^H \tag{2.38}$$

and

$$\boldsymbol{\upsilon} = \xi \mathbf{Y}\mathbf{d}, \tag{2.39}$$

we obtain (2.37) equivalent expression

$$\mathbf{RDP}^{-1}\mathbf{q} = \frac{1}{\xi}\boldsymbol{\Phi}\boldsymbol{\upsilon}, \tag{2.40}$$

from which follow unknown coefficients υ_m $(m = -M, \ldots, M)$ vector [162]

$$\boldsymbol{\upsilon} = \xi\boldsymbol{\Phi}^{+}\mathbf{RDP}^{-1}\mathbf{q}. \tag{2.41}$$

In this case, pseudo-inversed is matrix $\boldsymbol{\Phi}$, which elements

$$\Phi_{nm} = \xi \sum_{k\in Z} (t_{k+1} - t_k) e^{j(m-n)\frac{2\pi F_{max}}{M} t_k} \tag{2.42}$$

determine its Toeplitz and Hermitian type matrix structure and ensure fast pseudo-inverse. **[28]**, [162] The expression of coefficients υ_m follows from (2.39)

$$\upsilon_m = \xi \sum_{n\in Z} d_n e^{-jm\frac{2\pi F_{max}}{M} t_n} \tag{2.43}$$

Further, if (2.31) is used instead of $g(t)$, from (2.25) it is obtained:

$$\int_{-\infty}^{t} x(u)du = \xi \sum_{n\in Z} d_n \sum_{m=-M}^{M} e^{jm\frac{2\pi F_{max}}{M}(t-t_n)} \tag{2.44}$$

By considering (2.43) and transforming right side of (2.44), we obtain

$$\int_{-\infty}^{t} x(u)du = \sum_{m=-M}^{M} e^{jm\frac{2\pi F_{max}}{M}t} \xi \sum_{n\in Z} d_n e^{-jm\frac{2\pi F_{max}}{M}t_n} = \sum_{m=-M}^{M} \upsilon_m e^{jm\frac{2\pi F_{max}}{M}t} \tag{2.45}$$

Finally, by deriving right side of (2.45), the expression of reconstructed signal is obtained **[28]**, [162]

$$\hat{x}(t) = \frac{j2\pi F_{max}}{M} \sum_{m=-M}^{M} m\upsilon_m e^{jm\frac{2\pi F_{max}}{M}t} \tag{2.46}$$

By using this - fast signal recovery method, the signal reconstruction is up to 228 times faster (see simulation results in Section 4.1.1.2). Still, the signal reconstruction from time instants t_k is not possible in real time, since prior to reconstruction it is necessary to store time instants from $z(t)$ signal. In order to reconstruct the original signal in real-time, a separate section (Section 2.2.3) is dedicated to describe an idea [163] on how to reconstruct the signal, which is encoded by ASDM, in real time.

2.2.3 Real-time Signal Recovery

As outlined in Section 2.2.2, despite the fact that it is possible to reconstruct the original signal significantly faster, it is not possible to reconstruct the signal in real-time, since prior to reconstruction it is necessary to store ASDM output time instants t_k. Usually, in real-time systems, the signal is continuously sampled by ADC at the rate of f_s, and the ADC presents a new sample to the Digital Signal Processor (DSP) at this rate. In order to maintain a real-time operation, the DSP must perform all its required computation within the sampling interval, $1/f_s$, and present an output sample before arrival of the next sample from the ADC. [15]

In ASDM case, in order to reconstruct the signal in real-time, the reconstruction must be carried out in short time intervals $t \in [t_{mJ}, t_{mJ+L}]$, where $m = 0, 1, 2, \ldots$ designates the order number of the interval, but J determines the number of switchings, after which the reconstruction of the next interval can start, but L is a number of switching instants, which determines the length of the interval (see Fig. 2.7). [163]

Since the precision of the reconstructed signal fragment at the beginning and at the end of the fragment is low, the reconstructed signal $\hat{x}_m(t)$ is multiplied by corresponding window function: [163]

$$w_m(t) = \begin{cases} 0, & \text{if} \quad t \notin (\tau_m, \sigma_{m+1}], \\ \theta_m(t), & \text{if} \quad t \in (\tau_m, \sigma_m], \\ 1, & \text{if} \quad t \notin (\sigma_m, \tau_{m+1}], \\ 1 - \theta_{m+1}(t), & \text{if} \quad t \in (\tau_{m+1}, \sigma_{m+1}], \end{cases} \tag{2.47}$$

where $\tau_m = t_{mJ+M}$, $\sigma_m = t_{mJ+M+K}$ and

$$\theta_m(t) = \sin^2 \frac{\pi(t - \tau_m)}{2(\sigma_m - \tau_m)}. \tag{2.48}$$

After the multiplication, the signal fragment is different from zero only in the middle part of the interval, therefore the next interval is chosen after receiving $J = L - 2M - K$ switching time instants, thus ensuring overlapping of the intervals. In this case, increasing L not only improves the accuracy of the reconstruction, but also broadens window function $w_m(t)$ in time domain, and hence decreases its bandwidth Ω_w in frequency domain. It should be noted that enlarged size and decreased conditioning of $\mathbf{G}$ increase the computational load for calculating the pseudo-inverses $\mathbf{G}^+$ (see (2.22)). [163]

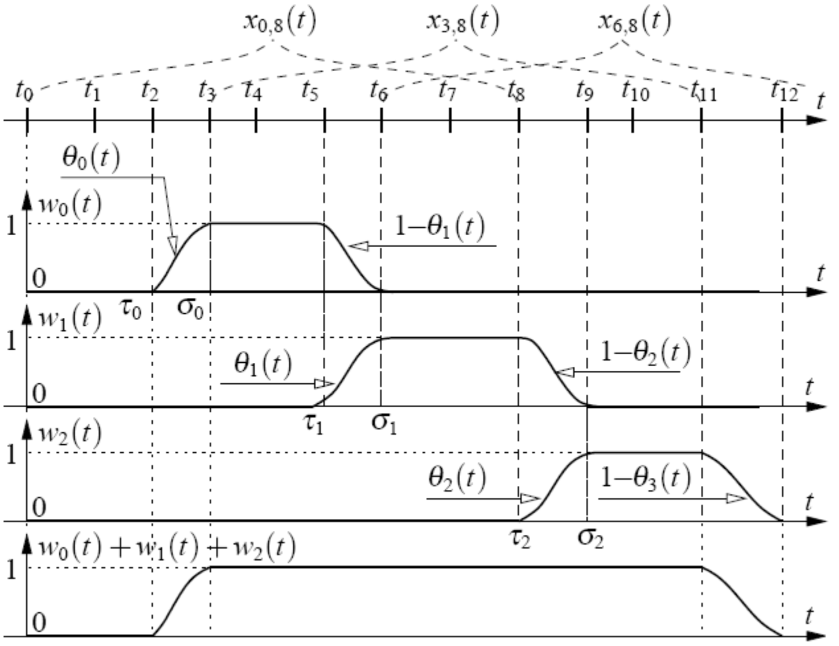


Figure 2.7: Real-time signal reconstruction by using interval approach [163]. In this particular case/figure, $t_0 = 0, L = 8, M = 2, K = 1 (J = 3)$

By combining all intervals, it is possible to obtain the whole reconstructed signal [163]:

$$\hat{x}_{L,M,K}(t) = \sum_{m \in Z} \hat{x}_m(t) w_m(t, L, M, K). \tag{2.49}$$

In this case, it can be assumed that the signal is recovered in real time if every last interval is reconstructed before the new sequence of J switching time instants have arrived (see Section 4.1.1.2).

The overall reconstruction error can be quantified by the following equation [163]:

$$e_{L,M,K}(t) = x(t) - \hat{x}_{L,M,K}(t). \tag{2.50}$$

and SNR, according to (2.50), after reconstruction is given by:

$$SNR_r = 10log_{10}\frac{P_s}{P_e} = 10log_{10}\frac{\int x^2(t)dt}{\int e^2(t)dt}. \tag{2.51}$$

In order to restore the initial signal frequency bandwidth, reconstructed signal $\hat{x}_{L,M,K}(t)$ is low pass filtered with a cut-off frequency Ω. Since the reconstruction spreads over the range $\omega \in (-\Omega_w - \Omega, \Omega_w + \Omega)$, low pass filtering also improves the overall accuracy. If $h(t)$ and $*$ denote the filter's impulse response and convolution, respectively, the error signal can be written as [163]:

$$\tilde{e}_{L,M,K}(t) = x(t) - \hat{x}_{L,M,K}(t) * h(t), \tag{2.52}$$

but SNR for reconstructed signal:

$$SNR_r = 10log_{10}\frac{P_s}{P_{\tilde{e}}} = 10log_{10}\frac{\int x^2(t)dt}{\int \tilde{e}^2(t)dt}. \tag{2.53}$$

2.3 Advantages and Disadvantages

As shown in Section 1.3, ASDM design, instead of other synchronous and asynchronous designs, in analog to digital converters (ADC) exhibit better properties such as low energy consumption and encoding complexity, etc. However, ASDM also have some disadvantages, for example, unnecessary high switching activity when it's applied to wide dynamic range signals (i.e. EEG signals) and complex signal reconstruction. In order to understand how to improve the ASDM, in this subsection an analysis on ASDM advantages and disadvantages are carried out in more details. Special emphasis will lay on ASDM disadvantages.

2.3.1 Advantages

The well-known benefits of ASDM are low energy consumption, immunity to metastable behavior, modular design, significant exclusion of electromagnetic interference (EMI), the absence of clock jitter, continuous-time signal processing, etc. (see Section 1.3) [164] [165] [166]

In the design and implementation of Brain Computer Interfaces (BCI), energy management and use of clocks are two main issues. First of all, the power dissipation due to analog to digital conversion is significant, therefore, it is crucial to find the best possible solution to reduce power consumption [167]. ASDM has promise for ultra-low power consumption because of the extreme simplicity of the required analog circuitry without any oversampling requirements [168]. Since the input signal is encoded in time and not in amplitude (voltage), it is possible to scale down the operating voltages and power consumption of ASDM, where speed/bandwidth is not a key limiting factor. Latest implementations show that it is possible to create a standard ASDM with power consumption less than $7.5nW$. [25]

Secondly, the presence of clocks, which are used in synchronous ADCs, in BCIs is problematic, since the required periodic clocks may cause EMI corrupting the analog signal (EEG) to be sampled [167]. In addition, in system on chip (SoC) implementations, these clocks consume a lot of power [169]. Even further, external clock will cause problems related to clock jitter. For example, in Sigma Delta modulator, the input of the quantizer at each clock moment has to be represented by a certain discrete value, thus introducing quantization noise [170]. By using clock-less ASDM, problems with clock jitter and high frequency components of the clock will be removed, thereby the spectrum of the signal will be "cleaner" and will have fewer distortion components at frequencies within the band of interest. [171]

Since ASDM is not using clock, it can be implemented in very simple analog circuit, which is much simpler than conventional ADCs circuits and allow to operate at low power levels, thus ensuring the low encoding complexity (see Section 1.3). The elimination of clocks also reduces device size. The latest implementations show that ASDM circuit will occupy as less as $0.026mm^2$ of a silicon die [146]. [172]

Finally, ASDM is immune to metastable behavior. With technology scaling, traditional high resolution ADC design in the voltage-domain becomes difficult due to the low supply voltage in the deep-submicrometer process. Noise would cause a meta-stability issue as the voltage resolution shrinks. However, for time-domain architecture, the time resolution benefits from the reduced transition time of digital circuits. [173]

2.3.2 Disadvantages

Despite the fact that ASDM has a long list of advantages, which makes it one of the best choices for EEG encoding and BCI application, it still has some disadvantages. For example, due to non-uniform sampling the reconstruction of the original signal $x(t)$, based on (2.22) is complicated (see Section 2.2). But, since the main aim is to reduce BCI power consumption and thus the complexity at the signal encoding part, the complexity shift from encoding part to decoding part is admissible.

Another ASDM disadvantage (inefficiency) is related to over-triggering, which occurs when ASDM is applied to wide dynamic range signals such as EEG signals. Due to wide dynamic range these signals have, an unnecessary high switching activity of ASDM circuit appears when the input signal amplitude is low (see Figure 2.8). This is due to ASDM circuit parameters (b

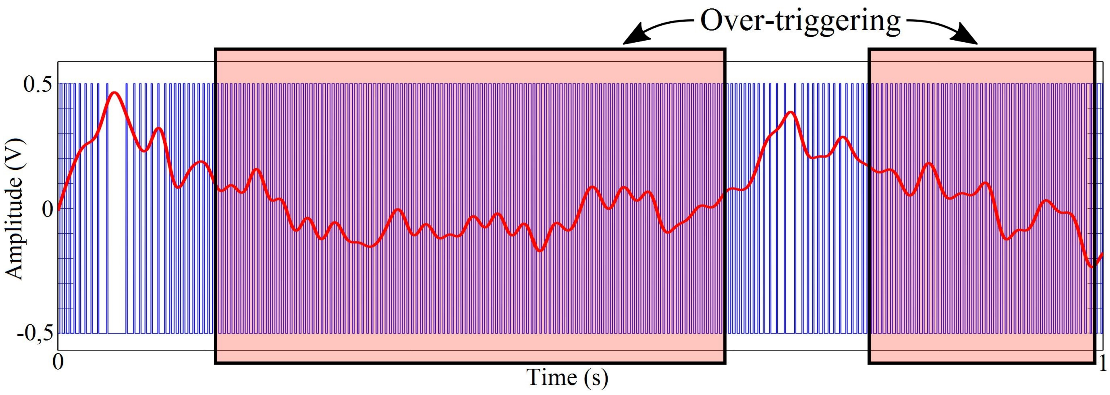


Figure 2.8: Switching activity of ASDM. EEG signal (red line) and ASDM trigger output signal (blue line)

and δ) are chosen considering the maximum value that is never exceeded by the signal. [27]

How to choose the parameters b and δ? If the signal being encoded is bandlimited to $[-\Omega, \Omega]$, then from Section 2.2.1 it follows that the necessary condition for perfect recovery of the original signal from the given time sequence t_k is:

$$\sup_{k\in Z}(t_{k+1} - t_k) \leq \frac{\pi}{\Omega} = T \tag{2.54}$$

That means τ_{max} can not exceed π/Ω, and the parameters b and δ can be chosen according to equation

$$\tau_{max} = \frac{2\kappa\delta}{b-c} = T \tag{2.55}$$

The minimum distance τ_{min} in this case becomes

$$\tau_{min} = \frac{2\kappa\delta}{b+c} = \frac{b-c}{b+c}T = \frac{1}{1+2/\alpha}T, \tag{2.56}$$

where $b = (1+\alpha)c$ and $\alpha > 0$. From (2.56) it follows: the smaller the coefficient α, the smaller the distance τ_{min} and thus the larger the number of trigger times. On the other hand, the larger the coefficient α, the smaller the difference between τ_{min} and τ_{max} and the more precision (more bits) is required to encode the time sequence t_k in order to recover the original signal. [27]

What is the inefficiency of having the constant parameter b? Let us assume we have a signal $x(t)$ with C being the maximum value of $|x(t)|$. The parameters b_1 and δ_1 are chosen as $b_1 = (1+\alpha)C$ and $\delta_1 = \alpha CT/(2\kappa)$ according to (2.55). Considering that instantaneous maximum value of $|x(t)|$ changes over time, and can be denoted as $c(t)$, the maximum and minimum distances τ_{max} and τ_{min} are time-varying as well:

$$\tau_{max1}(t) = \frac{2\kappa\delta_1}{b_1 - c(t)} = \frac{T}{1 + 1/\alpha - c(t)/(\alpha C)} \tag{2.57}$$

$$\tau_{min1}(t) = \frac{2\kappa\delta_1}{b_1 + c(t)} = \frac{T}{1 + 1/\alpha + c(t)/(\alpha C)} \tag{2.58}$$

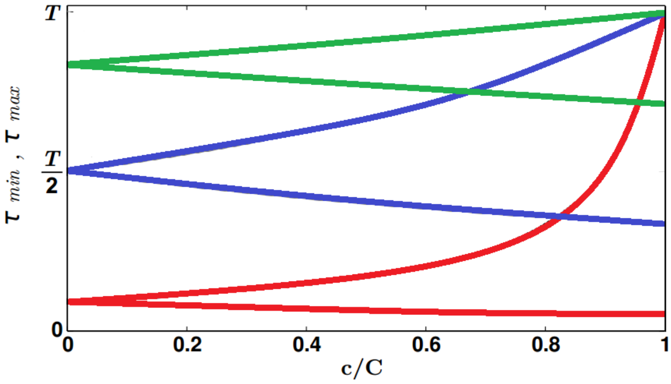


Figure 2.9: Distances τ_{max} (increasing lines) and τ_{min} (decreasing lines) depending on $c(t) \in [0, C]$ for three different $\alpha = \beta$ values 0.1 (red lines), 1 (blue lines) and 5 (green lines) with constant parameter b_1

The dependence of τ_{max1} and τ_{min1} on the value of $c(t) \in [0, C]$ is shown in Fig. 2.9 for different α values 0.1 (red lines), 1 (blue lines) and 5 (green lines). The decreasing lines correspond to τ_{min1} and increasing lines to τ_{max1}. As it can be seen, both distances τ_{min1} and τ_{max1} for small $c(t)$ values are less than T and thus over-triggering occurs, which is not necessary for reconstruction of the original signal. This can be reduced, if coefficient α is increased, however, too high α values lead to small differences $\tau_{max1} - \tau_{min1}$ and thus more precision is needed to encode the distances between consecutive trigger times. [27]

2.4 Summary and Conclusions

Section 2 is dedicated to the in depth analysis of Asynchronous Sigma-Delta modulator (ASDM). ASDM is a clock-less analog to digital converter (ADC), which converts amplitude information of the input signal $x(t)$ into time information or time sequence t_k in a very energy efficient way. The latest implementations show that it is possible to create a standard ASDM with power consumption less than $7.5nW$ [25], therefore it has a huge potential to reduce energy consumption in wireless Brain Computer Interface (BCI) systems.

The relationship between the switching instants t_k of the ASDM output $z(t)$ and the input signal $x(t)$ is given by the integral equation (see Section 2.1):

$$\int_{t_k}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - b(t_{k+1} - t_k)],$$

where κ, δ and b are ASDM circuit parameters, but t_k are switching time instants of $z(t)$.

The ASDM input signal $x(t)$ can be represented as

$$x(t) = \sum_{n \in Z} a_n g(t - \tau_n),$$

where $g(t) = \frac{\sin\Omega t}{\pi t}$ is the impulse response of an ideal low pass filter with cutoff frequency Ω, $\tau_n = \frac{t_k + t_{k+1}}{2}$ and a_n are the coefficients to be estimated in order to reconstruct the original signal. From both these equations follow that the unknown coefficient a_n values for signal reconstruction can be found by (see Section 2.2)

$$\mathbf{a} = \mathbf{G}^{+}\mathbf{q},$$

where $q_k = (-1)^k(2k\delta - b(t_{k+1} - t_k))$ and $G_{kn} = \int_{t_k}^{t_{k+1}} g(t - \tau_n)dt$.

Unfortunately, the signal recovery method described above is time and resource consuming and thus can not be used for real-time applications. Therefore, in Section 2.2.2 a method for fast signal recovery is described. In this case, expression which describes reconstructed signal is (see Section 2.2.2):

$$\hat{x}(t) = \frac{j2\pi F_{max}}{M} \sum_{m=-M}^{M} m\upsilon_m e^{jm\frac{2\pi F_{max}}{M}t},$$

where unknown coefficients υ_m can be found by:

$$\boldsymbol{\upsilon} = \xi \boldsymbol{\Phi}^{+}\mathbf{R}\mathbf{D}\mathbf{P}^{-1}\mathbf{q}$$

However, despite the fact that by using the fast signal recovery method it is possible to reconstruct the original signal significantly faster, it is not possible to reconstruct the signal in real-time, since prior to reconstruction it is necessary to store time instants from the ASDM output signal. In order to be able to reconstruct the signal in real time, short time interval reconstruction method should be used (see Section 2.2.3):

$$\hat{x}(t) = \sum_{m \in Z} \hat{x}_m(t) w_m(t),$$

where $w_m(t)$ is a window function, defined in (2.47).

Finally, in Section 2.3, ASDM advantages and disadvantages are described. The main benefits of ASDM are low energy consumption, immunity to metastable behavior, modular design, significant exclusion of electromagnetic interference (EMI), the absence of clock, low complicity, continuous-time signal processing, etc, which makes it suitable for BCI applications. But, the main inefficiency is related to over-triggering, which occurs when ASDM is applied to wide dynamic range signals such as EEG signals. Due to wide dynamic range that these signals have, an unnecessary high switching activity of ASDM circuit appears when the input signal amplitude is low. This is because the ASDM circuit parameters are chosen considering the maximum value that is never exceeded by the signal. Since the envelope of the signal changes over time, the proposition is: instead of constant value to consider the time-varying maximum value, which is also never exceeded by the signal. This could allow to reduce the over-triggering of the circuit and thus the power consumption of the whole wireless BCI system, since less switchings instants (events) will be necessary to be transmitted, which greatly reduces the power consumption of the transmitter. The proposed amplitude adaptive method is described in Section 3.

3. AMPLITUDE ADAPTIVE ASYNCHRONOUS SIGMA-DELTA MODULATOR

The concept of the Asynchronous Sigma-Delta modulator (ASDM) is described in Section 2. As it was shown in Section 2.3.2, due to wide dynamic range that electroencephalogram (EEG) signals have, an unnecessary high switching activity of ASDM circuit appears when the input signal amplitude is low, causing increased power consumption of wireless Brain Computer Interface (BCI) system. This is due to ASDM circuit parameters are chosen considering only the maximum value that is never exceeded by the signal. In recent years different studies and several applications of ASDM have been presented [173], [174], [175], [176], however, no signal-dependent adaptive enhancements have been shown.

Since envelope of the signal changes over time, the proposition of this work is, instead of constant value b, to consider the time-varying maximum value, which is also never exceeded by the signal. This would allow to reduce the over-triggering of the circuit and thus the power consumption of a wireless BCI system, since less switchings instants (events) will be necessary to be transmitted, which greatly reduces the power consumption of the transmitter.

The block diagram of proposed Amplitude-Adaptive Asynchronous Sigma-Delta modulator (AA-ASDM), which is an enhanced version of ASDM, is shown in Figure 3.1. In addition to ASDM circuit (see Fig. 2.2), there is an envelope detector with output $c(t)$ connected to the feedback loop. Now, instead of the equation (2.3), the relationship between the switching instants t_k of the AA-ASDM output $z(t)$ and the input signal $x(t)$ for $t_{k+1} > t_k$, and integers $k \in Z$, is given by the following equation:

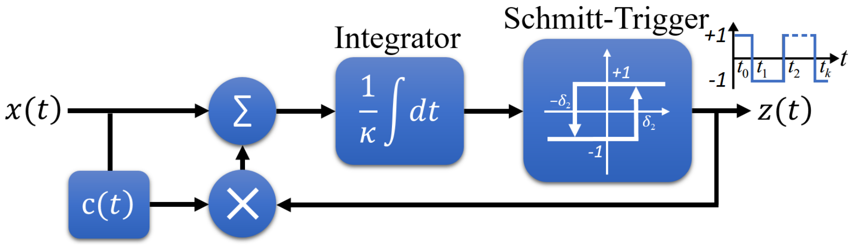


Figure 3.1: Amplitude Adaptive Asynchronous Sigma-Delta modulator block diagram

$$\int_{t_k}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - \int_{t_k}^{t_{k+1}} c(t)dt], \tag{3.1}$$

where κ and δ are AA-ASDM circuit parameters. By using the proposed method, the perfect recovery of the original signal is still possible, if $c(t)$ is known.

Based on this, two amplitude adaptive methods: 1) AA-ASDM with additional envelope encoding (Section 3.1); and 2) AA-ASDM without additional envelope encoding (Section 3.2); which are both particularly advantageous for signals with wide dynamic range, are proposed and described further in this section. In addition, at the end of this section a summary and conclusions are given.

3.1 AA-ASDM with additional envelope encoding

In this section, the proposed AA-ASDM with additional envelope encoding is described. First of all, signal encoding and decoding/recovery principles are described in Section 3.1.1 and Section 3.1.2.1. Then, in order to reconstruct the original signal in real-time, a method for fast signal reconstruction and a method for real-time signal reconstruction are given accordingly in Section 3.1.2.2 and Section 3.1.2.3.

3.1.1 Signal Encoding

As outlined in Section 2.3.2, having the constant parameter b is inefficient (see equations (2.57) and (2.58)). In order to avoid over-triggering, the proposition is instead of constant parameter b to use time-varying parameter $b_2(t)$ according to $c(t)$ to ensure

$$\tau_{max2}(t) = \frac{2\kappa\delta_2}{b_2(t) - c(t)} = const. = T, \tag{3.2}$$

where T is the Nyquist step and equals the maximum distance between two consecutive trigger switching times t_k and t_{k+1}. [27] In this case, the difference $b_2(t) - c(t)$ must be constant and thus $b_2(t)$ can be written as

$$b_2(t) = c(t) + \beta C, \tag{3.3}$$

where $\beta > 0$ is a constant, but C is the maximum value of $|x(t)|$. [27] The second parameter $\delta_2 = \beta CT/(2\kappa)$ follows from (3.2) and the minimum distance between two consecutive trigger switching times becomes

$$\tau_{min2}(t) = \frac{2\kappa\delta_2}{b_2(t) + c(t)} = \frac{T}{1 + 2c(t)/(\beta C)}. \tag{3.4}$$

The dependence of τ_{min2} on the value of $c(t) \in [0, C]$ is shown in Fig. 3.2 by colored lines for different β values: 0.1 (red line), 1 (blue line) and 5 (green line), but for τ_{min1} by black and gray lines for different β values: 0.1 (black line), 1 (dark gray line) and 5 (light gray line) (see also Fig. 2.9). As can be seen, $\tau_{min2} > \tau_{min1}$ and $\tau_{max2} > \tau_{max1}$ for all $c(t)$ and $\beta = \alpha$ values, which means the over-triggering reduces in comparison to the classical $b = const.$ case. Also the differences $\tau_{max2} - \tau_{min2}$ are larger for all $c(t) < C$ values. **[27]**

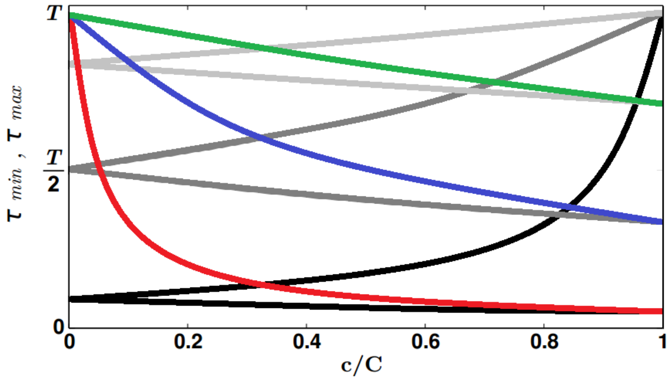


Figure 3.2: Distances τ_{max} (increasing lines) and τ_{min} (decreasing lines) depending on $c(t) \in [0, C]$ for three different $\alpha = \beta$ values 0.1 (red line), 1 (blue line) and 5 (green line) for AA-ASDM case (time-varying parameter $b_2(t)$), and $\alpha = \beta$ values 0.1 (black lines), 1 (dark gray lines) and 5 (light gray lines) for ASDM case (constant parameter b_1).

The block diagram of the proposed AA-ASDM with additional envelop encoding is shown in Figure 3.3. Now instead of (2.3) the following equation holds **[27]**:

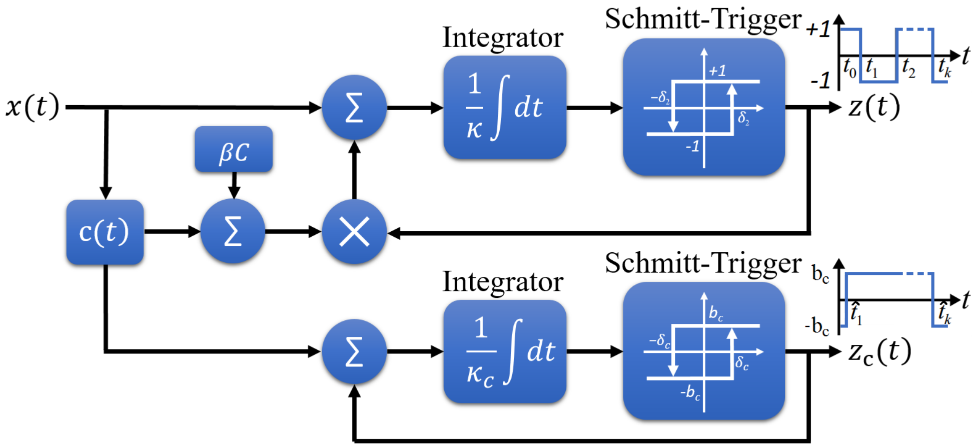


Figure 3.3: Amplitude Adaptive Asynchronous sigma-delta modulator (AA-ASDM) with additional envelop encoding block diagram

$$\int_{t_k}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - \beta C(t_{k+1} - t_k) - \int_{t_k}^{t_{k+1}} c(t)dt], \tag{3.5}$$

where $\beta > 0$ is a constant and C is the maximum value of $|x(t)|$, but $c(t)$ can be obtained from expression:

$$\int_{\hat{t}_k}^{\hat{t}_{k+1}} c(t)dt = (-1)^k[2\kappa_c\delta_c - b_c(\hat{t}_{k+1} - \hat{t}_k)] = \hat{q}_k. \tag{3.6}$$

As it follows from (3.5), the envelope signal $c(t)$ is also needed for recovery of $x(t)$, therfore it is encoded by another ASDM with circuit parameters κ_c, δ_c and b_c. Now, for AA-ASDM to be advantageous over ASDM, the signal $c(t)$ must have low frequencies in comparison to $x(t)$. This ensures the number of triggering points in $z_c(t)$ is much less than in $z(t)$. **[27]**

In this case, the complexity of signal recovery increases, but power consumption of the wireless BCI system decreases, due to fewer trigger switchings. The effectiveness of this AA-ASDM approach is estimated by simulations in Section 4.1.2.

Regardless of the decrease of the number of switchings, the perfect recovery of the original signal from the obtained time sequence is still possible (see Section 3.1.2). **[27]**

3.1.2 Signal Recovery

The signal reconstruction in AA-ASDM (with additional envelop encoding) is similar to one described in Section 2.2, but in this case, the envelope signal $c(t)$ is also needed for recovery of $x(t)$ (see Section 3.1.2.1). The speed of $x(t)$ reconstruction in AA-ASDM can be increased by using fast signal reconstruction approach described in Section 3.1.2.2. Also, AA-ASDM method can be used for real-time signal recovery as described in Section 3.1.2.3.

3.1.2.1 Signal Recovery from AA-ASDM Output Time Sequence

The signal $x(t)$ with time-varying envelope function $c(t)$ is recovered considering that both $x(t)$ and $c(t)$ are bandlimited to $[-\Omega_x, \Omega_x]$ and $[-\Omega_c, \Omega_c]$, respectively. The first step is recovery of $c(t)$ from the given time sequence $\hat{t}_k$ by finding the unknown coefficients o_n in

$$c(t) = \sum_{n \in Z} o_n h(t - \hat{\tau}_n), \tag{3.7}$$

where

$$h(t) = \frac{\sin\Omega_c t}{\pi t} = \frac{\Omega_c}{\pi}\text{sinc}(\Omega_c t) \tag{3.8}$$

and $\hat{\tau}_n = (\hat{t}_k + \hat{t}_{k+1})/2$. [162], [27] From (3.1) and (3.7) follows

$$\mathbf{o} = \mathbf{H}^{+}\hat{\mathbf{q}} \tag{3.9}$$

where $\mathbf{H}^{+}$ is the pseudoinverse of $\mathbf{H}$ and the elements of vector $\hat{\mathbf{q}}$ and matrix $\mathbf{H}$ are

$$\hat{q}_k = (-1)^k(2\kappa_c\delta_c - b_c(\hat{t}_{k+1} - \hat{t}_k)) \tag{3.10}$$

and

$$H_{kn} = \int_{\hat{t}_k}^{\hat{t}_{k+1}} h(t - \hat{\tau}_n)dt. \tag{3.11}$$

When $c(t)$ is found, the signal $x(t)$ is recovered from the given time sequence t_k by finding the unknown coefficients $\hat{a}_n$ in

$$x(t) = \sum_{n\in Z} \hat{a}_n g(t - \tau_n), \tag{3.12}$$

where

$$g(t) = \frac{\sin\Omega_x t}{\pi t} = \frac{\Omega_x}{\pi}\text{sinc}(\Omega_x t) \tag{3.13}$$

and $\tau_n = (t_k + t_{k+1})/2$. [162], [27] From (3.5) and (3.12) follows

$$\hat{\mathbf{a}} = \mathbf{G}^{+}\mathbf{q}, \tag{3.14}$$

where

$$q_k = (-1)^k\left(2\kappa\delta - \beta C(t_{k+1} - t_k) - \int_{t_k}^{t_{k+1}} c(t)dt\right) \tag{3.15}$$

and

$$G_{kn} = \int_{t_k}^{t_{k+1}} g(t - \tau_n)dt \tag{3.16}$$

By using this signal recovery method, signal reconstruction is time and resource consuming (see Section 4.1.1.2) thereby can be improved in order to reconstruct the original signal in real time. In next the Section 3.1.2.2, an algorithm for faster signal recovery is described.

3.1.2.2 Fast Signal Recovery

The signal recovery approach, described in Section 3.1.2.1, is not efficient enough for real-time signal reconstruction. This is due to time consuming calculation of the coefficients o_n and $\hat{a}_n$ according to (3.9) and (3.14).

In order to increase the speed of signal reconstruction, instead of finding the coefficients $\hat{a}_n$, which correspond to signal representation (3.12), it is more efficient to find the coefficients $\hat{d}_n$, which correspond to integral signal representation [162], **[27]**

$$\varsigma(t) = \int_{-\infty}^{t} x(u)du = \sum_{n\in Z} \hat{d}_n g(t-\tau_n) \tag{3.17}$$

From (3.17) it follows:

$$\int_{t_k}^{t_{k+1}} x(u)du = \sum_{n\in Z} \hat{d}_n [g(t_{k+1}-\tau_n) - g(t_k-\tau_n)] \tag{3.18}$$

Similarly, the integral envelope $c(t)$ representation can be written as

$$\int_{-\infty}^{t} c(u)du = \sum_{n=-\infty}^{\infty} \hat{o}_n h(t-\hat{\tau}_n).) \tag{3.19}$$

From (3.19) it follows:

$$\int_{\hat{t}_k}^{\hat{t}_{k+1}} c(u)du = \sum_{n\in Z} \hat{o}_n [h(\hat{t}_{k+1}-\hat{\tau}_n) - h(\hat{t}_k-\hat{\tau}_n)] \tag{3.20}$$

$$\int_{t_k}^{t_{k+1}} c(u)du = \sum_{n\in Z} \hat{o}_n [h(t_{k+1}-\hat{\tau}_n) - h(t_k-\hat{\tau}_n)] \tag{3.21}$$

As the right side of (3.6) equals $\hat{q}_k$, from (3.6) and (3.20) it follows that

$$\hat{q}_k = \sum_{n\in Z} \hat{o}_n [h(\hat{t}_{k+1}-\hat{\tau}_n) - h(\hat{t}_k-\hat{\tau}_n)]. \tag{3.22}$$

The equation (3.22) in matrix form can be written as

$$\hat{\mathbf{q}} = \hat{\mathbf{P}}\hat{\mathbf{H}}\hat{\mathbf{o}}, \tag{3.23}$$

where matrix $\hat{\mathbf{P}}$ and $\hat{\mathbf{H}}$ elements are

$$\hat{P}_{kn} = \delta_{k+1,n} - \delta_{k,n} \tag{3.24}$$

$$\hat{H}_{kn} = h(\hat{t}_k - \hat{\tau}_n), \tag{3.25}$$

where $\delta_{k,n}$ represents the Kronecker symbol: $\delta_{k,n} = 1$, if $k = n$, and $\delta_{k,n} = 0$, if $k \neq n$, but unknown coefficients $\hat{o}_n$ are obtained from (3.23) as:

$$\hat{\mathbf{o}} = \hat{\mathbf{H}}^{+}\hat{\mathbf{P}}^{-1}\hat{\mathbf{q}} \tag{3.26}$$

where the elements of inverse matrix $\hat{\mathbf{P}}^{-1}$ at row k and column n is -1, if $k \leq n$ and 0, if $k > n$. [162], **[28]**

From (3.5), (3.18) and (3.21) obtain

$$\sum_{n \in Z} \hat{d}_n [g(t_{k+1} - \tau_n) - g(t_k - \tau_n)] =$$

$$= (-1)^k [2\kappa\delta - \beta C(t_{k+1} - t_k)] + (-1)^{k+1} \sum_{n \in Z} \hat{o}_n [h(t_{k+1} - \hat{\tau}_n) - h(t_k - \hat{\tau}_n)]. \tag{3.27}$$

If

$$\tilde{q}_k = (-1)^k [2\kappa\delta - \beta C(t_{k+1} - t_k)], \tag{3.28}$$

then from (3.27) and (3.28) follows **[28]**

$$\sum_{n \in Z} \hat{d}_n [g(t_{k+1} - \tau_n) - g(t_k - \tau_n)] = \tilde{q} + (-1)^{k+1} \sum_{n \in Z} \hat{o}_n [h(t_{k+1} - \hat{\tau}_n) - h(t_k - \hat{\tau}_n)] \tag{3.29}$$

$$\sum_{n \in Z} \hat{d}_n [g(t_{k+1} - t\tau_n) - g(t_k - \tau_n)] = \tilde{q} + \sum_{n \in Z} \hat{o}_n [(-1)^{k+1} (h(t_{k+1} - \hat{\tau}_n) - h(t_k - \hat{\tau}_n))] \tag{3.30}$$

The equation (3.30) in matrix form can be written as **[28]**

$$\mathbf{P}\mathbf{G}\hat{\mathbf{d}} = \tilde{\mathbf{q}} + \tilde{\mathbf{P}}\mathbf{H}\hat{\mathbf{o}}, \tag{3.31}$$

where matrix **P**, **G**, $\tilde{\mathbf{P}}$ and **H** elements are

$$P_{kn} = \delta_{k+1,n} - \delta_{k,n} \tag{3.32}$$

$$G_{kn} = g(t_k - \tau_n) \tag{3.33}$$

$$\tilde{P}_{kn} = (\delta_{k+1,n} - \delta_{k,n}) \cdot (-1)^{k+1} \tag{3.34}$$

$$H_{kn} = h(t_k - \hat{\tau}_n). \tag{3.35}$$

From (3.26) and (3.31) it follows, that unknown coefficients $\hat{d}_n$ are obtained as [162], **[28]**:

$$\hat{\mathbf{d}} = \mathbf{G}^{+}\mathbf{P}^{-1} \cdot (\tilde{\mathbf{q}} + \tilde{\mathbf{P}}\mathbf{H}\hat{\mathbf{o}}) = \mathbf{G}^{+}\mathbf{P}^{-1} \cdot (\tilde{\mathbf{q}} + \tilde{\mathbf{P}}\mathbf{H}\hat{\mathbf{H}}^{+}\hat{\mathbf{P}}^{-1}\hat{\mathbf{q}}). \tag{3.36}$$

When all coefficients $\hat{d}_n$ are found, $\varsigma(t)$ in (3.17) can be calculated as **[28]**

$$\varsigma(t) = \sum_{n=1}^{N} \hat{d}_n g(t - \tau_n), \tag{3.37}$$

and then, original signal can be found from

$$x(t) = \frac{d\varsigma(t)}{dt}. \tag{3.38}$$

In order to increase the speed of signal reconstruction even more, it is necessary to fasten matrix **G** and **H** pseudoinversion in (3.36). This can be achieved by using the same method as described in Section 2.2.2.

Still, by using fast signal reconstruction method, the signal reconstruction from time instants t_k and $\hat{t}_k$ is not possible in real time, since prior to reconstruction it is necessary to store time instants from $z(t)$ and $z_c(t)$ signals. In order to reconstruct the original signal in real-time, a separate section (Section 3.1.2.3) is dedicated to describe how to reconstruct the signal, which is encoded by AA-ASDM with additional envelope encoding, in real time.

3.1.2.3 Real-Time Signal Recovery

In order to reconstruct the envelop as well as the original signal in real-time, the reconstruction must be carried out in short time intervals. The principle of the real-time short time interval signal reconstruction method is described in Section 2.2.3, based on [163].

In AA-ASDM case, first of all the envelop signal $c(t)$ must be reconstructed in short time intervals $\hat{t} \in [t_{mJ_c}, t_{mJ_c+L_c}]$, in order to reconstruct the original $x(t)$ signal (see Fig.3.4).

When the first interval of $c(t)$ is reconstructed, the real-time reconstruction of $x(t)$, in short time intervals $t \in [t_{mJ}, t_{mJ+L}]$, can start. In this particular example in Fig. 3.4, it is assumed that $c(t)$ frequency is four times lower than $x(t)$ frequency, therefore it is possible to reconstruct four $x(t)$ intervals by knowing one $c(t)$ interval. In practice, $c(t)$ frequency is up to twenty times lower than $x(t)$ frequency. The main drawback of this method applied to AA-ASDM approach is an increased delay at the beginning of the conversion. In this case, the delay introduced by reconstruction of $c(t)$ is between $\hat{t}_{10} - \hat{t}_0$ and $\hat{t}_{18} - \hat{t}_0$. Also, since it is necessary to reconstruct two signals, the AA-ASDM real-time reconstruction algorithm is more computing resource demanding than ASDM real time reconstruction algorithm.

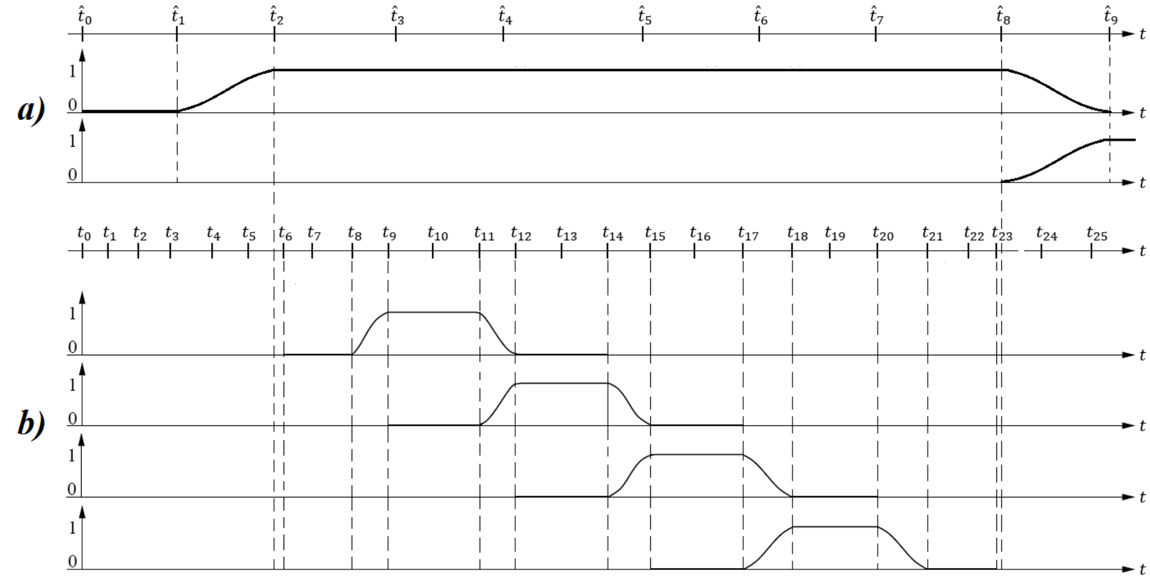


Figure 3.4: Real-time signal reconstruction by using interval approach. a) reconstruction of the envelop $c(t)$, $\hat{t}_0 = 0, L_c = 10, M_c = 1, K_c = 1$; b) reconstruction of the original signal $x(t)$, $t_0 = 0, L = 8, M = 2, K = 1$

Although AA-ASDM with additional envelop encoding is advantageous over ASDM in terms of switching activity (see Section 4.1.2), the efficiency of this method can be improved, if the envelope of the signal is not needed to be encoded and transmitted in order to recover the signal. Since the envelope depends on the signal itself, the appropriate reconstruction method could be developed for recovery from only one time sequence obtained at the output of the upper trigger in Fig. 3.3. This would also allow to avoid the delay, which is introduced by additional envelope signal reconstruction prior to the original signal reconstruction. This is the topic, which is further described in Section 3.2. [27]

3.2 AA-ASDM without additional envelope encoding

In Section 3.1, an improved version of ASDM, called AA-ASDM was presented, where by using time-varying envelope of the input signal in the feedback loop of ASDM, it is possible to reduce the switching activity of ASDM and thus the power consumption of the wireless BCI system. Regardless of this reduction the perfect recovery of the original signal from the obtained time sequences is still possible. [27] However, as concluded in Section 3.1.2.3, the efficiency of AA-ASDM can be improved if the time-varying envelope of the signal is not additionally encoded and transmitted in order to recover the original signal.

Further studies have led to a solution which not only solves the above-mentioned problem, but also allows to reduce the switching activity of the ASDM circuit even more. The principles of an improved version of AA-ASDM is described in Section 3.2.1. The proposed method shows how to choose the time-varying envelope in such a way, that there is no need to encode and transmit information about the time-varying envelope and to be able to recover the original signal from the obtained time sequence. In the proposed fast signal reconstruction method, Fourier series instead of sinc functions are used for signal representation, thus improving the speed of reconstruction while keeping the precision (see Section 3.2.2.1 and Section 3.2.2.2). A method for real-time reconstruction is described in Section 3.2.2.3.

3.2.1 Signal Encoding

The block diagram of the proposed AA-ASDM without additional envelop encoding is shown in Figure 3.5. [29]

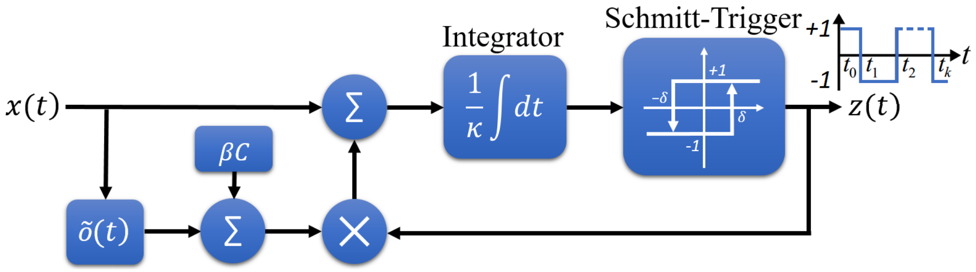


Figure 3.5: Amplitude Adaptive Asynchronous sigma-delta modulator (AA-ASDM) without additional envelop encoding block diagram

The relationship between the switching instants t_k of the AA-ASDM output $z(t)$ and the input signal $x(t)$ is given by the same equation as in Section 3.1.1 (see equation (3.5)):

$$\int_{tk}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - \beta C(t_{k+1} - t_k) - \int_{tk}^{t_{k+1}} \tilde{o}(t)dt], \tag{3.39}$$

where $x(t)$ is bounded in amplitude as $[-1, 1]$; κ, δ, β, C are known coefficients ($C = 1$, since $|x(t)| \leq 1$); and $\tilde{o}(t)$ is the estimated envelope of the input signal, where $\tilde{o}(t) \geq |x(t)|$.

In this case, the proposition is to choose the time-varying envelope function as [29]:

$$\tilde{o}(t) = const. + x^2(t). \tag{3.40}$$

By choosing $const. = 0.25$, the inequality $\tilde{o}(t) \geq |x(t)|$ holds for all $x(t) \in [-1, 1]$ (see Fig. 3.6), therefore the time-varying envelope $\tilde{o}(t)$ of the input signal $x(t)$ is selected as **[29]**:

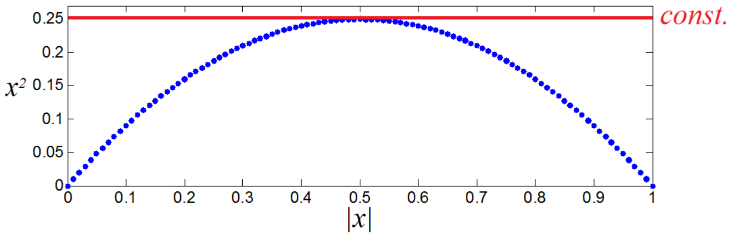


Figure 3.6: Selection of the constant value in (3.40)

$$\tilde{o}(t) = 0.25 + x^2(t) \geq x(t). \tag{3.41}$$

This function, as follows from the next Section 3.2.2, does not require any additional encoding scheme.

3.2.2 Signal Recovery

The signal reconstruction methods, described in Section 2.2 and Section 3.1.2, are not efficient, because sinc functions are not well suited for representing time-limited signals and integral values of sinc functions can no be analytically found, therefore the calculation of matrix **G** is both time consuming and not perfectly precise. The proposition is therefore to use Fourier series instead of sinc functions for the original signal representation. The proposed signal reconstruction method can also be used for real-time signal recovery as described further in sub-sections.

3.2.2.1 Signal Recovery from AA-ASDM Output Time Sequence

By inserting (3.41) into (3.39), a relationship between the output sequence $\{t_k\}_{k=1,2,\ldots,K}$ and the input signal $x(t)$ of AA-ASDM can be obtained **[29]**:

$$\int_{tk}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - \beta C(t_{k+1} - t_k) - \int_{tk}^{t_{k+1}} (0.25 + x^2(t))dt]. \tag{3.42}$$

The equation (3.42) can be expressed as:

$$\int_{tk}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - \beta C(t_{k+1} - t_k) - 0.25 \int_{tk}^{t_{k+1}} dt - \int_{tk}^{t_{k+1}} x^2(t)dt] \tag{3.43}$$

$$\int_{tk}^{t_{k+1}} x(t)dt + (-1)^k \int_{tk}^{t_{k+1}} x^2(t)dt = (-1)^k[2\kappa\delta - \beta C(t_{k+1} - t_k) - 0.25(t_{k+1} - t_k)] \quad (3.44)$$

$$\int_{tk}^{t_{k+1}} x(t)dt + (-1)^k \int_{tk}^{t_{k+1}} x^2(t)dt = (-1)^k[2\kappa\delta - (\beta C + 0.25)(t_{k+1} - t_k)]. \quad (3.45)$$

By denoting

$$\tilde{q}_k = (-1)^k[2\kappa\delta - (\beta C + 0.25)(t_{k+1} - t_k)], \quad (3.46)$$

the equation (3.45) becomes:

$$\int_{tk}^{t_{k+1}} x(t)dt + (-1)^k \int_{tk}^{t_{k+1}} x^2(t)dt = \tilde{q}_k. \quad (3.47)$$

By assuming that the input signal $x(t)$ can be represented as

$$x(t) = \sum_{n=0}^{N-1} \tilde{d}_n g_n(t), \quad (3.48)$$

where $\tilde{d}_n$ are unknown coefficients and $g_n(t)$ are the chosen base functions, the first term of (3.47) can be written as

$$\int_{tk}^{t_{k+1}} x(t)dt = \sum_{n=0}^{N-1} \tilde{d}_n \int_{tk}^{t_{k+1}} g_n(t)dt = \tilde{\mathbf{d}}^T \mathbf{g}_k, \quad (3.49)$$

where $\tilde{\mathbf{d}} = [\tilde{d}_0, \tilde{d}_1, \cdots, \tilde{d}_{N-1}]^T$ and

$$\mathbf{g}_k = \begin{bmatrix} \int_{tk}^{t_{k+1}} g_0(t)dt \\ \int_{tk}^{t_{k+1}} g_1(t)dt \\ \vdots \\ \int_{tk}^{t_{k+1}} g_{N-1}(t)dt \end{bmatrix}. \quad (3.50)$$

The second term of (3.47) can be written as

$$\int_{tk}^{t_{k+1}} x^2(t)dt = \int_{tk}^{t_{k+1}} \Big(\sum_{n=0}^{N-1} \tilde{d}_n g_n(t)\Big)^2 dt =$$

$$= \begin{bmatrix} \tilde{d}_0 \\ \tilde{d}_1 \\ \cdots \\ \tilde{d}_{N-1} \end{bmatrix}^T \left[\int_{tk}^{t_{k+1}} \begin{bmatrix} g_0(t) & g_0(t) & \cdots & g_0(t) \\ g_1(t) & g_1(t) & \cdots & g_1(t) \\ \vdots & \vdots & \ddots & \vdots \\ g_{N-1}(t) & g_{N-1}(t) & \cdots & g_{N-1}(t) \end{bmatrix} \circ \begin{bmatrix} g_0(t) & g_1(t) & \cdots & g_{N-1}(t) \\ g_0(t) & g_1(t) & \cdots & g_{N-1}(t) \\ \vdots & \vdots & \ddots & \vdots \\ g_0(t) & g_1(t) & \cdots & g_{N-1}(t) \end{bmatrix} dt \right] \begin{bmatrix} \tilde{d}_0 \\ \tilde{d}_1 \\ \vdots \\ \tilde{d}_{N-1} \end{bmatrix} =$$

$$= \begin{bmatrix} \tilde{d}_0 \\ \tilde{d}_1 \\ \cdots \\ \tilde{d}_{N-1} \end{bmatrix}^T \begin{bmatrix} \int_{tk}^{t_{k+1}} g_0(t)g_0(t)dt & \int_{tk}^{t_{k+1}} g_0(t)g_1(t)dt & \cdots & \int_{tk}^{t_{k+1}} g_0(t)g_{N-1}(t)dt \\ \int_{tk}^{t_{k+1}} g_1(t)g_0(t)dt & \int_{tk}^{t_{k+1}} g_1(t)g_1(t)dt & \cdots & \int_{tk}^{t_{k+1}} g_1(t)g_{N-1}(t)dt \\ \vdots & \vdots & \ddots & \vdots \\ \int_{tk}^{t_{k+1}} g_{N-1}(t)g_0(t)dt & \int_{tk}^{t_{k+1}} g_{N-1}(t)g_1(t)dt & \cdots & \int_{tk}^{t_{k+1}} g_{N-1}(t)g_{N-1}(t)dt \end{bmatrix} \begin{bmatrix} \tilde{d}_0 \\ \tilde{d}_1 \\ \vdots \\ \tilde{d}_{N-1} \end{bmatrix} =$$

$$= \tilde{\mathbf{d}}^T \cdot \hat{\mathbf{G}}_k \cdot \tilde{\mathbf{d}} \tag{3.51}$$

where $\circ$ denotes the Hadamard product, but $\hat{G}_{k_{mn}} = \int_{tk}^{t_{k+1}} g_m(t) g_n(t) dt$.

From (3.47), (3.49) and (3.51) the final equation corresponding to the time interval $t \in [t_k, t_{k+1}]$ follows **[29]**:

$$\tilde{\mathbf{d}}^T \cdot \mathbf{g}_k + (-1)^k \tilde{\mathbf{d}}^T \cdot \hat{\mathbf{G}}_k \cdot \tilde{\mathbf{d}} = \tilde{q}_k. \tag{3.52}$$

As there are total $K-1$ time intervals, then $K-1$ equations (3.52) are obtained, and the unknown coefficients $\tilde{\mathbf{d}}$ are found by minimizing the total error value **[29]**:

$$\sum_{k=1}^{K-1} \Big(\tilde{\mathbf{d}}^T \cdot \mathbf{g}_k + (-1)^k \tilde{\mathbf{d}}^T \cdot \hat{\mathbf{G}}_k \cdot \tilde{\mathbf{d}} - \tilde{q}_k \Big)^2. \tag{3.53}$$

The question is what base functions $g_n(t)$ to choose for representing the input signal $x(t)$, the bandwidth of which is limited to $\omega \in [-\Omega, \Omega]$. In Section 2.2 and Section 3.1.2, the base functions $g_n(t)$ were chosen to be sinc functions: $g_n(t) = \mathrm{sinc}(\Omega(t - \tau_n))$ [156], [163], however, two problems exist: 1) these functions are well suited for representing only time-unlimited signals; 2) calculation of $\mathbf{g}_k$ and $\hat{\mathbf{G}}_k$ is both time consuming and not perfectly precise since no analytical solutions of the integrals exist. In order to solve these problems, in [162], periodic sinc functions and integral signal representation is presented. The proposition of this work is to use Fourier series instead of sinc functions for the original signal representation, in order to increase speed of reconstruction even more, while keeping the precision in the same level. This approach is described in next Section 3.2.2.2.

3.2.2.2 Fast Signal Recovery

By using Fourier series, given the output sequence $\{t_k\}_{k=1,2,\ldots,K}$ with the corresponding time period $\Theta = t_K - t_1$, the input signal for $t \in [t_1, t_K]$ is expressed as

$$x(t) = \tilde{d}_0 + \sum_{m=1}^{M} \Big(\tilde{d}_m \cos(m\frac{2\pi}{\Theta}t) + \tilde{d}_{m+M} \sin(m\frac{2\pi}{\Theta}t) \Big), \tag{3.54}$$

where the upper limit M follows from the bandwidth Ω of the signal **[29]**:

$$M = \Big\lfloor \frac{\Omega\Theta}{2\pi} \Big\rfloor. \tag{3.55}$$

Such a representation of $x(t)$ is both well suited for expressing time-limited signals of length Θ and allows fast and precise calculation of $\mathbf{g}_k$ and $\hat{\mathbf{G}}_k$ (see simulation results in Section 4.1.3.2), the expressions of which are given in Appendix A.

3.2.2.3 Real-Time Signal Recovery

The real-time signal reconstruction in ”AA-ASDM without additional envelope encoding” is carried out in short time intervals as described in Section 2.2.3. The principle stays the same, only the signal reconstruction inside each of the windows w_m is different. [29] In this case a new signal recovery method, described in previous subsections (Section 3.2.2.1 and Section 3.2.2.2), is used instead of method described in Section 2.2.1 and Section 2.2.2.

3.3 Comparison of number of samples

Given a signal fragment $x(t)$, $t \in [0, \Theta]$, a number N_{ASDM} of switching instants $0 \leq t_0 < t_1 < \ldots < t_{N_{ASDM}} \leq \Theta$, where $t_0 - 0 < T$ and $\Theta - t_{N_{ASDM}} < T$, at the output of ASDM can be found from equation (2.3) by dividing its both sides by $(-1)^k$:

$$\int_{t_k}^{t_{k+1}} [(-1)^k x(t) + b] dt = 2\kappa\delta. \tag{3.56}$$

After summing the left side of (3.56) over all $k = 0, 1, \ldots, N_{ASDM} - 1$, it is obtained:

$$b(t_{N_{ASDM}} - t_0) + \sum_{k=0}^{N_{ASDM}-1} \int_{t_k}^{t_{k+1}} (-1)^k x(t) dt = 2\kappa\delta N_{ASDM}, \tag{3.57}$$

where $\Theta - 2T < t_{N_{ASDM}} - t_0 \leq \Theta$, therefore, if $\Theta \gg T$, then $t_{N_{ASDM}} - t_0 \approx \Theta$ and

$$N_{ASDM} \approx \frac{1}{2\kappa\delta} \Big(b\Theta + \sum_{k=0}^{N_{ASDM}-1} \int_{t_k}^{t_{k+1}} (-1)^k x(t) dt \Big). \tag{3.58}$$

By denoting $\bar{x}_{abs} = \frac{1}{\Theta} \int_0^{\Theta} |x(t)| dt$ and considering $-\bar{x}_{abs}\Theta \leq \sum_{k=0}^{N_{ASDM}-1} \int_{t_k}^{t_{k+1}} (-1)^k x(t) dt \leq \bar{x}_{abs}\Theta$, it follows:

$$\frac{\Theta}{2\kappa\delta}(b - \bar{x}_{abs}) \leq N_{ASDM} \leq \frac{\Theta}{2\kappa\delta}(b + \bar{x}_{abs}), \tag{3.59}$$

where $b > \bar{x}_{abs}$. If b is much larger than $\bar{x}_{abs}$, then $N_{ASDM} \approx \Theta b / (2\kappa\delta)$.

In case of AA-ASDM a relationship between the output sequence t_k, $k \in Z$, and the input signal $x(t)$ holds:

$$\int_{t_k}^{t_{k+1}} [x(t) + (-1)^k c(t)]dt = (-1)^k 2\kappa\delta. \tag{3.60}$$

By similar procedure, if both sides of (3.60) are divided by $(-1)^k$ and then summed over all $k = 0, 1, \ldots, N_{AA-ASDM} - 1$, a number $N_{AA-ASDM}$ of switching instants $0 \leq t_0 < t_1 < \ldots < t_{N_{AA-ASDM}} \leq \Theta$ at the output of AA-ASDM is found:

$$N_{AA-ASDM} \approx \frac{1}{2\kappa\delta}\Big(\bar{c}\Theta + \sum_{k=0}^{N_{AA-ASDM}-1} \int_{t_k}^{t_{k+1}} (-1)^k x(t)dt\Big), \tag{3.61}$$

from which:

$$\frac{\Theta}{2\kappa\delta}(\bar{c} - \bar{x}_{abs}) \leq N_{AA-ASDM} \leq \frac{\Theta}{2\kappa\delta}(\bar{c} + \bar{x}_{abs}), \tag{3.62}$$

where $\bar{c} = \frac{1}{\Theta}\int_0^{\Theta} c(t)dt$ is the mean value of $c(t)$.

From (3.59) and (3.62) a ratio between the minimum value of N_{ASDM} and the maximum value of $N_{AA-ASDM}$ can be obtained:

$$\frac{N_{ASDM\,min}}{N_{AA-ASDM\,max}} = \frac{b - \bar{x}_{abs}}{\bar{c} + \bar{x}_{abs}}. \tag{3.63}$$

By introducing the coefficients $\gamma_x = \bar{x}_{abs}/C$ and $\gamma_c = \bar{\tilde{o}}/C$, where $\bar{\tilde{o}} = \frac{1}{\Theta}\int_0^{\Theta} \tilde{o}(t)dt$ is the mean value of $\tilde{o}(t)$, then b and $\bar{c}$ can be expressed as $b = C + \alpha C = \bar{x}_{abs}(1 + \alpha)/\gamma_x$ and $\bar{c} = \bar{\tilde{o}} + \alpha C = \gamma_c C + \alpha C = \bar{x}_{abs}(\gamma_c + \alpha)/\gamma_x$, which, after inserting into (3.63), gives:

$$\frac{N_{ASDM\,min}}{N_{AA-ASDM\,max}} = \frac{1 - \gamma_x + \alpha}{\gamma_c + \gamma_x + \alpha}. \tag{3.64}$$

The coefficient γ_x is solely determined by $x(t)$ and its values are bounded: $\gamma_x \in (0, 1]$, since $\bar{x}_{abs} \leq C$. On the other hand, the coefficient γ_c additionally depends on the chosen $\tilde{o}(t)$ and its values $\gamma_c \geq \gamma_x$, since $\tilde{o}(t) \geq |x(t)|$. In AA-ASDM2 case, the maximum ratio is 4.

If the mean value $\bar{x}_{abs}$ is small in comparison to the maximum value C of $|x(t)|$, then $1 - \gamma_x$ in (3.64) is close to 1 and for small $\gamma_c \approx \gamma_x$ and α values the number $N_{AA-ASDM}$ will be significantly less than N_{ASDM}. In addition, $N_{ASDM\,min} > N_{AA-ASDM\,max}$ for all α, if $\gamma_c < 1 - 2\gamma_x$. It means that AA-ASDM is advantageous over ASDM for signals having high peak-to-peak amplitudes in comparison to their mean absolute values.

As α grows, the ratio (3.64) for small $\gamma_c \approx \gamma_x$ values decreases towards 1, however, since the trigger times t_k are measured with finite precision, increasing α values also reduce signal reconstruction quality.

3.4 Summary and Conclusions

As indicated in the conclusions of Section 2, Asynchronous Sigma-Delta modulator (ASDM) circuit is inefficient when applied to wide dynamic range signals, such as electroencephalogram (EEG) signals, i.e. an unnecessary high switching activity of ASDM circuit appears when the input signal amplitude is low, causing increased power consumption of the wireless Brain Computer Interface (BCI) system. This is due to ASDM circuit parameters are chosen considering the maximum value that is never exceeded by the signal. Since envelope of the signal changes over time, the proposition made in this Section is, instead of constant value, to use the time-varying maximum value, which is also never exceeded by the signal, i.e. to change the ASDM circuit parameter b according to $c(t)$ to ensure that maximum distance between two consecutive trigger switching times is:

$$\tau_{max2}(t) = \frac{2\kappa\delta_2}{b_2(t) - c(t)} = const. = T$$

In this case, the difference $b_2(t) - c(t)$ must be constant and thus $b_2(t)$ can be written as $b_2(t) = c(t) + \beta C$, where $\beta > 0$. The minimum distance becomes

$$\tau_{min2}(t) = \frac{2\kappa\delta_2}{b_2(t) + c(t)} = \frac{T}{1 + 2c(t)/(\beta C)}$$

The proposed method, called Amplitude Adaptive Asynchronous Sigma-Delta modulator (AA-ASDM) allows to reduce the over-triggering of the circuit and thus the power consumption of wireless BCI system. Regardless of this reduction the perfect recovery of the original signal from the obtained time sequence is still possible.

To implement this idea, in addition to ASDM circuit, there is an envelope detector with output $c(t)$ connected to the feedback loop. Now, the relationship between the switching instants t_k of the AA-ASDM output $z(t)$ and the input signal $x(t)$ for $t_{k+1} > t_k$, and integers $k \geq 0$, is given by the following equation:

$$\int\limits_{t_k}^{t_{k+1}} x(t)dt = (-1)^k[2\kappa\delta - \beta C(t_{k+1} - t_k) - \int\limits_{t_k}^{t_{k+1}} c(t)dt].$$

Based on the above mentioned, two amplitude adaptive methods: 1) AA-ASDM with additional envelope encoding; and 2) AA-ASDM without additional envelope encoding, are proposed and described.

AA-ASDM with additional envelope encoding

In this case, the envelope signal $c(t)$ is also needed for recovery of $x(t)$, therefore it is encoded by another ASDM. Now, for AA-ASDM to be advantageous over ASDM, the signal $c(t)$ must have low frequencies in comparison to $x(t)$.

In order to recover the encoded signal, the first step is recovery of envelope signal $c(t)$. When $c(t)$ is found, the original signal $x(t)$ is recovered from the given time sequence t_k by finding the unknown coefficients by the following equation:

$$\hat{\mathbf{a}} = \mathbf{G}^{+}\mathbf{q},$$

where $q_k = (-1)^k(2\kappa\delta - \beta C(t_{k+1} - t_k) - \int_{t_k}^{t_{k+1}} c(t)dt)$ and $G_{kn} = \int_{t_k}^{t_{k+1}} g(t - \tau_n)dt$.

By using this signal recovery method, the reconstruction is time and resource consuming therefore fast and real-time signal reconstruction methods are considered.

In order to increase the speed of signal reconstruction, instead of finding the coefficients $\hat{a}_n$, it is more efficient to find the coefficients $\hat{d}_n$, which correspond to integral signal representation:

$$\varsigma(t) = \int_{-\infty}^{t} x(u)du = \sum_{n \in Z} \hat{d}_n g(t - \tau_n)$$

The unknown coefficients $\hat{d}_n$ are obtained as:

$$\hat{\mathbf{d}} = \mathbf{G}^{+}\mathbf{P}^{-1} \cdot (\tilde{\mathbf{q}} + \tilde{\mathbf{P}}\mathbf{H}\hat{\mathbf{o}}) = \mathbf{G}^{+}\mathbf{P}^{-1} \cdot (\tilde{\mathbf{q}} + \tilde{\mathbf{P}}\mathbf{H}\hat{\mathbf{H}}^{+}\hat{\mathbf{P}}^{-1}\hat{\mathbf{q}}).$$

When all coefficients $\hat{d}_n$ are found, $\varsigma(t)$ can be calculated as $\varsigma(t) = \sum_{n=1}^{N} \hat{d}_n g(t - \tau_n)$, and then, original signal can be found as

$$x(t) = \frac{d\varsigma(t)}{dt}.$$

It is also shown that speed of signal reconstruction can be increased even more by fastening matrix **G** and **H** pseudoinversion (see Section 4.1.1.2).

In order to reconstruct the envelop as well as the original signal in real-time, the reconstruction is carried out in short time intervals. When the first interval of $c(t)$ is reconstructed, the reconstruction of $x(t)$, in short time intervals, can start. If $c(t)$ frequency is, for example, twenty times lower than $x(t)$ frequency, it is possible to reconstruct twenty $x(t)$ intervals by knowing one $c(t)$ interval. The main drawbacks of this method are an increased delay at the beginning of the conversion and a demand for increased computational resources.

Although ”AA-ASDM with additional envelop encoding” is advantageous over ASDM in terms of switching activity, the efficiency of this method can be improved, if the time-varying envelope of the signal is not additionally encoded and transmitted in order to recover the original signal.

AA-ASDM without additional envelope encoding

The relationship between the switching instants t_k of the AA-ASDM output $z(t)$ and the input signal $x(t)$ is the same as in ”AA-ASDM with additional envelope encoding”, only in this case, the proposition is to use the time-varying envelope function as $\tilde{o}(t) = 0.25 + x^2(t)$, where the inequality $\tilde{o}(t) \geq |x(t)|$ holds for all $x(t) \in [-1, 1]$. This function does not require any additional encoding scheme to be able to reconstruct the original signal with the same precision as with the ASDM (see Section 4.1.3).

The input signal $x(t)$ can be represented as $x(t) = \sum_{n=0}^{N-1} \tilde{d}_n g_n(t)$, where $\tilde{d}_n$ are unknown coefficients and $g_n(t)$ are the chosen base functions. The unknown coefficients $\tilde{\mathbf{d}}$ are found by minimizing the total error value:

$$\sum_{k=1}^{K-1} \left(\tilde{\mathbf{d}}^T \cdot \mathbf{g}_k + (-1)^k \tilde{\mathbf{d}}^T \cdot \hat{\mathbf{G}}_k \cdot \tilde{\mathbf{d}} - \tilde{q}_k \right)^2 .$$

The question was what base functions $g_n(t)$ to choose for representing the input signal $x(t)$. If the base functions $g_n(t)$ are chosen to be sinc functions, as in ASDM and ”AA-ASDM with additional envelop encoding” case, two problems occur: 1) these functions are not well suited for representing time-limited signals; 2) calculation of $\mathbf{g}_k$ and $\hat{\mathbf{G}}_k$ is both time consuming and not perfectly precise since no analytical solutions of the integrals exist. In order to solve these problems, the proposition of this work is to use Fourier series instead of sinc functions for the original signal representation.

By using Fourier series, given the output sequence $\{t_k\}_{k=1,2,\ldots,K}$ with the corresponding time period $\Theta = t_K - t_1$, the input signal for $t \in [t_1, t_K]$ is expressed as

$$x(t) = \tilde{d}_0 + \sum_{m=1}^{M} \left(\tilde{d}_m \cos(m \frac{2\pi}{\Theta} t) + \tilde{d}_{m+M} \sin(m \frac{2\pi}{\Theta} t) \right),$$

where the upper limit M follows from the bandwidth Ω of the signal $M = \left\lfloor \frac{\Omega\Theta}{2\pi} \right\rfloor$. Such a representation of $x(t)$ is both well suited for expressing time-limited signals of length Θ and allows fast and precise calculation of $\mathbf{g}_k$ and $\hat{\mathbf{G}}_k$. The real-time signal reconstruction is carried out by using the same approach as in ”AA-ASDM with additional envelope encoding” case.

”AA-ASDM without additional envelope encoding” allows not only to reduce the switching activity of the ASDM circuit even more, compared to ”AA-ASDM with additional envelope encoding”, but it also decreases the speed and delay of the signal reconstruction meanwhile keeping the desired precision (see Section 4.1.3).

The ratio between the minimum value of a number N_{ASDM} of switching time instants and the maximum value of $N_{AA-ASDM}$ can be obtained from the following expression:

$$\frac{N_{ASDM\,min}}{N_{AA-ASDM\,max}} = \frac{1 - \gamma_x + \alpha}{\gamma_c + \gamma_x + \alpha},$$

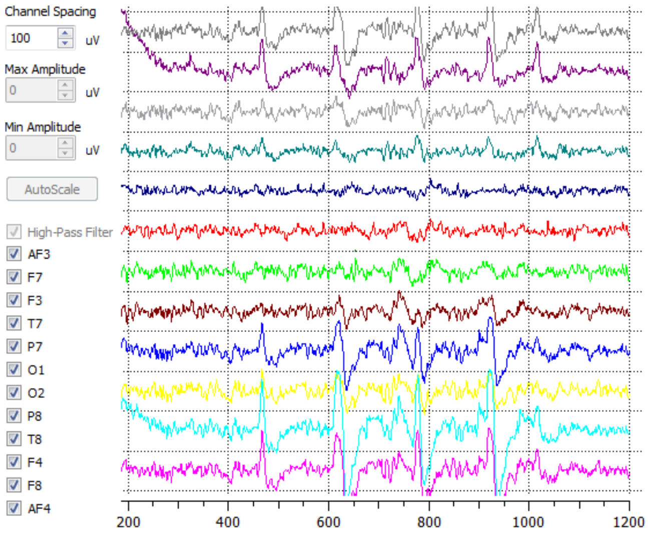
where $\gamma_x = \bar{x}_{abs}/C$ and $\gamma_c = \bar{\bar{o}}/C$ are coefficients. It follows that AA-ASDM is advantageous over ASDM for signals having high peak-to-peak amplitudes in comparison to their mean absolute values.

4. EXPERIMENTAL RESEARCH

In order to verify and assess the theory, developed in Section 3, in practice, in this Section various simulations, modeling and physical implementations are carried out. First of all, in Section 4.1, Asynchronous Sigma-Delta Modulator (ASDM) (as a reference design) and Amplitude Adaptive Asynchronous Sigma-Delta Modulator (AA-ASDM) are simulated in *Matlab* numerical computing environment. Then, in Section 4.2, ASDM and AA-ASDM are modeled in *SIMetrix* circuit simulation software. In Section 4.3, AA-ASDM together with other Brain Computer Interface (BCI) system's components are designed in *Altium Designer* Printed Circuit Board (PCB) design software and finally constructed as physical devices. In each of these subsections, a detailed description of tests as well as corresponding results are given. In addition, at the end of Section 4, a summary and conclusions are given.

4.1 Simulations

In this sub-section, both ASDM and AA-ASDM are simulated and tested in *Matlab* environment. Both approaches are tested on real electroencephalogram (EEG) signals acquired by the 14-channel *Emotiv EPOC* [9] headset. An excerpt from these signals can be seen in Fig. 4.1.



Channel Spacing 100 uV

Max Amplitude 0 uV

Min Amplitude 0 uV

☑ High-Pass Filter

☑ AF3

☑ F7

☑ F3

☑ T7

☑ P7

☑ O1

☑ O2

☑ P8

☑ T8

☑ F4

☑ F8

☑ AF4

Figure 4.1: EEG signals used for simulations

Considering typical EEG signal frequencies (see Section 1.1.2), all signals are low-pass filtered up to 49Hz ir order to remove the noise prior to encoding.

4.1.1 Asynchronous Sigma-Delta Modulator

Based on the theory, described in Section 2, ASDM *Matlab* functions for signal encoding, reconstruction, fast reconstruction and real-time reconstruction are developed, tested and assessed in this Section. All used *Matlab* functions can be found in Appendix B.

4.1.1.1 Signal Encoding

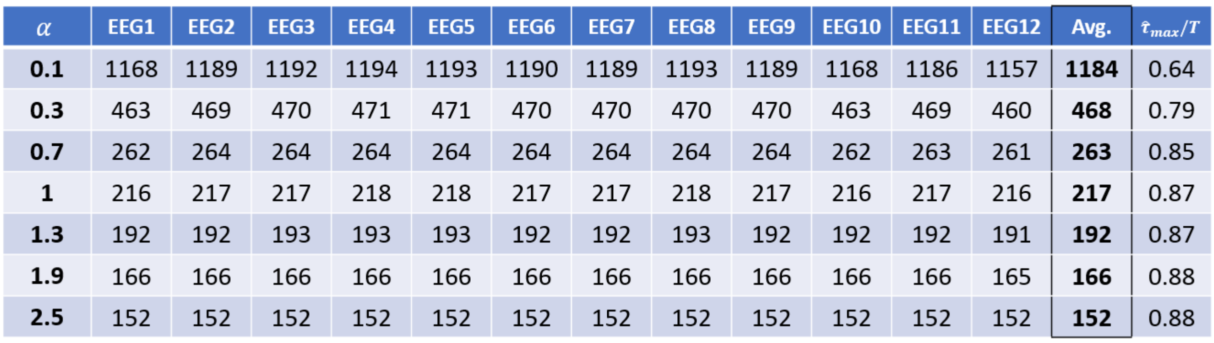
ASDM *Matlab* function for signal encoding is shown in Appendix B-1. First, considering inequation (2.4) $\tau_{min} = \frac{2\kappa\delta}{b+c} \leq t_{k+1} - t_k \leq \frac{2\kappa\delta}{b-c} = \tau_{max}$, the parameters of the ASDM must be set in such a way that maximum distance between two consecutive trigger switchings $t_{k+1} - t_k$ does not exceed the Nyquist step, i.e, $\tau_{max} \leq \frac{1}{2F_{max}} = T$. In this case, ASDM parameters are set as follows: $\kappa = 1, b = c + \alpha \cdot c$, where α is a constant (0.1, 0.3, 0.7, 1, 1.3, 1.9 or 2.5) and c is a maximum amplitude of absolute value of the EEG signal, but δ is calculated as

$$\delta = 0.9\frac{(b-c)T}{2\kappa} = 0.9\frac{\alpha c \frac{1}{2F_{max}}}{2\kappa} = 0.9\frac{\alpha c}{4F_{max}}, \tag{4.1}$$

where ”0.9” is added due to computing inaccuracies, otherwise, there might be occasions when actual measured maximum distances between two trigger switching time instants is $\hat{\tau}_{max} > T$. It introduces a small over-triggering, but ensures that $\hat{\tau}_{max}$ is less than T. **[27]**, **[29]**

The simulation results for different α values and EEG signals are shown in Table 4.1.

Table 4.1: Number of switching time instants per sec. for different α values and EEG signals



α	EEG1	EEG2	EEG3	EEG4	EEG5	EEG6	EEG7	EEG8	EEG9	EEG10	EEG11	EEG12	Avg.	$\hat{\tau}_{max}/T$
0.1	1168	1189	1192	1194	1193	1190	1189	1193	1189	1168	1186	1157	**1184**	0.64
0.3	463	469	470	471	471	470	470	470	470	463	469	460	**468**	0.79
0.7	262	264	264	264	264	264	264	264	264	262	263	261	**263**	0.85
1	216	217	217	218	218	217	217	218	217	216	217	216	**217**	0.87
1.3	192	192	193	193	193	192	192	193	192	192	192	191	**192**	0.87
1.9	166	166	166	166	166	166	166	166	166	166	166	165	**166**	0.88
2.5	152	152	152	152	152	152	152	152	152	152	152	152	**152**	0.88

As can be seen in Table 4.1, for low α values the average switching activity is high which is not recommended due to increased power consumption, but for too high α values variance of distances between consecutive trigger switching time instants reduces and more precision (more bits) is needed to measure the distances. The optimal choice could be $\alpha = 1$, when the minimum distance between consecutive trigger times is $T/3$. [27], [29]

The distances between consecutive trigger switching time instants when $\alpha = 1$ are shown in Fig. 4.2.

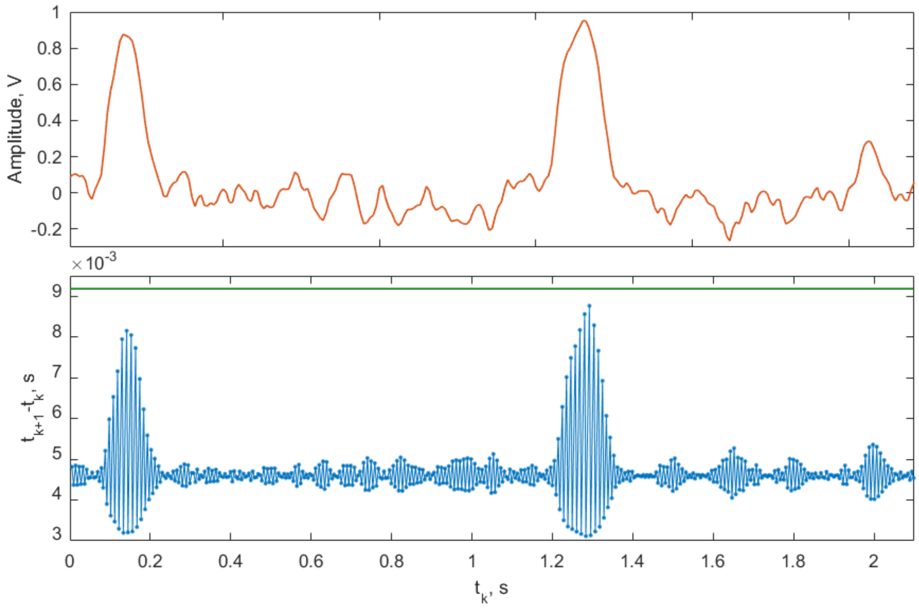


Figure 4.2: EEG signal (red line) and the obtained ASDM distances between consecutive trigger switching time instants (blue line) and Nyquist step (T) multiplied by 0.9 (green line)

As can be seen in Fig. 4.2, the maximum distances (equaling nearly T (green line)) are obtained only when the input EEG signal reaches its highest values at ≈0.15 and ≈1.3 seconds. Due to constant parameter b, the over-triggering occurs most of the time and the distances vary closely around $T/2$. This over-triggering causes an increased power consumption of a BCI system.

As can be seen in Table 4.1, $\hat{\tau}_{max}/T$ values are close to $0.9 \cdot T$, except for $\alpha = 0.1$ and $\alpha = 0.3$, where this parameter is less than 0.9. This is due to fact that the maximum absolute value c of the original signal is set to be above the actual maximum value, since in real life systems it is only normal to set the maximum and minimum amplitude values to be above the range of the expected input signal.

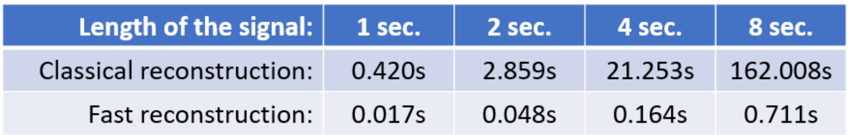
Since the maximum distance between two consecutive ASDM trigger switching time instants does not exceed Nyquist step, a sufficient condition to reconstruct the original signal from switching time instants t_k is fulfilled (see next sub-section).

4.1.1.2 Signal Recovery

As shown in Section 2.2, ASDM encoded signal can be recovered by using two approaches. The first one (*Classical reconstruction*) is based on expressions (2.20) and (2.22), and the second one (*Fast reconstruction*) on expressions (2.41) and (2.46). In order to reconstruct the signal in real time, an expressions (2.49) is used. *Matlab* functions for each of the approaches are shown in Appendixes B-2, B-3 and B-4 accordingly.

Experimental simulation results, when $\alpha = 1$, show that in average 0.420 seconds are required for *Classical reconstruction* algorithm and 0.017 seconds for *Fast reconstruction algorithm* to recover a 1 second long signal on a mid-range personal computer (PC) (see Table 4.2).

Table 4.2: Comparison of algorithms (Average time necessary to reconstruct the signal)



Length of the signal:	1 sec.	2 sec.	4 sec.	8 sec.
Classical reconstruction:	0.420s	2.859s	21.253s	162.008s
Fast reconstruction:	0.017s	0.048s	0.164s	0.711s

Since the reconstruction speed (time) can vary depending on the performance of the PC, the time values are compared in relative terms. In this case, *Fast reconstruction* algorithm is ≈ 25 times faster then *Classical reconstruction* algorithm, if the signal length is 1 second, and ≈ 61, ≈ 130 and ≈ 228 times faster, if the length of the signal is 2, 4 and 8 seconds, respectively. **[28]**

In order to evaluate the accuracy of both signal reconstruction methods, the average Signal-to-Noise radio (SNR) is calculated based on the simulation results. For *Classical reconstruction*:

$$SNR_{ASDM_C} = 10log_{10}\frac{P_s}{P_e} = 10log_{10}(\frac{E[x(t)]}{E[x(t)-\hat{x}(t)]}) = 132.33\ dB, \tag{4.2}$$

where E denotes the energy of the signal, $x(t)$ is the original signal, but $\hat{x}(t)$ the reconstructed signal. From expression (4.2), it follows that Effective Number of Bits are

$$ENOB_{ASDM_C} = \frac{SNR - 1,76}{6,02} = \frac{132.33 - 1,76}{6,02} \approx 22\ bits. \tag{4.3}$$

Accordingly, for *Fast reconstruction*: $SNR_{ASDM_F} = 131.49\ dB$, and $ENOB_{ASDM_F} \approx 22\ bits$.

Fig. 4.3 shows an example of the original signal and the reconstructed signal (visually it is not possible to see the difference), as well as the error signal, defined as the difference between the original and reconstructed signal.

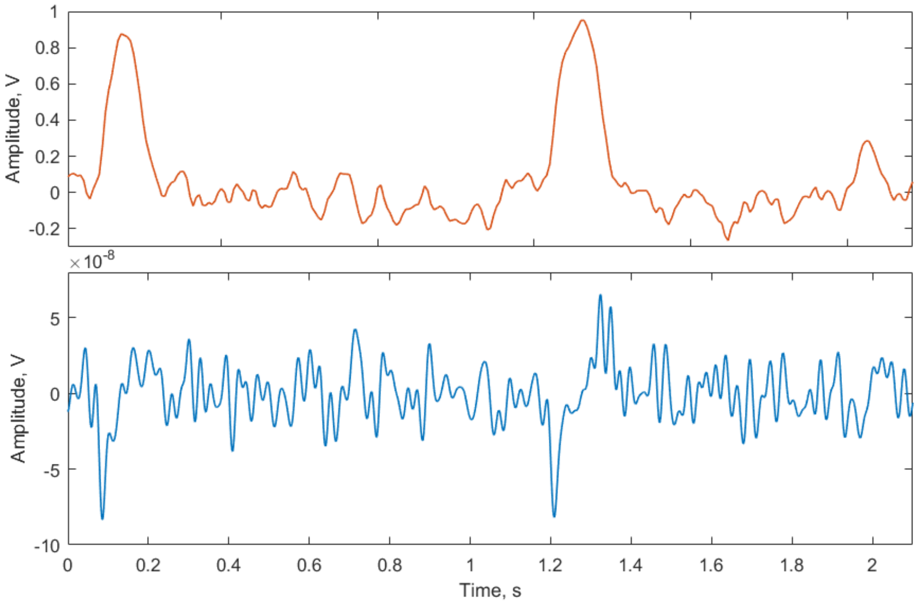


Figure 4.3: Original EEG signal (red line), reconstructed signal (green line) and error signal (blue line), which is the difference between original and reconstructed signal

As can be seen in Fig. 4.3 a), it is impossible to see the difference between the original and reconstructed signal due to fact that the error signal is in a range of 10^{-8}V.

The simulation results show that by using *Fast reconstruction* algorithm the reconstructed signal is slightly less accurate ($SNR_{ASDM_C} = 132.33\ dB$, $SNR_{ASDM_F} = 131.49\ dB$), but the time it takes to reconstruct the original signal is significantly faster (see Table 4.2).

In order to reconstruct the signal in real time, the reconstruction is carried out in short time intervals as described in Section 2.2.3. In this case, the parameters are chosen as follow: $L = 15$, $M = 2$, $K = 2$ and $J = L - 2M - K = 9$, which means that each time after receiving nine switching time instants t_k, a new signal fragment reconstruction begins (see Fig. 4.4). **[29]**

In this particular example, nine ASDM trigger switchings time instants are generated in $70ms$, but the signal reconstruction takes only $40ms$, which means that by using this method it is possible to reconstruct the signal in real time.

By using real-time reconstruction method, the SNR is also affected. The simulation results

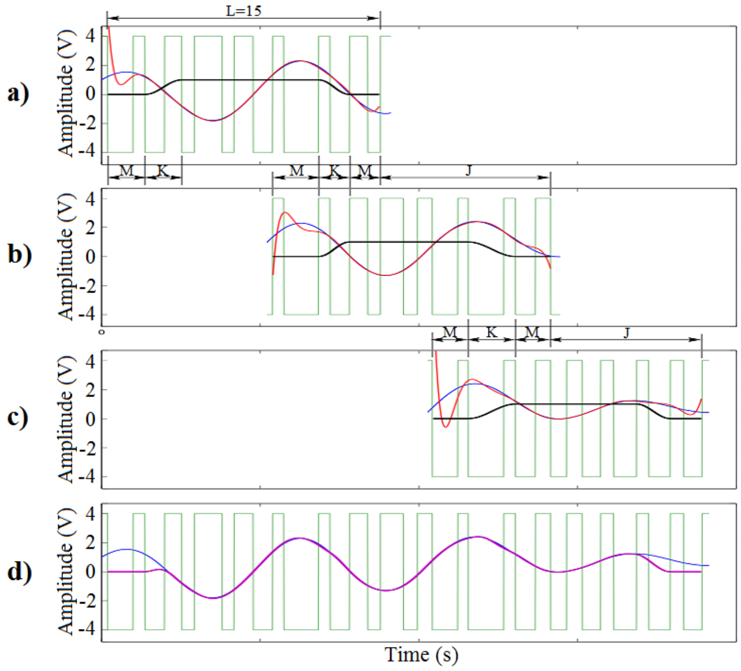


Figure 4.4: Reconstructed signal (d) by using three consecutive intervals: a), b) and c), where green line is ASDM output signal, red line - reconstructed signal in interval before it is multiplied by window function (black line), blue line - original signal (before encoding)

show that in this case the average SNR is:

$$SNR_{ASDM_{RT}} = 10log_{10}\frac{P_s}{P_e} = 10log_{10}(\frac{E[x(t)]}{E[x(t) - \hat{x}(t)]}) = 123.62\ dB, \tag{4.4}$$

while

$$ENOB_{ASDM_{RT}} = \frac{SNR - 1,76}{6,02} = \frac{123.62 - 1,76}{6,02} \approx 20\ bits. \tag{4.5}$$

4.1.2 AA-ASDM with additional envelope encoding

Based on the theory, described in Section 3.1, AA-ASDM with additional envelope encoding is simulated and assessed in this Section.

4.1.2.1 Signal Encoding and Reconstruction

Matlab function for signal encoding is shown in Appendix C-1. First, considering expressions (3.2) and (3.3), the parameters of the AA-ASDM must be set in such a way that maximum distance between two consecutive trigger switchings $t_{k+1} - t_k$ does not exceed the Nyquist step, i.e, $\tau_{max} \leq \frac{1}{2F_{max}}$. In this case, AA-ASDM parameters are set as follows: $\kappa = 1$, $b_2 = c(t) + \beta C$, where $\beta = \alpha$ is a constant (0.1, 0.3, 0.7, 1, 1.3, 1.9 or 2.5) and C is a maximum amplitude of absolute value of the EEG signal, but δ_2 is calculated from the equation $\delta_2 = 0.9\beta CT/2$. For envelope encoding, the parameters are set as $\kappa_c = 1$, $b_c = 2C$ and $\delta_c = C/4F_{max_c}$. **[27]**

The example of the input EEG signal and its estimated envelope function, when $\alpha = 0.1$, is shown in Fig. 4.5. In this case, the envelope function (solid blue line in Fig. 4.5) is acquired by finding peak values of the modulus of input signal, which are interpolated and then low pass filtered by a 5Hz filter. If more precise (more rapid) envelope (dashed line in Fig. 4.5) is used, then less triggering occurs at the output of the upper trigger (see Fig. 3.3), however, more switchings are required to encode the envelope.

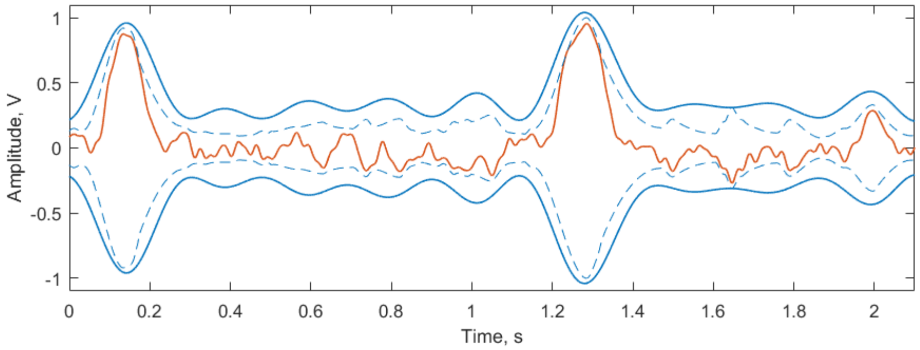


Figure 4.5: Input EEG signal (red line) and its low frequency envelope functions (blue lines)

The simulation results for different α values and EEG signals are shown in Table 4.3. **[29]**

Table 4.3: Total number of switching time instants per second for different $\alpha = \beta$ values for different EEG signals

$\alpha = \beta$	EEG1	EEG2	EEG3	EEG4	EEG5	EEG6	EEG7	EEG8	EEG9	EEG10	EEG11	EEG12	Avg.	τ_{max}/T
0.1	400	384	385	367	366	416	387	386	391	536	395	460	**406**	0.90
0.3	220	213	213	206	206	223	214	212	215	265	217	241	**220**	0.90
0.7	166	162	162	159	159	167	163	162	163	185	164	175	**166**	0.89
1	154	151	150	148	148	154	151	150	151	167	152	160	**153**	0.90
1.3	147	144	144	142	142	147	144	144	145	157	145	152	**146**	0.90
1.9	139	137	137	136	136	139	138	137	138	146	138	143	**139**	0.90
2.5	135	134	134	133	133	135	134	134	134	141	134	138	**135**	0.89

In this case the total number of switching time instants is composed of two numbers, the first number corresponds to the output of the upper trigger in Fig. 3.3, while the second to the output of the lower trigger. For example, for EEG1 signal, when $\alpha = \beta = 0.1$, the number of switching time instants of the upper trigger is 390, but for lower trigger 10, giving a total of 400 switchings per second. Since envelope signal is limited to 5Hz, for all coefficient $\alpha = \beta$ values and all EEG signals, for envelope encoding there are needed 10 switching time instants.

Comparing Table 4.1 and Table 4.3 it can be seen that it is possible to reduce the switching activity of ASDM, if instead of trigger output the time-varying envelope of the signal is used in the feedback loop of the circuit. This also can be seen in Fig. 4.6, where $\alpha = \beta = 0.1$.

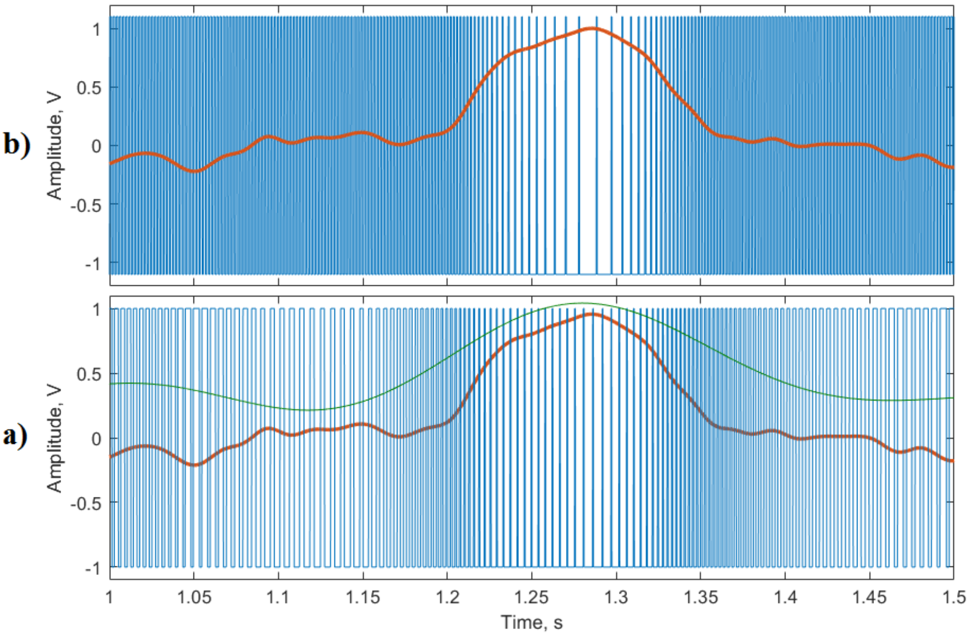


Figure 4.6: Fragment of the input EEG signal (red line) and the trigger output (blue line) in non-adaptive and amplitude adaptive cases. a) Switching activity of ASDM, b) Switching activity of AA-ASDM with additional envelope encoding

Knowing that the main power consumer of a BCI system is transmitter (see Section 1.1.3), by reducing the amount of information (switching time instants) to be transmitted, the power consumption of the transmitter will reduce proportionally. Thus, it is possible to calculate the energy saving, if AA-ASDM instead of ASDM is used **[29]** :

$$E = (1 - \frac{N_{\text{AA-ASDM}}}{N_{\text{ASDM}}}) \cdot 100\%, \tag{4.6}$$

where $N_{\text{AA-ASDM}}$ is the total number of switching time instants for AA-ASDM, but N_{ASDM} for ASDM. From Table 4.1 and Table 4.3 it can be calculated that AA-ASDM, for different $\alpha = \beta$ values, can achieve up to 65.69% energy saving for transmitter compared to ASDM (see further in the work Table 4.5). As the $\alpha = \beta$ values grow, the advantage (in switching activity) of AA-ASDM over ASDM becomes less, however, it is not recommended to choose too high these values since the variance of distances between consecutive trigger times reduces and more precision (more bits) is needed to measure the distances. On the other hand, too low $\alpha = \beta$ values are not recommended as well since high switching activity appears in both cases. [27]

The distances between consecutive trigger switching times when $\alpha = \beta = 1$ are shown in Fig. 4.7. The blue line corresponds to ASDM with the maximum distances (equaling nearly Nyquist step T) obtained when the signal $x(t)$ reaches its highest values at ≈0.15 and ≈1.3 seconds. Due to constant parameter $b = 2C$, the over-triggering occurs most of the time and the distances vary closely around $T/2$. On the contrary, the pink line, which corresponds to AA-ASDM with additional envelop encoding, shows that the maximum distances (equaling nearly $0.9 \cdot T$) occur all the time and the values $\hat{t}_{k+1} - \hat{t}_k$ are more spread. [27]

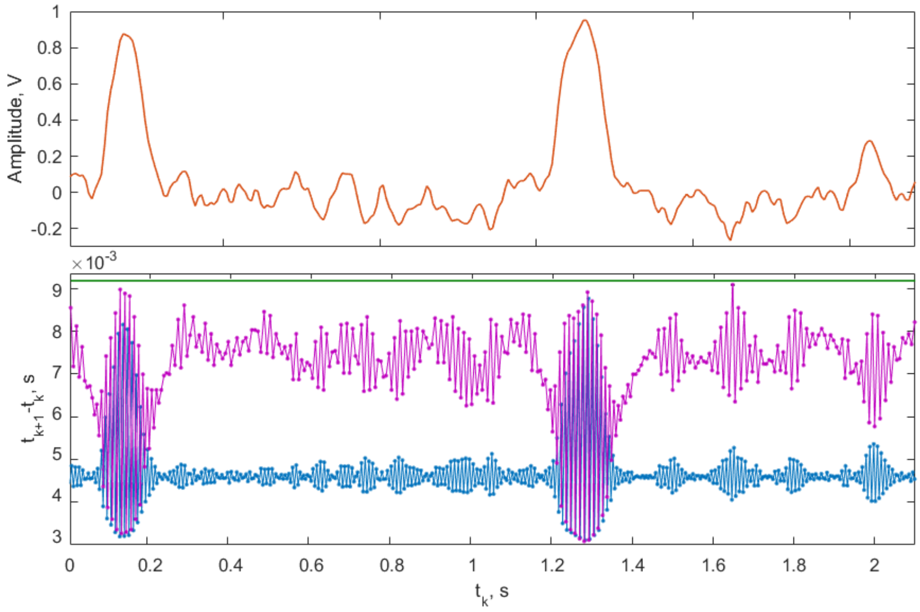


Figure 4.7: EEG signal (red line) and the obtained distances between consecutive trigger switching time instants for ASDM (blue line) and AA-ASDM (pink line), and Nyquist step (T) multiplied by 0.9 (green line)

In both cases, the maximum distance between consecutive trigger times is close and do not exceed $0.9 \cdot T$, which is a sufficient condition to reconstruct the original signal from the obtained reduced AA-ASDM time sequence. The signal reconstruction for AA-ASDM with additional envelop encoding is the same as in ASDM case (see Section 4.1.1) with the same accuracy.

Although AA-ASDM with additional envelop encoding is advantageous over ASDM in terms of number of samples, it has few drawbacks. For example, there is a need to transmit two signals instead of one and there is a delay introduced by additional envelope signal reconstruction prior to the original signal reconstruction. Even further, the physical implementation of the AA-ASDM with additional envelop encoding involves envelope detector which introduce additional delay to the system and synchronization with the original signal. Therefore, AA-ASDM without additional envelope encoding (see Section 4.1.3) is used in further research.

4.1.3 AA-ASDM without additional envelope encoding

Based on the theory, described in Section 3.2, ”AA-ASDM without additional envelope encoding” *Matlab* functions for signal encoding, fast and real-time reconstruction are developed, tested and assessed in this Section. All these *Matlab* functions are available in Appendix D.

4.1.3.1 Signal Encoding

AA-ASDM *Matlab* function for signal encoding is shown in Appendix D-1. Considering expressions (3.2) and (3.3), the parameters of the AA-ASDM must be set in a way that maximum distance between two consecutive trigger switchings $t_{k+1} - t_k$ does not exceed the Nyquist step, i.e, $\tau_{max} \leq \frac{1}{2F_{max}} = T$. AA-ASDM without additional envelope encoding parameters are set as in Section 4.1.2. In this case, the envelop function is set as $c(t) = 0.25 + x^2(t)$. **[29]**

The example of the input signal and its estimated envelope function, when $\alpha = 0.1$, is shown in Fig. 4.8, but simulation results of the average amount of triggering switching time instants

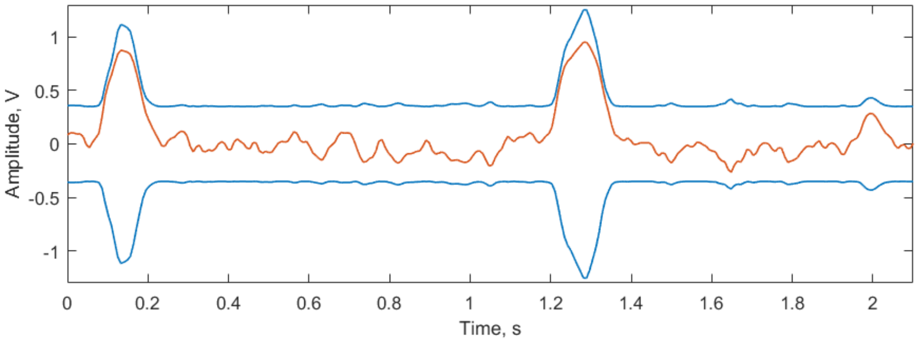


Figure 4.8: Input EEG signal (red line) and its envelope function (blue line)

per second for different α values and EEG signals are shown in Table 4.4.

Comparing Fig. 4.5 (AA-ASDM with additional envelope encoding (AA-ASDM1)) and Fig. 4.8 (AA-ASDM with additional envelope encoding (AA-ASDM2)), it can be seen that in AA-ASDM2 case the envelope is more precise and thus the number of trigger switching time instants is lower, as it can be also seen by comparing Table 4.1 and Table 4.4.

Table 4.4: Number of switching time instants per sec. for different α values and EEG signals

$\alpha = \beta$	EEG1	EEG2	EEG3	EEG4	EEG5	EEG6	EEG7	EEG8	EEG9	EEG10	EEG11	EEG12	Avg.	τ_{max}/T
0.1	353	365	376	380	380	379	376	376	369	354	352	367	**369**	0.89
0.3	200	199	201	201	201	201	201	201	200	200	197	205	**200**	0.90
0.7	152	150	150	150	150	150	150	150	150	152	150	155	**150**	0.90
1	140	138	138	138	138	138	138	138	138	140	138	142	**139**	0.89
1.3	134	132	131	131	131	131	131	131	132	134	132	136	**132**	0.90
1.9	127	125	125	125	125	125	125	125	125	127	125	128	**126**	0.90
2.5	123	122	122	121	121	121	122	122	122	123	122	124	**122**	0.90

The distances between consecutive trigger switching time instants, when $\alpha = \beta = 1$, are shown in Fig. 4.9. As can be seen in Fig. 4.9, the maximum distances (equaling nearly $0.9 \cdot T$

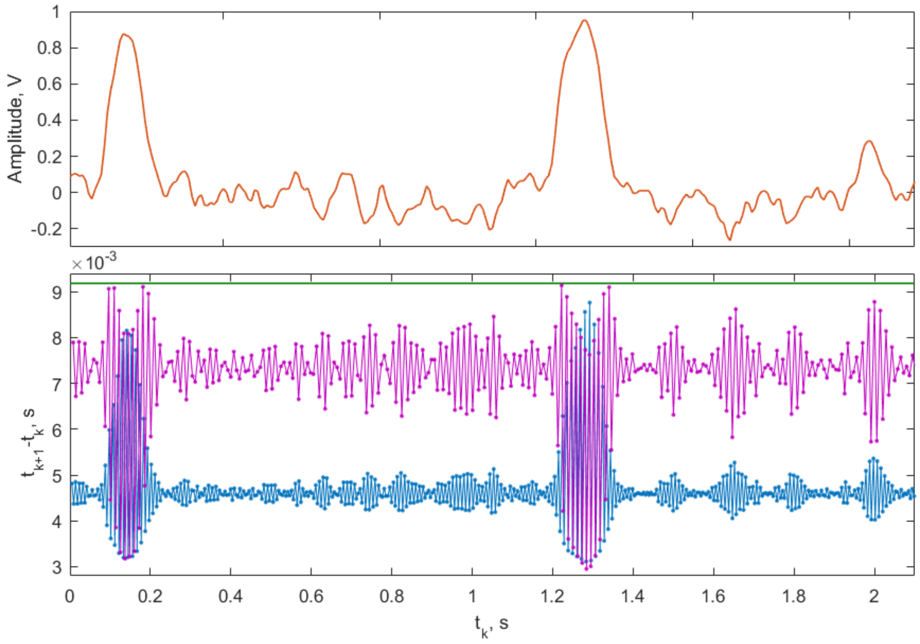
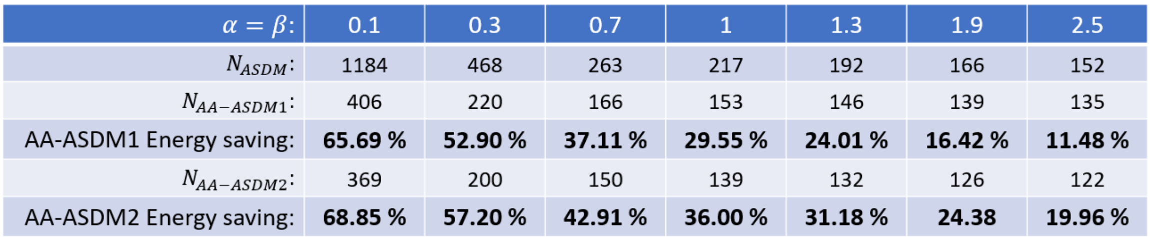


Figure 4.9: EEG signal (red line) and the obtained distances between consecutive trigger switching time instants for ASDM (blue line) and AA-ASDM (pink line), and Nyquist step (T) multiplied by 0.9 (green line)

(green line)) occur all the time (pink line), if compared to ASDM case (blue line), and the values $\hat{t}_{k+1} - \hat{t}_k$ are more spread. Since the maximum distance between consecutive trigger times does not exceed the Nyquist step, it is still possible to reconstruct the signal from the obtained reduced time sequence (see Section 4.1.3.2). [29]

As mentioned before, the main power consumer of a BCI system is a transmitter, where by reducing the average number of time codes needed to be transmitted, the average power consumption of the transmitter will reduce proportionally (see expression (4.7)). That means, the energy consumption of the whole BCI system reduces, if the average switching rate of the Schmitt trigger decreases. From expression (4.6), it is possible to calculate the energy saving, if AA-ASDM2 instead of ASDM is used. The simulation results show that it is possible to achieve up to 68.85% energy saving for transmitter compared to ASDM (see Table 4.5, where a comparison of ASDM, AA-ASDM1 and AA-ASDM2, based on Tables 4.1, 4.3 and 4.4 and expression (4.6) is shown).

Table 4.5: Comparison number of trigger switching time instants per second and corresponding energy saving of the transmitter for ASDM, AA-ASDM1 and AA-ASDM2



$\alpha = \beta$:	0.1	0.3	0.7	1	1.3	1.9	2.5
N_{ASDM}:	1184	468	263	217	192	166	152
$N_{AA-ASDM1}$:	406	220	166	153	146	139	135
AA-ASDM1 Energy saving:	**65.69 %**	**52.90 %**	**37.11 %**	**29.55 %**	**24.01 %**	**16.42 %**	**11.48 %**
$N_{AA-ASDM2}$:	369	200	150	139	132	126	122
AA-ASDM2 Energy saving:	**68.85 %**	**57.20 %**	**42.91 %**	**36.00 %**	**31.18 %**	**24.38**	**19.96 %**

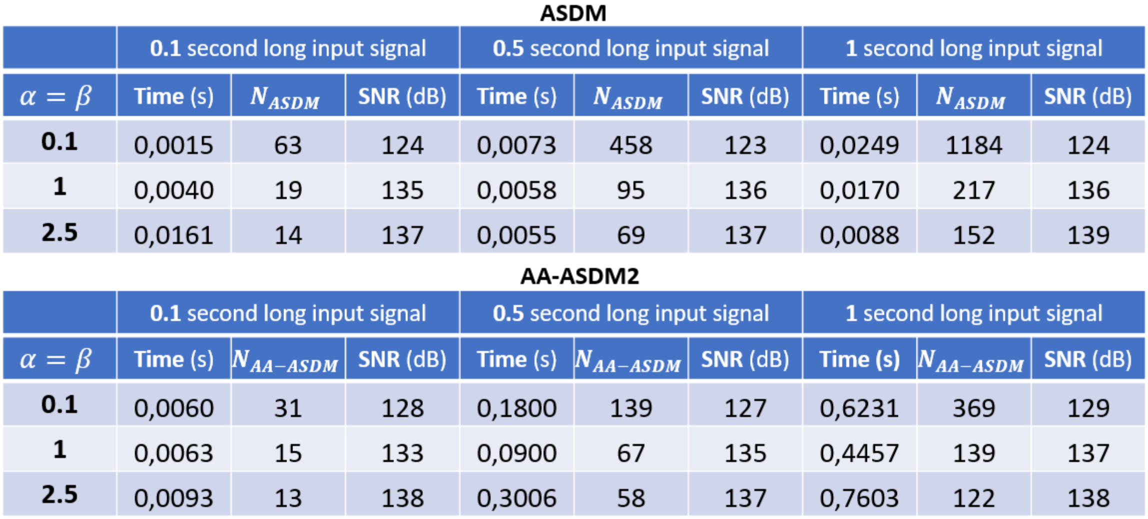
As the $\alpha = \beta$ values grow, the advantage (in switching activity) of AA-ASDM2 over ASDM becomes less, however, it is not recommended to choose too high these values since the variance of distances between consecutive trigger times reduces and more precision (more bits) is needed to measure the distances. Too low $\alpha = \beta$ values are not recommended as well since high switching activity appears in both cases.

4.1.3.2 Signal Recovery

Based on the Section 3.2.2.2, the simulation results of both fast and real-time signal reconstruction methods are shown in this section. *Matlab* functions for fast and real-time signal reconstruction are shown in Appendixes D-2 and D-3, accordingly.

For AA-ASDM, the experimental simulation results show, that it takes in average 0.09s to reconstruct one original signal fragment with a length of 0.5s, when $\alpha = \beta = 1$ (see Table 4.6)

Table 4.6: Comparison of ASDM and AA-ASDM2 reconstruction speed for different $\alpha = \beta$ values and signal lengths



ASDM

	0.1 second long input signal			0.5 second long input signal			1 second long input signal		
$\alpha = \beta$	Time (s)	N_{ASDM}	SNR (dB)	Time (s)	N_{ASDM}	SNR (dB)	Time (s)	N_{ASDM}	SNR (dB)
0.1	0,0015	63	124	0,0073	458	123	0,0249	1184	124
1	0,0040	19	135	0,0058	95	136	0,0170	217	136
2.5	0,0161	14	137	0,0055	69	137	0,0088	152	139

AA-ASDM2

	0.1 second long input signal			0.5 second long input signal			1 second long input signal		
$\alpha = \beta$	Time (s)	$N_{AA-ASDM}$	SNR (dB)	Time (s)	$N_{AA-ASDM}$	SNR (dB)	Time (s)	$N_{AA-ASDM}$	SNR (dB)
0.1	0,0060	31	128	0,1800	139	127	0,6231	369	129
1	0,0063	15	133	0,0900	67	135	0,4457	139	137
2.5	0,0093	13	138	0,3006	58	137	0,7603	122	138

As can be seen in Table 4.6, the reconstruction speed vary depending on the length of the fragment needed to be reconstructed. Also, it can be seen, that the signal reconstruction for AA-ASDM ir more time consuming than for ASDM. This is due to fact that the optimization algorithm, which minimizes the expression (3.53), is used to find the unknown coefficients needed for signal reconstruction. Besides that, it can be seen that by using AA-ASDM it is possible to achieve practically the same precision of the reconstructed signal as in the ASDM case. It can be concluded that the performance of AA-ASDM is admissible, since it fulfills the requirements set out in Section 1.2.

From Table 4.6, as well as from the conclusions at the end of Section 4.1.3.1, it is clear that the best option is to use $\alpha = \beta = 1$ and 0.5s long fragments to reconstruct the original signal with acceptable precision and reasonably small delay. Fig. 4.10 shows an example of the original signal and the reconstructed signal, as well as the error signal, defined as the difference between the original and reconstructed signal. The reconstruction is done by using real-time fragment reconstruction approach, described in Section 3.2.2.3 and shown in Fig. 4.4.

As can be seen in Fig. 4.10 b), the error signal lies within the range of 10^{-8}V. Usually the error increases in those regions where distance between two consecutive trigger switching time instants is closer to $0.9 \cdot T$. Still, as can be seen in Fig. 4.10 a), it is impossible to see the

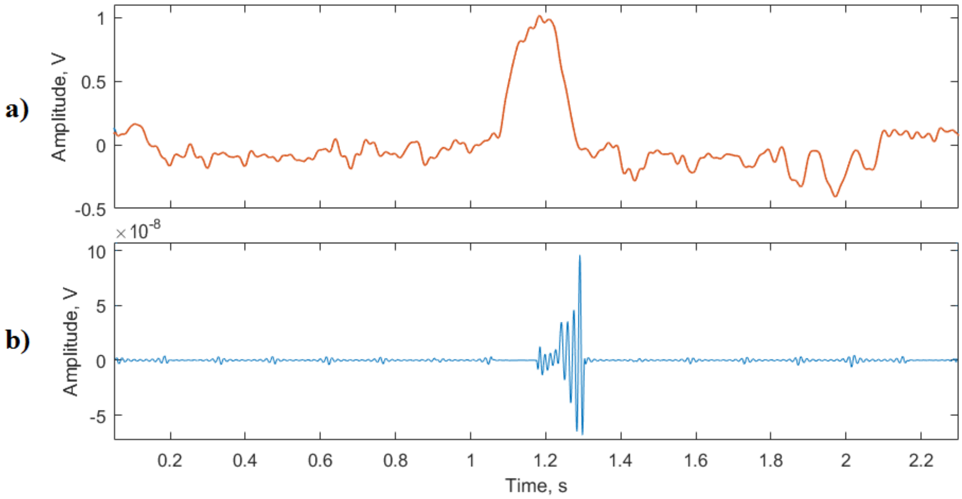


Figure 4.10: Original EEG signal (red line), reconstructed signal (green line) and error signal (blue line), which is the difference between original and reconstructed signal

difference between the original and reconstructed signal. Experimental simulations shows that the use of real-time reconstruction algorithm do not affect the average SNR value.

”AA-ASDM without additional envelope encoding” allows not only to reduce the switching activity of the ASDM circuit even more, if compared to ”AA-ASDM with additional envelope encoding”, it also has lower complexity of the encoding circuit, since it does not involve an additional encoding circuit for the envelope. Therefore, further in the next sections only this approach will be used for experimental research.

4.2 Modeling

As shown in simulations, in Section 4.1, by using AA-ASDM it is possible to reduce the switching activity of ASDM by up to 68.85%. In order to develop a working AA-ASDM hardware prototype and estimate its properties, first an electric circuit model with existing electronic components must be developed, simulated and analyzed for both ASDM and AA-ASDM. To reach this aim, *SIMetrix* circuit simulation software is used for transient analysis and power consumption estimation of ASDM, AA-ASDM and On-Off-Keying transmitter circuits.

4.2.1 Asynchronous Sigma-Delta Modulator

In order to understand the difference between ASDM and AA-ASDM in terms of hardware implementation complexity and power consumption, first an ASDM circuit, based on block diagram shown in Fig. 2.2, is developed, simulated and analyzed. Developed ASDM electric circuit is shown in Fig. 4.11. It consists of a voltage adder ($OpAmp1$), an integrator ($OpAmp2$) and a Schmitt trigger ($OpAmp3$), where resistors and capacitor determine the circuit parameters and switching activity. The circuit is powered by a ±5V DC voltage source. As shown in Section 1.1.2, the typical EEG signal frequency is up to 50Hz, therefore, in this case, a 50Hz sinusoidal signal with amplitude ±1V is used as a test signal, in order to cover the typical EEG signal bandwidth. Simulation results of the ASDM circuit are shown in Fig. 4.12.

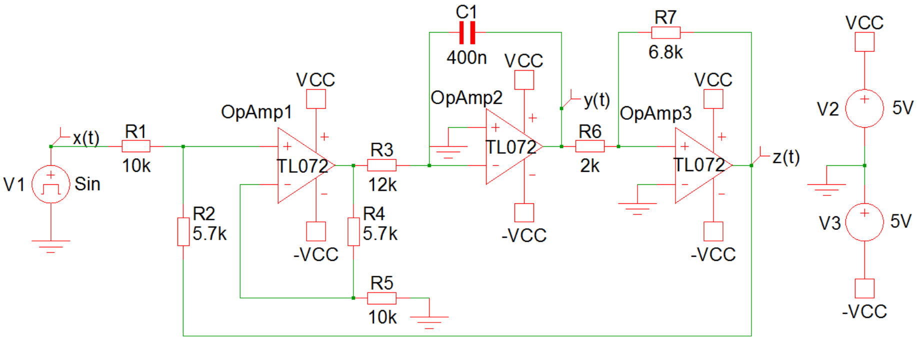


Figure 4.11: ASDM electric circuit

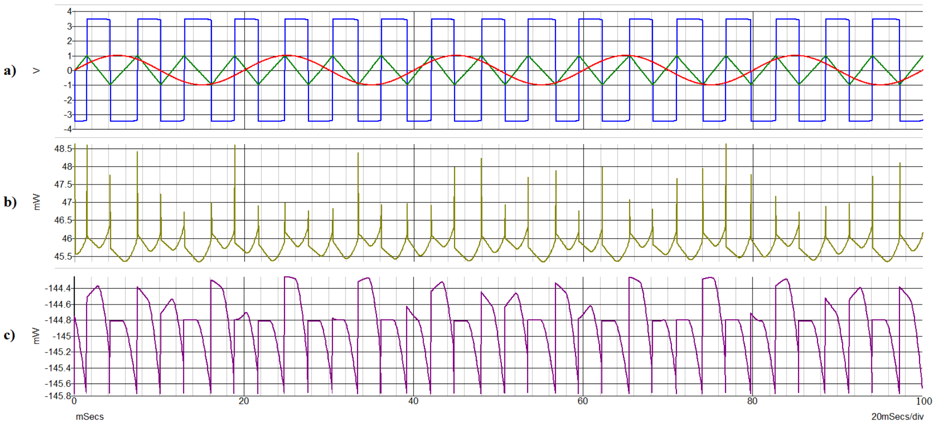
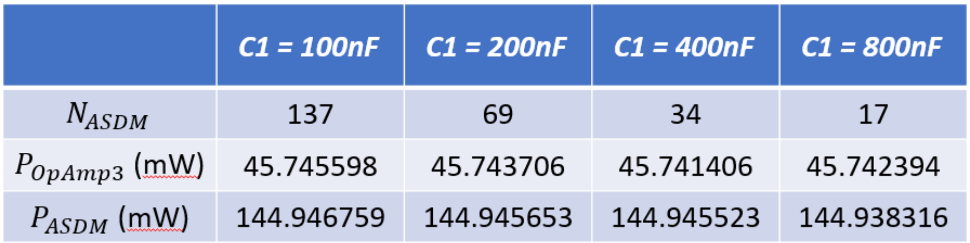


Figure 4.12: Operation of ASDM. a) Input signal (red line), integrator output (green line) and trigger output (blue line); Power consumption of b) the trigger and c) the whole circuit

As it can be seen in graph a), after the sinusoidal test signal (red line) is encoded, the output of the ASDM circuit (blue line) is a continuous rectangular pulse signal with an amplitude of ±3.5V. The power consumption of the ASDM output trigger ($OpAmp3$) in shown in graph b), but overall power consumption of the whole circuit in graph c).

ASDM circuit is simulated four times, each time with different capacitor $C1$ value of an integrator circuit. By changing $C1$ from 100nF to 800nF, the number N_{ASDM} of switching time instants of the ASDM circuit within the given time (100ms) increases too. The results are shown in Table 4.7.

Table 4.7: Comparison of the number of switching time instants and power consumption of the ASDM circuit for different capacitor $C1$ values



	C1 = 100nF	C1 = 200nF	C1 = 400nF	C1 = 800nF
N_{ASDM}	137	69	34	17
P_{OpAmp3} (mW)	45.745598	45.743706	45.741406	45.742394
P_{ASDM} (mW)	144.946759	144.945653	144.945523	144.938316

As shown in Table 4.7, by increasing the value of capacitor $C1$ from 100nF to 800nF, the number of switching time instants N_{ASDM} decreases from 137 to 17, but power consumption of the whole ASDM circuit remains nearly the same, since power consumption fluctuations of the trigger are minimal compared to its average power consumption. As shown in Fig. 4.12, ASDM output trigger consumes the most, when it changes the state from -3.5V to +3.5V.

It should be noted that simulations were made only to verify the working principles of the ASDM circuit and the power consumption value of ≈145mW is just indicative, as it is possible to develop ASDM circuit with 7.5nW power consumption as shown in [25], which is achieved by developing specialized circuit structures with decreased operating supply voltage, capacities, etc. and increased slew rates [177].

As mentioned before the main power consumer of a BCI system is transmitter, therefore it must be also modeled and simulated together with ASDM. This is a subject of the next subsection.

4.2.1.1 Wireless Data Transmission in ASDM case

As shown in Section 1.1.3 (see Fig. 1.6), modern BCI systems usually use wireless data transmitter in order to transmit the encoded signal to data processing/visualization device. In this case, to make ASDM based BCI system efficient, wireless data transmission must be event-driven, where the amount of transmitted information per second is proportional to the number of switching time instants t_k per second of the ASDM output.

One, very promising and most importantly energy efficient wireless data transmission approach is a transmission based on Ultra Wide Band (UWB) pulses. In this case, for each switching time instant t_k an UWB pulse is generated and transmitted to receiver. The fewer switching time instants, the lower power consumption of the transmitter. But, as mentioned in Section 1.1.3, not only the reception and reconstruction of the signal is very complex and usually inaccurate, but also generation of UWB pulses with different shapes is very complicated, thus not only limiting the number of channels which can be used for BCI system, but also increasing the complexity and the size of the circuit.

For this particular BCI application, due to its great properties, an On-Off Keying (OOK) technique can be used for wireless data transmission. It has several advantages - simple architecture, good performance in the presence of co-channel interference, robust when exposed to vibration and shock, and major criteria - small dimensions for on-head device implementation. [26] Simple OOK implementation for ASDM output transmission is shown in Fig. 4.13.

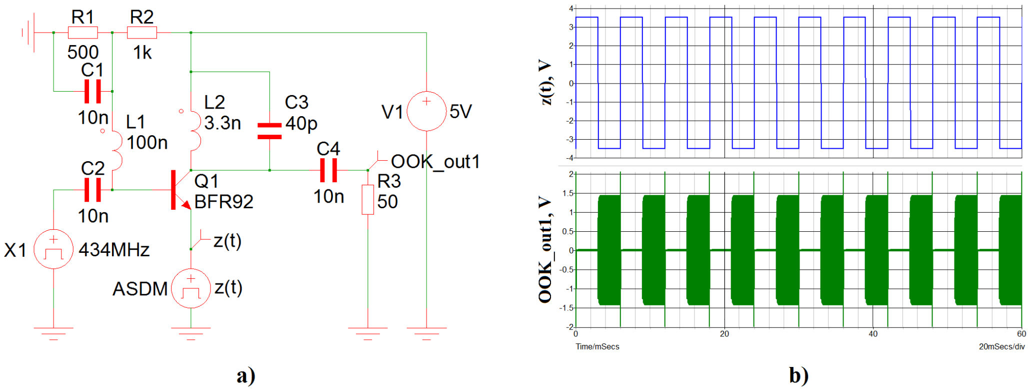


Figure 4.13: Operation of the On-Off-Keying (OOK) transmitter: a) electric circuit; b) ASDM output signal $z(t)$ and OOK output signal OOK_out1 driven by the output of ASDM

As can be seen in Fig. 4.13 b), the circuit is transmitting all the time ASDM output is negative, which is not energy efficient for the ASDM case, where it is needed to transmit a time information only, i.e., the switching time instants when ASDM output changes its state to opposite. This can be solved by a small modification of the OOK circuit, where a short pulses filled with carrier frequency are generated each time the ASDM output is changing its state to opposite. The modified OOK circuit is shown in Fig. 4.14.

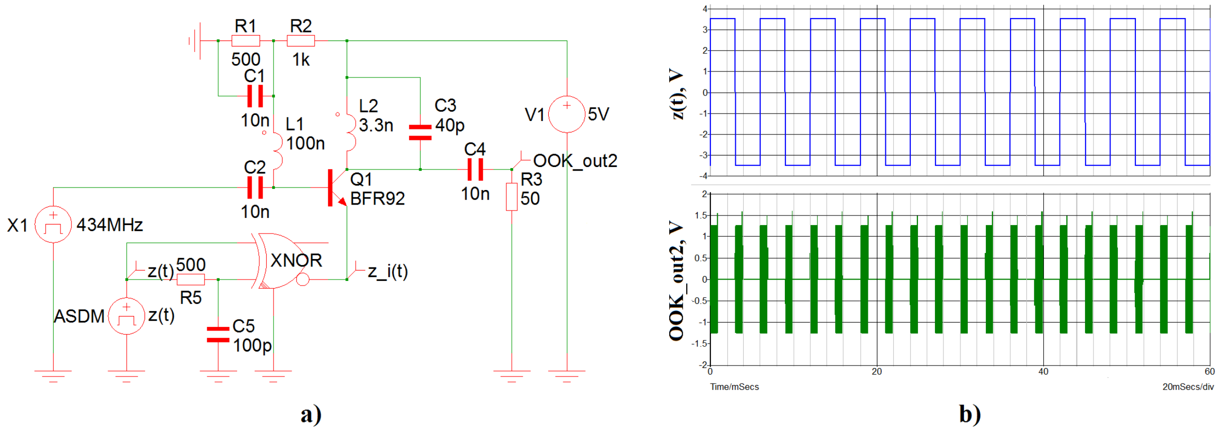


Figure 4.14: Operation of the modified On-Off-Keying (OOK) transmitter: a) electric circuit; b) ASDM output signal $z(t)$ and OOK output signal driven by the output of ASDM OOK_out2

In this case, the circuit is supplemented by a logic gate XNOR element, which consumes additional 45μW, and two passive elements, but the overall circuit's power consumption decreases up to 25 times depending on the number of switching instants N_{ASDM} (see Table 4.8).

Table 4.8: Comparison of power consumtion of OOK and modified OOK circuits for different ASDM circuit parameters

	C1 = 100nF	*C1 = 200nF*	*C1 = 400nF*	*C1 = 800nF*	*C1 = 1.6μF*	
N_{ASDM}	137	69	34	17	9	0
P_{ASDM_OOK} (mW)	562.40157	557.96274	546.57591	508.03407	463.37538	16.66667
$P_{ASDM\_OOK\_Mod}$ (mW)*	63.63890	40.52755	28.81639	22.67473	19.76925	16.68737

* - Pulse width: 110ns

As can be seen in Table 4.8, the average power consumption of both OOK and modified OOK circuits decrease if the number of ASDM switching time instants (N_{ASDM}) decreases. Assuming that ASDM output is a 50% duty cycle, it is expected that the average power consumption of the OOK circuit is not changing if the number of ASDM switching time instants decreases, however, it changes. This is due to fact that after each transmission this circuit (see Fig. 4.13) dissipates energy that was stored in LC oscillator (see Fig.4.15). Due to this, the higher the number of ASDM switching time instants (N_{ASDM}) within the given time, the higher the power consumption of the circuit. The relation is proportional.

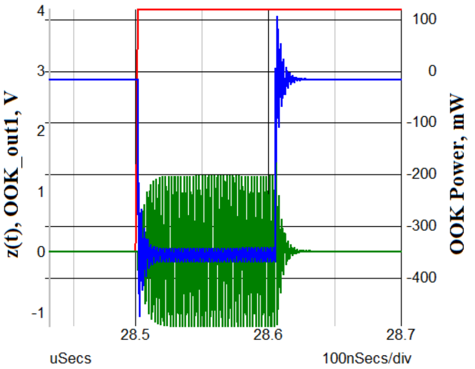


Figure 4.15: Pulse generation in OOK circuit: ASDM output signal $z(t)$ (red line), OOK output signal OOK_out1 (green line), power consumtion of the OOK circuit (blue line)

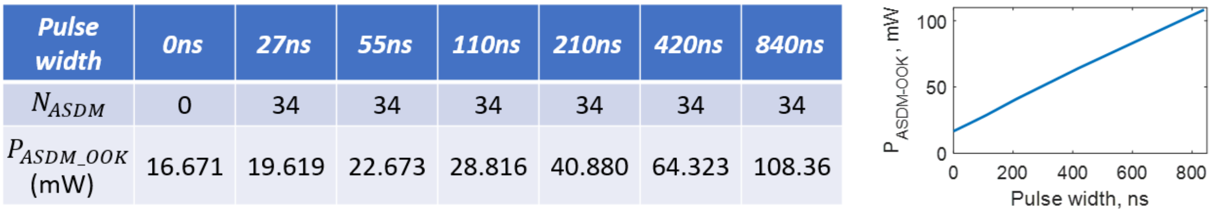
In the modified OOK circuit case (see Fig. 4.14), the power consumption of the circuit is directly proportional to the number of ASDM switching time instants (N_{ASDM}):

$$P_{OOKmod} = P_0(1 - n \cdot \frac{\tau_{tx}}{T_{tx}}) + n \cdot \frac{P_{tx} \cdot \tau_{tx}}{T_{tx}}, \tag{4.7}$$

where P_0 - is an average power consumption of the circuit, when LC is not oscillating, but P_{tx}, when LC is oscillating during and after each transmission; τ_{tx} is a length of the transmitting pulse (including decay time of oscillations), T_{tx} is a period in which the circuit is analyzed, but n is an average number of pulses during T_{tx}. From (4.7) it follows, if the the number of ASDM switching time instants is decreased, the average power consumption of the OOK circuit is also decreased in a linear relation. It should be noted that in this case the power consumption of the circuit when it is not transmitting is P_0 ≈16.6mW, which is very high, while modern wireless transmitters have P_0 in nW [178], [179], meaning that if the number of switching time instants are reduced by 50%, the power consumption of the transmitter will also decrease by ≈50%.

Besides that, the experimental test results show that the power consumption of the modified OOK circuit is also directly proportional to selected pulse width (see Table 4.9).

Table 4.9: Power consumption of the OOK circuit depending on selected pulse width



Pulse width	0ns	27ns	55ns	110ns	210ns	420ns	840ns
N_{ASDM}	0	34	34	34	34	34	34
P_{ASDM_OOK} (mW)	16.671	19.619	22.673	28.816	40.880	64.323	108.36

P_ASDM-OOK, mW
100
50
0
0
200
400
600
800
Pulse width, ns

In practical implementations, the length of the pulse must be chosen the shortest possible which can ensure reliable transmission.

4.2.2 AA-ASDM without additional envelope encoding

The electrical circuit of the AA-ASDM without additional envelope encoding is similar to the ASDM circuit, shown in Fig. 4.11, only in this case it is supplemented by an Analog Multiplier ($MLT04$) and a Voltage Follower ($OpAmp1$) (see Fig. 4.16). In order to make a comparison between ASDM and AA-ASDM circuits, the parameters in both circuits are set to be equal (the selected parameters can be found in Section 4.2.1).

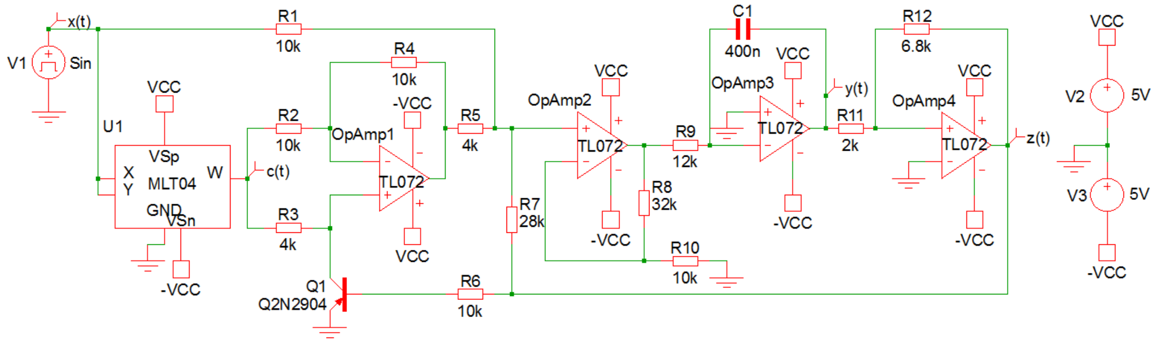


Figure 4.16: AA-ASDM electrical scheme

Simulation results of the AA-ASDM circuit are shown in Fig. 4.17.

Just as for ASDM circuit, the AA-ASDM circuit is simulated four times, each time with different capacitor $C1$ value (100nF, 200nF, 400nF and 800nF) of an integrator circuit. By changing $C1$, the number of switching time instants ($N_{AA-ASDM}$) of the AA-ASDM circuit within the given time (100ms) is changing too. The results are shown in Table 4.10.

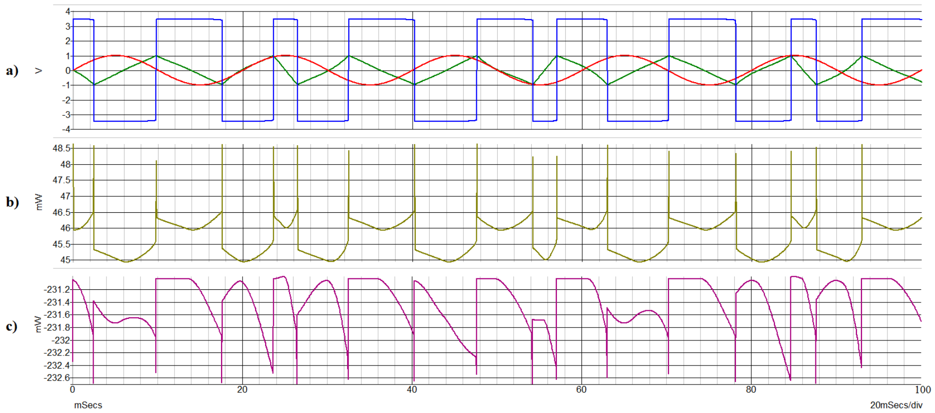
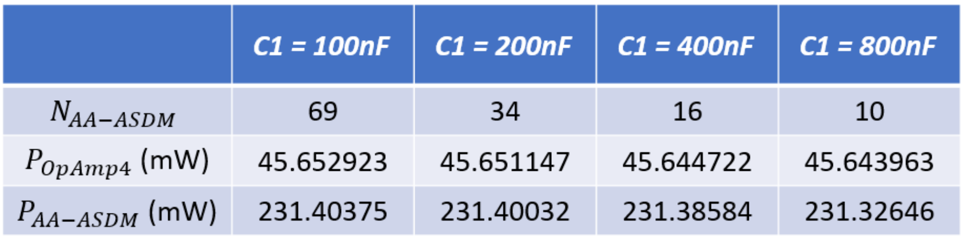


Figure 4.17: Operation of AA-ASDM. a) Input signal (red line), integrator output (green line) and trigger output (blue line); Power consumption of b) the trigger and c) the whole circuit

Table 4.10: Comparison of the number of switching time instants and power consumption of the AA-ASDM circuit for different capacitor $C1$ values



	C1 = 100nF	*C1 = 200nF*	*C1 = 400nF*	*C1 = 800nF*
$N_{AA-ASDM}$	69	34	16	10
P_{OpAmp4} (mW)	45.652923	45.651147	45.644722	45.643963
$P_{AA-ASDM}$ (mW)	231.40375	231.40032	231.38584	231.32646

As can be seen in Table 4.7 and Table 4.10, for equal capacitor $C1$ values, the number of switching time instants are different. For example, for $C1$=100nF, the number of switching time instants for ASDM is N_{ASDM}=137, but for AA-ASDM $N_{AA-ASDM}$=69. The number of switching time instants for AA-ASDM circuit can be even smaller, if signals with wide dynamic range is used (see Section4.1.3.1). In this modeling, a sinusoidal 50Hz test signal is used.

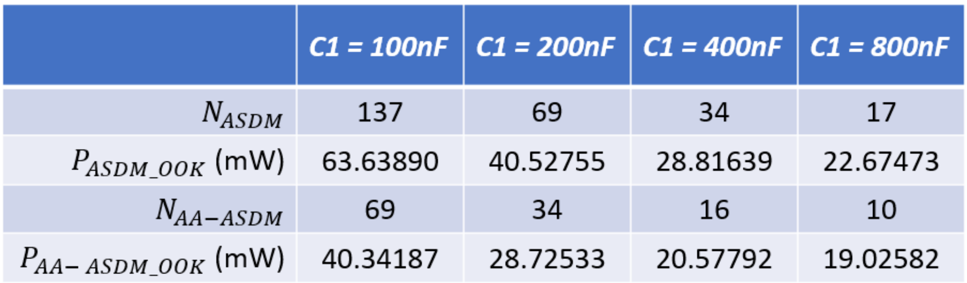
On the other hand, from Table 4.7 and Table 4.10 it can be seen that the power consumption of the AA-ASDM circuit is ≈60% higher (≈231mW), compared to ASDM (≈145mW). It should be noted that simulations were made only to verify the working principles of both ASDM and AA-ASDM circuits and the power consumption values are just indicative, as it is possible to develop an ASDM circuit with 7.5nW power consumption as shown in [25]. This means, if the AA-ASDM circuit is implemented in the same technology as shown in [25], the increase of power consumption by 60% compared to ASDM would give an overall power consumption of the AA-ASDM circuit: 12nW. As mentioned in Section 1.1.3 the main power consumer of a

BCI system is a transmitter therefore it is more important to reduce the amount of information to be transmitted. As shown in Section 4.2.1.1, the power consumption of the OOK transmitter can be reduced significantly by reducing the number of switching time instants to be transmitted, which can be achieved by using AA-ASDM instead of ASDM.

4.2.2.1 Wireless Data Transmission in AA-ASDM case

The transmitter, used for AA-ASDM is the same as shown in Fig. 4.14, only in this case, the OOK transmitter is driven by the AA-ASDM output, instead of ASDM. This means, fewer switching time instants and reduced power consumption of the OOK circuit. A comparison of the power consumption of the OOK circuit depending on if ASDM or AA-ASDM circuit's output is used as an input for the OOK transmitter is shown in Table 4.11.

Table 4.11: Comparison of the power consumption of the OOK circuit depending on if ASDM or AA-ASDM circuit's output is used as an input for the OOK transmitter



	C1 = 100nF	*C1 = 200nF*	*C1 = 400nF*	*C1 = 800nF*
N_{ASDM}	137	69	34	17
P_{ASDM_OOK} (mW)	63.63890	40.52755	28.81639	22.67473
$N_{AA-ASDM}$	69	34	16	10
$P_{AA-\ ASDM\_OOK}$ (mW)	40.34187	28.72533	20.57792	19.02582

As can be seen in Table 4.11, the reduction of the power consumption of the transmitter by using AA-ASDM, instead of ASDM, can reach up to ≈37% depending on the number of switching time instants. But, it is important to note that in this case the power consumption of the circuit when it is not transmitting is very high ≈16.6mW, while modern wireless transmitters have it in a range of nW [178], [179]. This means, if the the power consumption of the circuit, when it is not transmitting, would be for example 100nW, the reduction of the power consumption could reach up to ≈50% depending on the number of switching time instants or even more if signals with HDR is used (see Section 4.1.3), where AA-ASDM is particularly advantageous.

As mentioned in Section 1.1.3, modern wireless data transmitters consume ≈3mW. It follows that if AA-ASDM instead of ASDM is used, it is possible to reduce the power consumption of the transmitter by 1.5mW, while increasing the power consumption of the encoder just by 4.5nW, giving a total of ≈50% power consumption reduction in a wireless BCI system, leading to two times longer battery life and operation time of wireless system. These parameters can be increased even more if the system is used for wide dynamic range signals, such as EEG signals.

4.3 Practical Implementations

As shown in simulations in Section 4.1, by using AA-ASDM it is possible to reduce the switching activity of ASDM by 68.86%, while modeling shows that such an improvement comes with an increase in the power consumption of the AA-ASDM circuit, which is admissible, because the increase is small (4.5nW) compared to the reduction of power consumption in wireless transmitter (≈1.5mW). In this case, the reduction of switching time instants proportionally reduce the power consumption of the wireless transmitter (see Section 4.2.1.1). Now, the main challenge is to develop and test AA-ASDM based EEG data acquisition system hardware prototype. To reach this aim, *Altium Designer* PCB design software is used for designing all the necessary hardware components - EEG signal amplifier, AA-ASDM, wireless data transmitter and receiver. All these components are constructed with an aim of creating complete one channel BCI system. Developed hardware is tested and assessed on test signals.

4.3.1 Amplitude Adaptive Asynchronous Sigma-Delta modulator

Based on the AA-ASDM modeling results (see Section 4.2.2), a physical AA-ASDM based EEG data acquisition system is designed (see the block diagram in Figure 4.18). [26] It consists

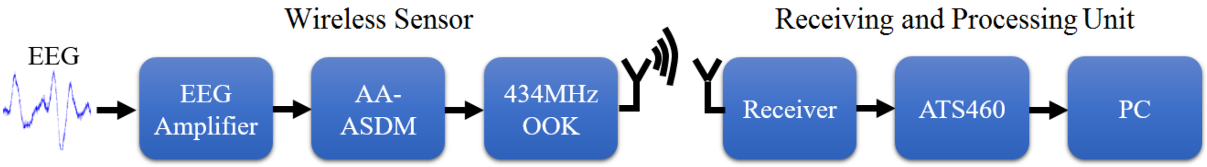


Figure 4.18: Full one channel block diagram

of a wireless sensor, which includes EEG signal amplifier, AA-ASDM and OOK transmitter, and receiving and processing unit, which includes super-heterodyne receiver, ATS460 digitizer and personal computer (PC). In the next subsections the description of each part follows.

4.3.1.1 AA-ASDM based Wireless Sensor for EEG data acquisition

Developed AA-ASDM based Wireless Sensor for EEG data acquisition consist of 3 parts (PCBs): EEG signal amplifier, AA-ASDM and OOK transmitter.

First, by using silver chloride (Ag/Cl) bridge electrodes and special conductive gel the weak EEG signal is obtained. As shown in Section 1.1.2, the typical measured EEG signal amplitude is between 0.5 - 100μV, therefore it must be amplified prior the encoding. The amplification is performed by the first PCB: EEG signal amplifier (see Fig. 4.19). First EEG signal enters a high CMRR (∼120dB) differential preamplifier (*INA128*) circuit with variable gain (up to g≈250), which includes protection circuit together with Driven Right Leg (DRL) circuit. Then, the signal is passed to 2nd stage non-inverting amplifier (g≈40), which in total gives an amplification of ∼10 000 times, amplifying EEG signal from ∼0.5-100μV to 0.005-1V. **[26]**

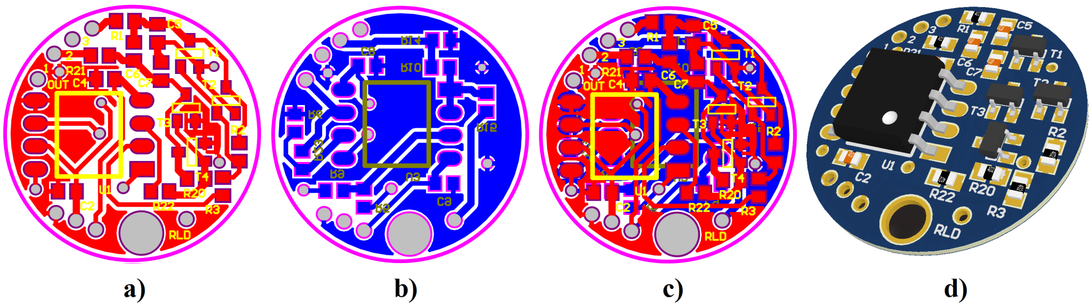


Figure 4.19: PCB of the designed EEG signal amplifier: a) top routing layer; b) bottom routing layer; c) top and bottom routing layers; d) 3D model

After signal amplification, it is time encoded by the second PCB: AA-ASDM (see Fig. 4.20). Based on the Fig. 4.16, AA-ASDM is implemented by using 4 operational amplifiers, analog

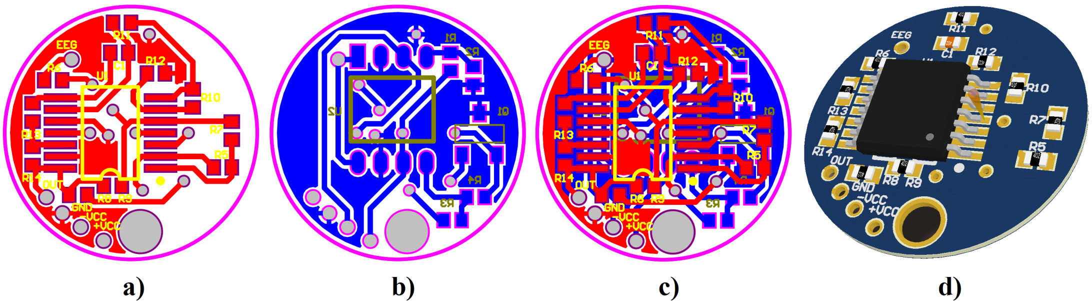


Figure 4.20: PCB of the designed AA-ASDM: a) top routing layer; b) bottom routing layer; c) top and bottom routing layers; d) 3D model

multiplier and few passive elements. The selection of these elements and set parameters is described in next Section 4.3.2. An example of AA-ASDM input signal $x(t)$ and corresponding output signal $z(t)$ from the developed hardware is shown in Fig. 4.21.

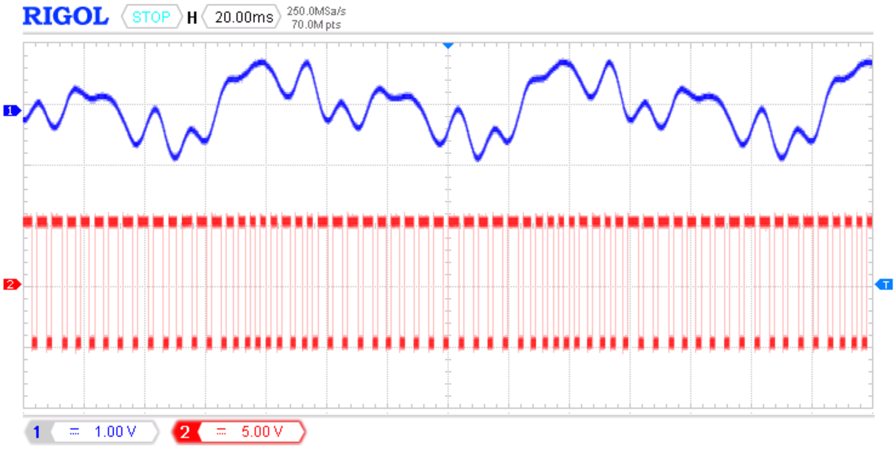


Figure 4.21: AA-ASDM input signal (blue line) and corresponding output signal (red line)

After time encoding, the AA-ASDM output signal $z(t)$ is wirelessly transmitted by using the third PCB: OOK transmitter (see Fig. 4.22). As shown in Section 4.2.1.1, it consist of a

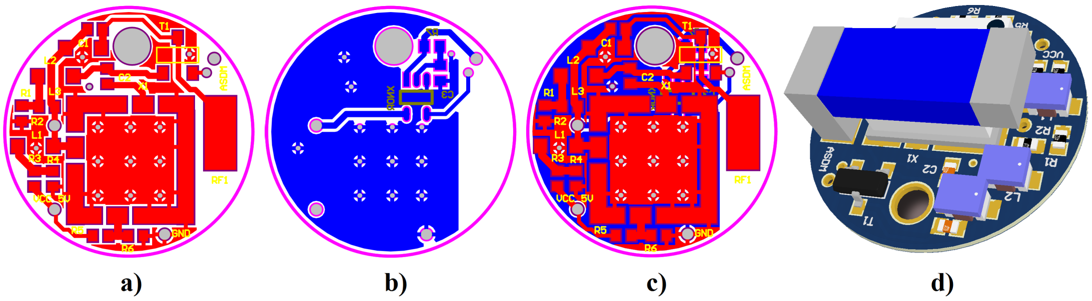


Figure 4.22: PCB of the designed OOK transmitter: a) top routing layer; b) bottom routing layer; c) top and bottom routing layers; d) 3D model

RC filter, XNOR gate, a simple transistor circuit for OOK manipulation and bandpass filter. An example of AA-ASDM output signal $z(t)$ and generated pulses at the output of the XNOR gate is shown in Fig. 4.23, while an example of one pulse and corresponding OOK output signal $s(t)$ is shown in Fig. 4.24.

In this case, by varying carrier frequency ω_n, for different AA-ASDM outputs $z_n(t)$, it is possible to create a multi-channel BCI system. Different carrier frequencies are obtained by voltage controlled oscillator and the modulated signal is transmitted by chip antenna.

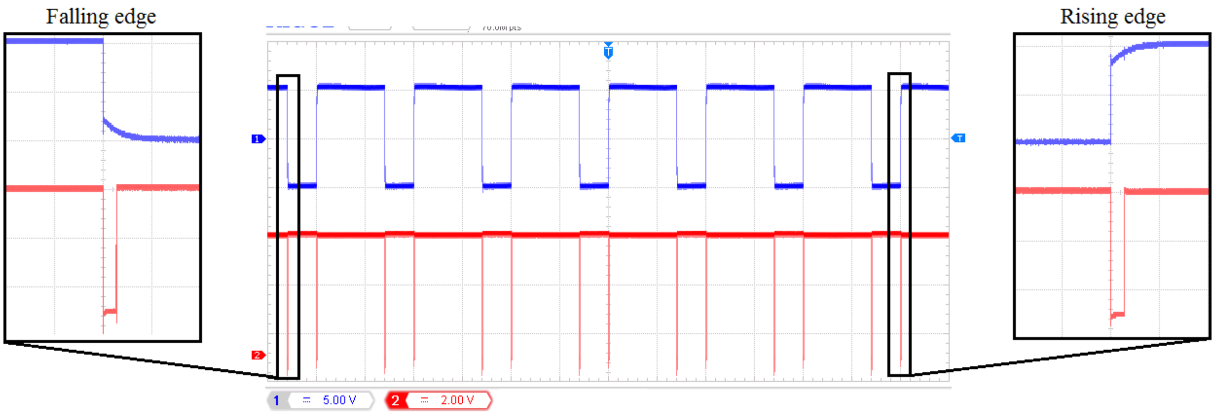


Figure 4.23: AA-ASDM output signal (blue line) and XNOR gate's output signal with pulses generated at each rising and falling edge of the AA-ASDM output signal (red line)

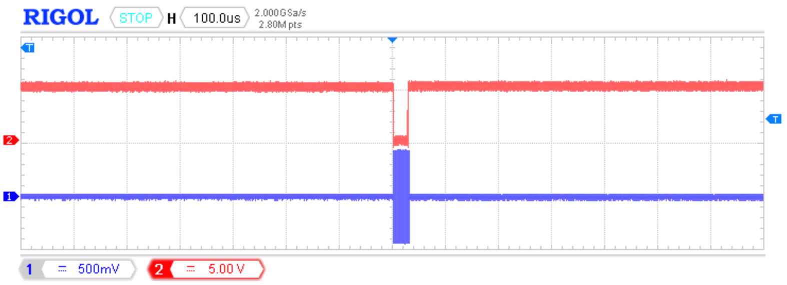


Figure 4.24: One pulse of the XNOR gate output signal (red line) and corresponding OOK output signal (blue line)

The transmitted signal $s_1(t)$ of the first channel can be written as $s_1(t) = z_1(t) \cdot cos(\omega_0 t)$, where ω_0 is the carrier frequency, which is different for each channel (ω_{01}, ω_{02}, ω_{0n} (an example of three channel case is depicted in Fig. 4.25)). [26]

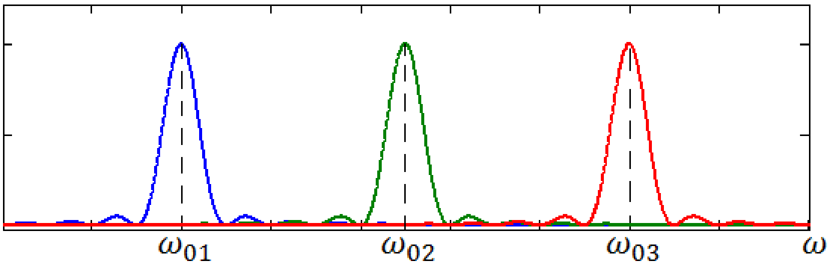


Figure 4.25: 3 channel spectrum example

By knowing the pulse width at the output of the XNOR gate, it's possible to calculate the occupied spectrum bandwidth. On one hand, the shorter the pulse, the wider the occupied bandwidth. On the other hand, the wider the pulse, the higher the power consumption of a transmitter.

The Wireless Sensor, which consists of EEG amplifier, AA-ASDM and OOK transmitter is fabricated in cylindrical 16x11mm package (see Fig. 4.26). [26] The sensor design allows to put it right on the electrode surface, thus can be used as an on-head sensor device. After EEG signal being amplified, encoded and transmitted, it also must be received and reconstructed (see Fig. 4.18). This is a subject of the next Section 4.3.1.2.

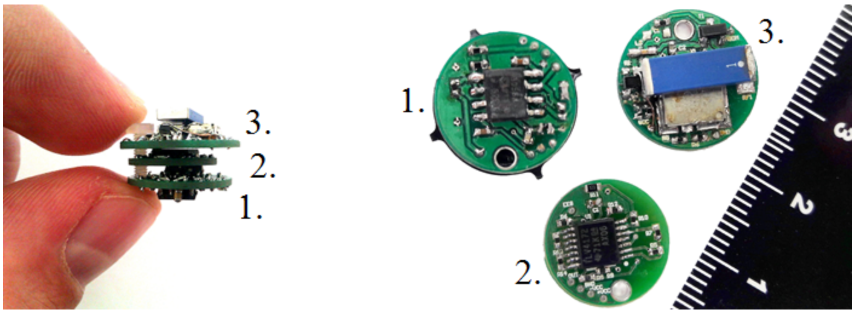


Figure 4.26: Designed wireless sensor: 1. EEG amplifier, 2. AA-ASDM, 3. OOK transmitter

4.3.1.2 Receiving and Processing Unit

The received signal is a sum of all transmitted signals: $s_r(t) = \sum_{n=1}^{N} s_n(t) \cdot cos(\omega_n t)$.

First, a super-heterodyne receiver (one receiver for all sensors) is developed (see Fig. 4.27) and used for high frequency signal $s_r(t)$ down-conversion to a fixed intermediate frequency (IF), which can be more easily processed than the original 434MHz carrier frequency. So, the output

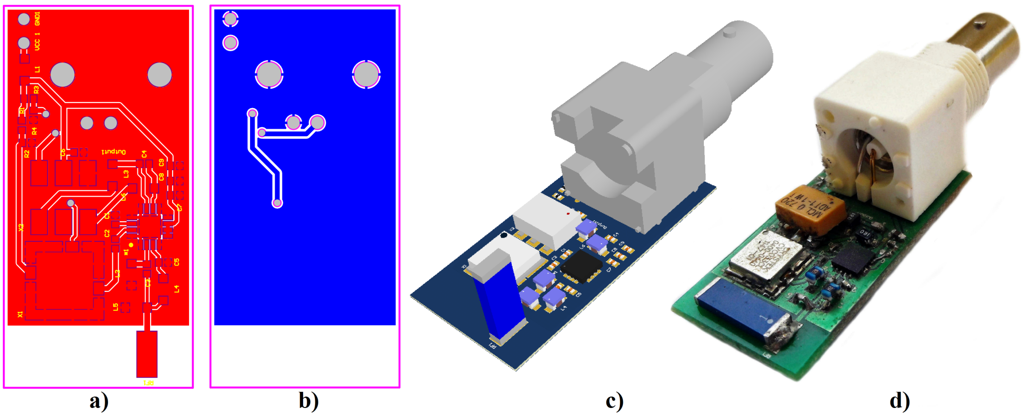


Figure 4.27: PCB of the designed OOK receiver: a) top routing layer; b) bottom routing layer; c) 3D model d) physical device

s_{IF} can be written as following: $s_{IF}(t) = s_r(t) \cdot cos(\omega_0 t)$, where ω_0 is a variable frequency. **[26]**

The super-heterodyne receiver consists of chip antenna, high linearity and low power down-converting mixer *LT5526*, voltage controlled oscillator (VCO), radio frequency transformer *ADT1-1WT* and 50Ω output matching circuit. **[26]**

After frequency down converting from ≈434 MHz to ≈1 MHz, IF signal is imported into PC using *ATS460* 14-bit, 125 MS/s digitizer. Further, all the signal processing is performed in *Matlab*. Each channel of the imported signal $s_{IF}(t)$ is bandpass filtered and processed in order to obtain AA-ASDM output switching time instants t_k.

In order to obtain AA-ASDM output switching time instants t_k from the band-passed signal, following steps are made: first, the band-passed signal is multiplied by the center frequency to obtain the envelope; then, peak values are found of the modulus of the acquired signal; further, all the values which are lower than the predefined threshold are set to be zero, thus eliminating the noise; finally, the switching time instants ar found in those locations, where the difference between two consecutive sample values is higher than the predefined threshold, or in other words, the transition between non-transmitting and transmitting state is found, and marked with the bars (see Fig. 4.28). These bars are the AA-ASDM switching time instants t_k from which the original signal, based on the algorithm shown in Section 4.1.3.2, can be reconstructed. In order to estimate the performance of the developed system, and most importantly of the AA-ASDM, the experimental setup is build and system tested (see next Section 4.3.2).

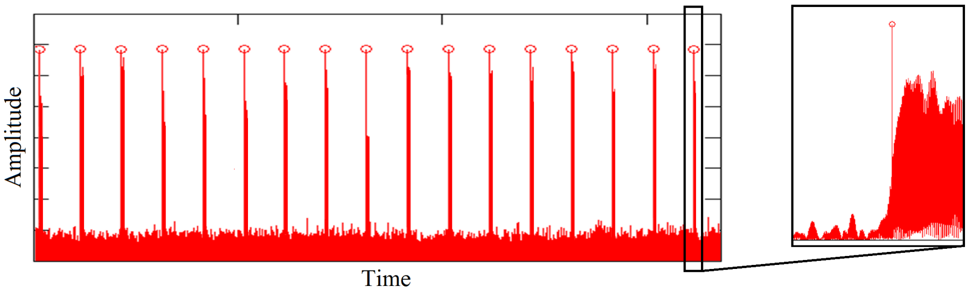


Figure 4.28: Received and band-passed OOK signal and found switching time instants

4.3.2 Experimental Setup and Tests

In order to test and assess the developed AA-ASDM based EEG data acquisition system, described in Section 4.3.1, first AA-ASDM must be calibrated. The block diagram of the calibration setup is shown in Fig. 4.29, while *Matlab* program for AA-ASDM calibration in Appendix E-1. The calibration setup consists of a signal generator, which generates test signals, physical AA-ASDM PCB, 2-channel ATS460 digitizer (20Msps) and PC.

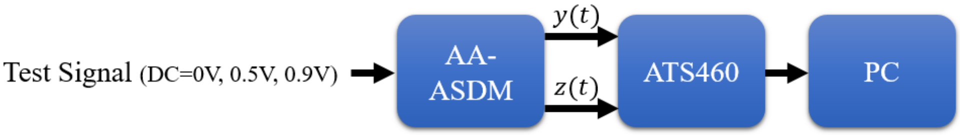


Figure 4.29: Block digram of the calibration setup

First, based on the Fig. 4.29, the outputs of the AA-ASDM integrator $y(t)$ and trigger $z(t)$ for three different DC input signal voltage levels (in this case, DC=0V, 0.5V and 0.9V) are digitized and imported into PC. Knowing the number and step of the quantization levels of the ATS460 digitizer, from acquired three $y(t)$ signals (see Fig. 4.30), the actual value of the AA-ASDM trigger parameters δ^+ and δ^- are acquired. In this case, $+\delta$=1.5V, but $-\delta$=-1.5V.

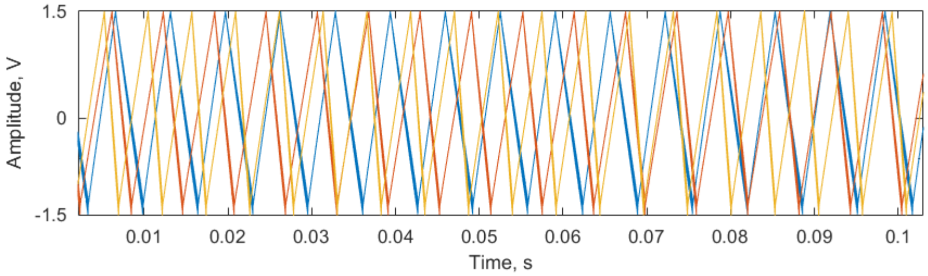


Figure 4.30: Outputs of the integrator for three different AA-ASDM input DC signal values (0V (blue line), 0.5V (red line) and 0.9V (yellow line))

From three imported AA-ASDM trigger $z(t)$ signals, it is possible to find the average $+b$ and $-b$ parameter values as well as the trigger switching time instants t_k for each of the $z(t)$ signals (see an example in Fig 4.31). In this case, $+b$=4.94V, but $-b$=-4.85V.

Further, from t_k values, the average maximum Δt_{max} and minimum Δt_{min} distance between two consecutive trigger switching time instants is calculated for all three signals. It should be

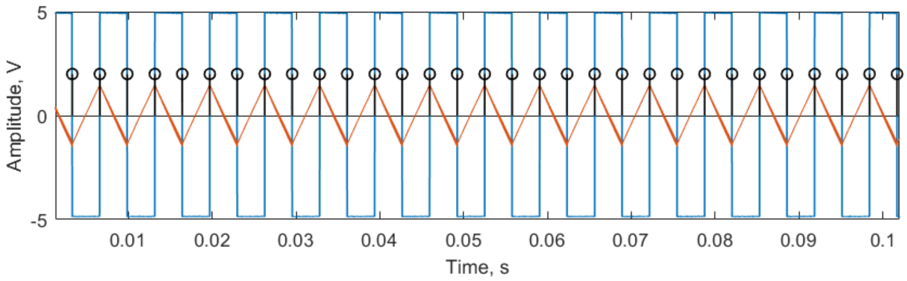


Figure 4.31: The output of the AA-ASDM PCB trigger (blue line), integrator (red line) and found switching time instants (black bars), when input signal is DC=0V

noted that for DC input there exist only one maximum and one minimum value (see an example of DC=0V in Fig. 4.32).

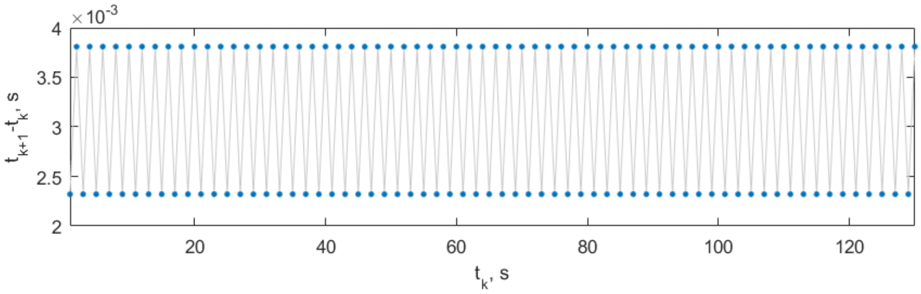


Figure 4.32: Distances between consecutive AA-ASDM trigger switching time instants when input signal is DC=0V

In order to verify if developed AA-ASDM PCB is working properly, acquired Δt_{max} and Δt_{min} values are compared to the theoretically calculated values. Theoretical values are calculated from the electrical scheme, shown in Fig. 4.33.

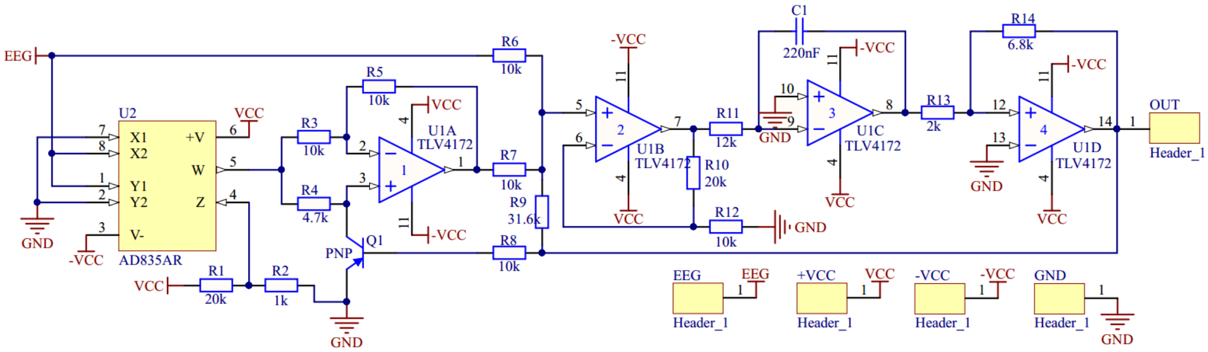


Figure 4.33: AA-ASDM electrical scheme of the actually developed PCB

First, the output of the Adder (U1B in Fig. 4.33) is found by following equation:

$$U1B_{out} = \left(1 + \frac{R10}{R12}\right)\left[u_1 \frac{R7||R6}{R9 + R7||R6} + u_2 \frac{R9||R6}{R7 + R9||R6} + u_3 \frac{R9||R7}{R6 + R9||R7}\right] = \\ = \left(1 + \frac{20}{10}\right)\left[u_1 \frac{5}{36.6} + u_2 \frac{7.6}{17.6} + u_3 \frac{7.6}{17.6}\right] = u_1 \cdot 0.4098 + u_2 \cdot 1.2951 + u_3 \cdot 1.2951, \tag{4.8}$$

where u_1 is the b parameter signal, u_2 the envelope signal $\tilde{o}(t)$ and u_3 is the input signal $x(t)$. From 4.8, we can denote the coefficients as a_1=0.4098 and a_2=1.2951.

Knowing values of a_1, a_2, $+b$, $-b$, δ^+ and δ^-, it is possible to calculate theoretical $\Delta t_{max/t}$ and $\Delta t_{min/t}$:

$$a_1 \int_{t_k}^{t_{k+1}} x(t)dt + a_1 \int_{t_k}^{t_{k+1}} (x^2(t) + 0.25)dt + a_2 \int_{t_k}^{t_{k+1}} b_2 dt = \frac{1}{\kappa}(\delta^+ - \delta^-), \tag{4.9}$$

$$a_1 \int_{t_{k+1}}^{t_{k+2}} x(t)dt - a_1 \int_{t_{k+1}}^{t_{k+2}} (x^2(t) + 0.25)dt - a_2 \int_{t_{k+1}}^{t_{k+2}} b_2 dt = -\frac{1}{\kappa}(\delta^+ - \delta^-). \tag{4.10}$$

where, $\frac{1}{\kappa} = \frac{1}{R11 \cdot C1}$ is an integrator constant.

If $x(t)$=0V, equations (4.9) and (4.10) can be written as:

$$a_1 \int_{t_k}^{t_{k+1}} (x^2(t) + 0.25)dt + a_2 \int_{t_k}^{t_{k+1}} b_2 dt = \frac{1}{\kappa}(\delta^+ - \delta^-), \tag{4.11}$$

$$a_1 \int_{t_{k+1}}^{t_{k+2}} (x^2(t) + 0.25)dt + a_2 \int_{t_{k+1}}^{t_{k+2}} b_2 dt = \frac{1}{\kappa}(\delta^+ - \delta^-). \tag{4.12}$$

From (4.11) and (4.12) it follows:

$$a_1(0.5 + x(t))^2 \Delta t_k + a_2 b_2 \Delta t_k = \frac{1}{\kappa}(\delta^+ - \delta^-), \tag{4.13}$$

$$a_1(x(t) - 0.5)^2 \Delta t_{k+1} + a_2 b_2 \Delta t_{k+1} = \frac{1}{\kappa}(\delta^+ - \delta^-). \tag{4.14}$$

Finally, from (4.13) and (4.14) it is possible to calculate the minimum and maximum distance between two consecutive trigger switching time instants:

$$\Delta t_k = \frac{\frac{1}{\kappa}(\delta^+ - \delta^-)}{a_1(x(t) + 0.5)^2 + a_2 b_2}, \tag{4.15}$$

$$\Delta t_{k+1} = \frac{\frac{1}{\kappa}(\delta^+ - \delta^-)}{a_1(x(t) - 0.5)^2 + a_2 b_1}. \tag{4.16}$$

In order to verify if AA-ASDM is working properly, the division of practically acquired maximum and minimum distances between two consecutive trigger switching time instants $\frac{\Delta t_{max}}{\Delta t_{min}}$ is compared to theoretically calculated $\frac{\Delta t_{k+1}}{\Delta t_k}$ a these values must be equal or almost equal:

$$\frac{\tau_{max}}{\tau_{min}} = 1.0046, \text{when } x(t) = 0V; 1.6416 \ (0.5V); 2.0654 \ (0.9V), \tag{4.17}$$

$$\frac{\Delta t_{k+1}}{\Delta t_k} = \frac{a_2 b_2 + a1(x(t)+0.5)^2}{a_2 b_1 + a1(x(t)-0.5)^2} = 1.0155, \text{when } \ x(t) = 0V; 1.6692 \ \ (0.5V).2.0778 \ \ (0.9V). \tag{4.18}$$

As can be seen above, the practical values are almost equal to theoretically calculated, which means the AA-ASDM is working properly. Also, the theoretically calculated integrator constant $\frac{1}{\kappa} = \frac{1}{R11 \cdot C1} = \frac{1}{12^3 \cdot 220^{-9}} = 378.7879$ should be almost equal to practically acquired:

$$\frac{1}{\kappa_k} = \frac{(\delta^+ - \delta^-)}{\Delta t_k (a_2 b_2 + a_1 (x(t)+0.5)^2)} = 390.3648, \tag{4.19}$$

when $x(t)$=0V; 389.5847 (0.5V); 385.8625 (0.9V),

$$\frac{1}{\kappa_{k+1}} = \frac{(\delta^+ - \delta^-)}{\Delta t_{k+1} (a_2 b_1 + a_1 (x(t)-0.5)^2)} = 394.5847, \tag{4.20}$$

when $x(t)$=0V; 395.9194 (0.5V); 388.1883 (0.9V). As can be seen, the difference from the theoretically calculated value is within the range of 4%. This is due to capacitor non-ideality, as it has ±10% capacitance tolerance. As experiments shows, the precision of the reconstruction is very sensitive to capacitor non-ideality and therefore the circuit must be calibrated first. In further experiments, the value of $\frac{1}{\kappa}$ is selected based on the practical measurements.

After calibration of the AA-ASDM and initial tests, the AA-ASDM circuit can be placed in the overall AA-ASDM based EEG data acquisition system for further experimental tests. A block diagram of the experimental setup is shown in Fig. 4.34.

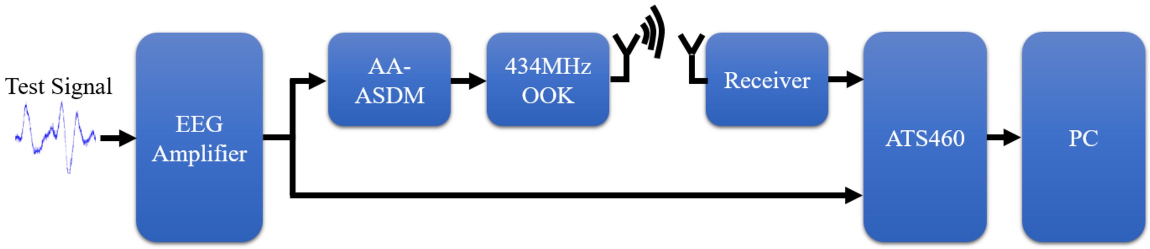


Figure 4.34: Block diagram of the experimental setup

Knowing the parameters of the AA-ASDM circuit and by using *Matlab* program shown in Appendix E-2, it is possible to acquire, process and reconstruct the original signal.

An examples of original and reconstructed signals is given in Fig. 4.35, while example of the distances between consecutive trigger switching time instants is shown in Fig. 4.36.

As can be seen in Fig. 4.35, it is hard to see the difference between the original and reconstructed signal, which means that it is possible to encode and qualitatively reconstruct the signal,

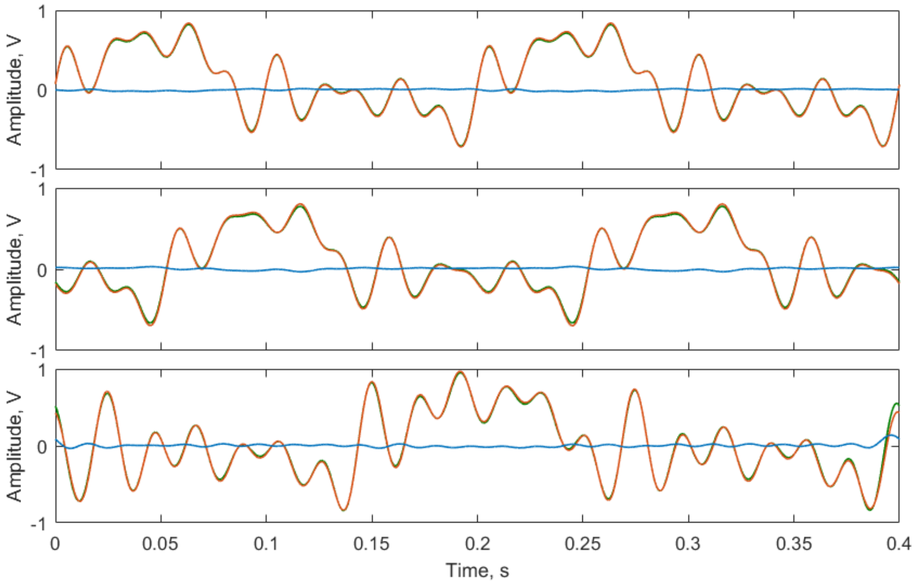


Figure 4.35: Original signal (red line), reconstructed signal (green line) and error signal (blue line), which is the difference between original and reconstructed signal

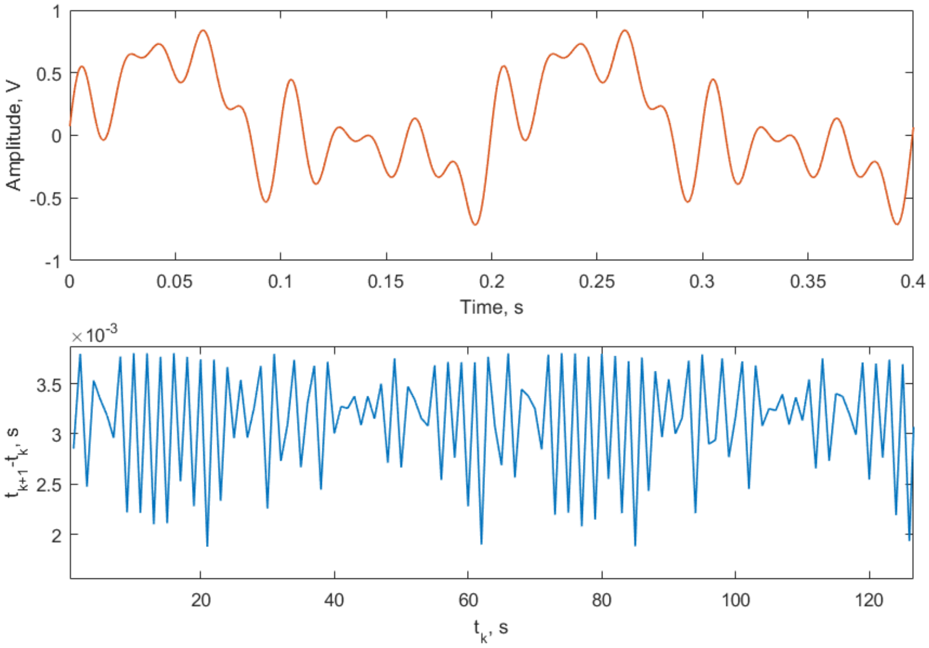


Figure 4.36: Original signal (red line) and obtained distances between consecutive trigger switching time instants (blue line)

by using AA-ASDM2. In order to quantitatively estimate the performance of AA-ASDM it must be implemented in a chip, with specialized circuit structures with decreased operating supply voltage, capacities, etc., with low comparator jitter, slew rate, DC gain, voltage saturation, excess loop delays and comparator offsets, etc. [177], [146], [180], [181] Also, as shown is [182], Time-to-digital converter (in this case *ATS460*) has a significant impact on the precision of the reconstructed signal. The precision of the reconstructed signal is directly proportional to the t_k measurement precision. [183]

4.4 Summary and Conclusions

In order to verify and assess the theory, developed in Section 3, in practice, in this Section various simulations, modeling and physical implementations of ASDM (as a reference design) and AA-ASDM were carried out.

Simulations

In Section 4.1, experimental simulation results confirmed the theory (in Section 2.3.2) that by using ASDM for EEG signal encoding, an unnecessary high switching activity of the ASDM circuit appears when input signal amplitude is low. For a 50Hz EEG signal, the average number of trigger switching time instants per second N_{ASDM}, depending on selected α value, vary between 152 and 1184. The maximum distances between two consecutive trigger switching time instants equaling nearly Nyquist step are obtained only when the input signal reaches its highest amplitude values. Since the maximum distance between two consecutive ASDM trigger switching time instants does not exceed Nyquist step, it is a sufficient condition to reconstruct the signal.

Experimental simulation results show that *Fast reconstruction* algorithm is ≈ 25 times faster then *Classical reconstruction* algorithm, if the signal length is 1 second, and ≈ 61, ≈ 130 and ≈ 228 times faster, if the length of the signal is 2, 4 and 8 seconds, respectively. As simulations show, both reconstruction approaches can ensure $\approx$24 bit resolution. But, by using real-time reconstruction method, the resolution is affected and drops to $\approx$22 bits.

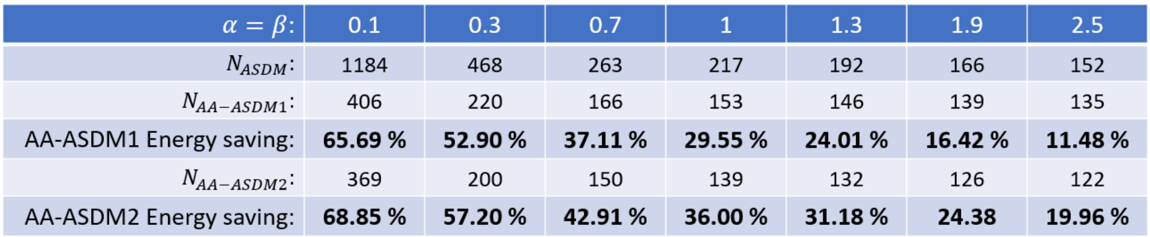
As expected from the theory, described in Section 3, the number of switching time instants decreases if proposed AA-ASDM instead of ASDM approach is used. The simulation results show that by using AA-ASDM1 (AA-ASDM with additional envelope encoding), the average number of trigger switching time instants per second $N_{AA-ASDM1}$, depending on selected $\alpha = \beta$ value, varies between 135 and 406, which is up to 65.69% less switchings then in ASDM case. The maximum distances between two consecutive trigger switching time instants (equaling nearly Nyquist step) occur all the time and are more spread. The signal reconstruction (fast and real-time) for AA-ASDM1 is exactly the same as in ASDM case with the same accuracy ≈22 bits. Since the main power consumer of a BCI system is a transmitter, by reducing the average number of time codes needed to be transmitted, the average power consumption of the transmitter decreases proportionally. Which means, also the energy consumption of the whole BCI system decreases.

Although AA-ASDM1 is advantageous over ASDM, there is a need to transmit two signals instead of one and there is a delay introduced by additional envelope signal reconstruction prior to the original signal reconstruction. Even further, the physical implementation of the AA-ASDM1 involves envelope detector which introduce additional delay to the system and synchronization with the original signal. Besides that the acquisition of the AA-ASDM1 envelop function is very difficult to obtain in practice, which complicates the physical implementation.

This can be overcome by using AA-ASDM2 (AA-ASDM without additional envelope encoding). In this case, the gain is even bigger and can reach up to 68.85% less switching time instants ($N_{AA-ASDM2}$) compared to ASDM. But this gain comes with more time consuming signal reconstruction due to optimization algorithm. Still, the performance of AA-ASDM2 is admissible, since reconstruction algorithms can reconstruct the signal in real-time on PC. Even further, AA-ASDM2 has lower complexity of the encoding circuit, since it does not involve an additional encoding circuit for the envelope, as it is in AA-ASDM1 case.

The comparison of ASDM, AA-ASDM1 and AA-ASDM2 is given in Table below:

Table: Comparison number of trigger switching time instants per second and corresponding energy saving of the transmitter for ASDM, AA-ASDM1 and AA-ASDM2



$\alpha = \beta$:	0.1	0.3	0.7	1	1.3	1.9	2.5
N_{ASDM}:	1184	468	263	217	192	166	152
$N_{AA-ASDM1}$:	406	220	166	153	146	139	135
AA-ASDM1 Energy saving:	**65.69 %**	**52.90 %**	**37.11 %**	**29.55 %**	**24.01 %**	**16.42 %**	**11.48 %**
$N_{AA-ASDM2}$:	369	200	150	139	132	126	122
AA-ASDM2 Energy saving:	**68.85 %**	**57.20 %**	**42.91 %**	**36.00 %**	**31.18 %**	**24.38**	**19.96 %**

Since AA-ASDM2 is particularly advantageous over ASDM if used for wide dynamic range signals, such as EEG signals, it could be effectively used also for other wide dynamic range signals, for example, electromyogram, electrooculogram, electrocardiogram, seismic and other signals.

Modeling

In order to understand the difference between ASDM and AA-ASDM2 in terms of hardware implementation complexity and power consumption, in Section 4.2, first an ASDM circuit is developed, modeled and analyzed. ASDM circuit consists of three OpAmps and few passive elements which determine the circuit parameters and switching activity. The modeling results show that ASDM circuit consumes $\approx$144mW, but it should be noted that modeling were made only to verify the working principles of the circuit, where power consumption value is just indicative, as it is possible to develop ASDM circuit with 7.5nW power consumption by developing specialized circuit structures with decreased operating supply voltage, capacities, etc. and increased slew rates.

Since the main power consumer of a BCI system is transmitter, it is also modeled and simulated together with ASDM. For this particular BCI application, due to its great properties, an On-Off Keying (OOK) technique is used for wireless data transmission. It has several advantages - simple architecture, good performance in the presence of co-channel interference, robust when exposed to vibration and shock, and major criteria - small dimensions for on-head device implementation.

Although power consumption of the ASDM circuit does not changes based on the average number of trigger switchings, the average power consumption of presented event-driven OOK circuit decrease proportionally if the number of ASDM switching time instants (N_{ASDM}) decreases. Also, the power consumption of the presented OOK circuit varies depending on selected pulse width, during which the LC is oscillating thus ensuring the data transmission.

The electrical circuit of the AA-ASDM2 is similar to the ASDM circuit, only in this case it is supplemented by an Analog Multiplier and a Voltage Follower OpAmp. The power consumption of the AA-ASDM2 circuit is $\approx$60% higher ($\approx$231mW), compared to ASDM, but it should be noted that if the AA-ASDM2 circuit would be implemented in the same technology as shown in [25], the increase of power consumption by 60% compared to ASDM would give an overall power consumption of the AA-ASDM2 circuit: 12nW.

The simulation results show that the reduction of the power consumption of the transmitter by using AA-ASDM2, instead of ASDM, can reach up to $\approx$37% depending on the number of switching time instants. But, it is important to note that in this case the power consumption of the circuit when it is not transmitting is very high $\approx$16.6mW, while modern wireless transmitters have it in a range of nW. This means, if the the power consumption of the circuit, when it is not transmitting, would be for example 100nW, the reduction of the power consumption could reach up to ≈50% depending on the number of switching time instants.

Since modern wireless data transmitters consume ≈3mW, it follows that if AA-ASDM2 instead of ASDM is used, it is possible to reduce the power consumption of the transmitter by 1.5mW, while increasing the power consumption of the encoder just by 4.5nW, giving a total of ≈50% power consumption reduction in a wireless BCI system, leading to 2 times longer battery life and operation time of wireless system. These parameters can be increased even more if the system is used for wide dynamic range signals, such as EEG signals.

Practical Implementation

In Section 4.3, a hardware prototype of AA-ASDM2 based EEG data acquisition system is developed and tested. The system consists of a wireless sensor (fabricated in cylindrical 16x11mm package), which includes EEG signal amplifier, AA-ASDM2 and OOK transmitter, and receiving and processing unit, which includes super-heterodyne receiver, ATS460 digitizer and personal computer (PC). Developed PCBs are shown in Fig. 4.37.

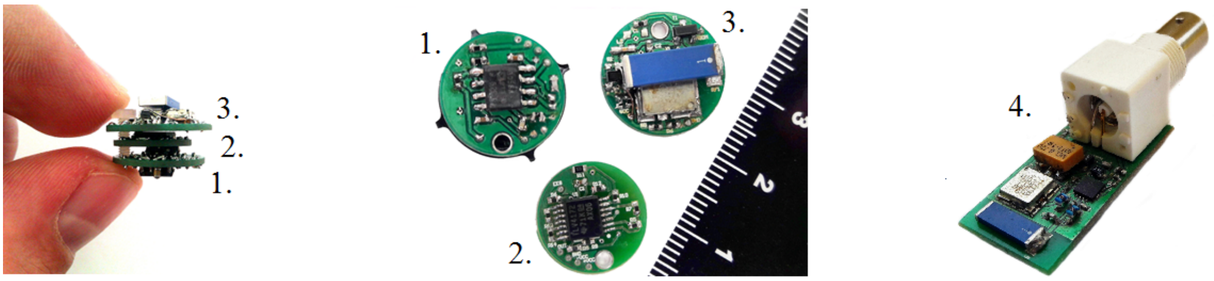


Figure 4.37: Developed AA-ASDM2 based EEG data acquisition system's components: 1. EEG amplifier, 2. AA-ASDM, 3. OOK transmitter, 4. receiver

First, by using electrodes the weak EEG signal is obtained. Then by using the first PCB: EEG signal amplifier, the signal is amplified ∼10 000 times. After EEG signal amplification, it is time encoded by the second PCB: AA-ASDM2. The output signal of the AA-ASDM2 is wirelessly transmitted by using the third PCB: OOK transmitter. By varying carrier frequency for different AA-ASDM2 outputs, it is possible to create a multi-channel BCI system. There is a trade-off with the selection of the transmitting pulse width. On one hand, the shorter the pulse, the wider the occupied spectrum bandwidth. On the other hand, the wider the pulse, the higher the power consumption of a transmitter. After EEG signal being amplified, encoded and transmitted, it is received by a super-heterodyne receiver (one receiver for all sensors) where high frequency signal is down-converted to a fixed intermediate frequency (IF). IF signal is imported into PC, by using *ATS460* 14-bit, 125 MS/s digitizer, for further signal processing. At the end of signal processing AA-ASDM2 output switching time instants t_k are obtained and signal reconstruction, based on the algorithm shown in Section 4.1.3.2, can begin.

In order to estimate the performance of the developed system, and most importantly of the AA-ASDM2, the experimental setup is build and system tested. First, to verify if the developed AA-ASDM2 PCB is working properly, it is calibrated. The calibration setup consists of a signal generator, which generates test signals, physical AA-ASDM2 PCB, 2-channel ATS460 digitizer (20Msps) and PC. In order to calibrate the circuit, the outputs of the AA-ASDM2 integrator and trigger for three different DC input signal voltage levels (0V, 0.5V and 0.9V) are acquired, digitized and imported into PC for further processing. At the end of signal processing/calibration, the actual AA-ASDM2 integrator and trigger parameter values κ, $+b$, $-b$, δ^+ and δ^- are obtained and inserted in the signal reconstruction algorithm.

After the calibration, AA-ASDM2 circuit is placed in the overall AA-ASDM2 based EEG data acquisition system for further experimental tests. Knowing the parameters of the AA-ASDM2 circuit and by using *Matlab* program, it is possible to acquire, process and reconstruct the original signal.

Visually it is hard to see the difference between the original and reconstructed signal, which means that it is possible to encode and qualitatively reconstruct the signal, by using AA-ASDM2. In order to quantitatively estimate the performance of AA-ASDM it must be implemented in a chip, with specialized circuit structures with decreased operating supply voltage, capacities, etc., with low comparator jitter, slew rate, DC gain, voltage saturation, excess loop delays and comparator offsets, etc. Also, Time-to-digital converter (in this case *ATS460*) has a significant impact on the precision of the reconstructed signal. The precision of the reconstructed signal is directly proportional to the t_k measurement precision.

5. CONCLUSIONS

The main aim of this thesis to develop an improved method for signal encoding based on ASDM, which allows to reduce the power consumption of the wireless BCI system, while maintaining the desired signal quality has been reached.

According to defined tasks, following results are achieved:

- based on the literature review and analysis of electroencephalogram (EEG) signals and brain computer interface (BCI) systems, the author has defined the requirements for Analog-to-digital converters (ADC). These requirements are: energy efficiency, encoding complexity, resolution and sampling rate;
- based on defined requirements, the author has reviewed, analyzed and described a literature on synchronous and asynchronous ADCs as well as selected the most appropriate ADC for EEG signal encoding and BCI systems as whole; Due to its great properties, Asynchronous Sigma Delta modulator (ASDM) is selected;
- the author has extensively analyzed, researched and described the ASDM and identified the points of improvement. The main inefficiency of the ASDM is related to over-triggering, which occurs when ASDM is applied to wide dynamic range signals such as EEG signals. Due to wide dynamic range that these signals have, a high switching activity of ASDM circuit appears when the input signal amplitude is low;
- in order to improve te ASDM, the author has proposed and in detail described a new method, called Amplitude-Adaptive Asynchronous Sigma-Delta modulator (AA-ASDM), which allows to reduce the over-triggering of the circuit and thus the power consumption of the whole wireless BCI system, while maintaining the desired signal quality;
- in order to test and assess the proposed method, the author has simulated (in *Matlab*), modeled (in *SIMetrix*), designed (in *Altium Designer*) and developed the AA-ASDM. The main achieved results are provided and assessed, and conclusions given;
- the author has developed and tested a complete one channel BCI system. The main achieved results are provided and assessed, and conclusions given.

Scientific results of the research have been published in the following papers:

- Ozols K., Shavelis R., *Amplitude Adaptive ASDM without Envelope Encoding*, 2016 24th

European Signal Processing Conference (EUSIPCO), Budapest, 2016, pp. 165-169.

- Ozols K., *Implementation of reception and real-time decoding of ASDM encoded and wirelessly transmitted signals*, 2015 25th International Conference Radioelektronika (RADIOELEKTRONIKA), Pardubice, 2015, pp. 236-239.

- Ozols K., Greitans M., Shavelis R., *Amplitude Adaptive Asynchronous Sigma-Delta Modulator*, 2013 8th International Symposium on Image and Signal Processing and Analysis (ISPA 2013), Trieste, 2013, pp. 460-464

- Ozols K., Greitans M., Shavelis R., *EEG Data Acquisition System Based on Asynchronous Sigma-Delta Modulator*, 2012 13th Biennial Baltic Electronics Conference, Tallinn, 2012, pp. 183-186.

Based on the results, described above, it can be concluded that the main aim of this work is achieved.

APPENDIX

APPENDIX A

MATHEMATICAL EXPRESSIONS FOR AA-ASDM

A-1 Vector $\mathbf{g}_k$

Vector $\mathbf{g}_k$ is defined as in (3.50):

$$\mathbf{g}_k = \begin{bmatrix} \int_{tk}^{t_{k+1}} g_0(t)dt \\ \int_{tk}^{t_{k+1}} g_1(t)dt \\ \vdots \\ \int_{tk}^{t_{k+1}} g_{N-1}(t)dt \end{bmatrix}, \quad \text{where}$$

$$g_n(t) = \begin{cases} 1 & \text{if} \quad n = 0 \\ \text{cos}n\frac{2\pi}{T}t & \text{if} \quad n \in [1, M] \\ \text{sin}(n-M)\frac{2\pi}{T}t & \text{if} \quad n \in [M+1, N-1] \end{cases} \qquad \text{where} \quad M = \frac{N-1}{2}.$$

Considering

$\int_{t_k}^{t_{k+1}} dt = t_{k+1} - t_k$

$\int_{t_k}^{t_{k+1}} \text{cos}n\frac{2\pi}{T}tdt = \frac{T}{n2\pi}\Big[\text{sin}n\frac{2\pi}{T}t_{k+1} - \text{sin}n\frac{2\pi}{T}t_k\Big]$, where $n \in [1, M]$

$\int_{t_k}^{t_{k+1}} \text{sin}n\frac{2\pi}{T}tdt = -\frac{T}{n2\pi}\Big[\text{cos}n\frac{2\pi}{T}t_{k+1} - \text{cos}n\frac{2\pi}{T}t_k\Big]$, where $n \in [M+1, N-1]$

and

$$\int_{t_k}^{t_{k+1}} g_n(t)dt = \begin{cases} t_{k+1} - t_k & \text{if} \quad n = 0 \\ \frac{T}{n2\pi}\Big[\text{sin}n\frac{2\pi}{T}t_{k+1} - \text{sin}n\frac{2\pi}{T}t_k\Big] & \text{if} \quad n \in [1, M] \\ -\frac{T}{n2\pi}\Big[\text{cos}n\frac{2\pi}{T}t_{k+1} - \text{cos}n\frac{2\pi}{T}t_k\Big] & \text{if} \quad n \in [M+1, N-1] \end{cases}$$

from the previous expressions follows

$$\mathbf{g}_k = \begin{bmatrix} t_{k+1} - t_k \\ \frac{T}{1\cdot 2\pi}\Big(\text{sin}1\frac{2\pi}{T}t_{k+1} - \text{sin}1\frac{2\pi}{T}t_k\Big) \\ \vdots \\ \frac{T}{M\cdot 2\pi}\Big(\text{sin}M\frac{2\pi}{T}t_{k+1} - \text{sin}M\frac{2\pi}{T}t_k\Big) \\ -\frac{T}{1\cdot 2\pi}\Big(\text{cos}1\frac{2\pi}{T}t_{k+1} - \text{cos}1\frac{2\pi}{T}t_k\Big) \\ \vdots \\ -\frac{T}{M\cdot 2\pi}\Big(\text{cos}M\frac{2\pi}{T}t_{k+1} - \text{cos}M\frac{2\pi}{T}t_k\Big) \end{bmatrix}.$$

A-2 Matrix $\hat{\mathbf{G}}_k$

Matrix $\hat{\mathbf{G}}_k$ elements are given as $\hat{G}_{k_{mn}} = \int_{tk}^{t_{k+1}} g_m(t)g_n(t)dt$, where $m = 0, 1, ..., N-1$ and $n = 0, 1, ..., N-1$:

$$\hat{\mathbf{G}}_k = \begin{bmatrix} \int_{tk}^{t_{k+1}} g_0(t)dt & \int_{tk}^{t_{k+1}} g_0(t)g_1(t)dt & \int_{tk}^{t_{k+1}} g_0(t)g_2(t)dt & \cdots & \int_{tk}^{t_{k+1}} g_0(t)g_{N-1}(t)dt \\ \int_{tk}^{t_{k+1}} g_1(t)g_0(t)dt & \int_{tk}^{t_{k+1}} g_1(t)g_1(t)dt & \int_{tk}^{t_{k+1}} g_1(t)g_2(t)dt & \cdots & \int_{tk}^{t_{k+1}} g_1(t)g_{N-1}(t)dt \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \int_{tk}^{t_{k+1}} g_{N-1}(t)g_0(t)dt & \int_{tk}^{t_{k+1}} g_{N-1}(t)g_1(t)dt & \int_{tk}^{t_{k+1}} g_{N-1}(t)g_2(t)dt & \cdots & \int_{tk}^{t_{k+1}} g_{N-1}(t)g_{N-1}(t)dt \end{bmatrix}.$$

Considering $g_n(t)$ in Appendix A-1, it follows:

$$\hat{\mathbf{G}}_k = \begin{bmatrix} c_0 & c_1 & c_2 & \cdots & c_M & s_1 & s_2 & \cdots & s_M \\ c_1 & c_1c_1 & c_1c_2 & \cdots & c_1c_M & c_1s_1 & c_1s_2 & \cdots & c_1s_M \\ c_2 & c_2c_1 & c_2c_2 & \cdots & c_2c_M & c_2s_1 & c_2s_2 & \cdots & c_2s_M \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ c_M & c_Mc_1 & c_Mc_2 & \cdots & c_Mc_M & c_Ms_1 & c_Ms_2 & \cdots & c_Ms_M \\ s_1 & s_1c_1 & s_1c_2 & \cdots & s_1c_M & s_1s_1 & s_1s_2 & \cdots & s_1s_M \\ s_2 & s_2c_1 & s_2c_2 & \cdots & s_2c_M & s_2s_1 & s_2s_2 & \cdots & s_2s_M \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ s_M & s_Mc_1 & s_Mc_2 & \cdots & s_Mc_M & s_Ms_1 & s_Ms_2 & \cdots & s_Ms_M \end{bmatrix},$$

the elements of which are determined by the corresponding integral values:

$$c_0 = \int_{t_k}^{t_{k+1}} dt = t_{k+1} - t_k$$

$$c_n = \int_{t_k}^{t_{k+1}} \cos n\tfrac{2\pi}{T}tdt = \tfrac{T}{n2\pi}\left[\sin n\tfrac{2\pi}{T}t_{k+1} - \sin n\tfrac{2\pi}{T}t_k\right]$$

$$s_n = \int_{t_k}^{t_{k+1}} \sin n\tfrac{2\pi}{T}tdt = -\tfrac{T}{n2\pi}\left[\cos n\tfrac{2\pi}{T}t_{k+1} - \cos n\tfrac{2\pi}{T}t_k\right]$$

$$c_mc_n = \int_{t_k}^{t_{k+1}} \cos m\tfrac{2\pi}{T}t \cdot \cos n\tfrac{2\pi}{T}tdt = \tfrac{1}{2}\int_{t_k}^{t_{k+1}} \cos(m-n)\tfrac{2\pi}{T}tdt + \tfrac{1}{2}\int_{t_k}^{t_{k+1}} \cos(m+n)\tfrac{2\pi}{T}tdt =$$

$$= \tfrac{T}{4\pi(m-n)}\sin(m-n)\tfrac{2\pi}{T}t\Big|_{t_k}^{t_{k+1}} + \tfrac{T}{4\pi(m+n)}\sin(m+n)\tfrac{2\pi}{T}t\Big|_{t_k}^{t_{k+1}} =$$
$$= \tfrac{T}{4\pi(m-n)}\left[\sin\tfrac{2\pi(m-n)}{T}t_{k+1} - \sin\tfrac{2\pi(m-n)}{T}t_k\right] + \tfrac{T}{4\pi(m+n)}\left[\sin\tfrac{2\pi(m+n)}{T}t_{k+1} - \sin\tfrac{2\pi(m+n)}{T}t_k\right],$$

from which it follows, that:

$$c_mc_n = \int_{t_k}^{t_{k+1}} \cos m\tfrac{2\pi}{T}t \cdot \cos n\tfrac{2\pi}{T}tdt =$$
$$= \begin{cases} \tfrac{t_{k+1}-t_k}{2} + \tfrac{T}{4\pi(m+n)}\left[\sin\tfrac{2\pi(m+n)}{T}t_{k+1} - \sin\tfrac{2\pi(m+n)}{T}t_k\right], & \text{if } m = n \\ \tfrac{T}{4\pi(m-n)}\left[\sin\tfrac{2\pi(m-n)}{T}t_{k+1} - \sin\tfrac{2\pi(m-n)}{T}t_k\right] + \tfrac{T}{4\pi(m+n)}\left[\sin\tfrac{2\pi(m+n)}{T}t_{k+1} - \sin\tfrac{2\pi(m+n)}{T}t_k\right], & \text{if } m \neq n \end{cases}$$

$$\begin{aligned} s_m s_n &= \int_{t_k}^{t_{k+1}} \sin m\tfrac{2\pi}{T}t \cdot \sin n\tfrac{2\pi}{T}tdt = \tfrac{1}{2}\int_{t_k}^{t_{k+1}} \cos(m-n)\tfrac{2\pi}{T}tdt - \tfrac{1}{2}\int_{t_k}^{t_{k+1}} \cos(m+n)\tfrac{2\pi}{T}tdt = \\ &= \tfrac{T}{4\pi(m-n)}\left[\sin\tfrac{2\pi(m-n)}{T}t_{k+1} - \sin\tfrac{2\pi(m-n)}{T}t_k\right] - \tfrac{T}{4\pi(m+n)}\left[\sin\tfrac{2\pi(m+n)}{T}t_{k+1} - \sin\tfrac{2\pi(m+n)}{T}t_k\right], \end{aligned}$$

from which it follows, that:

$$s_m s_n = \int_{t_k}^{t_{k+1}} \sin m\tfrac{2\pi}{T}t \cdot \sin n\tfrac{2\pi}{T}tdt =$$
$$= \begin{cases} \tfrac{t_{k+1-t_k}}{2} - \tfrac{T}{4\pi(m+n)}\left[\sin\tfrac{2\pi(m+n)}{T}t_{k+1} - \sin\tfrac{2\pi(m+n)}{T}t_k\right], & \text{if} \;\; m=n \\ \tfrac{T}{4\pi(m-n)}\left[\sin\tfrac{2\pi(m-n)}{T}t_{k+1} - \sin\tfrac{2\pi(m-n)}{T}t_k\right] - \tfrac{T}{4\pi(m+n)}\left[\sin\tfrac{2\pi(m+n)}{T}t_{k+1} - \sin\tfrac{2\pi(m+n)}{T}t_k\right], & \text{if} \;\; m \neq n \end{cases}$$

$$\begin{aligned} c_m s_n &= \int_{t_k}^{t_{k+1}} \cos m\tfrac{2\pi}{T}t \cdot \sin n\tfrac{2\pi}{T}tdt = \tfrac{1}{2}\int_{t_k}^{t_{k+1}} \sin(n-m)\tfrac{2\pi}{T}tdt - \tfrac{1}{2}\int_{t_k}^{t_{k+1}} \sin(m+n)\tfrac{2\pi}{T}tdt = \\ &= \tfrac{T}{4\pi(m-n)}\cos(n-m)\tfrac{2\pi}{T}t\Big|_{t_k}^{t_{k+1}} - \tfrac{T}{4\pi(m+n)}\cos(m+n)\tfrac{2\pi}{T}t\Big|_{t_k}^{t_{k+1}} = \\ &= \tfrac{T}{4\pi(m-n)}\left[\cos\tfrac{2\pi(m-n)}{T}t_{k+1} - \cos\tfrac{2\pi(m-n)}{T}t_k\right] - \tfrac{T}{4\pi(m+n)}\left[\cos\tfrac{2\pi(m+n)}{T}t_{k+1} - \cos\tfrac{2\pi(m+n)}{T}t_k\right], \end{aligned}$$

from which it follows, that:

$$c_m s_n = \int_{t_k}^{t_{k+1}} \cos m\tfrac{2\pi}{T}t \cdot \sin n\tfrac{2\pi}{T}tdt =$$
$$= \begin{cases} -\tfrac{T}{4\pi(m+n)}\left[\cos\tfrac{2\pi(m+n)}{T}t_{k+1} - \cos\tfrac{2\pi(m+n)}{T}t_k\right], & \text{if} \;\; m=n \\ \tfrac{T}{4\pi(m-n)}\left[\cos\tfrac{2\pi(m-n)}{T}t_{k+1} - \cos\tfrac{2\pi(m-n)}{T}t_k\right] - \tfrac{T}{4\pi(m+n)}\left[\cos\tfrac{2\pi(m+n)}{T}t_{k+1} - \cos\tfrac{2\pi(m+n)}{T}t_k\right], & \text{if} \;\; m \neq n \end{cases}$$

APPENDIX B

FUNCTIONS FOR SIGNAL ENCODING/DECODING WITH ASDM

B-1 *Matlab* Function for Signal Encoding with ASDM

```
function [tk,ASDM_int,ASDM_trig]=ASigmaDeltaM(y,t,d,b)
% Function for signal encoding with asynchronous sigma-delta modulator (ASDM)

% tk - switching instants of the ASDM trigger output
% ASDM_int - output signal of the ASDM integrator
% ASDM_trig - output signal of the ASDM trigger
% y - input signal
% t - time vector
% d - ASDM trigger hysteresis parameter
% b - ASDM trigger hysteresis parameter

tk=[]; %vector, which will consist of ASDM trigger switching instants
dt=mean(diff(t)); %discretization step of the input signal
ASDM_int=(y(1)+b)*dt; ASDM_trig=b;
for k=2:length(y)
    ASDM_int=[ASDM_int,ASDM_int(end)+(y(k)+b)*dt];
    ASDM_trig=[ASDM_trig,b];
    if abs(ASDM_int(end))>d && abs(ASDM_int(end-1))<=d
        b=-b; tk=[tk,t(k)];
    end
end
```

An example of how to use *ASigmaDeltaM* function:

```
load(’EEG.mat’); % uploads fourteen (14) EEG signals
signal=ch4(1:256)’; % selects 4-th channel
Fd=128; % discretization frequency of the EEG signal
Flow=0; Fhigh=49; % Limiting signal frequency band from Flow to Fhigh
M=64;
[x,y]=IdLFilter(signal,Fd,Flow,Fhigh,M);
FdNew=Fd*M; % discretization frequency after filtering
t=[0:length(y)-1]/FdNew;
b=180+180*0.1; % parameter b should be choosen as follow: 2*max(abs(y))
d=0.9*(b-180)/4/Fmax;; % parameter d should be: d<=(b-max(abs(y)))/4/Fhigh
[tk,ASDM_int,ASDM_trig]=ASigmaDeltaM(y,t,d,b); % use ”ASigmaDeltaM” function
plot(t,y,t,ASDM_int,t,ASDM_trig) % plots input, integrator and trigger signals
```

B-2 *Matlab* Function (1): Signal Decoding from ASDM Output Switching Instants

```
function [Sat,t] = ASDMrec1(tk,d,b,Fmax)
% Signal decoding from asynchronous sigma-delta modulator output switching instants

% Sat - vector of the reconstructed signal
% t - time vector
% tk - switching instants of the ASDM trigger output
% d - ASDM trigger hysteresis parameter
% b - ASDM trigger hysteresis parameter
% Fmax - max frequency of the reconstruction functions

if nargin<5, % default time vector
    t=tk(1):1/80000/Fmax:tk(end);
end;

fmax=1/2/max(diff(tk)); % maximum allowable frequency
fprintf('Fmax can be no greater than %4.4f [Hz] \n', fmax);

sl=tk(1):1/2/Fmax:tk(end);
n=1:length(tk)-1;
q=(-1)ˆ.n.*(2*d-b*(tk(n+1)-tk(n))); %vector q

G=[]; dt=1/2/32/Fmax; t=tk(1):dt:tk(end);

for n=1:length(sl); % calculation of the G matrix elements
    g=sinc(2*Fmax*(t-sl(n)));
    for m=1:length(tk)-1;
        G(m,n)=sum(g(t<tk(m+1)&t>=tk(m)))*dt;
    end
end
c=pinv(G)*q'; % coefficient vector c

Sat=0; % reconstructed/decoded signal
for n=1:length(c);
    Sat=Sat+c(n)*sinc(2*Fmax*(t-sl(n)));
end
```

An example of how to use *ASDMrec1* function:

```
% paramters tk, d, b follow from Appendix B-1

[Sat,tt]=ASDMrec1(tk,d,b,50);
plot(t,y,tt,Sat,'r') % reconstructed signal (red color)
```

B-3 *Matlab* Function (2): Signal Decoding from ASDM Output Switching Instants

```
function [Sat,t] = ASDMrec3(tk,d,b,Fmax,t)
% Fast Signal decoding from asynchronous sigma-delta modulator output switching instants

% Sat - vector of the reconstructed signal
% t - time vector
% tk - switching instants of the ASDM trigger output
% d - ASDM trigger hysteresis parameter
% b - ASDM trigger hysteresis parameter
% Fmax - max frequency of the reconstruction functions

if nargin<5, % default time vector
    t=tk(1):1/8/Fmax:tk(end);
end;

fmax=1/2/max(diff(tk)); % maximum allowable frequency
if fmax<Fmax
    fprintf(’Fmax should not exceed %4.4f [Hz] \n’, fmax);
end;
M=2*ceil((tk(end)-tk(1))*Fmax);
alfa=2*Fmax/(2*M+1);
n=1:length(tk); tk=[tk,0];
q=(-1).^n.*(2*d-b*(tk(n+1)-tk(n))); q=q’; % vector q
D=diag(tk(2:end)-tk(1:end-1));
tk(end)=[];
Pm1=-1+zeros(length(q),length(q));
Pm1=triu(Pm1);
m=-M:M;
X=exp(-1i*m’*2*pi*Fmax/M*tk);
A=alfa*X*D*X’;
bm=alfa*pinv(A)*X*D*Pm1*q; % coefficient vector bm
Sat=0; % reconstructed/decoded signal
for n=1:length(m);
    Sat=Sat+m(n)*bm(n)*exp(1i*m(n)*2*pi*Fmax/M*t);
end
Sat=real(Sat*1i*2*pi*Fmax/M);
```

An example of how to use *ASDMrec3* function:

```
% paramters tk, d, b follow from Appendix B-1

[Sat,tt]=ASDMrec3(tk,d,b,50);
plot(t,y,tt,Sat,’r’) % reconstructed signal (red color)
```

B-4 *Matlab* Function: Real-Time Signal Decoding from ASDM Output Switching Time Instants

```
function [Sat,t] = ASDMrecFragm(tk,d,b,Fmax,Fd)
% Real-time Signal decoding from ASDM output switching instants

% Sat - vector of the reconstructed signal
% t - time vector
% tk - switching instants of the ASDM trigger output
% d - ASDM trigger hysteresis parameter
% b - ASDM trigger hysteresis parameter
% Fmax - max frequency of the reconstruction functions
% Fd - discretization frequency of the reconstructed signal

fmax=1/2/max(diff(tk)); %maximum permissible frequency
fprintf('Fmax can be no greater than %4.4f [Hz] \n', fmax);

if nargin<5, %default Fd value
    Fd=Fmax*8;
end;

L=30; M=2; K=2; %Window function parameters

for k=1:L-K-2*M:length(tk)+M-L
    if k+L>length(tk)
        L=length(tk)-k;
    end;

    ttk=tk(k:k+L);
    sl=ttk(1):1/2.5/Fmax:ttk(end);

    n=k:length(ttk)+k-2;
    q=(-1)ˆ.n.*(2*d-b*(ttk(n+1-k+1)-ttk(n-k+1))); %vector q
    G=[];
    for n=1:length(sl); %calculation of G matrix elements
        G(:,n)=sinc(2*Fmax*(ttk(2:end)-sl(n)))-sinc(2*Fmax*(ttk(1:end-1)-sl(n)));
    end
    c=pinv(G)*q'; %vector c (coefficients)
    xat=0;

    if k==1
        tt=ttk(1):1/Fd:ttk(end);
    else
        tt=tt(tt>=ttk(1));
        tt=[tt(1:end-1),tt(end):1/Fd:ttk(end)];
    end;

    for n=1:length(c);
        xat=xat+c(n)*sinc(2*Fmax*(tt-sl(n)));
    end

    xat=diff(xat)*Fd;
    xat=[xat,2*xat(end)-xat(end-1)]; %reconstructed signal in interval n

    if k==1 %reconstructed signal multiplication with window function and adding
        xat(tt<ttk(1+M))=0;
        ind1=tt>=ttk(1+M)&tt<=ttk(1+M+K);
        wind1=sin(pi/2*(tt(ind1)-ttk(1+M))/(ttk(1+M+K)-ttk(1+M)))^.2;
        xat(ind1)=xat(ind1).*wind1;
        ind2=tt>=ttk(1+L-M-K)&tt<=ttk(1+L-M);
        wind2=1-sin(pi/2*(tt(ind2)-ttk(1+L-M-K))/(ttk(1+L-M)-ttk(1+L-M-K)))^.2;
        xat(ind2)=xat(ind2).*wind2;
        xat(tt>ttk(1+L-M))=[];
        Xat=xat;
    else
        wL=length(wind2);
        ind1=tt>=ttk(1+M)&tt<=ttk(1+M+K);
        xat(ind1)=xat(ind1).*(1-wind2);
        ind2=tt>=ttk(1+L-M-K)&tt<=ttk(1+L-M);
        wind2=1-sin(pi/2*(tt(ind2)-ttk(1+L-M-K))/(ttk(1+L-M)-ttk(1+L-M-K)))^.2;
        xat(ind2)=xat(ind2).*wind2;
        xat(tt<ttk(1+M)|tt>ttk(1+L-M))=[];
        Xat(end-wL+1:end)=Xat(end-wL+1:end)+xat(1:wL);
        Xat=[Xat,xat(wL+1:end)]; %adding to overall reconstructed signal
    end;
end;

t=tk(1):1/Fd:tk(1)+(length(Xat)-1)*1/Fd;
[x,Sat]=IdLFilter(Xat,Fd,0,Fmax,1); %reconstruction of the signal’s initial frequency band
```

An example of how to use *ASDMrecFragm* function:

```
% paramters tk, d, b follow from Appendix B-1

[Sat1,tt1] = ASDMrecFragm(tk,d,b,Fhigh,FdNew);
plot(t,y,tt1,Sat1,’r’,tt1,Sat1-interp1(t,y,tt1),’k’)
```

APPENDIX C

FUNCTIONS FOR SIGNAL ENCODING/DECODING WITH AA-ASDM WITH ADDITIONAL ENVELOPE ENCODING

C-1 *Matlab* Function for Signal Encoding with AA-ASDM1

```
 function [tk,ASDM_int,ASDM_trig,okk]=ASigmaDeltaM3(y,t,d,b)
% Function for signal encoding with amplitude adaptive asyncrhonous sigma-delta modulator
% with additional envelop encoding (AA-ASDM1)

% tk - switching instants of the AA-ASDM1 trigger output
% ASDM_int - output signal of the AA-ASDM1 integrator
% ASDM_trig - output signal of the AA-ASDM1 trigger
% y - input signal
% t - time vector
% d - AA-ASDM1 trigger hysteresis parameter
% b - AA-ASDM1 trigger hysteresis parameter

tk=[]; %vector, which will consist of AA-ASDM1 trigger switching instants
dt=mean(diff(t)); %discretization step of the input signal
ASDM_int=(y(1)+b(1))*dt; ASDM_trig=1; ok=1;okk=ok;
for k=2:length(y)
    ASDM_int=[ASDM_int,ASDM_int(end)+(y(k)+ok*b(k))*dt];
    ASDM_trig=[ASDM_trig,ok];
    if abs(ASDM_int(end))>d && abs(ASDM_int(end-1))<=d
        ok=-ok; tk=[tk,t(k)];
    end
    okk=[okk,ok];
end
```

An example of how to use *ASigmaDeltaM3* function:

```
load(’EEG.mat’); % uploads fourteen (14) EEG signals
signal=ch4(1:256)’; % selects 4-th channel
Fd=128; % discretization frequency of the EEG signal
Flow=0; Fhigh=49; % limiting signal frequency band from Flow to Fhigh
M=64;
[x,y]=IdLFilter(signal,Fd,Flow,Fhigh,M); %filtering
y = y./180; %signal rationing
FdNew=Fd*M; % discretization frequency after filtering
t=[0:length(y)-1]/FdNew;
```

```
s=size(t,2);
tau=26e-6:1e-6:499e-6; % variable time constant
sizetau=size(tau,2);
output=zeros(474,s); % output of the envelope detector
y2=abs(y);
for j=1:sizetau % simulation of RC circuit for envelope detection
        for k=1:s-1
            if(y2(k)<output(j,k))
            output(j,k+1)=output(j,k)*exp(-1e-7/tau(j));
        else
            output(j,k+1)=y2(k);
        end
    end
end
RC_coef = 1; % RC coeficient
apl = output(RC_coef,:); % envelope
apl(1)=2*apl(2)-apl(3);
[pks,locs]=findpeaks(abs(apl)); % find envelope’s peak values and its location
Zer = zeros(size(apl));
Zer(locs) = pks; % leave only peak values in the vector
tt=t;
tt(Zer<2e-4)=[]; % adjustin to threshold
Zer2=Zer;
Zer2(Zer<2e-4)=[]; % adjustin to threshold
Zer2 = interp1([t(1),tt,t(end)],[apl(1),Zer2,apl(end)],t,’pchip’);
apl2 = Zer2;
if min(apl2-y)<0 % acquiring a new envelope
    apl2=apl2+abs(min(apl2-y));
end
[apl3,yyyy]=IdLFilter(apl2,1/mean(diff(t)),0,5,1); % evelope’s filtering
% max envelope frequency: 5Hz
apl3 = apl3+0.15; % off-set
C = 1; % max. input signal value

% y(t) original signal encoding:
beta = 0.1; % coefficient
b = apl3+beta*C; % adaptive hysteresis parameter ”b”
d = 0.9*(beta*C*(1/2/Fhigh))/2;
[tk,ASDM_int,ASDM_trig,okk]=ASigmaDeltaM3(y,t,d,b); % use ”ASigmaDeltaM3” func.
% apl3(t) envelope encoding:
b2=0.1*C;
d2=0.025;
[tk2,ASDM_int2,ASDM_trig2]=ASigmaDeltaM(apl3,t,d2,b2); % use ”ASigmaDeltaM” func.
plot(t,y,t,ASDM_int,t,ASDM_trig) % plots input, integrator and trigger signals
```

APPENDIX D

FUNCTIONS FOR SIGNAL ENCODING/DECODING WITH AA-ASDM WITHOUT ADDITIONAL ENVELOPE ENCODING

D-1 *Matlab* Function for Signal Encoding with AA-ASDM2

```
 function [y_filtered,tk]=AA_ASigmaDeltaM_Precise(y,Fs,Fmax,delta,beta)
% Function for signal encoding with amplitude adaptive asyncrhonous sigma-delta modulator
% without additional envelop encoding (AA-ASDM2)

% y_filtered - bandlimited (up to Fmax) input signal
% tk - switching instants of the AA-ASDM2 trigger output
% y - input signal
% Fs - discretization frequency
% Fmax - maximum frequency of the y signal's Fourier series (FS)
% delta - AA-ASDM1 trigger hysteresis parameter
% beta - AA-ASDM1 trigger hysteresis parameter

M=length(y);
T=M/Fs; %length of the signal
w=2*pi/T; %FS fundamental frequency (rad/s)

sp=fft(y); %DFT spectrum of y, from which follows FS coefficients
N=floor(Fmax/Fs*length(y)); %upper bound of FS

if N>=floor(M/2) %if the bound >= Fs/2
    N=floor(M/2);
    if rem(M,2)==1
        c=[sp(end-N+1:end),sp(1:N+1)]/M; %FS coefficients -N,...,N
        nn=(-N:N);
    else
        c=[sp(end-N+2:end),sp(1:N+1)]/M; %FS coefficients -N+1,...,N
        nn=(-N+1:N);
    end
else %if the bound < Fs/2
    c=[sp(end-N+1:end),sp(1:N+1)]/M; %FS coefficients -N,...,N
    nn=(-N:N);
end

y_filtered=real(c*exp(1i*w*nn'*(0:M-1)/Fs));
```

```
y_f_max=max(abs(y_filtered));

if y_f_max>1 %limiting the amplitude from -1 to 1
    y_filtered=y_filtered/y_f_max;
    c=c/y_f_max;
end;

if beta<0
    fprintf(’Warning: beta < 0, therefore set to 1 \n’);
    beta=1;
end

if delta>beta/4/Fmax;
    fprintf(’Warning: delta > beta/4/Fmax, therefore set to 0.9*beta/4/Fmax \n’);
    delta=0.9*beta/4/Fmax;
end

tk=0; %switching instants of the AA-ASDM2 trigger output
tkp1_end=1/2/Fmax*1.2; %max time value to which will be intersections searched
tkp1=linspace(0,tkp1_end,1000);
tkr=tk;
k=1;

MN=repmat(nn’,1,length(nn))+repmat(nn,length(nn),1);
MN0ind=(MN==0);
MN_i=-1i/w./MN;

while tkr<T
    cn=c./nn/1i/w.*exp(1i*w*nn*tkr);
    cn(end-N)=c(end-N);
    MN_i_exp=MN_i.*exp(1i*w*MN*tkr);

    ut1=0; %left side of the expression
    ut2=(-1)^k*(2*delta-(beta+0.25)*tkp1(1)); %right side of the expression
    dut_previous=ut1-ut2;

    for r=2:length(tkp1)
        yexp=exp(1i*w*nn’*tkp1(r))-ones(length(nn),1);
        yexp(end-N,1)=tkp1(r);
        ut11=real(cn*yexp); %integrated signal

        hv=MN_i_exp.*(exp(1i*w*MN*tkp1(r))-ones(length(nn),length(nn)));
        hv(MN0ind)=tkp1(r);
        ut12=real(c*hv*c.’); %integrated signal squared
        ut1=ut11+(-1)k*ut12; %left side of the expression
        ut2=(-1)k*(2*delta-(beta+0.25)*tkp1(r)); %right side of the expression
        dut=ut1-ut2;

        if sign(dut) =sign(dut_previous)
            tkr=-dut_previous/(dut-dut_previous)*(tkp1(r)-tkp1(r-1))+tkp1(r-1)+tkr;
            tk=[tk,tkr];
            k=k+1;
            break;
        else
            dut_previous=dut;
        end
    end
end

if tk(end)>T
    tk(end)=[];
end
```

An example of how to use *AA_ASigmaDeltaM_Precise* function:

```
load(’EEG.mat’); % uploads fourteen (14) EEG signals
signal=ch4(1:256)’; % selects 4-th channel
Fd=128; % discretization frequency of the EEG signal
Fmax=49; % limiting signal frequency band Fmax
beta=1;
delta=0.9*beta/4/Fmax;
[yf2,tk2]=AA_ASigmaDeltaM_Precise(signal,Fs,Fmax,delta,beta); %use the function
```

D-2 *Matlab* Function (1): Signal Decoding from AA-ASDM Output Switching Instants

```
 function [Sat,t] = AA_ASDM_rec_FS_Exp(tk,delta,beta,Fmax,t,f0_FS)
%Signal decoding from AA-ASDM2 output switching time instants

% Sat - reconstructed signal
% t - time vector
% tk - switching instants of the AA-ASDM2 trigger output
% delta - AA-ASDM2 trigger hysteresis parameter
% beta - AA-ASDM2 trigger hysteresis parameter
% Fmax - maximum frequency of FS functions
% f0_FS - FS fundamental frequency

if nargin<6 %default FS fundamental frequency
    T=tk(end)-tk(1); % length of the signal (period)
    f0_FS=1/T;
else
    if f0_FS<1/(tk(end)-tk(1))/2
        f0_FS=1/(tk(end)-tk(1))/2;
        fprintf('Warning: f0_FS may be too small, therefore set to default value. \n');
    end
end

if nargin<5, % default time vector
    t=tk(1):1/8/Fmax:tk(end);
end;

fmax=1/2/max(diff(tk)); %maximum permissible frequency
fprintf('Fmax can be no greater than %4.4f [Hz] \n', fmax);

n=1:length(tk)-1;
q=(-1)ˆ.n.*(2*delta-(0.25+beta)*diff(tk)); %vector q

w=2*pi*f0_FS; %FS fundamental frequency (rad/s)
N=floor(Fmax/f0_FS); %upper bound of FS
nn=-N:N;

G=-1i/w*repmat(1./nn',1,length(tk)-1).*(exp(1i*w*nn'*tk(2:end))-exp(1i*w*nn'*tk(1:end-1)));
G(N+1,:)=diff(tk); %matrix G

MN=repmat(nn',1,length(nn))+repmat(nn,length(nn),1);
MN0ind=(MN==0);
MN_i=-1i/w./MN;
```

```
Ghat=zeros(length(nn),length(nn),length(tk)-1);
Gexp1=exp(1i*w*MN*tk(1));

for k=1:length(tk)-1
    Gexp2=exp(1i*w*MN*tk(k+1));
    Ghat_k=MN_i.*(Gexp2-Gexp1);
    Ghat_k(MN0ind)=tk(k+1)-tk(k);
    Ghat(:,:,k)=Ghat_k;
    Gexp1=Gexp2;
end

c0=0.0+0.0*1i+zeros(1,length(nn));
[c,uu]=lsqnonlin(@(x)AA_ASDM_rec_FS_Exp_Fragm_Func(x,G,Ghat,q,n),c0);
Sat=real(c*exp(1i*w*nn’*t));
```

An example of how to use *AA_ASDM_rec_FS_Exp* function:

```
[Sat_AA_ASDM, ∼]=AA_ASDM_rec_FS_Exp(tk1,delta,beta,Fmax,t2,f0_FS);
```

D-3 *Matlab* Function (2): Signal Decoding from AA-ASDM Output Switching Instants

```
 function [Sat,t] = AA_ASDM_rec_FS_CosSin_Fragm(tk,delta,beta,Fmax,t,LMK,f0_FS)
% Real-time signal decoding from AA-ASDM2 output switching time instants

% Sat - reconstructed signal
% t - time vector
% tk - switching instants of the AA-ASDM2 trigger output
% delta - AA-ASDM2 trigger hysteresis parameter
% beta - AA-ASDM2 trigger hysteresis parameter
% Fmax - maximum frequency of FS functions
% f0_FS - FS fundamental frequency
% [L,M,K] - window function parameters

if nargin<7 %default FS fundamental frequency
    f0_FS=1/(tk(end)-tk(1));
else
    if f0_FS<1/(tk(end)-tk(1))/2
        f0_FS=1/(tk(end)-tk(1))/2;
        fprintf('Warning: f0_FS may be too small, therefore set to default value. \n');
    end
end
if nargin<6 %window function parameters L > 2*M+K
    L=20; M=2; K=2;
else
    L=LMK(1);
    M=LMK(2);
    K=LMK(3);
end;
if nargin<5, % default time vector
    t=tk(1):1/8/Fmax:tk(end);
end;
fmax=1/2/max(diff(tk)); %maximum permissible frequency
fprintf('Fmax can be no greater than %4.4f [Hz] \n', fmax);
flag=0;
for k=1:L-2*M-K:length(tk) % each next interval is reconstructed after L-2*M-K switchings
    if k+L-2*M-K+L>length(tk)
        L=length(tk)-k;
        flag=1;
    end;
    ttk=tk(k:k+L);
    n=k:length(ttk)+k-2;
    q=(-1)ˆ.n.*(2*delta-(0.25+beta)*diff(ttk)); %vector q
    [G,Ghat,T,NN]=AA_ASDM_rec_FS_CosSin_Fragm_matrices(ttk,Fmax,f0_FS);
    c0=0.0+zeros(2*NN+1,1); %initial coefficients cn
    [c,uu]=lsqnonlin(@(x)AA_ASDM_rec_FS_CosSin_Fragm_Func(x,G,Ghat,q,n),c0);
    if k==1
        tt=t(t>=ttk(1)&t<=ttk(end));
    else
        tt=t(t>ttk(1)&t<=ttk(end));
    end;
    xat=c(1); %reconstructed fragment in interval n
    for kk=2:NN+1
        xat=xat+c(kk)*cos(2*pi/T*(kk-1)*tt)+c(kk+NN)*sin(2*pi/T*(kk-1)*tt);
    end;
    if k==1 %reconstructed fragment multiplication with window function and adding
        xat(tt<ttk(1+M))=0;
        ind1=tt>=ttk(1+M)&tt<=ttk(1+M+K);
        wind1=sin(pi/2*(tt(ind1)-ttk(1+M))/(ttk(1+M+K)-ttk(1+M)))^.2;
        xat(ind1)=xat(ind1).*wind1;
        ind2=tt>=ttk(1+L-M-K)&tt<=ttk(1+L-M);
        wind2=1-sin(pi/2*(tt(ind2)-ttk(1+L-M-K))/(ttk(1+L-M)-ttk(1+L-M-K)))^.2;
        xat(ind2)=xat(ind2).*wind2;
        xat(tt>ttk(1+L-M))=[];
        Xat=xat;
    else
        wL=length(wind2);
        ind1=tt>=ttk(1+M)&tt<=ttk(1+M+K);
        xat(ind1)=xat(ind1).*(1-wind2);
        ind2=tt>=ttk(1+L-M-K)&tt<=ttk(1+L-M);
        wind2=1-sin(pi/2*(tt(ind2)-ttk(1+L-M-K))/(ttk(1+L-M)-ttk(1+L-M-K)))^.2;
        xat(ind2)=xat(ind2).*wind2;
        xat(tt<ttk(1+M)|tt>ttk(1+L-M))=[];
        Xat(end-wL+1:end)=Xat(end-wL+1:end)+xat(1:wL);
        Xat=[Xat,xat(wL+1:end)]; %adding to the overall reconstructed signal
    end;
    if flag
        break
    end
end;
Sat=[zeros(size(t(t<tk(1)))), Xat];
Sat=[Sat,zeros(1,length(t)-length(Sat))];
```

An example of how to use *AA_ASDM_rec_FS_CosSin_Fragm* function:

```
[Sat_AA_ASDM_Fragm_2,uu]=AA_ASDM_rec_FS_CosSin_Fragm(tk1,delta,beta,Fmax,t2,LMK,1.2);
plot(t2,yf,t2,Sat_AA_ASDM_Fragm_2,t2,yf-Sat_AA_ASDM_Fragm_2)
```

APPENDIX E

PHYSICAL AA-ASDM2 BASED EEG DATA ACQUISITION SYSTEM

E-1 *Matlab* program for AA-ASDM calibration

```
 load('AAdata01032017.mat') %load acquired data from the digitizer
int_dc00=AAint0; %input DC = 0V
trig_dc00=AAtrig0;
int_dc05=AAint05; %input DC = 0.5V
trig_dc05=AAtrig05;
int_dc09=AAint09; %input DC = 0.9V
trig_dc09=AAtrig09;
t=(0:length(int_dc00)-1)/20e6; %Digitizer sampling rate 20MSPS

figure
plot(t,int_dc00,t,int_dc05,t,int_dc09) %read max and min delta values from the graph
max_delta=10625; %from graph
min_delta=5625; %from graph

delta_plus=1.5; %actual delta+ value from oscilloscope
delta_minus=-1.5; %actual delta- value from oscilloscope

u=(max_delta*delta_minus-min_delta*delta_plus)/(delta_minus-delta_plus); %average value
gamma=delta_plus/(max_delta-u); %scale factor

int00=(int_dc00-u)*gamma; %convert to actual amplitude values
trig00=(trig_dc00-u)*gamma;
int05=(int_dc05-u)*gamma;
trig05=(trig_dc05-u)*gamma;
int09=(int_dc09-u)*gamma;
trig09=(trig_dc09-u)*gamma;

b2=abs(mean(trig00(trig00>0))); %find mean b values
b1=abs(mean(trig00(trig00<0)));

ind=find(abs(diff(diff(trig00)>0))); %max un min index
tk00=t(ind(find(abs(diff(trig00(ind)))>3)+1)); %trigger switching time instants

ind=find(abs(diff(diff(trig05)>0)));
tk05=t(ind(find(abs(diff(trig05(ind)))>3)+1));
```

```
ind=find(abs(diff(diff(trig09)>0)));
tk09=t(ind(find(abs(diff(trig09(ind)))>3)+1));

dtk00=diff(tk00); %difference between switching time instants
dtk05=diff(tk05);
dtk09=diff(tk09);

thr1=3.28e-3; %threshold. comes from dtk graphs
dtk00_max=mean(dtk00(dtk00>thr1)); %find average max distance between switchings
dtk00_min=mean(dtk00(dtk00<thr1)); %find average min distance between switchings
thr2=3e-3;
dtk05_max=mean(dtk05(dtk05>thr2));
dtk05_min=mean(dtk05(dtk05<thr2));
thr3=2.6e-3;
dtk09_max=mean(dtk09(dtk09>thr3));
dtk09_min=mean(dtk09(dtk09<thr3));

a1=1.2951; %coefficient a1. comes from theoretical calculations
a2=0.4098; %coefficient a2

% does actual dtk_max/dtk_min values corresponds to theoretically calculated?
[dtk00_max/dtk00_min, (a2/a1*b2+(0+0.5ˆ)2)/(a2/a1*b1+(0-0.5ˆ)2)]
[dtk05_max/dtk05_min, (a2/a1*b2+(0.5+0.5ˆ)2)/(a2/a1*b1+(0.5-0.5ˆ)2)]
[dtk09_max/dtk09_min, (a2/a1*b2+(0.9+0.5ˆ)2)/(a2/a1*b1+(0.9-0.5ˆ)2)]

R=12e3; % R11
C=220e-9; %C1
alfa_theoretical=1/R/C %integrator constant 1/κ

% practically acquired integrator constants
alfa_max=(delta_plus-delta_minus)/(a1*(0+0.5ˆ)2+a2*b2)/dtk00_min
alfa_min=(delta_plus-delta_minus)/(a1*(0-0.5ˆ)2+a2*b1)/dtk00_max
alfa_max=(delta_plus-delta_minus)/(a1*(0.5+0.5ˆ)2+a2*b2)/dtk05_min
alfa_min=(delta_plus-delta_minus)/(a1*(0.5-0.5ˆ)2+a2*b1)/dtk05_max
```

E-2 *Matlab* Function for signal decoding from real AA-ASDM based system

```
 %Set 14-bit ATS460 digitizer parameters Volts/DIV = mV and CLK
V_Div = 5e-3; %mV
Fs = 20e6; %20Ms/s

V_full = V_Div*8; %full range of the ATS460 digitizer
koef = 16384/V_full; % coefficient for acquiring actual amplitude values

data1 = csvread(’EEGsystemsData.csv’); %read data from ATS460 digitizer
data2 = data1(:,1)’;
y = (data2-8192)./koef; %Set 0 level and actual amplitude values

t = 0:(1/Fs):(length(y)-1)/Fs; %time vector

%Band-pass filter
ZF = 0.35e6; % bottom cut-off freq.
AF = 2.39e6; % top cut-off freq.
CF = 1.367e6; % Center freq.

sp = fft(y); %spectrum
fr = [0:length(y)-1]*Fs/length(y); %frequency vector
ind1 = fr<ZF; %all frequency values below ZF
n1 = length(fr(ind1));
sp2 = sp;
sp2(ind1) = 0; %zeroing
sp2(end-n1+2:end) = 0; %zeroing
ind2 = fr>=ZF & fr<=AF; %all frequency values above ZF and below AF
n2 = length(fr(ind2));
sp2(n1+n2+1:end-n1-n2+1) = 0; %zeroing
yf = ifft(sp2);

yy = cos(2*pi*CF(1)*t);
yj = yf.*yy; %evelope

[pks,locs]=findpeaks(abs(yj));
Zer = zeros(size(yj));
Zer(locs) = pks; %zeroing

tt=t;
tt(Zer<0.5e-3)=[]; % remove all values below threshold 0.4e-4
Zer2=Zer;
Zer2(Zer<0.5e-3)=[];
```

```
dtt = diff(tt); %vector with distance between switching time instants values
ind = dtt>1e-3;
tk = tt(ind);
tk = sort([tt(ind),tt(find(ind)+1)]); %switching time instants
tl = tk;
dt = 0.0001;
t = tl(1):dt:tl(end);

x=data1(:,2)’%original signal

t2=t(1:1000:end);
alfa=390; %parameters comes from Appendix E-1
delta=(delta_plus-delta_minus)/2/alfa/a1; %parameters comes from Appendix E-1
beta=[b1 b2]*a2/a1; %parameters comes from Appendix E-1
Fmax=60;

[xf, ]=IdLFilter(x,Fs,0,Fmax,1); %Filtering
xf=(xf(1:1000:end)-u)*gamma;

%Signal reconstruction. Function from Appendix D-2
[Sat_AA_ASDM, ]=AA_ASDM_rec_FS_Exp(tk(2:end),delta,beta,Fmax,t2);
plot(t2,xf,t2,Sat_AA_ASDM)

ErrInterval=[t2(1)+0.02 t2(end)-0.02]; %SNR
ind=t2>ErrInterval(1)&t2<ErrInterval(2);
SNR_AA_ASDM=10*log10(sum(xf(ind)^.2)/sum((xf(ind)-Sat_AA_ASDM(ind))^.2))
```

BIBLIOGRAPHY

[1] M.F. Bear, B.W. Connors, and M.A. Paradiso. *Neuroscience: Exploring the Brain, Third Edition*. Lippincott Williams & Wilkins, 2007, pp. 857, ISBN: 978-0781760034.

[2] S. Sanei and J.Chambers. *EEG Signal Processing*. Wiley, 2013, pp. 312, ISBN: 978-1118691236.

[3] A.E. Hassanien and A.T. Azar. *Brain-Computer Interfaces: Current Trends and Applications*. Springer International Publishing, 2014, pp. 416, ISBN: 978-3319109787.

[4] B. Graimann, B.Z. Allison, and G. Pfurtscheller. *Brain-Computer Interfaces: Revolutionizing Human-Computer Interaction*. The Frontiers Collection. Springer, 2010, pp. 407, ISBN: 978-3642020919.

[5] C. Guger, B.Z. Allison, and G. Edlinger. *Brain-Computer Interface Research: A State-of-the-Art Summary*. SpringerBriefs in Electrical and Computer Engineering. Springer, 2013, pp. 129, ISBN: 978-3642360831.

[6] I. Iturrate, J. Antelis, and J. Minguez. *Synchronous EEG Brain-Actuated Wheelchair with Automated Navigation*. Robotics and Automation. ICRA '09. IEEE International Conference on, 2009, pp. 2318-2325.

[7] K. Feong-Hun, F. Biessmann, and L. Seong-Whan. *Reconstruction of Hand Movements from EEG Signals Based on Non-Linear Regression*. Brain-Computer Interface (BCI), International Winter Workshop on, 2014, pp. 1-3.

[8] R. Rojas, D. Goehring, D. Latotzky, and M. Wang. *Semi-autonomous Car Control Using Brain Computer Interfaces*. Proceedings of the 12th International Conference IAS-12. Advances in Intelligent Systems and Computing, Volume 194., 2013, pp. 393-408.

[9] Emotiv EPOC headset. A high resolution, multi-channel eeg system. http://emotiv.com/, 2016. [Online; accessed 01-March-2016].

[10] BioRadio. Wireless physiology monitor. https://glneurotech.com/bioradio/bioradio-specifications/, 2016. [Online; accessed 01-March-2016].

[11] B-Alert X24 EEG System. 24 channel wireless eeg headset. http://www.advancedbrainmonitoring.com/xseries/x24/, 2016. [Online; accessed 03-November-2016].

[12] S. Gao, Y. Wang, X. Gao, and B. Hong. *Visual and Auditory Brain Computer Interfaces*. Biomedical Engineering, IEEE Transactions on, 2014, pp. 1436-1447.

[13] X. Fan, L. Bi, T. Teng, H. Ding, and Y. Liu. *A Brain-Computer Interface-Based Vehicle Destination Selection System Using P300 and SSVEP Signals*. Intelligent Transportation Systems, IEEE Transactions on, 2015, pp. 274-283.

[14] L. Zhu, H. Chen, X. Zhang, K. Guo, S. Wang, Y. Wang, W. Pei, and H. Chen. *Design of Portable Multi-Channel EEG Signal Acquisition System*. Biomedical Engineering and Informatics, BMEI '09. 2nd International Conference on, 2009, pp. 1-4.

[15] Analog Devices Inc. Engineeri. *Data Conversion Handbook (Analog Devices)*. Newnes, 2004, pp. 976, ISBN: 978-0750678414.

[16] M.J.M. Pelgrom. *Analog-to-Digital Conversion, Second Edition*. SpringerLink : Bücher. Springer, 2012, pp. 546, ISBN: 978-1461413707.

[17] I. Ahmed. *Pipelined ADC Design and Enhancement Techniques (Analog Circuits and Signal Processing)*. Springer, 2010, pp. 200, ISBN: 978-9048186518.

[18] P. Carbone, S. Kiaei, and F. Xu. *Design, Modeling and Testing of Data Converters (Signals and Communication Technology)*. Springer, 2014, pp. 430, ISBN: 978-3642396540.

[19] I. Bilinskis. *Digital Alias-free Signal Processing*. UK: Wiley, 2007, pp. 454, ISBN: 978-0470027387.

[20] E. Allier, G. Sicard, L. Fesquet, and M. Renaudin. *A New Class of Asynchronous A/D Converters Based on Time Quantization*. Asynchronous Circuits and Systems, Proceedings. Ninth International Symposium on, 2003, pp. 196-205.

[21] M. Renaudin. *Asynchronous Circuits and Systems: a Promising Design Alternative*. in Journal of Microelectronic Engineering, 2000, pp. 133-149.

[22] A. Yakovlev, D. Kinniment, and B. Gao. *Synchronous and Asynchronous A-D Conversion*. in IEEE Transactions on VLSI Systems, vol.8, 2000, pp. 217-220.

[23] S. Senay, L.F. Chaparro, S. Mingui, R. Sclabassi, and A. Akan. *Asynchronous Signal Processing for Brain-Computer Interfaces*. Electrical and Electronics Engineering, 2009. ELECO 2009. International Conference on, 2009, pp. II-30-II-35.

[24] A.A. Deshmukh, R. Deshmukh, and R.Patrikar. *Low Power Asynchronous Sigma-Delta Modulator Using Hysteresis Level Control.* VLSI (ISVLSI), 2011 IEEE Computer Society Annual Symposium on, 2011, pp. 353-354.

[25] G.D. Colletta, O.O. Dutra, L.H.C. Ferreira, and T.C. Pimenta. *An ultra-low-power first-order asynchronous sigma-delta modulator for biomedical applications.* SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), 2013 IEEE, 2013, pp. 1-2.

[26] K. Ozols, M. Greitans, and R. Shavelis. *EEG Data Acquisition System Based on Asynchronous Sigma-Delta Modulator.* Electronics Conference (BEC), 2012 13th Biennial Baltic, 2012, pp. 183-186.

[27] K. Ozols, R. Shavelis, and M. Greitans. Amplitude adaptive asynchronous sigma-delta modulator. In *2013 8th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pages 467–470. IEEE, 2013.

[28] K. Ozols. Implementation of reception and real-time decoding of asdm encoded and wirelessly transmitted signals. In *Radioelektronika (RADIOELEKTRONIKA), 2015 25th International Conference*, pages 236–239. IEEE, 2015.

[29] K. Ozols and R. Shavelis. Amplitude adaptive asdm without envelope encoding. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 165–169, 2016.

[30] I.A. Mirza, A. Tripathy, S. Chopra, M. D'Sa, K. Rajagopalan, A. D'Souza, and N. Sharma. *Mind-controlled wheelchair using an EEG headset and arduino microcontroller.* Technologies for Sustainable Development (ICTSD), 2015 International Conference on, 2015, pp. 1-5.

[31] M. Bohan. *Action Potential.* President and Fellows of Harvard College and MCB-HHMI Outreach, 2005.

[32] *Anatomy and Physiology.* US National Cancer Institute's Surveillance, Epidemiology and End Results (SEER) Program.

[33] R. Rhoades and D. R. Bell. *Medical Physiology: Principles for Clinical Medicine.* Lippincott Williams & Wilkins, 2009, pp. 816, ISBN: 978-0781768528.

[34] D. Purves, G. J. Augustine, D. Fitzpatrick, W. C. Hall, A. S. LaMantia, and L. E. White. *Neuroscience, Fifth Edition*. Sinauer Associates, Inc., 2011, pp. 759, ISBN: 978-0878936953.

[35] C. L. Dickter and P. D. Kieffaber. *EEG Methods for the Psychological Sciences*. SAGE Publications Ltd;, 2013, pp. 176, ISBN: 978-1446283004.

[36] K. Brodmann and L. J. Garey. *Brodmann's: Localisation in the Cerebral Cortex*. Springer; 2006 edition, 2005, pp. 298, ISBN: 978-0387269177.

[37] M. Dow. *3D representation of brodmann areas*. Brain Development Lab, University of Oregon, 2009, http://lcni.uoregon.edu/~dow/Space_software/renderings.html.

[38] G. Buzsaki. *Rhythms of the Brain*. Oxford University Press; 1 edition, 2011, pp. 464, ISBN: 978-0199828234.

[39] http://www.avstim.com/AboutBrainwaves.html.

[40] M. H. Libenson. *Practical Approach to Electroencephalography*. Saunders; 1 edition, 2012, pp. 464.

[41] S. Sanei. *Adaptive Processing of Brain Signals*. Wiley, 2013, pp. 472, ISBN: 978-1118622148.

[42] V. Botella-Soler, M. Valderrama, B. Crépon, V. Navarro, and M. Le Van Quyen. *Large-scale cortical dynamics of sleep slow waves*. Public Library of Science, PloS one, 2012, 7.2: e30757.

[43] J. Filipe and A. Fred. *Agents and Artificial Intelligence*. Springer Berlin Heidelberg, 2011, pp. 445, ISBN: 978-3642299650.

[44] M.E. Sabourin, S. D. Cutcomb, H. J. Crawford, and K. Pribram. *EEG correlates of hypnotic susceptibility and hypnotic trance: spectral analysis and coherence*. International Journal of Psychophysiology, 1990, 10(2), pp. 125-142.

[45] BR. Cahn and J. Polich. *Meditation states and traits: EEG, ERP, and neuroimaging studies*. Psychol Bull, 2006, 132(2), pp. 180-211.

[46] D. Siever. *The application of audio-visual entrainment for the treatment of seasonal affective disorder*. Biofeedback, 2004, 32(3), pp. 32-35.

[47] W. H. Miltner, C. Braun, M. Arnold, H. Witte, and E. Taub. *Coherence of gamma-band EEG activity as a basis for associative learning*. Nature, 1999, 397(6718), pp. 434-436.

[48] K. Roebuck. *Brain-Computer Interface: High-impact Emerging Technology - What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. Emereo Publishing, 2012, pp. 214, ISBN: 978-1743044612.

[49] J. G. Webster. *Medical Instrumentation Application and Design*. Wiley; 4 edition, 2009, pp. 720, ISBN: 978-0471676003.

[50] B. Basanta. *Neural Amplifier for Biopotential Acquisition System*. LAP Lambert Academic Publishing, 2014, pp. 96, ISBN: 978-3659541247.

[51] S. Li, X. Zhou, S. Zhang, H. Yao, Q. Li, V. Behravan, A. Natarajan, Z. Hong, and P. Y. Chiang. A 0.45v 687pw low noise amplifier front-end with 1.73 nef for energy-scavenging iot sensors. In *2016 IEEE MTT-S International Wireless Symposium (IWS)*, pages 1–4, March 2016.

[52] S. Chakraborty, A. Pandey, S. K. Saw, and V. Nath. A 6nw cmos operational amplifier for bio-medical and sensor applications. In *Communication Technologies (GCCT), 2015 Global Conference on*, pages 242–245, April 2015.

[53] M. Ghamati and M. Maymandi-Nejad. A low-noise low-power mosfet only electrocardiogram amplifier. In *2013 21st Iranian Conference on Electrical Engineering (ICEE)*, pages 1–5, May 2013.

[54] S. Wang, Y. Wang, L. Chen, J. Wang, X. Liu, L. Ye, R. Huang, and H. Liao. A 192nw inverter-based chopper instrumentation amplifier for micropower ecg applications. In *Solid-State and Integrated Circuit Technology (ICSICT), 2014 12th IEEE International Conference on*, pages 1–3, Oct 2014.

[55] T. K. H. Roy and T. H. Teo. A 0.9v l00nw rail-to-rail sar adc for biomedical applications. In *Integrated Circuits, 2007. ISIC '07. International Symposium on*, pages 481–484, Sept 2007.

[56] B. Kayaalti, Ö. Cerid, and G. Dundar. A 0.18-μm current-mode asynchronous sigma-delta modulator design.

[57] F. Cannillo, E. Prefasi, L. Hernández, E. Pun, F. Yazicioglu, and C. Van Hoof. 1.4 v 13μw 83db dr ct-$\sigma\delta$ modulator with dual-slope quantizer and pwm dac for biopotential signal acquisition. In *ESSCIRC (ESSCIRC), 2011 Proceedings of the*, pages 267–270. IEEE, 2011.

[58] Microsemi ZL70250. Ultra-low-power rf sub-ghz transceiver. http://www.microsemi.com/products/ultra-low-power-wireless/sub-ghz-radio-transceivers/zl70250#overview, 2016. [Online; accessed 03-November-2016].

[59] Texas Instruments CC1350. Simplelink™ ultra-low power dual band wireless microcontroller. http://www.ti.com/product/cc1350, 2016. [Online; accessed 03-November-2016].

[60] Nordic Semiconductor nRF24L01. Ultra low power 2.4ghz rf transceiver ic. http://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01, 2016. [Online; accessed 03-November-2016].

[61] Texas Instruments CC2650MODA. Simplelink™ bluetooth® low energy wireless mcu module. http://www.ti.com/product/CC2650MODA/, 2016. [Online; accessed 03-November-2016].

[62] P. P. Mercier, S. Bandyopadhyay, A. C. Lysaght, K. M. Stankovic, and A. P. Chandrakasan. A 78 pw 1 b/s 2.4 ghz radio transmitter for near-zero-power sensing applications. In *ESSCIRC (ESSCIRC), 2013 Proceedings of the*, pages 133–136, Sept 2013.

[63] A. L. Mansano, Y. Li, S. Bagga, and W. A. Serdijn. An asynchronous event-driven data transmitter for wireless ecg sensor nodes. In *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*, pages 404–407, Oct 2014.

[64] S. K. Dhar, Q. D. Hossain, and P. K. Bhowmik. Low power cmos ir-uwb transmitter for bio-sensing meeting fcc restrictions. In *Informatics, Electronics Vision (ICIEV), 2013 International Conference on*, pages 1–5, May 2013.

[65] M. Y. Ilchenco, V. I. Kalinin, T. N. Narytnik, and V. A. Cherepenin. Wireless uwb ecologically friendly communications at 70 nanowatt radiation power. In *Microwave and Telecommunication Technology (CriMiCo), 2011 21th International Crimean Conference*, pages 355–356, Sept 2011.

[66] E.A. Martinez-Garcia, E. Gallegos, and K.S. Jaichandar. *Telepresence by deploying an avatar robot with brain-robot interfacing*. Industrial Electronics and Applications (ICIEA), 2012 7th IEEE Conference on, 2012, pp. 144-149.

[67] N. Cao and Washington University in St. Louis. *Estimating Extended Brain Sources Using EEG and Diffuse Optical Tomography*. Washington University in St. Louis, 2007, pp. 144, ISBN: 978-0549465089.

[68] W. Klonowski. *Everything you wanted to ask about EEG but were afraid to get the right answer*. Nonlinear Biomedical Physics, 2009, 3:2. doi:10.1186/1753-4631-3-2.

[69] M.B. Hamaneh, N. Chitravas, K. Kaiboriboon, S.D. Lhatoo, and K.A. Loparo. *Automated Removal of EKG Artifact From EEG Data Using Independent Component Analysis and Continuous Wavelet Transformation*. Biomedical Engineering, IEEE Transactions on, 2014, vol.61, no.6, pp.1634,1641.

[70] B. Yang and L. He. *Removal of ocular artifacts from EEG signals using ICA-RLS in BCI*. Electronics, Computer and Applications, 2014 IEEE Workshop on, 2014, pp. 544-547.

[71] C. Zhang, L. Siyao, and A.K. Abdullah. *A New Blind Source Separation Method to Remove Artifact in EEG Signals*. Instrumentation, Measurement, Computer, Communication and Control (IMCCC), 2013 Third International Conference on, 2013, pp. 1430-1433.

[72] A. Torok, I. Sulykos, K. Kecskes-Kovacs, G. Persa, P. Galambos, A. Kobor, I. Czigler, V. Csepe, P. Baranyi, and F. Honbolygo. Comparison between wireless and wired eeg recordings in a virtual reality lab: Case report. In *Cognitive Infocommunications (CogInfoCom), 2014 5th IEEE Conference on*, pages 599–603, Nov 2014.

[73] V. Mihajlovic, B. Grundlehner, R. Vullers, and J. Penders. Wearable, wireless eeg solutions in daily life applications: What are we missing? *Biomedical and Health Informatics, IEEE Journal of*, 19(1):6–21, Jan 2015.

[74] A. Jantsch. *Implementation of Re-configurable Analog Circuitry on Field Programmable Analog Array for Vital Signs Acquisition and Communication System (VACS) Platform*. PhD thesis, Talasila Indira Priyadarshini, Department of Microelectronics and Information Technology Royal Institute of Technology, Stockholm, Sweden, 2006.

[75] W. A. Kester. *Data conversion handbook*. Newnes, 2005.

[76] J. Kaur and S. Kansal. Study of various adcs and compare their performance and parameters. *International Journal of Advanced Engineering Research and Technology (IJAERT)*, 3(3):88–96, March 2015.

[77] R.K. Agarwal and N. Jaipalan. A low-power flash quantizer using inverter threshold reference and dual mode operation. In *Electronics and Communication Systems (ICECS), 2014 International Conference on*, pages 1–5, Feb 2014.

[78] M. Kumar and S. Varshney. A 4.2gs/s 4-bit adc in 45nm cmos technology. In *Microelectronics and Electronics (PrimeAsia), 2013 IEEE Asia Pacific Conference on Postgraduate Research in*, pages 24–28, Dec 2013.

[79] M. Eslami, M. Taherzadeh-Sani, and F. Nabki. A 1-v 690 μw 8-bit 200 ms/s flash-sar adc with pipelined operation of flash and sar adcs in 0.13μm cmos. In *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 289–292. IEEE, 2015.

[80] D. Lee, J. Yoo, K. Choi, and J. Ghaznavi. Fat tree encoder design for ultra-high speed flash a/d converters. In *Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on*, volume 2, pages II–87. IEEE, 2002.

[81] L Hsun-Cheng and J.A. Abraham. A novel low power 11-bit hybrid adc using flash and delay line architectures. In *Design, Automation and Test in Europe Conference and Exhibition (DATE), 2014*, pages 1–4, March 2014.

[82] J. Wu, F. Li, W. Li, C Zhang, and Z. Wang. A 14-bit 200ms/s low-power pipelined flash-sar adc. In *Circuits and Systems (MWSCAS), 2015 IEEE 58th International Midwest Symposium on*, pages 1–4, Aug 2015.

[83] M.M. Ayesh, S. Ibrahim, and M.M. Aboudina. A 15.5-mw 20-gsps 4-bit charge-steering flash adc. In *Circuits and Systems (MWSCAS), 2015 IEEE 58th International Midwest Symposium on*, pages 1–4, Aug 2015.

[84] Y.Y. Hsieh and Z.M. Lin. An 8-bit 1.42gs/s 0.54mw cmos flash adc. In *Information, Communications and Signal Processing (ICICS) 2011 8th International Conference on*, pages 1–4, Dec 2011.

[85] M. Figueiredo. *Reference-Free CMOS Pipeline Analog-to-Digital Converters*. Analog Circuits and Signal Processing, Springer Science+Business Media New York, 2013.

[86] S. Sumathi and P. Surekha. *LabVIEW based Advanced Instrumentation Systems*. Labview Based Advanced Instrumentation Systems. Springer Berlin Heidelberg, 2007.

[87] Texas Instruments. *Choose the right A/D converter for your application*. http://www.ti.com/europe/downloads/Choose

[88] P. T. F. Kwok and H. C. Luong. Power optimization for pipeline analog-to-digital converters. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 46(5):549–553, May 1999.

[89] Y. Lim and M. P. Flynn. A 100 ms/s, 10.5 bit, 2.46 mw comparator-less pipeline adc using self-biased ring amplifiers. *IEEE Journal of Solid-State Circuits*, 50(10):2331–2341, Oct 2015.

[90] C.K. Hsu and T.C. Lee. A single-channel 10-b 400-ms/s 8.7-mw pipeline adc in a 90-nm technology. In *Solid-State Circuits Conference (A-SSCC), 2015 IEEE Asian*, pages 1–4, Nov 2015.

[91] Y. Lim and M. P. Flynn. 26.1 a 1mw 71.5db sndr 50ms/s 13b fully differential ring-amplifier-based sar-assisted pipeline adc. In *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, pages 1–3, Feb 2015.

[92] Y. Chai and J. T. Wu. A cmos 5.37-mw 10-bit 200-ms/s dual-path pipelined adc. *IEEE Journal of Solid-State Circuits*, 47(12):2905–2915, Dec 2012.

[93] N. Dolev, M. Kramer, and B. Murmann. A 12-bit, 200-ms/s, 11.5-mw pipeline adc using a pulsed bucket brigade front-end. In *VLSI Circuits (VLSIC), 2013 Symposium on*, pages C98–C99, June 2013.

[94] H. Adel, M. Sabut, and M. M. Louerat. 1.1-v 200 ms/s 12-bit digitally calibrated pipeline adc in 40 nm cmos. In *Circuits and Systems (ISCAS), 2015 IEEE International Symposium on*, pages 2281–2284, May 2015.

[95] J. W. Nam, Y. D. Jeon, S. J. Yun, T. M. Roh, and J. K. Kwon. A 12-bit 100-ms/s pipelined adc in 45-nm cmos. In *SoC Design Conference (ISOCC), 2011 International*, pages 405–407, Nov 2011.

[96] V. Saxena. Pipelined analog-to-digital converters.

[97] Synopsys. Designware 12-bit, 110 msps adc. https://www.synopsys.com/dw/doc.php/ds/a/dwc_adc_12b_110mhz_14.pdf?fe=y, 2016. [Online; accessed 17-March-2016].

[98] T. Li, L. Liu, Y. Wang, Y. Zhang, and X. Wang. Power optimization in a 16-bit 200 msps pipeline adc. In *Solid-State and Integrated Circuit Technology (ICSICT), 2012 IEEE 11th International Conference on*, pages 1–3, Oct 2012.

[99] M. Brandolini, Y. J. Shin, K. Raviprakash, Tao Wang, Rong Wu, H. M. Geddada, Yen-Jen Ko, Yen Ding, Chun-Sheng Huang, Wei-Ta Shih, Ming-Hung Hsieh, A. W. T. Chou, Tianwei Li, A. Shrivastava, D. Y. C. Chen, B. J. J. Hung, G. Cusmai, Jiangfeng Wu, M. M. Zhang, Yuan Yao, G. Unruh, A. Venes, Hung Sen Huang, and Chun-Ying Chen. A 5 gs/s 150 mw 10 b sha-less pipelined/sar hybrid adc for direct-sampling systems in 28 nm cmos. *IEEE Journal of Solid-State Circuits*, 50(12):2922–2934, Dec 2015.

[100] S. Mathur. *Microprocessor 8085 and Its Interfacing*. PHI Learning Pvt. Ltd., 2010.

[101] W. Gao, D. Gao, C. Hu-Guo, T. Wei, and Y. Hu. Design and characteristics of an integrated multichannel ramp adc using digital dll techniques for small animal pet imaging. *IEEE Transactions on Nuclear Science*, 58(5):2161–2168, Oct 2011.

[102] E. Delagnes, D. Breton, F. Lugiez, and R. Rahmanifard. A low power multi-channel single ramp adc with up to 3.2 ghz virtual clock. *IEEE Transactions on Nuclear Science*, 54(5):1735–1742, Oct 2007.

[103] J. Gu and N. McFarlane. Low power current mode ramp adc for multi-frequency cell impedance measurement. In *Circuits and Systems (MWSCAS), 2012 IEEE 55th International Midwest Symposium on*, pages 1016–1019, Aug 2012.

[104] P. Harpe. Asynchronous digital slope analog-to-digital converter and method thereof, February 5 2013. US Patent 8,368,578.

[105] W. Gao, D. Gao, C. Hu-Guo, and Y. Hu. Design of a 12-bit 2.5 ms/s integrated multichannel single-ramp analog-to-digital converter for imaging detector systems. *IEEE Transactions on Instrumentation and Measurement*, 60(6):1942–1951, June 2011.

[106] M. O. Shaker and M. A. Bayoumi. A 6-bit 130-ms/s low-power tracking adc in 90 nm cmos. In *Circuits and Systems (MWSCAS), 2010 53rd IEEE International Midwest Symposium on*, pages 304–307, Aug 2010.

[107] Y. Huang, H. Schleifer, and D. Killat. A current mode 6-bit self-clocked tracking adc with adaptive clock frequency for dc-dc converters. In *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, pages 145–148, May 2013.

[108] Analog Devices. Low cost, complete 12-bit resolver-to-digital converter. http://www.analog.com/media/en/technical-documentation/data-sheets/AD2S90.pdf, 2016. [Online; accessed 17-March-2016].

[109] S. Cha, C. Jeong, C. Yoo, and J. Kih. Digitally controlled phase locked loop with tracking analog-to-digital converter. In *Asian Solid-State Circuits Conference, 2005*, pages 377–380, Nov 2005.

[110] P. Harikumar, J. J. Wikner, and A. Alvandpour. A 0.4 v, sub-nw, 8-bit 1 ks/s sar adc in 65 nm cmos for wireless sensor applications. *IEEE Transactions on Circuits and Systems II: Express Briefs*, PP(99):1–1, 2016.

[111] D. Zhang and A. Alvandpour. A 3-nw 9.1-enob sar adc at 0.7 v and 1 ks/s. In *ESSCIRC (ESSCIRC), 2012 Proceedings of the*, pages 369–372, Sept 2012.

[112] A. Bannon, C. P. Hurrell, D. Hummerston, and C. Lyden. An 18 b 5 ms/s sar adc with 100.2 db dynamic range. In *VLSI Circuits Digest of Technical Papers, 2014 Symposium on*, pages 1–2, June 2014.

[113] John Errington's Data Conversion Website. Analog to Digital converters (ADC's). http://www.skillbank.co.uk/SignalConversion/adc.htm/, 2016. [Online; accessed 17-March-2016].

[114] Linear Technology. Ltc2380-24 datasheet, 24-bit, 1.5msps/2msps, low power sar adc with integrated digital filter. http://cds.linear.com/docs/en/datasheet/238024f.pdf, 2016. [Online; accessed 17-March-2016].

[115] Analog Devices. Ad7766 datasheet, 24-bit, 8.5 mw, 109 db, 128 ksps/64 ksps/32 ksps adcs. http://www.analog.com/media/en/technical-documentation/data-sheets/AD7766.pdf, 2016. [Online; accessed 17-March-2016].

[116] Linear Technology. Ltc2379-18 datasheet, 18-bit, 1.6msps, low power sar adc with 101.2db snr. http://www.linear.com/product/LTC2379-18, 2016. [Online; accessed 17-March-2016].

[117] Maxim Integrated. Max11156 datasheet, 18-bit, 500ksps, ±5v sar adc with internal reference in tdfn. https://datasheets.maximintegrated.com/en/ds/MAX11156.pdf, 2016. [Online; accessed 17-March-2016].

[118] L. Kull, J. Pliva, T. Toifl, M. Schmatz, P. A. Francese, C. Menolfi, M. Braendli, M. Kossel, T. Morf, T. M. Andersen, and Y. Leblebici. A 110 mw 6 bit 36 gs/s interleaved sar adc for 100 gbe occupying 0.048 mm2 in 32 nm soi cmos. In *Solid-State Circuits Conference (A-SSCC), 2014 IEEE Asian*, pages 89–92, Nov 2014.

[119] N. Suda, PV Nishanth, D. Basak, D. Sharma, and R.P. Paily. A 0.5-v low power analog front-end for heart-rate detector. *Analog Integrated Circuits and Signal Processing*, 81(2):417–430, 2014.

[120] A. Fazli Yeknami and A. Alvandpour. A 0.7-v 600-nw 87-db sndr dt- $\sigma\delta$; modulator with partly body-driven and switched op-amps for biopotential signal acquisition. In *Biomedical Circuits and Systems Conference (BioCAS), 2012 IEEE*, pages 336–339, Nov 2012.

[121] Analog Devices. 12-bit low power sigma-delta adc. http://www.analog.com/media/en/technical-documentation/data-sheets/AD7170.pdf, 2016. [Online; accessed 18-March-2016].

[122] Analog Devices. 32-bit, 10 ksps, sigma-delta adc with 100 µs settling and true rail-to-rail buffers. http://www.analog.com/media/en/technical-documentation/data-sheets/AD7177-2.pdf, 2016. [Online; accessed 18-March-2016].

[123] Texas Instruments. Ads126x 32-bit, precision, 38-ksps, analog-to-digital converter (adc) with programmable gain amplifier (pga) and voltage reference. http://www.ti.com/lit/ds/symlink/ads1262.pdf, 2016. [Online; accessed 18-March-2016].

[124] Analog Devices. 10 mhz bandwidth, 640 msps dual continuous time sigma-delta modulator. http://www.analog.com/media/en/technical-documentation/data-sheets/AD9267.pdf, 2016. [Online; accessed 18-March-2016].

[125] Analog Devices. 3-channel, low noise, low power, 16-/24-bit, Σ-Δ adc with on-chip in-amp. http://www.analog.com/media/en/technical-documentation/data-sheets/AD7798_7799.pdf, 2016. [Online; accessed 18-March-2016].

[126] Texas Instruments. Ads1293 low-power, 3-channel, 24-bit analog front-end for biopotential measurements. http://www.ti.com/lit/ds/symlink/ads1293.pdf, 2016. [Online; accessed 18-March-2016].

[127] R. Elliott. Zero crossing detectors and comparators. https://www.scribd.com/doc/97388393/Zero-Crossing-Detectors-and-Comparators, 2016.

[128] Logan B.F. *Information in the zero crossings of band-pass signals*. The Bell Systems Technical Journal, 1977, vol. 56. pp. 487–510.

[129] F. Marvasti. *Nonuniform Sampling: Theory and Practice*. Information Technology: Transmission, Processing and Storage. Springer US, 2012.

[130] M.A. Nazario and C. Saloma. Signal recovery in sinusoid-crossing sampling by use of the minimum-negativity constraint. *Applied optics*, 37(14):2953–2963, 1998.

[131] J Surber and L McHugh. Single-chip direct digital synthesis vs. the analog pll. *Analog Dialogue*, 30(3):12–13, 1996.

[132] E. Murphy and C. Slattery. Ask the application engineer—33 all about direct digital synthesis. *Analog Devices*, 38:1–5, 2004.

[133] Texas Instruments. Tlv3201, 40-ns, micropower, push-pull output comparators. http://www.ti.com/lit/ds/symlink/tlv3201.pdf, 2016. [Online; accessed 10-April-2016].

[134] Y. Li, D. Zhao, and W.A Serdijn. A sub-microwatt asynchronous level-crossing adc for biomedical applications. *IEEE transactions on biomedical circuits and systems*, 7(2):149–157, 2013.

[135] M. Greitans and R. Shavelis. Extended fourier series for time-varying filtering and reconstruction from level-crossing samples. In *21st European Signal Processing Conference (EUSIPCO 2013)*, pages 1–5. IEEE, 2013.

[136] K. Kozmin, J. Johansson, and J. Delsing. Level-crossing adc performance evaluation toward ultrasound application. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 56(8):1708–1719, 2009.

[137] C. Weltin-Wu and Y. Tsividis. An event-driven clockless level-crossing adc with signal-dependent adaptive resolution. *IEEE Journal of Solid-State Circuits*, 48(9):2180–2190, 2013.

[138] E. Allier, G. Sicard, L. Fesquet, and M. Renaudin. A new class of asynchronous a/d converters based on time quantization. In *Asynchronous Circuits and Systems, 2003. Proceedings. Ninth International Symposium on*, pages 196–205. IEEE, 2003.

[139] K. Kozmin, J. Johansson, and J. Delsing. Level-crossing adc performance evaluation toward ultrasound application. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 56(8):1708–1719, 2009.

[140] M. Miskowicz. Send-on-delta concept: an event-based data reporting strategy. *sensors*, 6(1):49–63, 2006.

[141] Y.S. Suh. Send-on-delta sensor data transmission with a linear predictor. *Sensors*, 7(4):537–547, 2007.

[142] K. Staszek, S. Koryciak, and M. Miskowicz. Performance of send-on-delta sampling schemes with prediction. In *2011 IEEE International Symposium on Industrial Electronics*, pages 2037–2042. IEEE, 2011.

[143] A. A. Lazar, E. K. Simonyi, and L. T. Tóth. *Time Encoding of Bandlimited Signals, an Overview*. Electronics Conference (BEC), 2012 13th Biennial Baltic, 2005.

[144] W. Hussain, H. Fakhoury, P. Desgreys, Y. Blaquière, and Y. Savaria. An asynchronous delta-modulator based a/d converter for an electronic system prototyping platform. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(6):751–762, June 2016.

[145] A.a. Deshmukh and R.B. Deshmukh. Design of resolution/power controllable asynchronous sigma-delta modulator. *EURASIP Journal on Advances in Signal Processing*, 2016(1):105, 2016.

[146] S Ouzounov, E Roza, H Hegt, G Van Der Weide, and A Van Roermund. An 8mhz, 72 db sfdr asynchronous sigma-delta modulator with 1.5 mw power dissipation. In *VLSI Circuits, 2004. Digest of Technical Papers. 2004 Symposium on*, pages 88–91. IEEE, 2004.

[147] S. Ouzounov, E. Roza, J.A. Hegt, G. van der Weide, and A. HM van Roermund. Analysis and design of high-performance asynchronous sigma-delta modulators with a binary quantizer. *IEEE Journal of Solid-State Circuits*, 41(3):588–596, 2006.

[148] L.H.C. Ferreira and S.R. Sonkusale. *A 0.25-V 28-nW 58-dB Dynamic Range Asynchronous Delta Sigma Modulator in 130-nm Digital CMOS Process*. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, 2014, pp. 1.

[149] G. D. Colletta. Uma arquitetura de modulação sigma-delta assíncrona em ultra-baixa potência para aplicações biomédicas. 2015.

[150] R. Shavelis. Sampling and waveform reconstruction of signals on the basis of minimax approach. In *Proc. DASP Workshop*, pages 57–61, 2007.

[151] I. Homjakovs, M. Hashimoto, T. Onoye, and T. Hirose. Signal-dependent analog-to-digital conversion based on minimax sampling. In *2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1–4. IEEE, 2011.

[152] I. Homjakovs, M. Hashimoto, T. Onoye, and T. Hirose. Signal-dependent analog-to-digital converter based on minimax sampling. In *SoC Design Conference (ISOCC), 2012 International*, pages 120–123. IEEE, 2012.

[153] C.H. Van Berkel, M.B. Josephs, and S.M. Nowick. Applications of asynchronous circuits. *Proceedings of the IEEE*, 87(2):223–233, 1999.

[154] Nazario M.A. *Signal Recovery in Sinusoid-Crossing Sampling by use of the Minimum-Negativity Constraint*. Applied Optics, 1998, vol. 37(14), pp. 2953–2963.

[155] R. Shavelis. *Signālu diskretizācijas un atjaunošanas paņēmienu izpēte*. Promocijas darbs. Rīga: [RTU], 2013, pp. 132.

[156] A.A. Lazar and L.T. Toth. *Time Encoding and Perfect Recovery of Bandlimited Signals*. Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on, 2003, vol 6, pp. 709-712.

[157] S. Senay, L. F. Chaparro, M. Sun, and R. J. Sclabassi. *Adaptive Level-Crossing Sampling and Reconstruction*. 18th European Signal Processing Conference (EUSIPCO-2010), 2010, pp. 1296-1300.

[158] A.A. Lazar and L.T. Toth. *Perfect Recovery and Sensitivity Analysis of Time Encoded Bandlimited Signals*. IEEE transactions on circuit and systems -I: Regular papers, 2004, vol 51, No. 10 pp. 2060-2073.

[159] E. Roza. Analog-to-digital conversion via duty-cycle modulation. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 44(11):907–914, Nov 1997.

[160] D. Wei and J. G. Harris. *Signal reconstruction from spiking neuron models*. ISCAS (5), 2004, pp. 353-356.

[161] C.E. Shannon. *Communication in the presence of noise*. Proc. of the IRE., 1949. vol. 37(1), pp. 10–21.

[162] A.A. Lazar, E.K. Simonyi, and L.T. Toth. *Fast recovery algorithms for time encoded bandlimited signals*. Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on, 2005. iv/237-iv/240 Vol. 4.

[163] A. A. Lazar, E. K. Simonyi, and L. T. Toth. *A real-time algorithm for time decoding machines*. Signal Processing Conference (EUSIPCO'2006), 2006 14th European, 2006, pp.1-5.

[164] D. Koscielnik and M. Miskowicz. *Asynchronous Sigma-Delta analog-to digital converter based on the charge pump integrator*. Analog Integrated Circuits and Signal Processing, 2008, 55.3: pp. 223-238.

[165] M. Renaudin. *Asynchronous circuits and systems: a promising design alternative*. Microelectronic engineering, 2000, pp. 133-149.

[166] D. Kinniment, A. Yakovlev, and B. Gao. *Synchronous and asynchronous AD conversion*. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, 2000, 8.2: pp. 217-220.

[167] S. Senay, L.F. Chaparro, Mingui Sun, and R. Sclabassi. *Time-frequency multiplexing for time-encoded signals from brain-computer interfaces*. Signal Processing Conference, 2009 17th European, 2009, pp. 1181-1185.

[168] D. Wei, V. Garg, and J. G. Harris. *An asynchronous delta-sigma converter implementation*. Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on, 2006, pp. 4.

[169] H.Y. Yang and R. Sarpesbkar. *A bio-inspired ultra-energy efficient ADC for biomedical applications*. IEEE Trans. on circuits and systems, 2004, vol. 53, pp. 1533-1538.

[170] S. Ouzounov, Engel Roza, J.A. Hegt, G. van der Weide, and A.H.M. van Roermund. *Analysis and design of high-performance asynchronous sigma-delta Modulators with a binary quantizer*. Solid-State Circuits, IEEE Journal of, 2006, 41.3: pp. 588-596.

[171] B. Schell and Y. Tsividis. *A Continuous-Time ADC/DSP/DAC System With No Clock and With Activity-Dependent Power Dissipation*. Solid-State Circuits, IEEE Journal of, 2008, 43.11: pp. 2472-2481.

[172] S. Yan Ng. *A continuous-time asynchronous Sigma Delta analog to digital converter for broadband wireless receiver with adaptive digital calibration technique*. [Ph.D. thesis], Department of Electrical and Computer Engineering, Ohio State University, 2009, pp. 137.

[173] J. Zhang, C. Wang, S. Diao, and F. Lin. *A Low-Power VCO based ADC with asynchronous sigma-delta modulator in 65nm CMOS*. Design Automation Conference (ASP-DAC), 2015 20th Asia and South Pacific, 2015, pp. 38-39.

[174] N. Tavangaran, D. Bruckmann, R. Kokozinski, and K. Konrad. *Continuous Time digital systems with Asynchronous Sigma Delta Modulation*. Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European, 2012, pp. 225-229.

[175] D. Koscielnik, M. Miskowicz, and M. Jableka. *Natural compression and expansion characteristics of asynchronous Sigma-Delta ADC*. Signal Processing and Communication Systems (ICSPCS), 2011 5th International Conference on, 2011, pp. 1-8.

[176] E. Alarcon, D. Fernandez, A. Garcia i Tormo, J. Madrenas, and A. Poveda. *Continuous-time CMOS adaptive asynchronous $\Sigma\Delta$; modulator approximating low-fs low-inband-error on-chip wideband power amplifier*. Circuits and Systems (ISCAS), 2011 IEEE International Symposium on, 2011, pp. 301-304.

[177] N.H.E. Weste and D.M. Harris. *CMOS VLSI design: a circuits and systems perspective*. Pearson Education India, 2005.

[178] Cmostek. Cmt2119aw datasheet, 240 - 960 mhz (g)fsk/ook transmitter. http://www.hoperf.com/upload/rfchip/CMT2119AW.pdf, 2017. [Online; accessed 31-January-2017].

[179] Analog Devices. Adf7901 datasheet, high performance ism band ook/fsk transmitter ic. http://www.analog.com/media/en/technical-documentation/data-sheets/ADF7901.pdf, 2017. [Online; accessed 31-January-2017].

[180] Y. Jun, Z. Zhaofeng, W. Jun, W. Chao, C. Zhenhai, Q. Wenrong, and Y. Yintang. Continuous time sigma delta adc design and non-idealities analysis. *Journal of Semiconductors*, 32(12):125007, 2011.

[181] P. Aguirre, V. Camargo, H. Klimach, A. Susin, and C. Prior. Behavioral modeling of continuous-time $\delta\sigma$ modulators in matlab/simulink. In *2013 IEEE 4th Latin American Symposium on Circuits and Systems (LASCAS)*, pages 1–4, Feb 2013.

[182] A. A. Lazar and L. T. Toth. Perfect recovery and sensitivity analysis of time encoded bandlimited signals. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 51(10):2060–2073, Oct 2004.

[183] W. Chen and C. Papavassiliou. Asynchronous sigma-delta modulator with noise shaping. *Electronics Letters*, 49(24):1520–1522, November 2013.