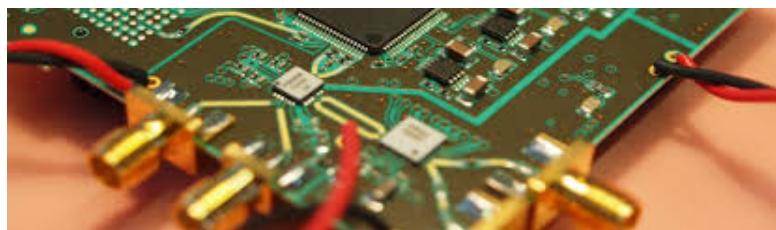


A Terabit Sampling System with a Photonics Time-Stretch ADC

Master Thesis
of

Olena Manzhura

at the Institute for Data Processing and Electronics (IPE)



Reviewer: Prof. Dr. Anke-Susanne Müller (LAS)
Second Reviewer: Dr. Michele Caselle (IPE)

15.11.2020 – 13.08.2021

Declaration

I hereby declare that I wrote my master thesis on my own and that I have followed the regulations relating to good scientific practice of the Karlsruhe Institute of Technology (KIT) in its latest form. I did not use any unacknowledged sources or means, and I marked all references I used literally or by content.

Karlsruhe, 13.08.2021, _____
Olena Manzhura

Approved as an exam copy by

Karlsruhe, 13.08.2021, _____
Prof. Dr. Anke-Susanne Müller (LAS)

Abstract

Zusammenfassung

Résumé

Contents

1. Introduction	1
1.1. Time-Stretching Techniques in Physics	1
1.2. Motivation	1
1.2.1. THz Science and Coherent Synchrotron Radiation	2
1.2.1.1. Micro-Bunching Instabilities	3
1.2.2. Longitudinal Bunch Profile Diagnostics	4
1.2.3. Potential Usage: ANR-DFG ULTRASYNC project?	4
1.3. Objective	5
1.4. General Architecture of a Photonic Time-Stretch Data Acquisition System	5
1.4.1. Photonic Time-Stretch Front-End	5
1.4.2. Analog-To-Digital Converter	6
2. State Of The Art Read-Out-Systems	21
2.1. Real-Time Oscilloscope	21
2.2. Karlsruhe Pulse Taking Ultra-fast Readout Electronics (KAPTURE)	21
2.2.1. General Architecture	21
2.2.2. Analog front-end	22
2.2.3. Sampling board	22
2.2.4. FPGA-GPU architecture	25
2.2.5. KAPTURE-2	25
3. Architecture Of The New Read-Out-System	27
3.1. Optical part	27
3.2. Front-End Card	27
3.3. Readout Card	28
3.4. Selection Of The Card	28
4. Design Of The Front-End Sampling Card	31
4.1. Schematics	31
4.1.1. Connectors	34
4.1.2. Sampling-Channel	36
4.1.3. Clocking	41
4.1.4. Digital-To-Analog-Converter Balun Channel	44
4.1.5. Power Supply	44
4.2. Layout	45
4.2.1. Printed Circuit Board (PCB) Structures Overview	45
4.2.2. PCB Substrate	46
4.2.3. Floor Planning	46
4.2.4. Transmission lines	46
4.3. Production	49

5. Selecting and Analyzing the Back-End Readout Card	51
5.1. Xilinx Zynq UltraScale+ RFSoC ZCU216 Evaluation Card	51
5.2. Features	53
5.3. Evaluation Tool	53
5.4. Firmware	53
5.4.1. RF Data Converter	53
5.4.2. SoC	53
5.4.3. RDMA over Converged Ethernet (RoCE)	53
5.4.4. System Integration	53
6. Results	55
7. Conclusion and Outlook	57
7.1. Measurements with the developed front-end card	57
Acknowledgements	59
Appendix	61
A. Characteristic Impedance Of Coplanar Waveguides	61
B. QuickStart Guide for Evaluation of ZCU216 Board	62
C. 3D model of front-end card	62
D. Code	62

List of Figures

1.1.	Use case of the system	2
1.2.	Basic scheme of an electron storage ring (redrawn from [?])	2
1.3.	Electromagnetic spectrum	3
1.4.	Placeholder [?]	4
1.5.	Electrons interact with their own radiation [?]	4
1.6.	Electro-Optical Time-Stretch Technique [?]	6
1.7.	Transfer function of ideal, 3-bit ADC	7
1.9.	Example for appropriate sampling timing when using Sample-And-Hold-Amplifier	7
1.8.	Aliasing	9
1.10.	Track-And-Hold-Amplifier schematic and principle [?]	10
1.11.	Quantization noise as function of time (redrawn from [?])	11
1.12.	Effects of Offset and Fain error in ADC	13
1.13.	Placeholder: Integral and Differential Nonlinearity Distortion [?]	14
1.14.	Placeholder: SFDR [?]	15
1.15.	Placeholder: Effects of aperture jitter and SJNR [?]	17
1.16.	Time Interleaving	19
1.17.	Placeholder: Offset-Mismatch in Interleaving [?]	19
1.18.	Placeholder: Gain-Mismatch in Interleaving [?]	20
1.19.	Placeholder: Timing-Mismatch in Interleaving [?]	20
1.20.	Placeholder: Timing-Mismatch in Interleaving [?]	20
2.1.	THz measurement with KAPTURE	22
2.2.	General architecture of the KAPTURE front-end sampling card (cf. [?, p.2])	23
2.3.	Signal and sampled points S_1 to S_4	23
2.4.	Photo of KAPTURE with highlighted main components. [?, p. 61]	24
2.5.	Comparison between KAPTURE and KAPTURE-2[?]	25
3.1.	General concept of the new readout system	27
3.2.	Placeholder: General schema of THERESA. For presentation purposes only four channels are shown.	28
3.3.	Placeholder: Discrete vs IC	29
4.1.	Capacitor equivalent circuit	33
4.2.	Impedance response of a real capacitor	33
4.3.	Rendering of FMC+ connector	34
4.4.	Edge-Mount RF SMA connector	35
4.5.	LPAM 8x50 connector	35
4.6.	LPAM 6x20 connector	35
4.7.	HMC5640 THA Schematic	37
4.8.	NB6L295 Delay Chip Schematic	38
4.9.	LVPECL driver topology	39

4.10. NB6L295 Delay Chip Schematic	40
4.11. Placeholder	41
4.12. Schematics of the LMX2594	42
4.13. Track-And-Hold Timing diagram	43
4.14. DAC-channel with balun. Signal travels from right to left.	44
4.15. Via types	45
4.16. Polaris Solver	46
4.17. Coplanar Waveguide with Ground	47
4.18. Characteristic impedance Z_o vs effective dielectric constant ϵ_r of a surface coplanar waveguide with ground	48
4.19. Edge-Coupled Coplanar Waveguide	48
4.20. Differential Offset Coplanar Waveguide	49
 5.1. ZCU216 evaluation board	52
5.2. RFSoC block diagram	52
 7.1. wow	57

List of Tables

1.1.	Some KARA parameters [?]	3
4.1.	HMC5640 Characteristics	36
4.2.	NB6L295 Characteristics	39
4.3.	NB6L295 Characteristics	42
4.4.	Power consumption of components on the board	44

1. Introduction

Observing non-repetitive and statistically rare signals that occur on short timescales requires fast real-time measurements that exceed the speed, precision and record length of conventional digitizers. Photonic time-stretch is a data acquisition method that overcomes the speed limitations of electronic digitizers and enables continuous ultrafast single-shot spectroscopy, imaging, reflectometry, terahertz and other measurements at refresh rates reaching billions of frames per second with non-stop recording spanning trillions of consecutive frames. The technology has opened a new frontier in measurement science unveiling transient phenomena in non-linear dynamics such as optical rogue waves and soliton molecules, and in relativistic electron bunching. It has also created a new class of instruments that have been integrated with artificial intelligence for sensing and biomedical diagnostics. We review the fundamental principles and applications of this emerging field for continuous phase and amplitude characterization at extremely high repetition rates via time-stretch spectral interferometry.

A new generation of instruments based on photonic time-stretch is opening the path to measuring and understanding the behavior of non-stationary and rare phenomena in ultrafast systems, and harvesting their potential for practical applications such as metrology, machine vision and biomedicine. While these instruments rely on optics for the transformation of a signal's timescale, they are able to operate with either electrical, optical, or terahertz inputs.

1.1. Time-Stretching Techniques in Physics

1.2. Motivation

When doing science with terahertz radiation and in some laser applications, ultrafast events need to be characterized. Commercially available real-time equipment is limited in bandwidth (240 GS/s, LeCroy LabMaster 10-100Zi) and cannot be used to characterize these events in the pico- and femto-second regime. In this thesis you will evaluate a real-time measurement technique based on the photonic time-stretch analog-to-digital converter (TS-ADC) for the measurement of ultrafast signals with a sampling-rate which exceeds TS/s. The principle is to use a photonic front-end that effectively “slows down” the analog signal in time before it is digitized by fast Analog-To-Digital-Converters (ADCs).

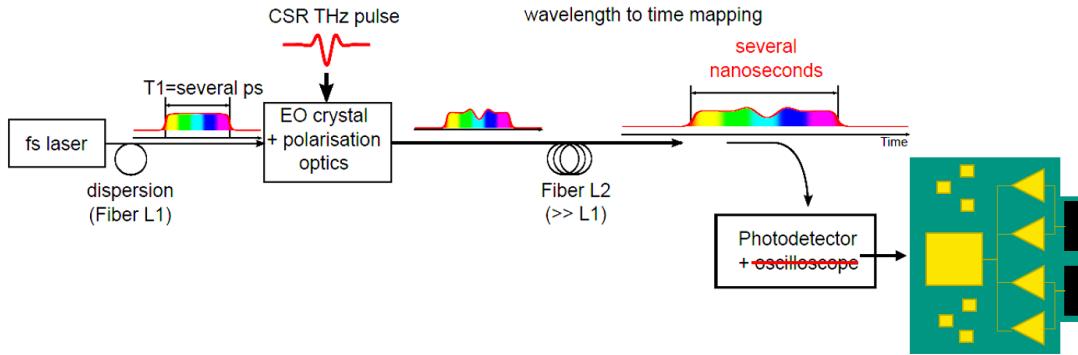


Figure 1.1.: Use case of the system

1.2.1. THz Science and Coherent Synchrotron Radiation

Synchrotron Radiation (SR) is produced in synchrotron radiation facilities (like electron storage rings) by accelerating relativistic electrons. Emission of SR occurs, when electron beams are bent or deflected with dipole magnets or using an undulator¹.

Figure 1.2 shows the general scheme of an electron storage ring. Electrons, or rather electron bunches, are generated with an electron gun and are accelerated to almost the speed of light by a linear accelerator (LINAC). After being broad up to their nominal energy in a booster, the bunches are then injected into the storage ring. In the ring, the path of the electron bunches is altered by bending magnets, guiding them on a circular trajectory. Due to emission of SR at each bend, the electrons lose energy, which has to be compensated for. This is done by accelerating them with an electric field inside a Radio Frequency (RF) cavity. Not shown in the drawing are the beamlines, which lead the SR radiation, or rather chosen wavelength ranges, through an optical system to the respective user experiments. [?] [?]

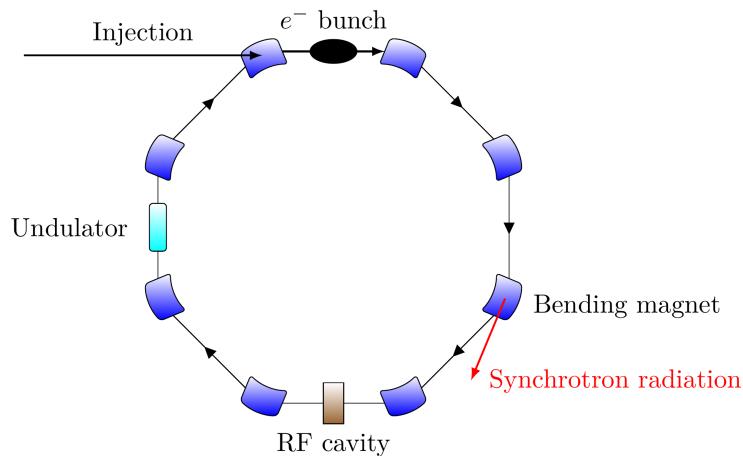


Figure 1.2.: Basic scheme of an electron storage ring (redrawn from [?])

Karlsruhe Research Accelerator

- Located at the Karlsruhe Institute of Technology (KIT)

¹Undulators are used to make the electrons oscillate by generating a periodic magnetic field

- Up to 184 electron bunches can be filled into the storage ring with a distance of 2 ns (500 MHz) between two adjacent bunches
- Operated by the Institute of Beam Physics and Technology (IBPT)
- Microtron, Booster Synchrotron, and Storage Ring

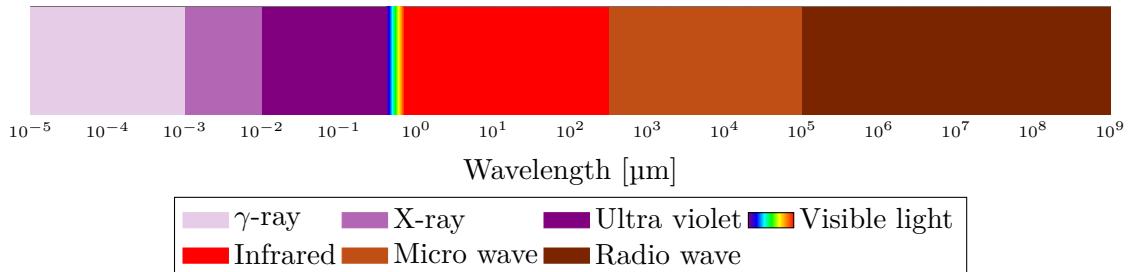
Table 1.1.: Some KARA parameters [?]

Parameter	Value
Beam energy	2.5 GeV
Circumference	110 m
Revolution Frequency (one electron)	2.7 MHz
Minimum bunch spacing	
multi-bunch	2 ns
single-bunch	368 ns
Bunch length	
normal operation	45 ps
short bunch	2 ps

The range of SR reaches from hard X-rays down to the infrared region of the electromagnetic spectrum (see Figure 1.3). In contrast to other sources, it has properties like:

- high intensity
- high collimation
- polarisation
- well-defined timing of pulses

Due to this properties, synchrotrons are used for microscopy, spectroscopy, and time-resolved experiments in such fields like condensed matter physics, biology, material science and many more.

**Figure 1.3.:** Electromagnetic spectrum

1.2.1.1. Micro-Bunching Instabilities

Increasing demands in current and future accelerators call for higher brilliance of the emitted radiation. This is achieved by increased photon flux and reduction of the transverse emittance. For longitudinal coherence, the electron bunches are shortened, which results in emission of Coherent Synchrotron Radiation (CSR) at frequencies up to the THz range.

However, this introduces complex dynamics, as the electrons interact with their own radiation. This manifests into the so called micro-bunching instability. The formation of micro-structures (in the sub-millimeter to centimeter range) in the longitudinal density profile of the electron bunches. Being on the one side a limitation to the stable operation of the overall system at high current density/short bunch length mode. On the other side, these instabilities can be potential sources of brilliant THz radiation. A thorough

understanding of these dynamics is necessary to control the emission in this spectral domain which enables usage in experiments. [?, ?]

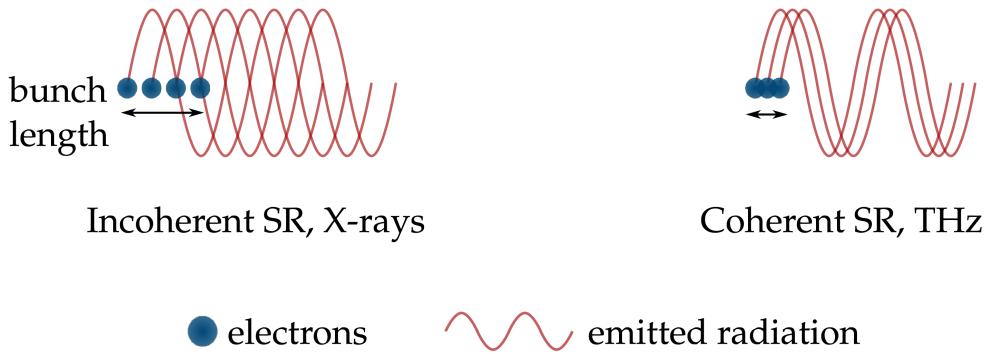


Figure 1.4.: Placeholder [?]

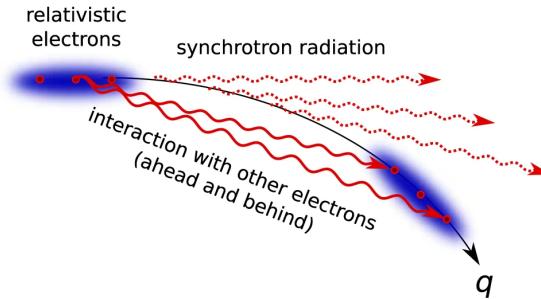


Figure 1.5.: Electrons interact with their own radiation [?]

1.2.2. Longitudinal Bunch Profile Diagnostics

1.2.3. Potential Usage: ANR-DFG ULTRASYNC project?

From Serge's email:

For accelerator applications, we have also a project (ANR-DFG ULTRASYNC project) with KARA and SOLEIL. There is the question of control (i.e., suppression) of the bursts occurring during the microbunching instability. The objective is to obtain a high power and stable coherent emission (at SOLEIL and KARA). For the moment, the only experimental test has been made using a relatively simple feedback:

- a bolometer signal as the feedback loop input
- a low-cost FPGA (redpitaya) that sends the feedback on the accelerating voltage

There are limitations in the maximal bunch charge in the accelerator. So an open question is whether measuring each THz pulse using EO sampling + time-stretch may help to solve this open problem. In clear:

- EO sampling + time-stretch as in Eléonore's thesis
- association with the new FPGA-based system
- finding adequate feedback that is programmed in the FPGA

may solve the problem, and allow the control to succeed at the highest currents in SOLEIL. Target would be ca. 15 mA for 1 bunch (and feedback control presently works to a little more than ca. 10 mA).

1.3. Objective

Design of a system to be used with photonics time-stretch technique. Fast, continuous sampling of signals, integratable in already existing high data-throughput architectures.

1.4. General Architecture of a Photonic Time-Stretch Data Acquisition System

In this section, a general architecture of a Photonic Time-Stretch Data Acquisition System (DAQ) is described. Such a system consists of a photonic front-end, which covers the time-stretching method and conversion of photons into electrical values. In the following, the general theory of the time-stretching technique is given together with some photo-detector fundamentals.

Furthermore, such a system contains an ADC which converts the analog values into digital signals that can be processed by a computing unit. A short overview of the basic ADC-theory is given, as well as of the most prominent figures of merit. Knowledge and understanding of these is necessary to define and/or evaluate the overall performance of the converter.

Last but not least a DAQ needs an appropriate computing system for collection, processing and visualization of the acquired data. As this part is use-case specific, there doesn't exist a general "theory" which can be described.

1.4.1. Photonic Time-Stretch Front-End

The working principle of the optic time-stretch technique can be described in two steps (see Figure 1.6).

First, a short laser pulse (duration typically hundreds of femtoseconds) propagates in a dispersive medium, e.g. an optical fiber of length L_1 (see Figure 1.6). This results in a chirped laser pulse of the duration

$$T_1 = \Delta\lambda D_1 L_1 \quad (1.1)$$

with the optical bandwidth of the laser pulse $\Delta\lambda$ and the dispersion parameter D_1 of the fiber.

The next step is the time-to-wavelength-mapping, where a temporal intensity modulation is imprinted on the chirped pulse. In this step, the laser pulse co-propagates with another pulse, e.g. a tera Hertz (THz) pulse from CSR (duration in the range of picoseconds), in an Electro-Optic (EO) crystal. Due to the Pockels effect² the THz pulse causes a time-dependent birefringence in the crystal. This modulates the polarization of the laser pulse.

After that, the modulated chirped pulse propagates through another dispersive medium, a fiber of the length L_2 . In this way, the temporal modulation of the pulse is further stretched to the duration T_2 , which is long enough for detection with photodetectors and the digitizing with ADCs. [?]

The factor, by which the pulse is slowed down, is calculated as

$$M = 1 + \frac{L_1}{L_2}. \quad (1.2)$$

²The Pockels effect describes the phenomenon of occurring and change of existing birefringence in an electric field, which is linearly proportional to the electric field strength." [?]

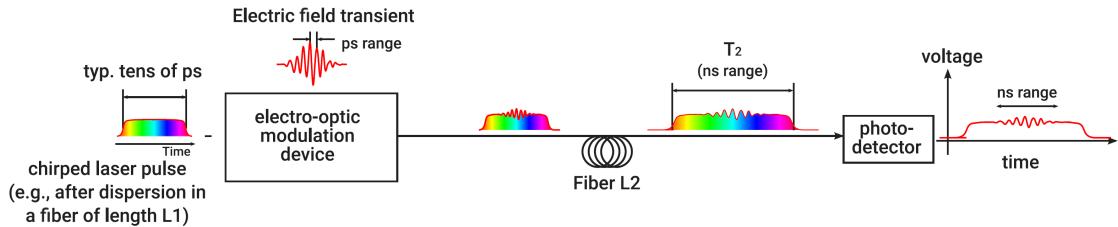


Figure 1.6.: Electro-Optical Time-Stretch Technique [?]

Detector/Diode

A photodetector converts the incoming optical signal into an electrical signal, so it is called an O/E converter. The detection and subtraction between the two stretched signals is performed using a amplified balanced photodetector (photoreceiver) from Discovery Semiconductors, with a 20 GHz bandwidth and 2800 V/W gain (specified at 1500 nm). The two differential outputs of the detector are sent to a Lecroy LabMaster 10i oscilloscope with 36 GHz bandwidth, 80GS/s sample rate on each channel and a memory of 256Mega samples.

Balanced Detection

TODO

Sensitivity Increase

TODO

1.4.2. Analog-To-Digital Converter

ADCs are used to translate analog voltages into digital signals, which can be processed by information processing, computing, data transmission and control systems. The translation can be seen as encoding a continuous-time analog input (voltage) into a series of discrete, N -bit words. This process, which is also called *sampling*, can be expressed as

$$V_{\text{In}} = V_{\text{FS}} \sum_{k=0}^{N-1} \frac{b_k}{2^{k+1}} + \epsilon \quad (1.3)$$

with V_{In} being the input voltage, V_{FS} the full-scale voltage, b_k the individual output bits and ϵ the quantization error (described in section 1.4.2). Figure 1.7 shows the ideal transfer function of a 3-bit ADC. As one can see, each digital N -bit word corresponds to a range of input voltage values (*code width*), which is centered around a *code center*. The input voltage is resolved to the code of the nearest code center.

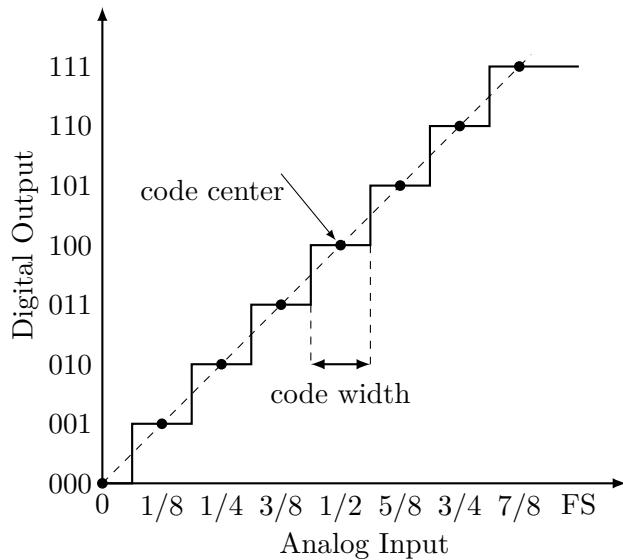


Figure 1.7.: Transfer function of an ideal, 3-bit ADC (redrawn from [?])

Sampling Theory

An ADC samples an analog signal with a sample frequency f_s . This frequency has to be chosen in such way, that the original signal can be fully reconstructed. The *Nyquist criteria* states, that in order to accurately represent a band-limited, continuous signal

$$y(t) \circ— Y(f) \quad \text{with} \quad Y(f) = 0|_{f > B/2} \quad (1.4)$$

it has to be sampled with a frequency f_s respecting

$$f_s > B \quad \text{or} \quad f_s > 2f_a \quad (1.5)$$

with f_a being the highest frequency contained in the signal. [?, ?]

Violation of this rule leads to *aliasing*.

Sample-And-Hold-Amplifier

ADCs need a certain amount of time to sample the input signal. If the level of the analog signal changes by more than one Least Significant Bit (LSB) during this period, this can result in large errors in the output signal. Therefore, so called Sample-And-Hold-Amplifier (SHA) are used in front of the ADC to hold the input level constant for the needed amount of time. The ADC sampling time needs to be timed in such way, that the analog-to-digital conversion falls into the hold period of the SHA and does not exceed into the sample period, for example like shown in ?. Thus, the upper frequency limitation is not determined by the ADC itself, but rather by the aperture jitter, bandwidth, distortion, etc. of the SHA. [?]

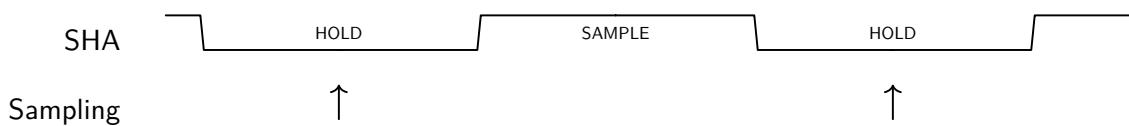
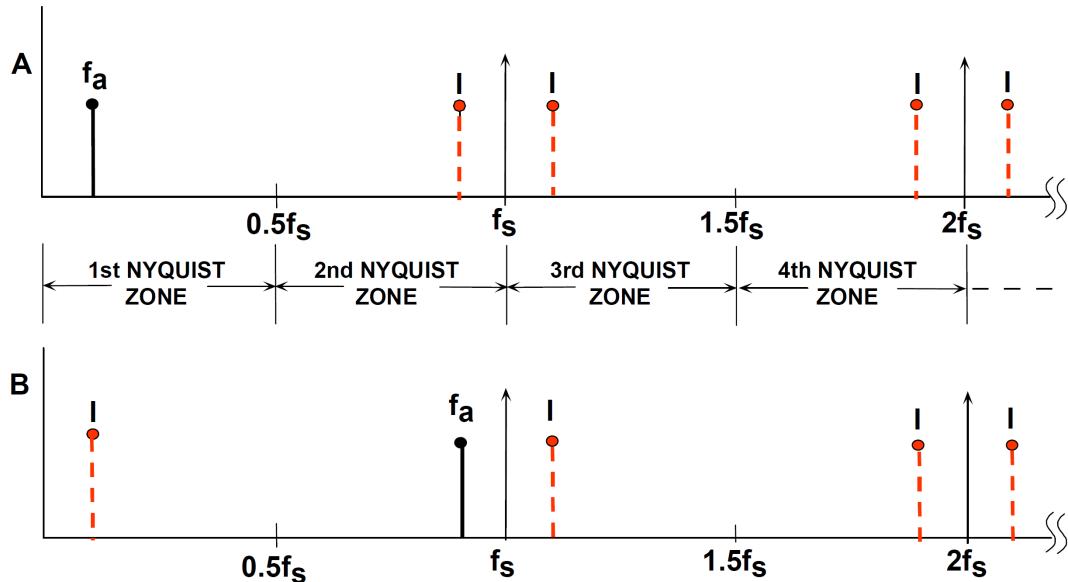


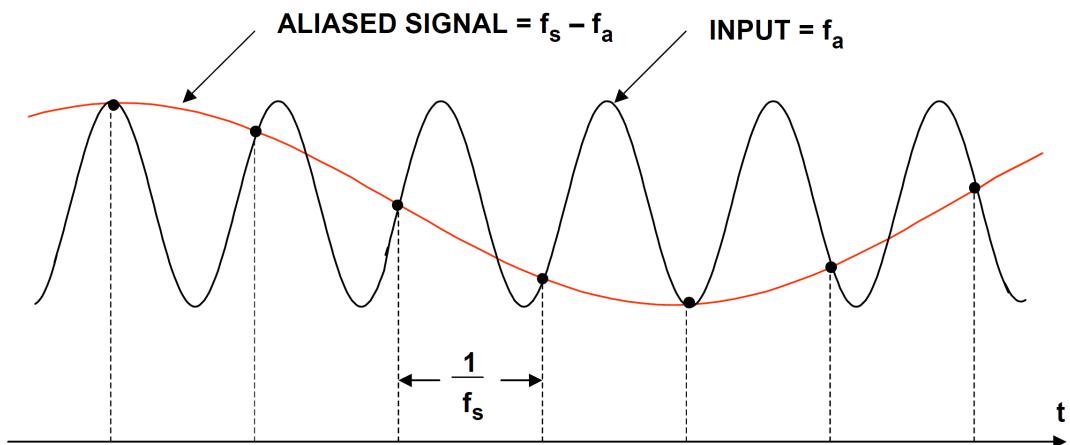
Figure 1.9.: Example for appropriate sampling timing when using Sample-And-Hold-Amplifier

In addition to the SHA, there is also the Track-And-Hold-Amplifier (THA). Instead of a sample period, the THA has a track period, where the output of the amplifier tracks the

input signal (see also Figure 1.10). When switching to hold mode, the signal at this instant is held. This is opposed to the SHA, where the output during sample mode is actually not defined and is set to the value of the input signal, only when switching into hold mode. In the track mode interval of the output waveform (positive differential clock voltage), the device behaves as a unity-gain amplifier that replicates the input signal at the output stage, subject to the input bandwidth and the output amplifier bandwidth limitations. At the positive to negative clock transition of the device, it samples the input signal with a very narrow sampling time aperture and holds the output relatively constant during the negative clock interval at a value that is representative of the signal at the instant of sampling. [?]



(a) Sampling in frequency domain



NOTE: f_a IS SLIGHTLY LESS THAN f_s

(b) Aliasing in time domain

Figure 1.8.: Analog signal with frequency f_a sampled at f_s respecting (A) and not respecting (B) the Nyquist criteria. Figure 1.8b shows the effect of case B in time domain. [?]

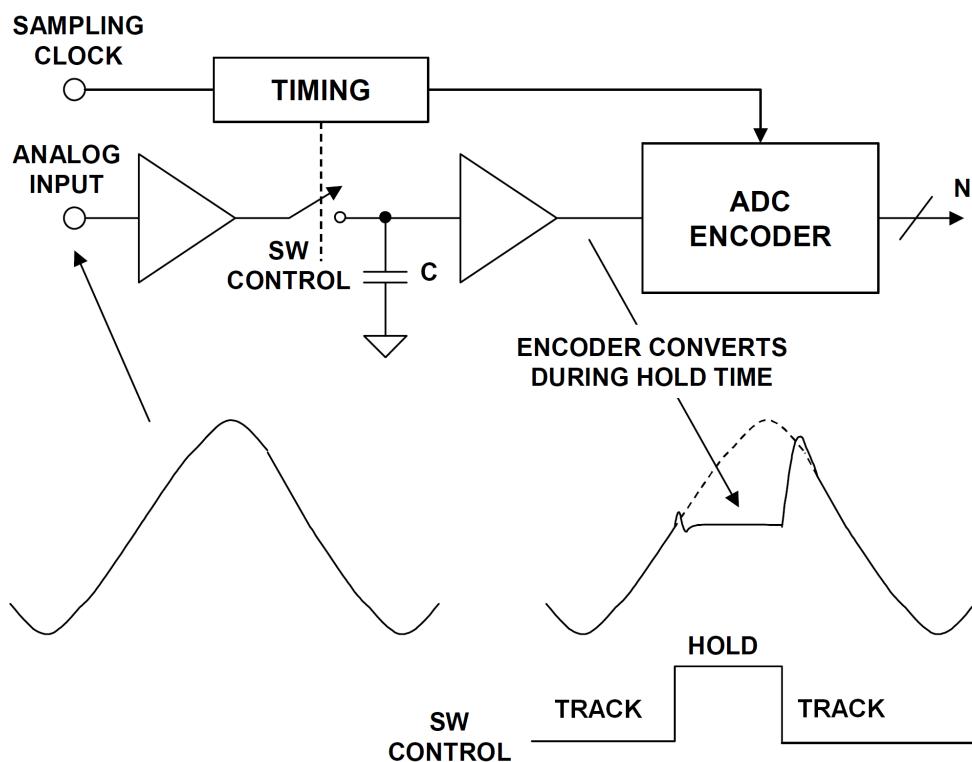


Figure 1.10.: Track-And-Hold-Amplifier schematic and principle [?]

Characteristics of Analog-To-Digital-Converters

For an ideal converter, the number of bits would be sufficient to fully characterize it. Real ADCs however differ from the ideal behavior by introducing static and dynamic imperfections. Different applications have different requirements, which leads to a number of specifications. These can be divided into the following categories [?]:

- Quantization Noise
- Static parameters
- Frequency-domain dynamic parameters
- Time-domain dynamic parameters

This section provides an overview of these figures of merit. Which of these are needed to specify the necessary performance of the ADC has to be chosen for each application accordingly.

Quantization Noise

Even in an ideal N -bit converter there will be errors during the quantization, which behave like noise. The reason is that each N -bit word represents a certain range of analog input values, which is 1 LSB wide and centered around a code center (see Figure 1.7) [?]. The input voltage is assigned to the word of the nearest code center. This means that there will always be a difference between the corresponding voltage of the respective digital code $x_q(t)$ and the actual analog input voltage $x(t)$. This difference is called the *quantization error*. For an equidistant quantization, the quantization error for a code width q is (see [?])

$$|e_q(t)| = |x(t) - x_q(t)| \leq \frac{q}{2} \quad (1.6)$$

Assuming the error voltage uncorrelated and uniformly distributed, the theoretical (maximum) Signal-To-Noise-Ratio (SNR) of this *quantization noise* can be calculated. In the time domain, the quantization error $e(t)$ can be approximated with a sawtooth signal:

$$e(t) = st, \quad -\frac{q}{2s} < t < \frac{q}{2s} \quad (1.7)$$

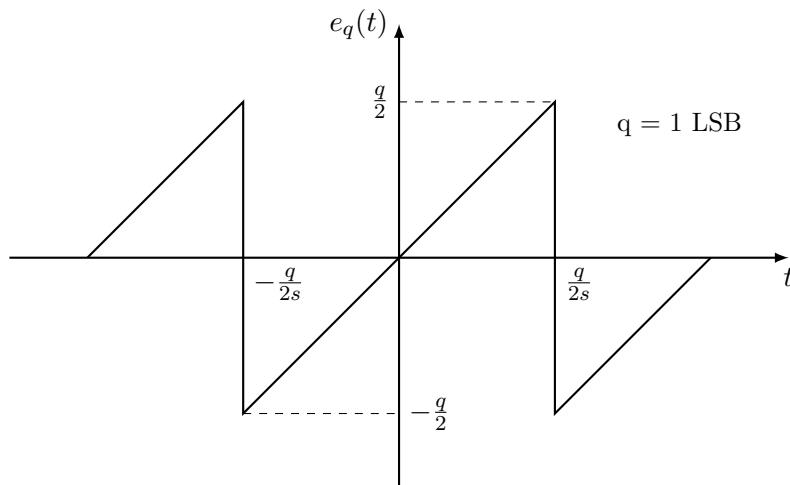


Figure 1.11.: Quantization noise as function of time (redrawn from [?])

The power of the quantization noise, which is assumed to be uncorrelated and broadband, can be calculated as the mean-square e_{rms}^2 of $e(t)$ [?]:

$$P_{QN} = e_{\text{rms}}^2 = \overline{e^2(t)} = \frac{s}{q} \int_{-q/2s}^{+q/2s} (st)^2 dt = \frac{s^3}{q} \left[\frac{t^3}{3} \right]_{-\frac{q}{2s}}^{+\frac{q}{2s}} = \frac{q^2}{12} \quad (1.8)$$

To calculate the maximal SNR of an ideal converter, a full-scale input sine wave is assumed:

$$u(t) = u_s \sin(2\pi ft) = \frac{2^N q}{2} \sin(2\pi ft) = 2^{N-1} q \sin(2\pi ft) \quad (1.9)$$

With the effective value of the signal amplitude

$$u_{\text{eff}} = \frac{u_s}{\sqrt{2}} = \frac{2^{N-1} q}{\sqrt{2}} \quad (1.10)$$

SNR

$$\text{SNR} = \frac{P_{\text{signal}}}{P_{\text{noise}}} = \frac{u_{\text{eff}}^2}{e_{\text{rms}}^2} = \frac{2^{2N-2} q^2 / 2}{q^2 / 12} = 2^{2N} \cdot 1.5. \quad (1.11)$$

In decibel:

$$\text{SNR}_{\text{dB}} = 10 \log (2^{2N} \cdot 1.5) = 6.02N + 1.76 \quad (1.12)$$

[?] [?]

Static parameters

Static parameters are specifications, which can be measured at low speed/DC.

Accuracy

Accuracy is the total error with which an ADC can convert a known voltage, which includes the effects of:

- Quantization error
- Gain error
- Offset error
- Nonlinearities

[?]

Resolution

Resolution is the number of bits N of the ADC. Depending from the resolution are the size of the LSB, which in its turn determines the dynamic range, code widths and quantization error.

Dynamic Range

The *dynamic range* represents the ratio between smallest possible output (LSB voltage) and the largest possible output (full-scale voltage). It can be calculated as

$$20 \log 2^N \approx 6N. \quad (1.13)$$

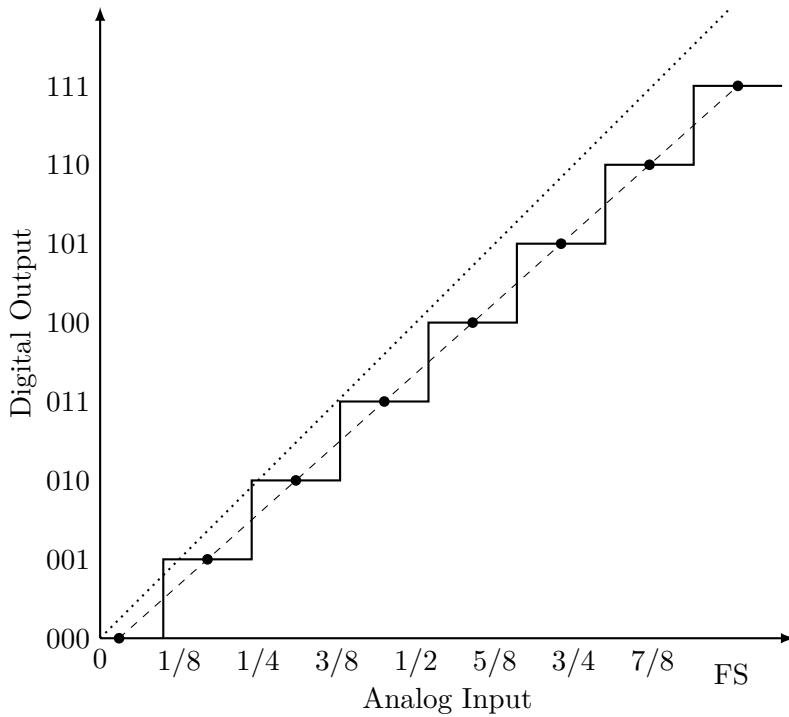


Figure 1.12.: Offset and Gain Error in the ADC characteristic transfer function. Notice the difference between the line going through the code centers (dashed) and the line of an ideal ADC (dotted)

Offset and Gain Error

The *offset error* is the deviation of the first transition voltage from the ideal $1/2$ LSB.

Gain Error defines the deviation of the slope of the line going through the zero and full-scale point of the transfer function.

These errors can easily be corrected by calibration. Figure 1.12 visualizes the effects of both offset and gain error.

Integral and Differential Non-linearity Distortion

Integral Nonlinearity (INL) in the transfer function is the distance of the code centers from the ideal line. It results from the integral nonlinearities of the front-end, SHA and also the ADC itself. [?] These nonlinearities depend on the input signal amplitude. [?]

Differential Nonlinearity (DNL) is the deviation in code width from the ideal width of 1 LSB. This nonlinearity stems exclusively from the encoding process in the ADC. [?] It not only depends on the input signal amplitude, but also on the positioning along the transfer function. [?]

Frequency-Domain Dynamic Parameters

Any real ADC is subject to noise distortion. *Noise* includes any unwanted (random or deterministic) signal components, which interfere with the measuring of the desired signal. Examples are quantization noise or random fluctuations due to thermal noise.

Distortion is the term for the change of the original signal.

In an ADC (with built-in SHA) there are a couple of sources, which introduce noise and distortion:

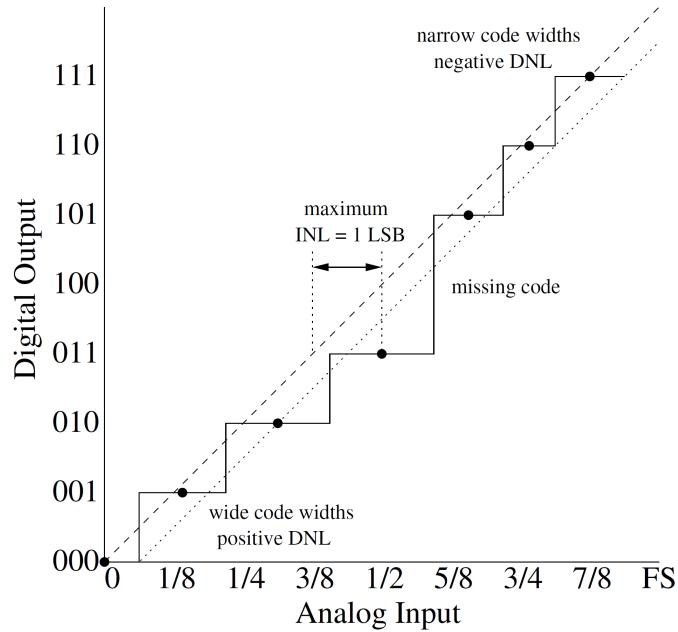


Figure 1.13.: Placeholder: Integral and Differential Nonlinearity Distortion [?]

- **Input Stage:** Wideband noise, non-linearity and bandwidth limitation
- **SHA:** Non-linearity, aperture jitter (see paragraph about Time-Domain Dynamic Performances) and bandwidth limitation
- **ADC:** Quantization noise, non-linearity

In the following, an overview of the metrics for quantification of these imperfections is given.

Signal-to-Noise-and-Distortion Ratio

Signal-to-Noise-and-Distortion Ratio (SINAD) (also called SNDR or S/N+D) denotes the ratio between the Root-Mean-Square (RMS) of the signal amplitude to the mean value of the Root-Sum-Square (RSS) of all other spectral components, including harmonics, but excluding DC (0 Hz). SINAD is a good indication over the general dynamic performance of the ADC, as it includes all contributions from noise and distortion.

Effective-Number-Of-Bits

The *Effective-Number-Of-Bits (ENOB)* expresses the SINAD in terms of bits. It can be calculated as

$$\text{ENOB} = \frac{\text{SINAD} - 1.76 \text{ dB}}{6.02 \text{ dB/bit}}. \quad [?] \quad (1.14)$$

This is derived from solving the equation of the "ideal SNR" Equation 1.12 for the number of bits N and substituting SNR with SINAD. This however means, that this parameter assumes a full-scale input signal. Expressing the ENOB for a smaller signal amplitude requires measuring the SINAD at this level and a correction factor. [?]

Spurious-Free Dynamic Range

Spurious-Free Dynamic Range (SFDR) indicates the dynamic range of the converter, which can be used, before there is interference or distortion from spurious components with the fundamental signal. [?] The SFDR is calculated as the RMS value of the fundamental

signal to the RMS value of the worst spurious signal, i.e. the highest spur in the spectrum. It is measured over the whole Nyquist-Bandwidth (DC (0 Hz) to $f_s/2$, f_s being the ADC sampling rate). The spur may or may not be a harmonic of the fundamental signal. [?] [?]

The SFDR is an important characteristic in the sense, that it indicates the smallest signal which can still be distinguished from a strong interfering signal. [?]

Figure 1.14 illustrates the SFDR in terms of decibels relative to full scale (dBFS) and decibels relative to the carrier (dBc).

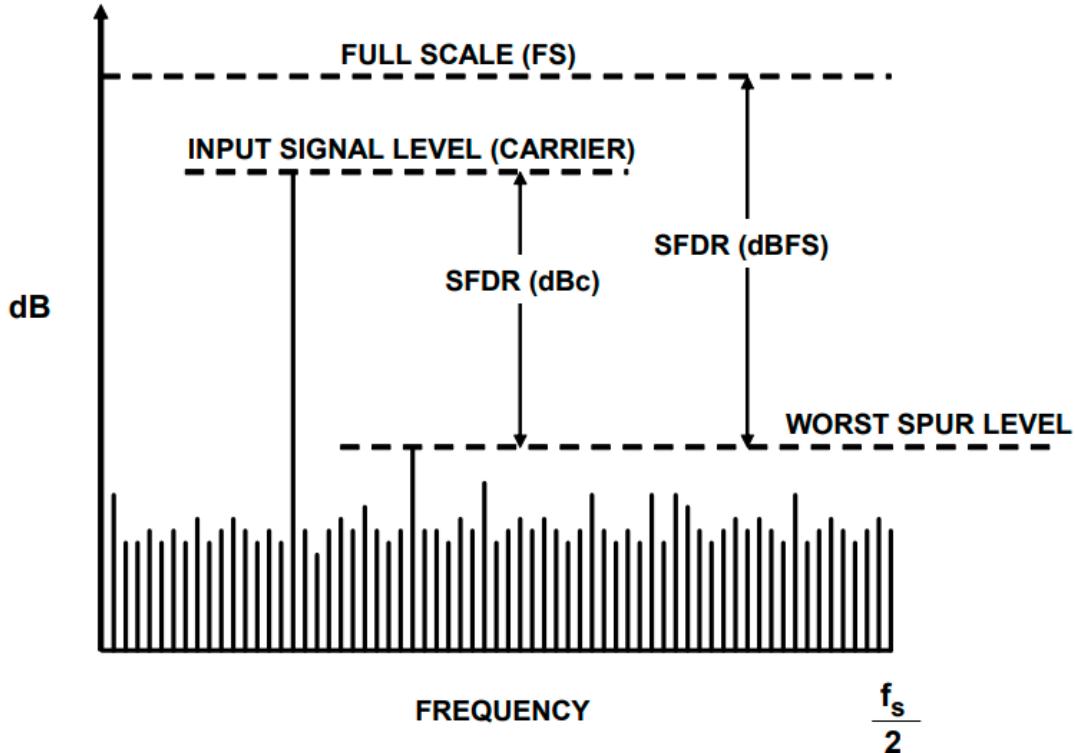


Figure 1.14.: Placeholder: SFDR [?]

Total Harmonic Distortion

The *Total Harmonic Distortion* describes the ratio of the RMS sum of the first five harmonic components (or aliased versions of them) to the RMS of the considered fundamental signal. [?]

Effective Resolution Bandwidth

Effective Resolution Bandwidth denotes the frequency of the input signal, at which the SINAD has fallen by 3dB ($\cong 0.5$ bit in terms of ENOB) compared to the SINAD at lower frequency range. [?]

Analog Input Bandwidth

Analog Input Bandwidth is the analog input frequency at which the power of the fundamental is reduced by 3dB with respect to the low-frequency value.

Full-Linear Bandwidth

The *Full-Linear Bandwidth* is defined as the frequency at which the slew-rate (SR) of the SHA starts to distort the input signal by a specified value. [?] The slew-rate is defined as the rate of how much the voltage v changes against time t :

$$\text{SR} = \frac{dv}{dt} \quad (1.15)$$

A SR of 1 V/μs for example means, that the output of the amplifier can not change more than 1 V over the course of 1 μs.[?]

Time-Domain Dynamic Parameters

Time-Domain Dynamic parameters describe the deviation of the converter's behavior from the ideal one in time domain.

Aperture Delay

Aperture Delay (or *aperture time*) is defined as delay between the triggering of the converter (e.g. rising edge of the sampling clock) and the actual conversion of the input voltage into the digitized value. [?]

Aperture Jitter

Aperture jitter describes the sample-to-sample variation in aperture delay. Jitter can cause significant error in the voltage and decreases the overall SNR of a converter. Especially for high-speed ADCs jitter poses a limit in performance.

Assuming a full-scale sinus-wave V_{in} as input signal with

$$V_{\text{in}} = V_{\text{FS}} \sin(\omega t) \quad (1.16)$$

the maximal slope of this signal is then

$$\left. \frac{dV_{\text{in}}}{dt} \right|_{\text{max}} = \omega V_{\text{FS}} \quad (1.17)$$

Aperture jitter Δt_{rms} occurring during the sampling of this maximal slope produces the RMS voltage error

$$\Delta V_{\text{rms}} = \omega V_{\text{FS}} \Delta t_{\text{rms}} = 2\pi f V_{\text{FS}} \Delta t_{\text{rms}}. \quad (1.18)$$

As variations in aperture time occur randomly, these errors behave like a random noise source. This way, a Signal-to-Jitter-Noise-Ratio (SJNR) can be defined as

$$\text{SJNR} = 20 \log \left(\frac{V_{\text{FS}}}{\Delta V_{\text{rms}}} \right) = 20 \log \left(\frac{1}{2\pi f V_{\text{FS}}} \right) \quad (1.19)$$

The voltage error due to jitter and the SJNR for different aperture jitter values are shown in Figure 1.15.

Transient Response

The *transient response* denotes the settling time of an ADC until full accuracy ($\pm 1/2$ LSB)

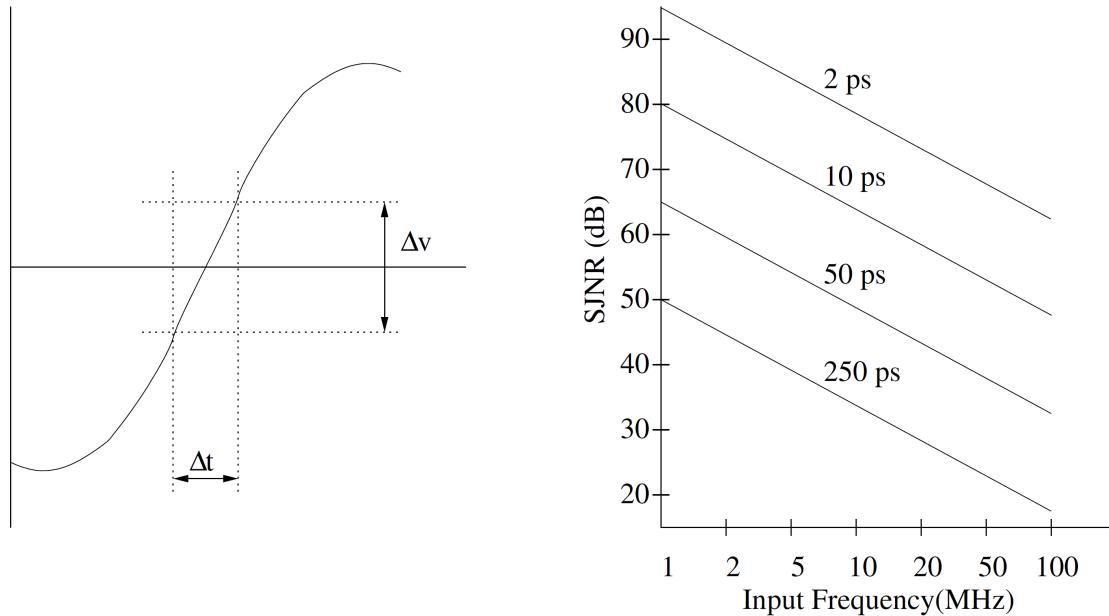


Figure 1.15.: Placeholder: Effects of aperture jitter and SJNR [?]

Time Interleaving

In the *Time Interleaving* technique multiple ADCs are used in such way, that allows to sample data at a faster rate, than the respective sample rate of each individual ADC. The principle is based on time-multiplexing an array of M identical ADCs (see Figure 1.16), each sampling at $f_c = f_s/M$ individually. This means, the ADCs are clocked in such a way, that they start their respective conversion cycle shortly one after another. At time t_0 the first ADC starts converting the input signal $V_i(t_0)$, after a time delay Δt_i the second starts converting the signal $V_i(t_0 + t_i)$, the third converts $V_i(t_0 + 2t_i)$ and so on. After the M -th ADC has sampled the signal $V_i(t_0 + (M - 1)t_i)$, the whole cycle starts anew with the first converter. [?]

Challenges

Spurs appear in the spectrum. There are several reasons for this.

First reason is the *offset mismatch* between the ADCs. Each ADC has an DC offset value. Considering as example an interleaving structure with two ADCs and a constant input voltage: when the samples are acquired back and forth between the two ADCs, the resulting output will switch back and forth between two levels due to the different offset levels. This output switches at the frequency $f_s/2$ and therefore introduces an additional frequency component in the spectrum (see Figure 1.17). The magnitude of the spur depends on the offset difference between the ADCs. [?]

Besides of the offset also the gain of the converters can be mismatched. This *gain mismatch* has a frequency component to it, which in case of an input signal of the frequency f_{in} results in a spur at $f_s/2 \pm f_{in}$ (see Figure 1.18). [?]

In the time domain, *timing mismatch* due to group delay in the analog circuitry of the ADC and clock skew³ can occur. The group delay in analog circuitry can vary between the converters. Furthermore, the clock skew has on the one hand an aperture uncertainty component at each of the ADCs and on the other hand a component related to the accuracy

³Difference in arrival time of the clock signal at different components.

of the clock phases, which are input to each converter. [?] This mismatch also produces a spur at $f_s/2 \pm f_{in}$ (see Figure 1.19).

The last possible mismatch is the *bandwidth mismatch*, which contains both gain and phase/frequency component (see Figure 1.20). Due to bandwidth mismatch, different gain values at different frequencies can be seen. An additional timing component causes different delays for signals at different frequencies through each ADC. Just like gain and timing mismatch, the bandwidth mismatch causes a spur at $f_s/2 \pm f_{in}$.

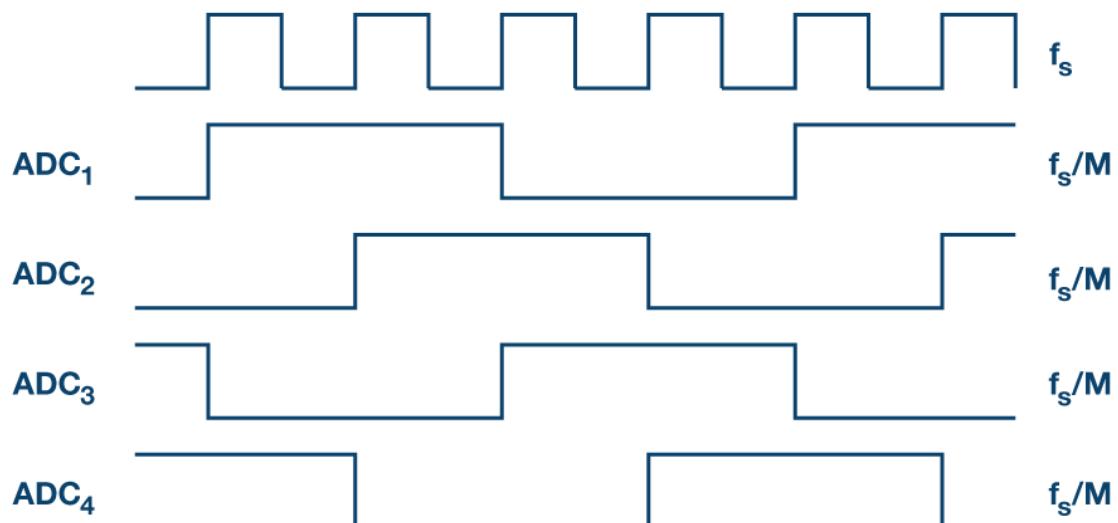
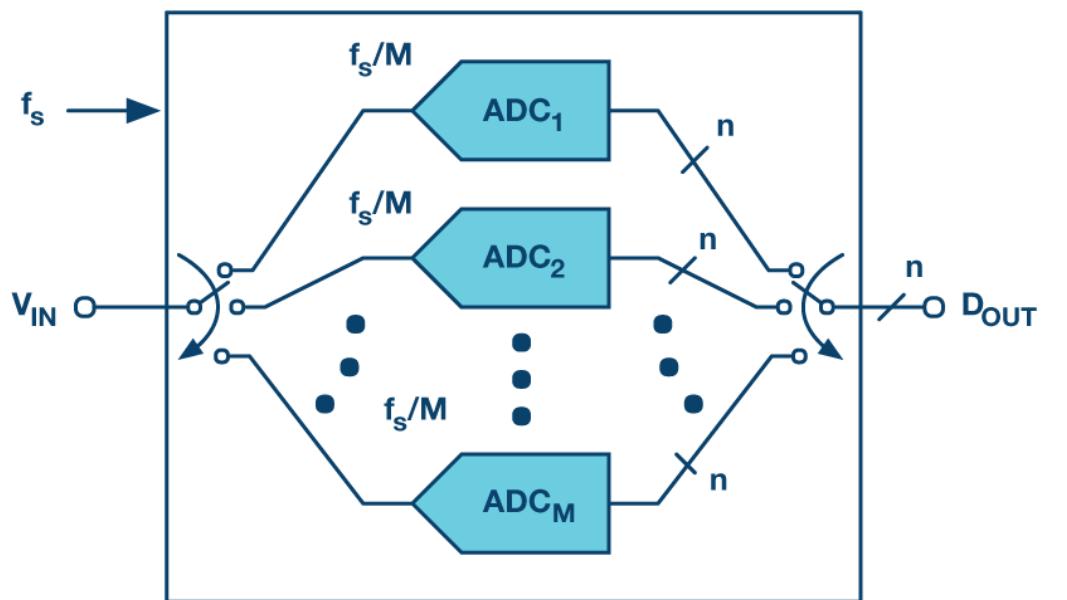


Figure 1.16.: Placeholder: An array of M time interleaved N -bit ADCs with example of clocking scheme for the case of $M = 4$ [?]

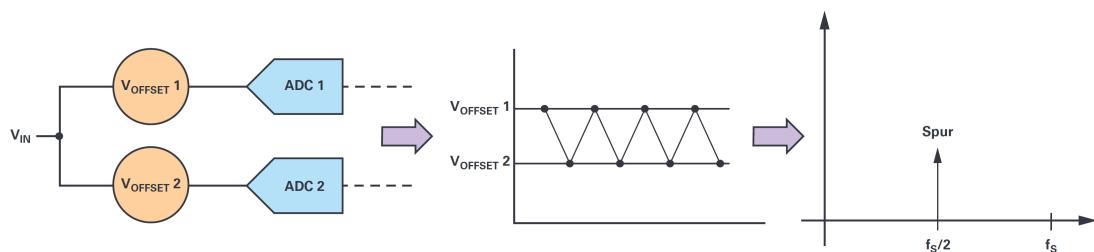


Figure 1.17.: Placeholder: Offset-Mismatch in Interleaving [?]

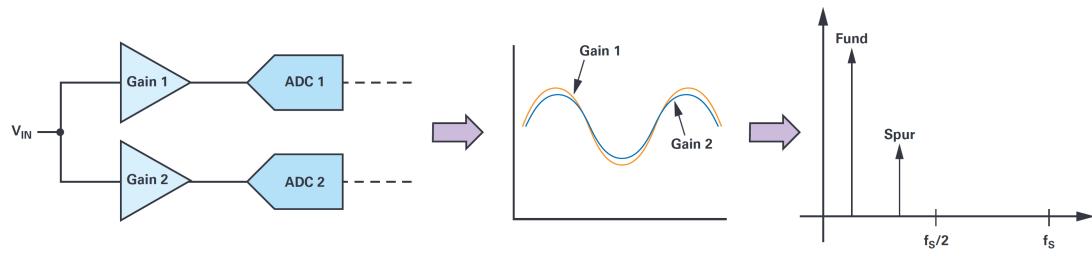


Figure 1.18.: Placeholder: Gain-Mismatch in Interleaving [?]

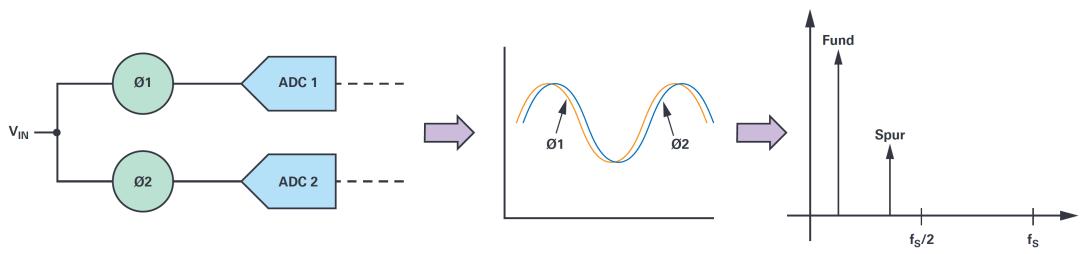


Figure 1.19.: Placeholder: Timing-Mismatch in Interleaving [?]

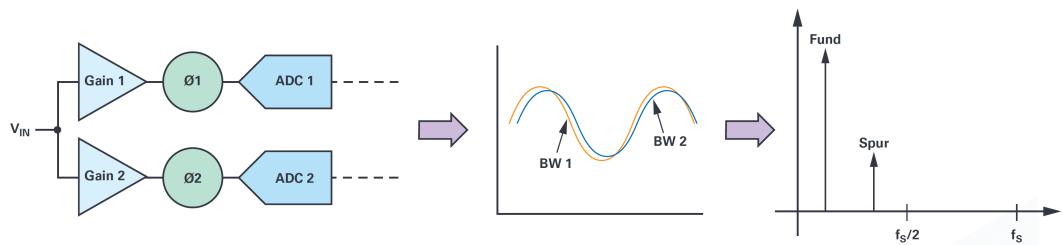


Figure 1.20.: Placeholder: Timing-Mismatch in Interleaving [?]

2. State Of The Art Read-Out-Systems

This section describes commercially available real-time oscilloscopes, as well as the KAPTURE system, which is in use at Karlsruhe Research Accelerator (KARA) for THz diagnostics. Understanding the architecture of the latter will help for the development of the new system, as the basic concept of KAPTURE is reused there.

2.1. Real-Time Oscilloscope

- KeySight Infiniium MXR-Series Real-Time Oscilloscopes
- Rohde&Schwarz DPO70000SX
- LeCroy LabMaster 10-100Zi

2.2. KAPTURE

KAPTURE is a fast readout system developed at the Institute for Data Processing and Electronics (IPE) for THz diagnostics at KARA. It is designed to digitize the pulses generated by THz detectors at each revolution, only sampling the pulse shapes without the "empty" signal in between. The system is able to sample pulses with a Full Width At Half Maximum (FWHM) between a few tens to a hundred picoseconds with a minimal sample time of 3 ps. [?]

2.2.1. General Architecture

The system consists of two parts: the sampling front-end card and a Field Programmable Gate Array (FPGA) readout card. In Figure 2.1 the setup for THz radiation measurements with KAPTURE is shown.

The incoming radiation is fed into a detector, which converts the incident photons into an electrical signal. This signal is then amplified in a wide-band Low-Noise-Amplifier (LNA). The latter splits the detector signal into four identical signals, which are then propagated to the sampling front-end card. The card consists of four parallel sampling channels with adjustable sample time, each containing a THA and an ADC. This card is connected to a read-out card, which has two tasks: programming the components on the front-end card (FPGA-part) and sending the acquired data to a PC/DAQ system. [?]

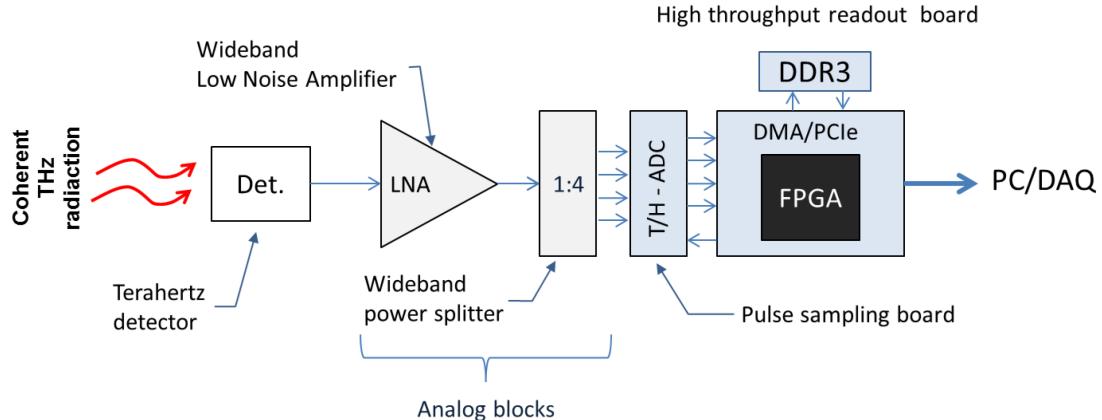


Figure 2.1.: THz radiation measurement setup with KAPTURE(cf. [?])

2.2.2. Analog front-end

Due to the high bandwidth nature of the detector signal, the analog front-end of the system has to be wideband as well to be able to sample the signal with picosecond resolution.

The used LNA is based on a commercial GaAs Microwave Monolithic Integrated Circuit (MMIC) which operates from DC to 50 GHz. It is needed to compensate the insertion loss¹ due to the following power splitter.

Classical power-splitters are not intrinsically wideband ([?]). For that reason, an wideband power-splitter was developed at IPE which fulfills the bandwidth requirements. The designed power-splitter works up to 100 GHz with an insertion loss of 8 dB and a return loss² of about 20 dB at 50 GHz.[?]

2.2.3. Sampling board

The general structure of the board with the power splitter is shown in Figure 2.2.

Four identical signals from the power-splitter are fed into four channels, consisting of a respective THA unit and a 12-bit ADC sampling at 500 MS/s. The sampling time of each unit can be adjusted individually with a delay chip with a resolution of 3 ps (maximal delay range: 100 ps). The delay chips are programmed with the FPGA on the readout card. The clock signal is provided by KARA, which is cleared from jitter by a Phase-Locked-Loop (PLL). This ensures the synchronization of the ADCs with the RF system. The cleaned clock signal is distributed to the delay chips via fan-out buffer. [?] In this way, the pulse can be "locally sampled" by adjusting the different delay with a maximum rate of 330 GS/s possible. A simplified representation of the local sampling of the signal is shown in Figure 2.3.

Figure 2.4 shows a photo of the system setup at KARA

¹Insertion loss is the loss of signal power which occurs, when a signal passes through a component.

²Return loss is the loss of signal power due to reflection by a discontinuity in the transmission line.

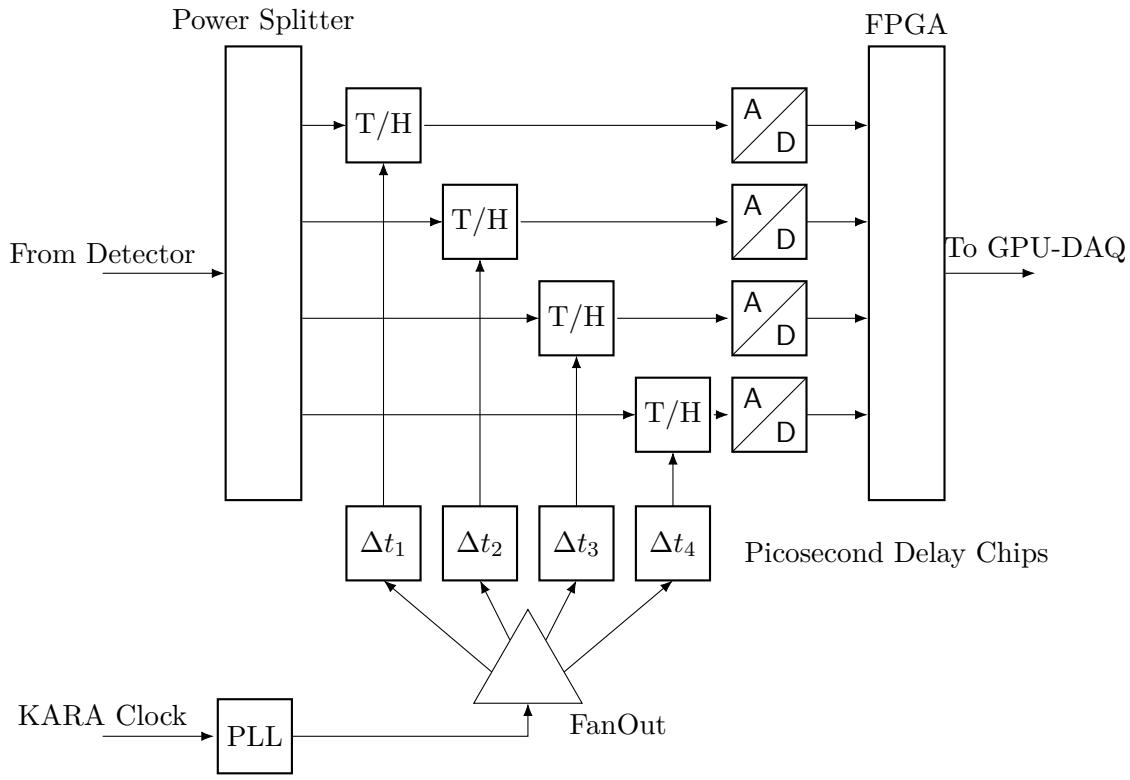


Figure 2.2.: General architecture of the KAPTURE front-end sampling card (cf. [?, p.2])

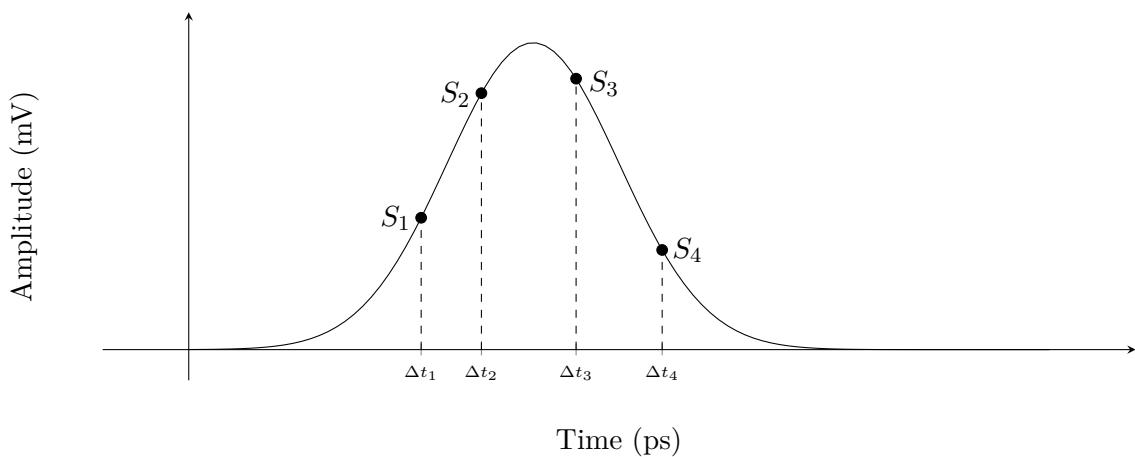


Figure 2.3.: Signal and sampled points S_1 to S_4

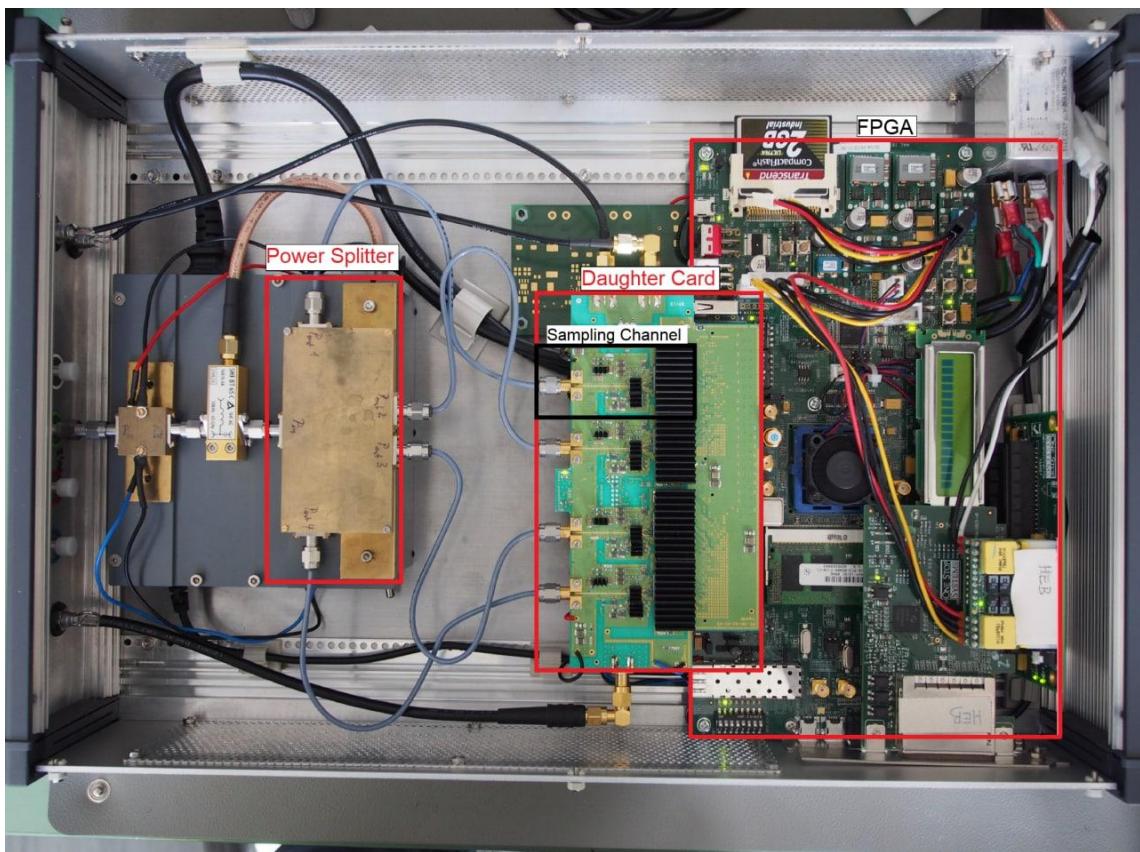


Figure 2.4.: Photo of KAPTURE with highlighted main components. [?, p. 61]

2.2.4. FPGA-GPU architecture

In order to keep a continuous data acquisition the necessary bandwidth is

$$12\text{bits} \cdot 4 \text{ samples} \cdot 500 \text{ MHz} = 24\text{Gb/s} \quad (2.1)$$

To ensure high data throughput, the readout board is based on a bus master DMA architecture which is connected to PCI Express (PCIe) end-point logic. This ensure a throughput of up to 2 GByte/s. [?]

2.2.5. KAPTURE-2

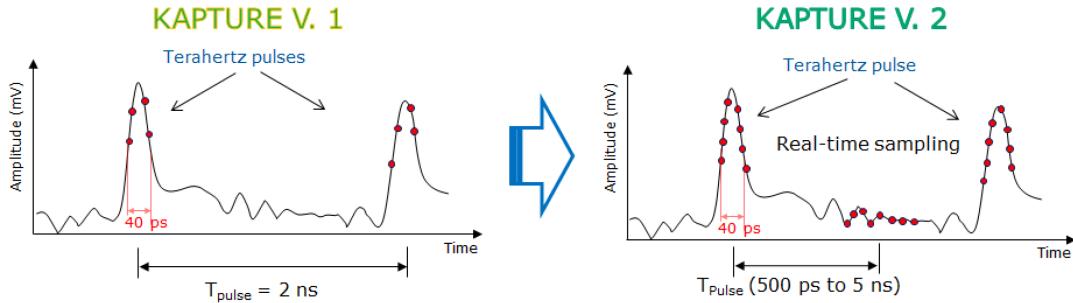


Figure 2.5.: Comparison between KAPTURE and KAPTURE-2[?]

3. Architecture Of The New Read-Out-System

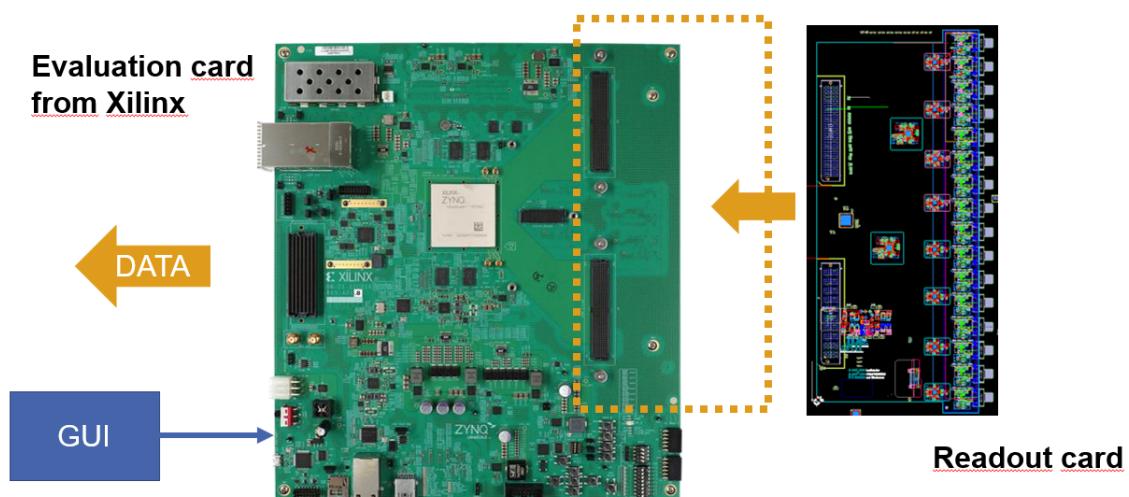


Figure 3.1.: General concept of the new readout system

3.1. Optical part

3.2. Front-End Card

Figure 3.2 shows the general schema of the sampling system, reduced to four channels for presentation purposes.

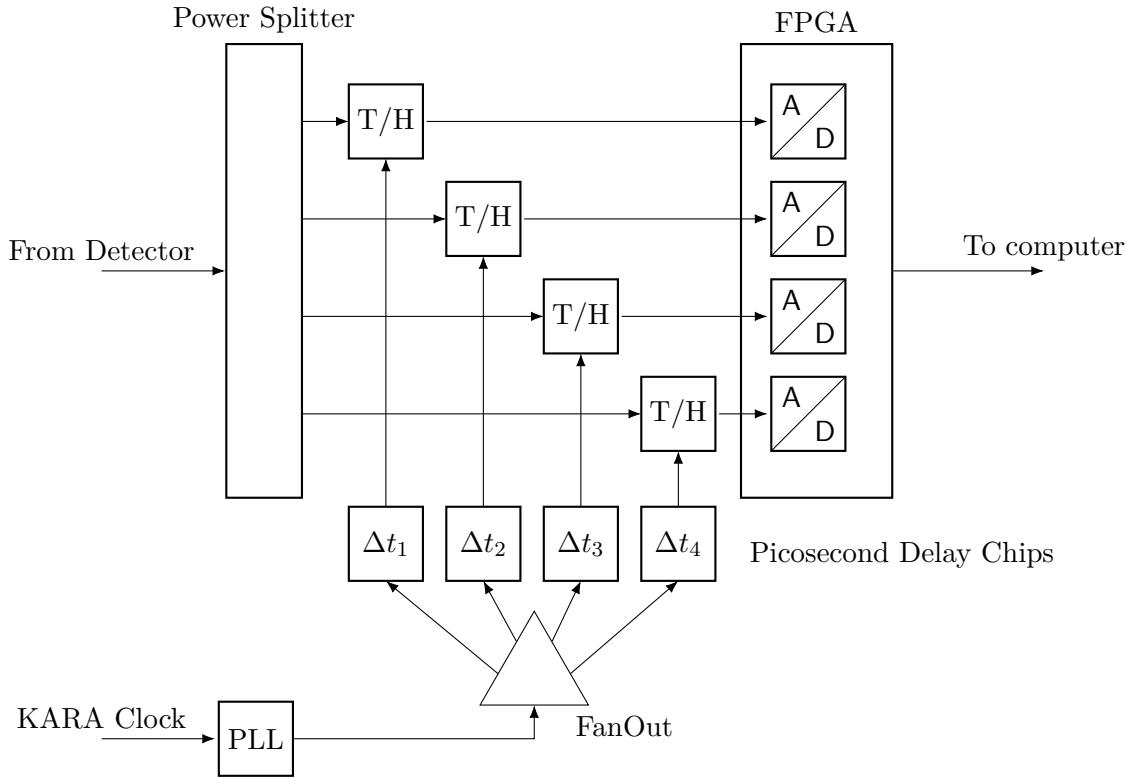


Figure 3.2.: Placeholder: General schema of THERESA. For presentation purposes only four channels are shown.

3.3. Readout Card

3.4. Selection Of The Card

When selecting the Readout Card, following criteria need to be considered:

- Integrated ADCs
- Number, resolution and bandwidth of ADCs
- Peripheral connections
- Flexibility/Customization
- Suitable connectivity for high-data-throughput

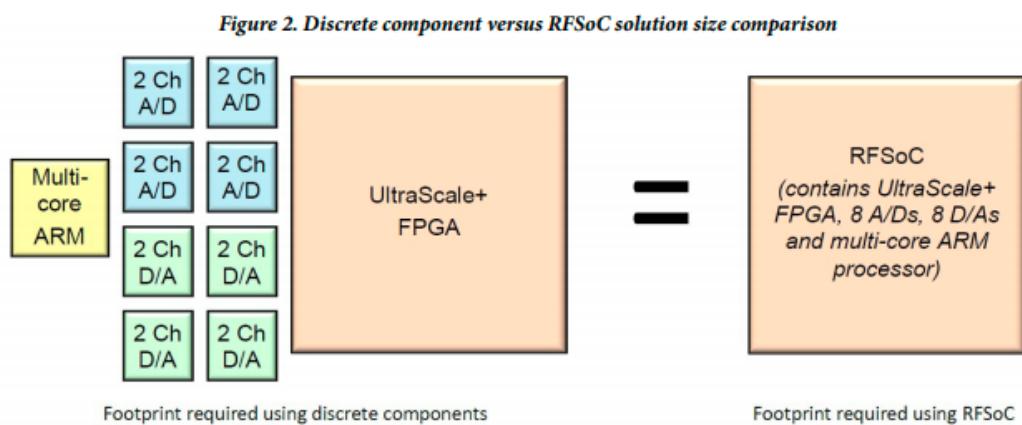


Figure 3.3.: Placeholder: Discrete vs IC

4. Design Of The Front-End Sampling Card

In the following, the process of designing the front-end sampling card is described. Designing a PCB is a process of two steps: circuit design and layout design. The software tool used to cover both of these steps is PADS xDx Designer (for schematic capture) and PADS Layout/Router (for PCB layout design) from Mentor Graphics (Siemens).

4.1. Schematics

Without knowing which components are needed and how they are interconnected, it is impossible to manufacture any board, no matter how high or low the level of complexity is. The main purpose of a schematic is thus to provide a documentation about the necessary components and in which way they should be connected. Furthermore, a schematic provides a starting point for automatic translation of the components into the layout design tool. During the creation of the schematics, the following points have to be taken into consideration:

- Components: Which components are needed and what are the performance requirement? Especially for high-speed components carefully considering specifications like signal rise and fall times, jitter, skew, etc. is crucial to achieve the overall expected performance.
- Connection/Periphery: How many pins are available for peripheral connection? Many components have an interface for programming (e.g. Serial Peripheral Interface (SPI)) which requires several pins. Especially for boards with a lot of components this can quickly become an issue.
- Signaling interfaces of the components: Additional circuitry might be needed for interfacing between two different components. Some signaling interfaces, like Low Voltage Differential Signaling (LVDS), require a specific voltage level, which might result in the need of voltage level translators.
- Common mode voltage: Keep in mind the different common mode voltages at input/output pins of different components and placing decoupling capacitors if needed.
- Filtering: Consider placing additional filtering for power supplies, as well as recommended filters from manufacturers of the components.
- Power Supply: Choose suitable power supplies/voltage regulators. How many of them are needed?
- Size and Packaging: Packaging of the components. Obviously the size matters, as space on the board is limited. But package also introduces additional capacitance/inductance, which can be a problem for precise filtering circuits.

- Power dissipation: Components, especially voltage regulators, might need coolers. This might not pose any problems for components which are located on the top side of the board. However, components on the bottom side might create an issue, if the designed PCB should be mounted on another board.
- Availability: Check if the components are still available and if they can be delivered in the given project time.

Decoupling techniques

Probably the most important part in schematics design is proper decoupling of power supplies, as Integrated Circuits (ICs) require a stable voltage on the power supply pin for optimal performance. Any ripple¹ or noise can substantially degrade the performance of the ICs, i.e. by decreasing the noise margin². Usually, manufacturers give information about proper decoupling circuits for their component in the data sheet. Is this not the case, there are basic rules of thumb which can be followed to ensure good decoupling. [?]

Basically, two types of voltage variations on the power supply pin can be distinguished: low frequency and high frequency variation. Low frequency variation occurs e.g. due to devices (or parts of them) being enabled/disabled or in the event of data traffic or processing. The current draw during these occurrences can't be compensated immediately by the voltage regulator providing the supply voltage, which leads to voltage level sagging. Time frames of this noise vary in the range of milliseconds up to days. High frequency variation results from switching events in the device, occurring in the range of the clock frequency and the corresponding harmonics up to about 5 GHz. Spikes due to Electro-Magnetic Interference (EMI) are also a source of high frequency variation and need to be compensated for. [?]

Ideally, one capacitor, which acts like a low-pass filter, should be enough to mitigate these variations. A real capacitor however has parasitics and thus can in general not be modeled by a "pure" capacitive behavior, especially for high-frequency applications. Additional resistances and inductance need to be considered [?]:

- A parallel resistance R_P , which shunts the nominal capacitance (C), representing insulation resistance or leakage.
- A series resistance R_S , or Equivalent-Series-Resistance (ESR), which represents the plates and the leads of the capacitor.
- A series inductance L_S , or Equivalent-Series-Inductance (ESL), that models the inductance of the plates and leads of the capacitor.
- A parallel resistance and capacitance, R_D and R_C , which model the effect called dielectric absorption. This denotes the phenomenon, that a capacitor which has been charged for a long time, doesn't fully discharge when briefly discharged. Dielectric absorption can be detrimental for high-precision use-cases, for power supply decoupling this effect doesn't have to be considered.

Considering all these effects leads to the equivalent circuit shown in Figure 4.1. It can be seen that this forms a RLC circuit, meaning the capacitor won't have the ideal behavior over all frequency range. Indeed, a real capacitor shows an impedance response as seen in Figure 4.2, which resembles one of a band stop, rather than a low pass. Typical capacitive behavior is seen in region (I). Region (II) shows the influence of the ESR, which is why there is a residual impedance at the lowest point. Region (III) showcases the effect of

¹ Ripple is the Alternating Current (AC)-voltage superimposed on an otherwise Direct Current (DC)-voltage.

² Noise margin defines the difference between the useful signal and noise. A sufficient noise margin is necessary to guarantee that the output signal will still be correctly interpreted, even if some noise is added to the signal.

the ESL. To "extend" the capacitive behavior over a wider frequency range, at least two capacitors are placed

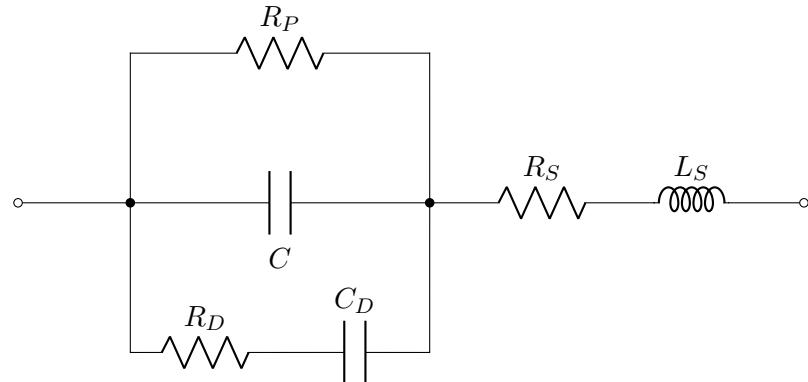


Figure 4.1.: Equivalent circuit of a real capacitance (redrawn from [?])

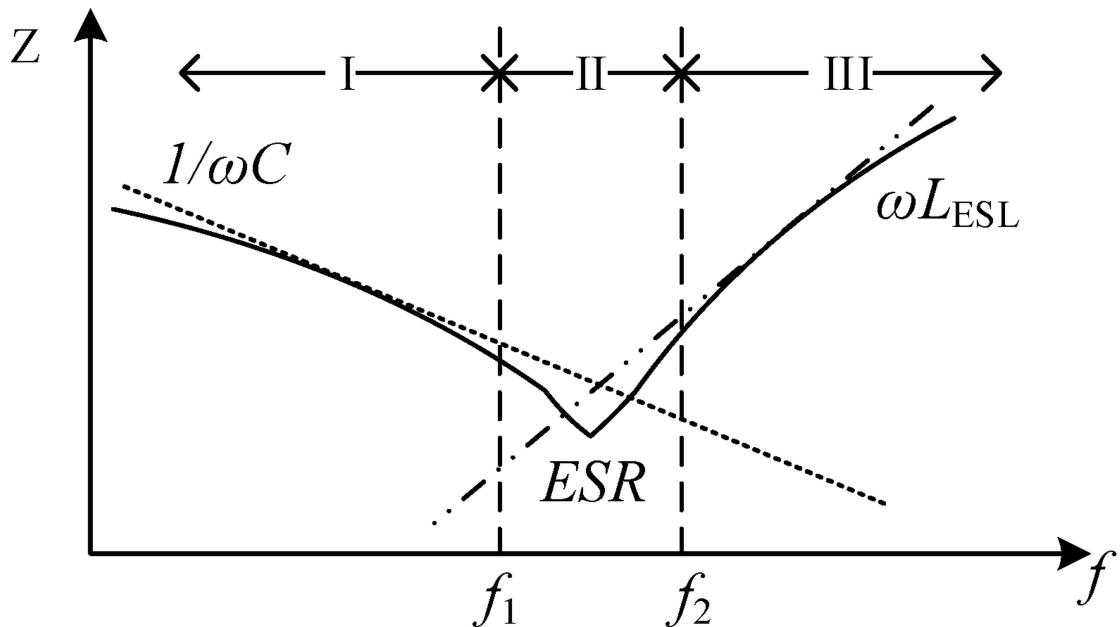


Figure 4.2.: Qualitative impedance response of a real capacitance [?]

To deal with the low frequency variation, a large capacitor (typical values: 10 - 100 μF) is placed next to the chip, not more than 5 cm (≈ 2 inch.) away. The role of this capacitor is to be a charge supply for the instantaneous needs of the device, i.e. "taking the role" of the voltage regulator until the latter can adjust to the changed current draw. [?]

A small capacitor (typical values: 0.01 - 0.1 μF), placed as close as possible to the power pins of the component. This capacitor should bypass the high frequency variation on the power supply line. [?]

All capacitors should be connected through via³ or short trace to a large area, low impedance ground plane⁴. This way the inductance due to connection traces is minimized. [?]

An optional ferrite bead in series with the supply pin keeps external high frequency from the device and the internally generated noise from the rest of the board. [?]

³See subsection 4.2.1

⁴See subsection 4.2.1

Separating Analog and Digital Ground

TODO

4.1.1. Connectors

The number and type of connectors is primarily defined by the read-out card, on which the sampling board is mounted. The different connector types serve different purposes, which can be organized into three categories.

Digital Control Signals

For digital control signals (i.e. SPI, enable signals, ...) and clocking a VITA 57.4 FMC+ connector from *SAMTEC* is used (see Figure 4.3).

FPGA Mezzanine Card (FMC) is a standard defined by VMEbus International Trade Association (VITA) to provide a standard mezzanine card⁵ form factor, connectors, and modular interface to a FPGA located on a base board (carrier card).^[?] The FMC+ standard extends the pin count and throughput of the present high-speed interfaces.

This connector provides 560 pins arranged in a 14x40 array, 80 of which are additional high-speed interfaces, located on either side of the connector (therefore this connector type is also called High Serial Pin Count Extension (HSPCe) connector, as opposed to the HSPC connector which doesn't have additional rows). 160 of them are data-pins configurable by the user. They can be used as single-ended or differential pins. Clocking capable pins can be used to propagate clocking from the mezzanine to the carrier board.

Furthermore, the connector provides pins for power supply from carrier board to mezzanine card [?]:

- VADJ: Voltage adjustable from 0 to 3.3 V (max. 4 A, max. 1000 μ F capacitive load)
- 3.3 V (max. 3 A, max. 1000 μ F capacitive load)
- 12 V (max. 1 A, max. 1000 μ F capacitive load)

An assembly drawing of the connectors is shown in Figure 4.3.

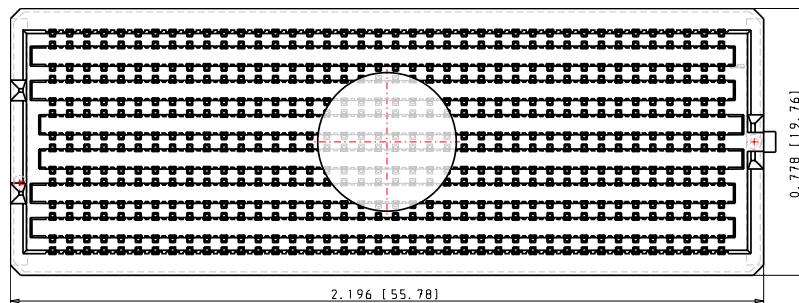


Figure 4.3.: Part drawing of FMC+ connector [?]

Analog Signals

The signal from the detector is provided to the THAs through SubMiniature version A (SMA)⁶ RF connectors from *molex*, which are mounted at the edge of the board. Figure 4.4 shows a 3D model of this connector type.

⁵A PCB which is plugged on a plug-in board. [?]

⁶Coaxial RF connector

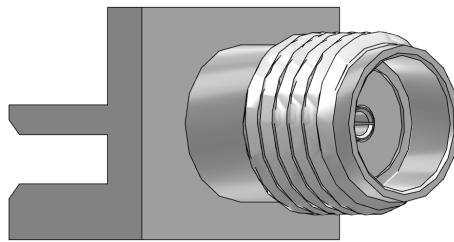


Figure 4.4.: 3D model of the edge-mount RF SMA connector from *molex* [?]

On the read-out board two RFMC 2.0 (RF Mezzanine Card) interface connectors are provided. The connectors used are LPAF (Low Profile Array, Female) connectors from *SAMTEC* with 400 pins arranged in a 8x50 array. One is dedicated for transmitting signals from the mezzanine card to the on-board ADCs, the other provides the analog output from the on-board Digital-To-Analog-Converters (DACs)⁷ to the mezzanine card. On the sampling board, the "male" (LPAM) counterpart of the connectors is used (see Figure 4.5).

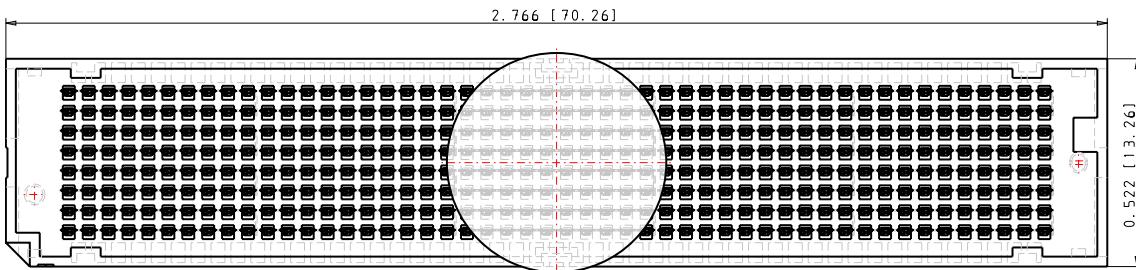


Figure 4.5.: Part drawing of a LPAM 8x50 connector

Clock Signals

The clock from the PLLs on the sampling board are propagated in different ways. The reference clock for the FPGA is propagated through the FMC+ connector. Clocking for ADCs and DACs is provided through a 6x20 LPAM connector (see Figure 4.6). For interleaving

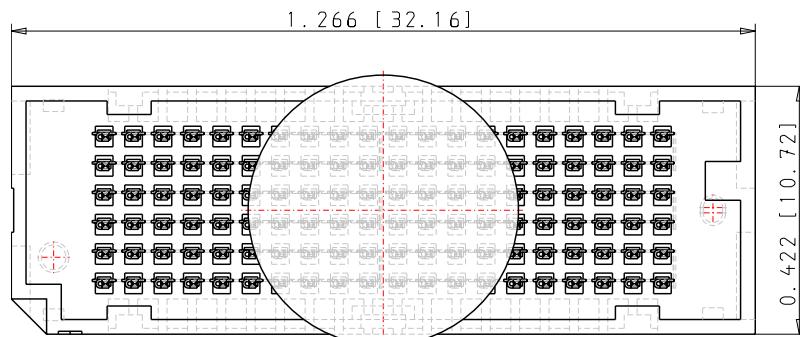


Figure 4.6.: Part drawing of LPAM 6x20 connector

⁷A DAC translates digital values into an analog signal.

4.1.2. Sampling-Channel

The sampling channel consists of the THA, which is driven by a delay chip.

Track-And-Hold-Amplifier

The THA used is the same as in KAPTURE. The component was chosen such that jitter lies in the range of hundreds of femtoseconds. [?] According to the data sheet [?], the component shows the characteristics shown in Table 4.1.

Table 4.1.: Specifications of the HMC5640 THA

Parameter	Min	Typ.	Max	Unit
Analog Inputs				
Differential FS Range		1		Vpp ¹
Common mode voltage	-0.1	0	0.1	V
Clock Inputs				
DC Differential High Voltage (Track Mode)	20	40	2000	mV
DC Differential Low Voltage (Hold Mode)	-2000	-40	-20	mV
Common mode voltage	-0.5	0	0.5	V
Analog Outputs				
Differential FS Range		1		Vpp
Common mode voltage		0		V
Track-to-Hold/Hold-to-Track Switching				
Aperture Delay		-6		ps
Random Aperture Jitter (FS, 1 GHz)		< 70		fs
Settling time ² (to 1 mV)		116		ps

¹Volt peak-to-peak

²*Settling time* is the interval between the internal track-hold transition and the time when the output signal is settled within the specified value.

As the analog input to the THA is single-ended, a 50Ω termination on the unused input pin has been added, as recommended in the data sheet.[?]

At the power pins, decoupling capacitors and a ferrite bead were placed. The THA is a crucial component, as it samples the detector signal, therefore any possible noise should be reduced to a minimum.

Delay Chip

The delay chip is used to create a delay in the clocking signal, which then goes to the THA chip. For the selection of the delay chip, the most important characteristic, apart from jitter, is the delay step size and delay range.

The delay step size must be small enough, such that the ADC interleaving technique section 1.4.2 can be implemented. The ADCs on the read-out card sample at a maximal sample rate of 2.5 GS/s, meaning during the time

$$t_s = \frac{1}{2.5 \text{ GS/s}} = 400 \text{ ps} \quad (4.1)$$

all 16 ADCs have to be clocked one time. This means, a delay step can not be greater than $400 \text{ ps}/16 = 25 \text{ ps}$.

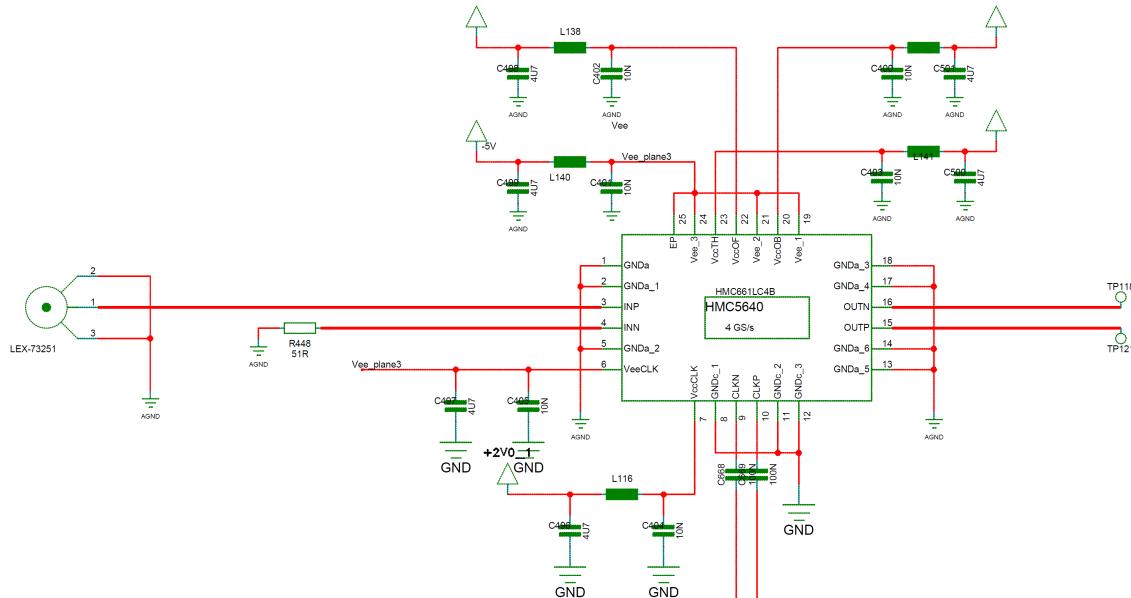


Figure 4.7.: HMC5640 THA schematic

With the HMC856 programmable delay chip from *Analog Devices*, which is also used for the KAPTURE sampling, a minimal step size of 3 ps [?] is possible. However, one draw-back is the maximal delay range of 100 ps. Considering a signal, which is stretched over several nanoseconds, this range limits the possibility to freely chose the resulting timing resolution. The major problem is the programming interface of the chip, which consists of five differential Current Mode Logic (CML) inputs. This means, one chip already takes up 10 pins. For in total 16 necessary delay chips, this results in 160 pins occupied only by the delay chips. This occupies all pins of the FMC+ connector (see subsection 4.1.1) available for the user.

A better candidate is the dual channel programmable delay chip NB6L295 from *ON Semiconductor*. This chip provides two programmable delay channels, therefore effectively reducing the chip count by half. With a delay step size of 11 ps it is still suitable for the targeted interleaving method, covering a total delay range from 3.2 ns to 8.8 ns per delay channel. The chip is programmed via Serial Data Interface (SDI), which only requires 3 pins and one enable pin. Thus, the total number of pins used by the delay chips is $4 \cdot 8 = 32$, which is a significant reduction compared to the HMC856.

Outputs

The output of the delay chip is using a Low-voltage positive emitter-coupled logic (LVPECL) signaling interface, which is based on an open-emitter topology (see Figure 4.9). This requires a path to DC, which is achieved by adding 140Ω resistors.

As the output will be connected to the THA, it is necessary to check the compatibility of the maximum amplitude and common mode. According to the data sheet [?], the voltage level of the output can vary between $V_{cc} - 1825$ mV and $V_{cc} - 825$ mV (see Table 4.2). Maximal voltage amplitude acceptable by the THA inputs is 2000 mV (see Table 4.1). When using a supply voltage of $V_{cc} = 3.3$ V, which is e.g. provided by the read-out card through the FMC+ connector, this leads to a maximum output level of 2475 mV. This succeeds the limit given by the THA. Therefore, for V_{cc} a smaller voltage should be considered. In this design a voltage of $V_{cc} = 2.5$ V is chosen, which guarantees that the amplitude lies within the range 675 mV – 1675 mV.

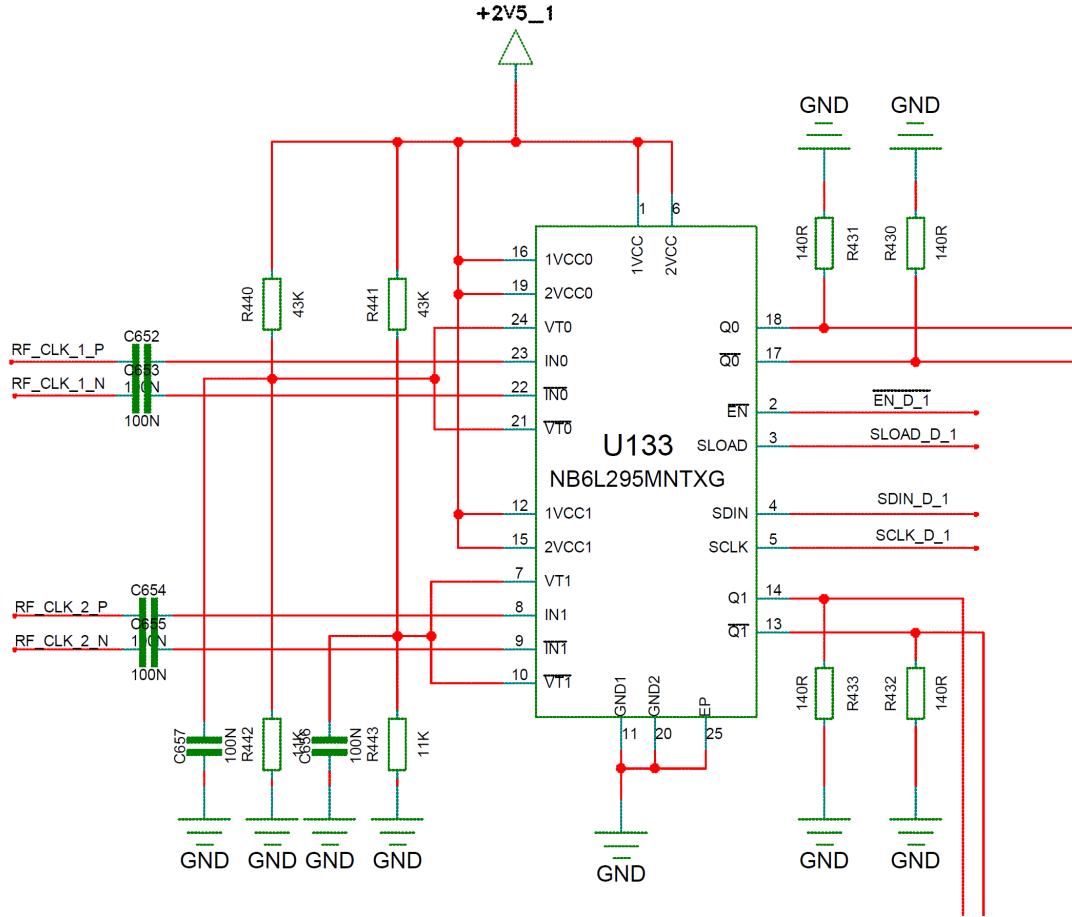


Figure 4.8.: NB6L295 schematic

The common mode voltage V_{CM} is calculated as

$$V_{CM} = \frac{675 \text{ mV} + 1675 \text{ mV}}{2} = 1175 \text{ mV}. \quad (4.2)$$

This is higher than the maximal input common mode voltage of 0.1 V of the THA (see Table 4.1). AC coupling is therefore necessary in this case.

Inputs

When driving the inputs with a LVPECL driver (output from the preceding PLL), the VT_x and \bar{VT}_x pins of the delay chip need to be connected to $V_{cc} - 2 \text{ V}$ (see Figure 4.10). In case of $V_{cc} = 2.5 \text{ V}$, this results in a voltage level of 0.5 V. To avoid using an additional voltage regulator, this voltage level is achieved by using a simple voltage divider connected to V_{cc} . Choosing the resistor values to $43 \text{ k}\Omega$ and $11 \text{ k}\Omega$ results in a voltage of

$$V_{cc} \frac{11 \text{ k}\Omega}{11 \text{ k}\Omega + 43 \text{ k}\Omega} = 0.5093 \text{ V} \approx 0.5 \text{ V} \quad (4.3)$$

A 100 nF capacitor is put in parallel for power supply decoupling.

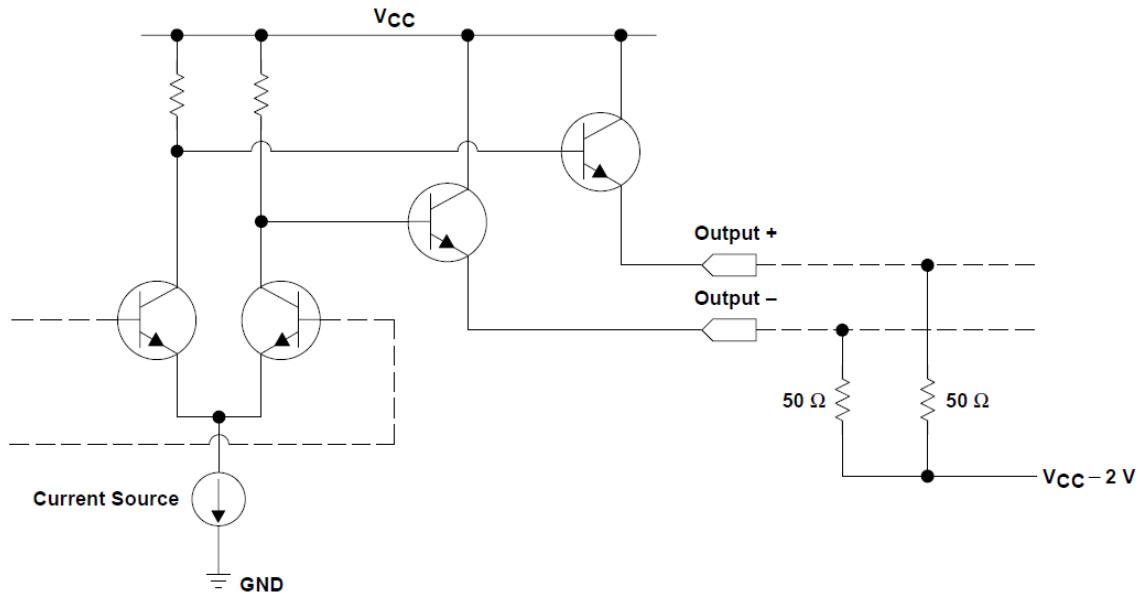


Figure 4.9.: LVPECL driver topology. Left side shows the emitter-follower based driver. On the right, an example biasing with resistors is shown. [?]

Table 4.2.: Specifications of the NB6L295 delay chip [?]

Parameter	Min	Typ.	Max	Unit
Outputs				
Output HIGH Voltage	$V_{cc} - 1075$	$V_{cc} - 950$	$V_{cc} - 825$	mV
Output LOW Voltage	$V_{cc} - 1825$	$V_{cc} - 1725$	$V_{cc} - 1625$	mV
Common mode voltage	-0.1	0	0.1	V
AC Characteristics				
Random Clock Jitter RMS		3	10	ps
Output Rise/Fall Times (@50 MHz)	85	120	170	ps
Serial Clock Input Frequency (50% Duty Cycle ¹)			20	MHz
Minimum Pulse width SLOAD	1			ns

¹Percentage of the ratio of pulse width and total period of the waveform.

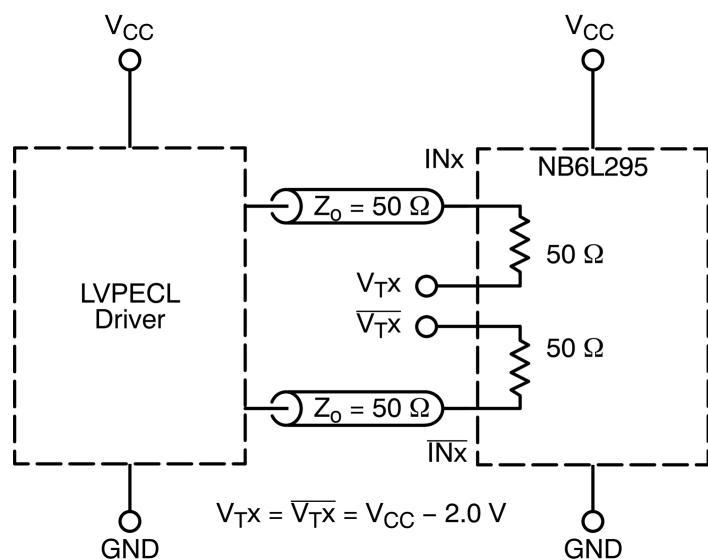


Figure 4.10.: LVPECL recommendations for NB6L295 [?]

4.1.3. Clocking

TODO:

- Brief theory about PLL -> VCO, Loop Filter, Charge pump, etc.
- Maybe some simulation with the PLLatinum Tool (used for calculation of the noise, jitter, etc. over the frequency)

Clocking distribution is designed as shown in Figure 4.11. The LMK0480x Low-Noise Clock Jitter Cleaner PLL from *Texas Instruments* cleans the incoming reference clock coming from the system (e.g. from KARA) for high temporal accuracy [?]. It is used with an external Voltage-Controlled Crystal Oscillator (VCXO) from *ABRACON*. The LMK0480x has only 12 outputs, not enough for the 16 THAs and additional clocking for FPGA, ADC and DAC, especially considering that the outputs are divided into six groups à two outputs. Outputs in one group have the same configuration (frequency, phase, ...), which means that effectively only six different outputs are available. A low noise clock distribution fanout buffer, the HMC987LP5E from *Analog Devices* is therefore used for distributing the clock signal to the delay chips. As one fanout buffer has eight outputs, two chips are needed to cover all channels. One output of the PLL is propagated to the FMC+ connector as reference clock for the FPGA. Up until this part, this architecture is not different from the one on the KAPTURE sampling board.

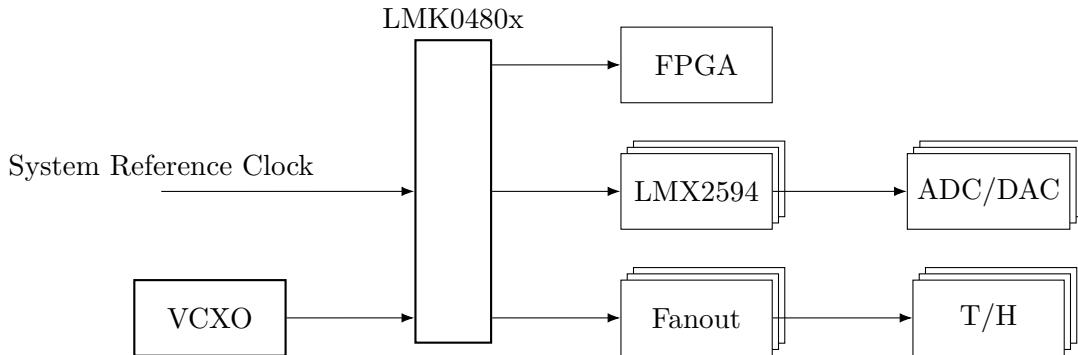


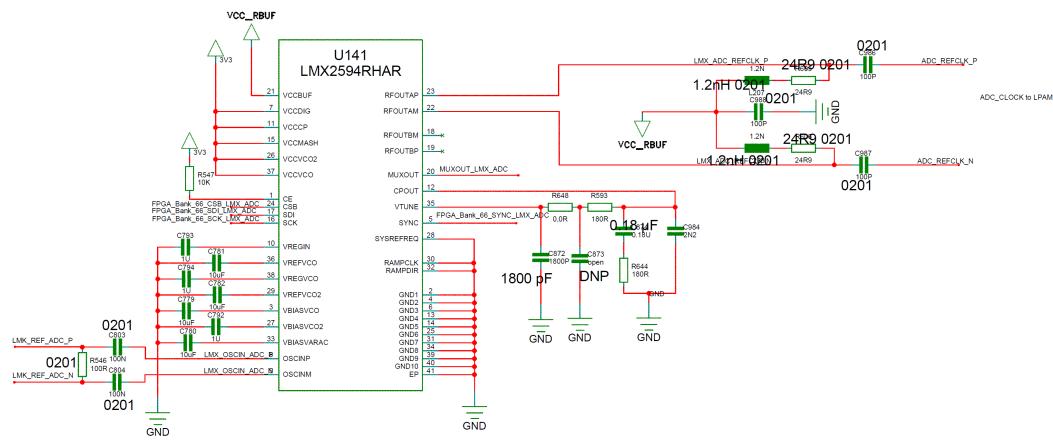
Figure 4.11.: Placeholder

The maximum output frequency of the LMK0480 is 1536 MHz, not enough to clock the ADCs at maximum sampling rate (2.5 GS/s). A second PLL is therefore needed. The read-out card is provided with an RF Clock add-on card, which is necessary to generate the high-frequency clocks for the converters. The schematic is provided by the manufacturer and is used as reference for the design. According to the user guide [?], the parameters for the loop filter are as shown in Table 4.3.

TODO: Explain with help of timing diagram, why

Table 4.3.: Specifications of the NB6L295 delay chip [?]

Parameter	Value
VCO Gain	239 MHz/V
Loop Bandwidth	32.7 kHz
Phase Margin	69 deg
Effective Charge Pump Gain	3 mA
Phase Detector Frequency	24.576 MHz
VCXO Frequency	Designed for 15 GHz
Loop filter components	
$C_{1,LF}$	2200 pF
$C_{2,LF}$	180 nF
$C_{3,LF}$	1800 pF
R_2	160 Ω
$R_{3,LF}$	180 Ω

**Figure 4.12.:** Schematics of the LMX2594

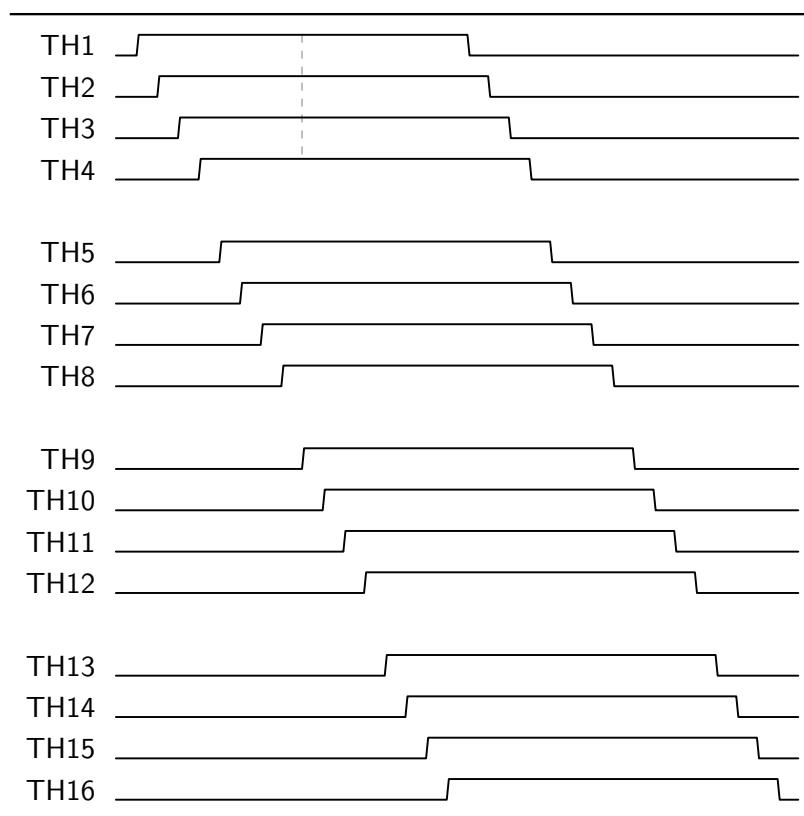


Figure 4.13.: THA Timing diagram. Shows the clocking of the THA, high level = HOLD, low level = TRACK. Dashed line represents the sampling of the ADC.

4.1.4. Digital-To-Analog-Converter Balun Channel

For test purposes, two DAC channels from the read-out card are routed on the sampling board. In this way, test signals can be generated right on the read-out card, without the need for an external signal generator.

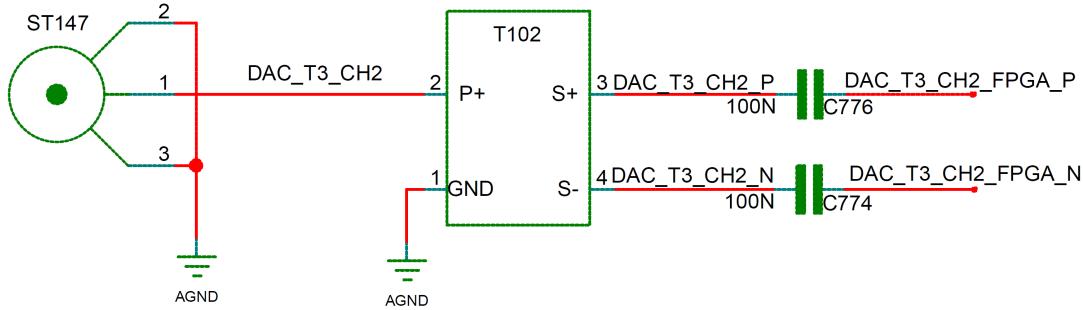


Figure 4.14.: DAC-channel with balun. Signal travels from right to left.

4.1.5. Power Supply

On the KAPTURE sampling board, the ADP1708 from *Analog Devices* is used to provide a power supply for the THAs. For the Track-And-Hold amplifiers a new power supply unit, the ADP1741 from *Analog Devices*, should be used. It is necessary to think about the amount of power supply chips needed. As a rule of thumb, the power supply should provide twice the maximum power needed by the components it drives. [?] The power consumption/maximum current for the respective components on the sampling board is listed in Table 4.4.

Table 4.4.: Power consumption of components on the board

Component	V_{cc} (V)	I_{max} (A)	P_{max} (W)	#parts	I_{tot}^1 (A)
HMC5649 THA	2	0.221	0.442	16	3.536
	-5	-0.242	1.21		3.872
HMC856 (Delay)	-3.3	0.185	-0.611	16	2.96
HMC987LP5E (Fan-Out buffer)	3.3	0.234 ²	0.772	2	0.468
LMC0480 PLL	3.3	0.590 ³	1.947	1	0.590
VCXO	3.3	0.03	0.198	1	0.03

¹for 16 ADCs

²All Outputs and RF-Buffer

³All CLKS

The maximal current which the ADP1741 can provide @2 V is 2 A. This means, with one Track-And-Hold amplifier requiring a maximal current of 0.221 A, one ADP1741 can handle four units according to the rule mentioned beforehand ($I_{max_ADP1741} = 2 \text{ A} > 2 * I_{tot}, I_{tot} = 4 * 0.221 \text{ A} = 0.884 \text{ A}$).

4.2. Layout

4.2.1. PCB Structures Overview

An overview over the basic structures on a PCB is given.

Traces

A *trace* is a strip of metal, which establishes an electrical connection and carries signals between two (or more) points in the horizontal plane of a PCB. [?]

Planes

Plane denotes an uninterrupted area of metal, which covers the whole PCB layer. If this area only covers only part of the layer, it is called a *planelet*. These areas provide power distribution across the PCB and present an important transmission medium for the return current⁸. [?]

Vias

A via is metal-plated hole, which is used to route a trace in vertical direction, i.e. from the PCB outer layer to the inner layers. They carry signals and power. Three types of vias are [?]:

- Blind via: A blind via connects the surface layers with at most three layers below.
- Buried via: A buried via only connects internal layers.
- Through via: A through via goes from one PCB surface to another and is used to connect any layer.

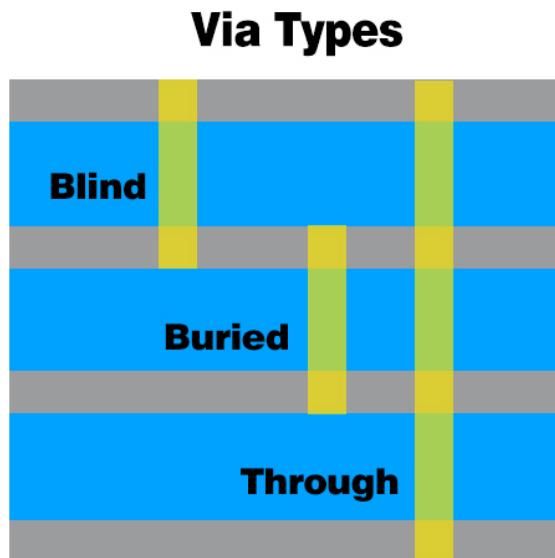


Figure 4.15.: Visualization of via types [?]

In this design only blind and through vias are used.

⁸Any current, which is injected into the components/boards, needs a return path, as otherwise there is no closed circuit.

4.2.2. PCB Substrate

TODO Megtron6 Laminate R-5775 Prepreg R-5670

$\epsilon_r = 3.61$ at

4.2.3. Floor Planning

TODO

4.2.4. Transmission lines

TODO

Transmission lines carry high-frequency signals, therefore the geometry of them is important, as this affects the impedance. For single-ended the waveguide characteristic impedance should be 50Ω , for differential signals 100Ω . For slow signals not that crucial, but for sensitive, high-speed signals, e.g. clocking signals, proper calculation is very important to ensure signal integrity and reduce reflection and damping.

For PCBs usually coplanar waveguides are used for signal propagation. The characteristic impedance depends on the dielectric, the trace width, separation between traces and separation between the signal traces and the ground planes/traces. Formulas to calculate the characteristic impedance are quite lengthy and not easy to solve. Luckily, tools⁹ exist to quickly calculate the geometric values needed for appropriate impedance. For the design, the Si9000e PCB field solver from *Polaris* (see Figure 4.16) is used to calculate the necessary trace widths, separations, etc.

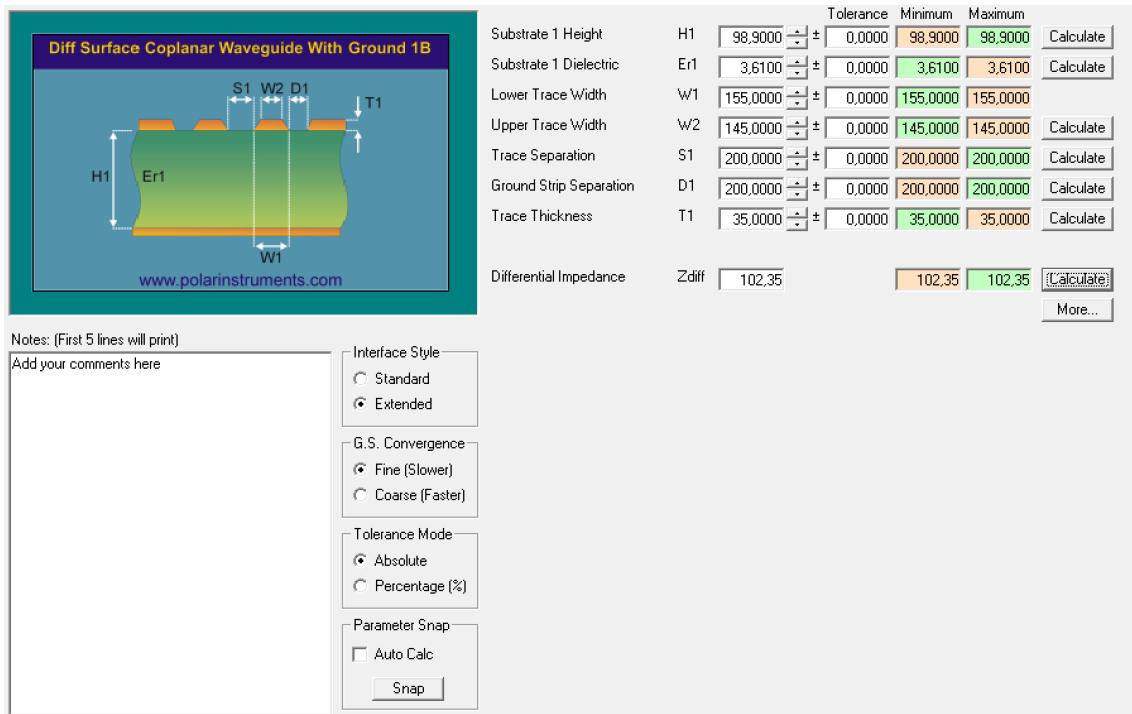


Figure 4.16.: Polaris Solver

Three geometries of waveguides are used in this design which are described in the following. Furthermore, geometric dimensions calculated with the Si9000e tool are presented. In principle, the geometries can be taken from the design of the sampling card of KAPTURE

⁹which are quite expensive though, if a high variety of geometries is necessary

system. As the dielectric constant of the substrate is slightly different (KAPTURE: 3.52, here: 3.61), the impedance has to be recalculated to check whether the characteristic value impedance still lies in the 10% tolerance.

Surface Coplanar Waveguide with Ground

The surface coplanar waveguide has the geometry shown in Figure 4.17. The single trace of the thickness t and width a lies between two ground planes on a dielectric of thickness h and the effective dielectric constant ϵ_r . Another ground plane is located at the bottom of the dielectric. Separation between trace and ground plane is calculated as $(b - a)/2 := d$.

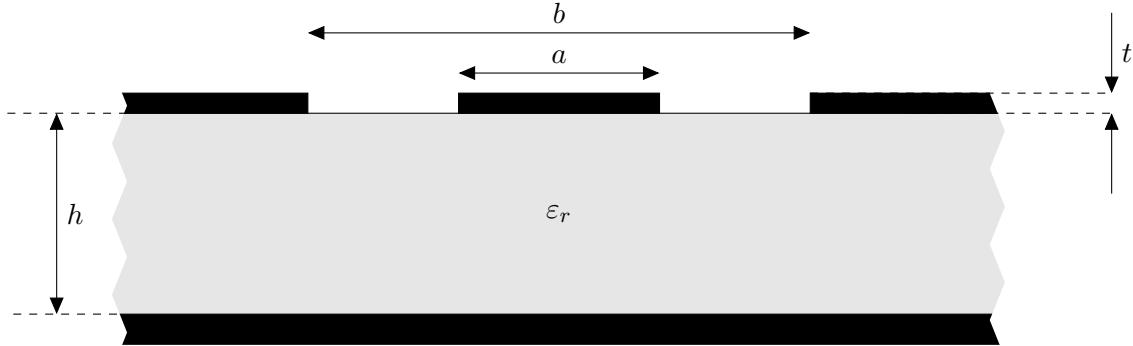


Figure 4.17.: Coplanar Waveguide with Ground

Trace width is assumed to be $a = 180 \mu\text{m}$. In the tool, an upper and a lower trace width can be specified, therefore taking into account manufacturing processes. As the exact upper trace width is not known, both are assumed to be $180 \mu\text{m}$. Trace-To-Ground Separation, or "Ground Strip Separation", is defined by the manufacturing technology of the PCB process: $d = 250 \mu\text{m}$. With $h = 98.9 \mu\text{m}$, $\epsilon_r = 3.61$ (at 10 GHz) this results in a characteristic impedance of 52.90Ω . This lies well in the tolerance area.

Over the frequency range, the value of the effective dielectric constant changes from 3.71 (at 1 GHz) to 3.61 (at 10 GHz). As the tool provides the possibility to calculate the impedance versus a changing parameter, the influence of a changing dielectric was calculated. As can be seen in ??, with higher effective dielectric constant, the characteristic impedance decreases (see ??).

The only parameters, which are not defined by the manufacturing process and therefore can be altered, are the ground strip separation and the trace width. When altering the trace width, the tool automatically assumes an upper trace width, which is $25 \mu\text{m}$.

Differential Pairs on Surface

Differential Pairs between Layers

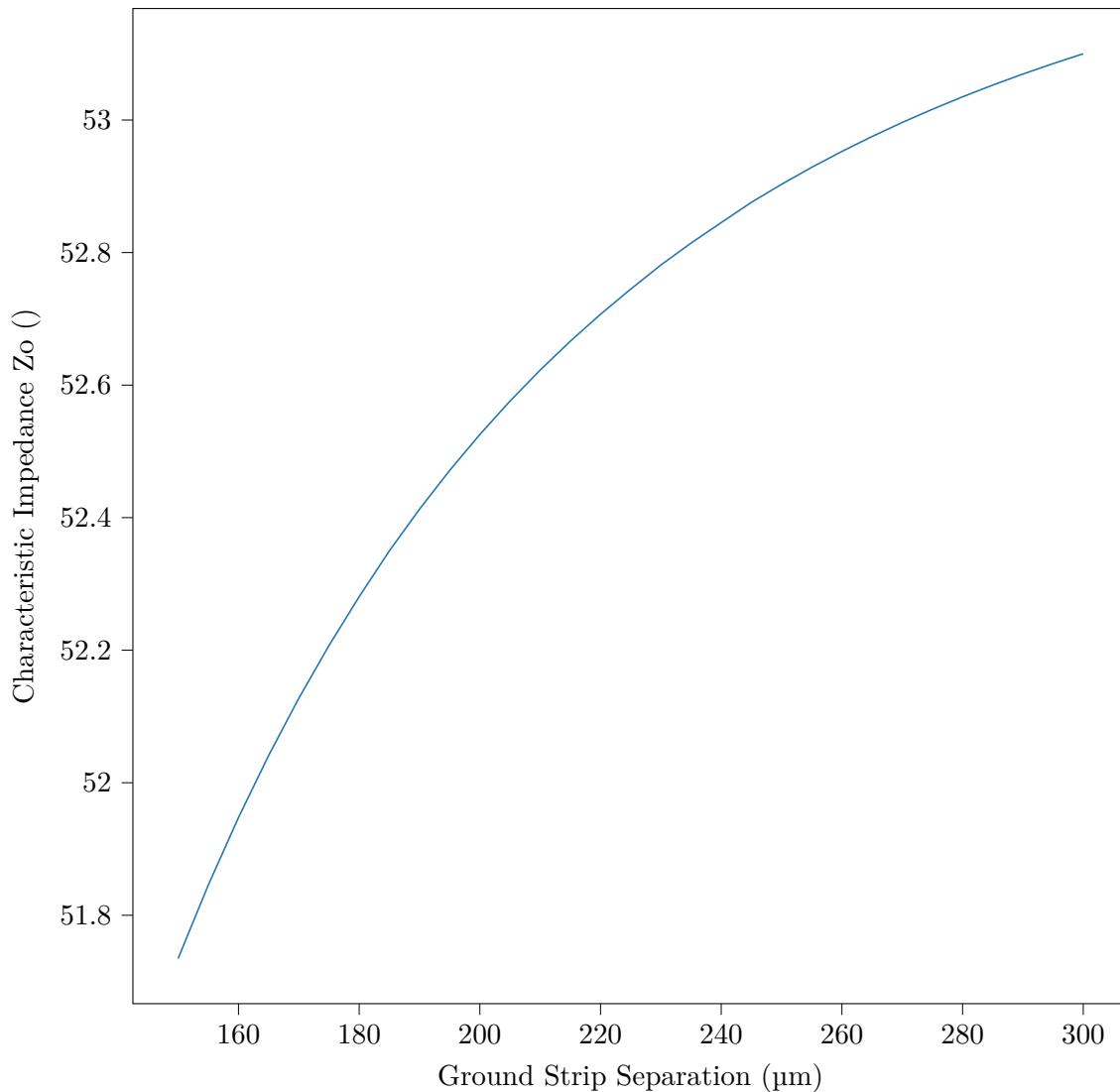


Figure 4.18.: Characteristic impedance Z_o vs effective dielectric constant ϵ_r of a surface coplanar waveguide with ground

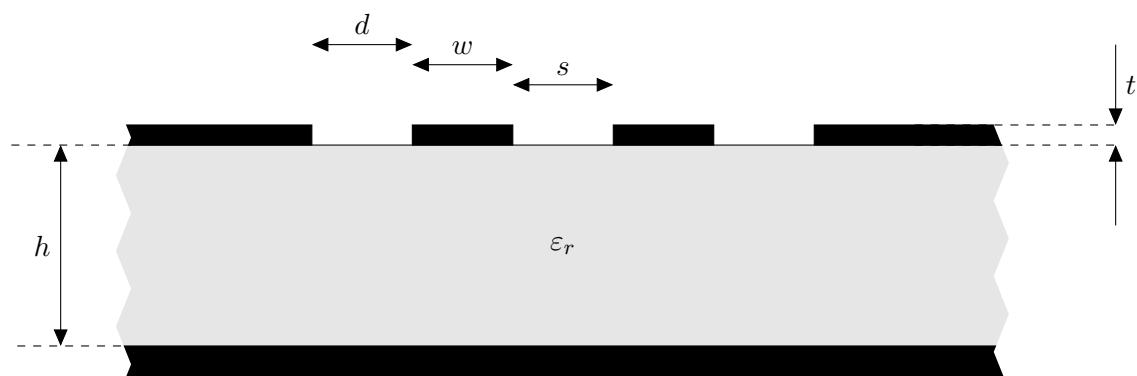


Figure 4.19.: Edge-Coupled Coplanar Waveguide

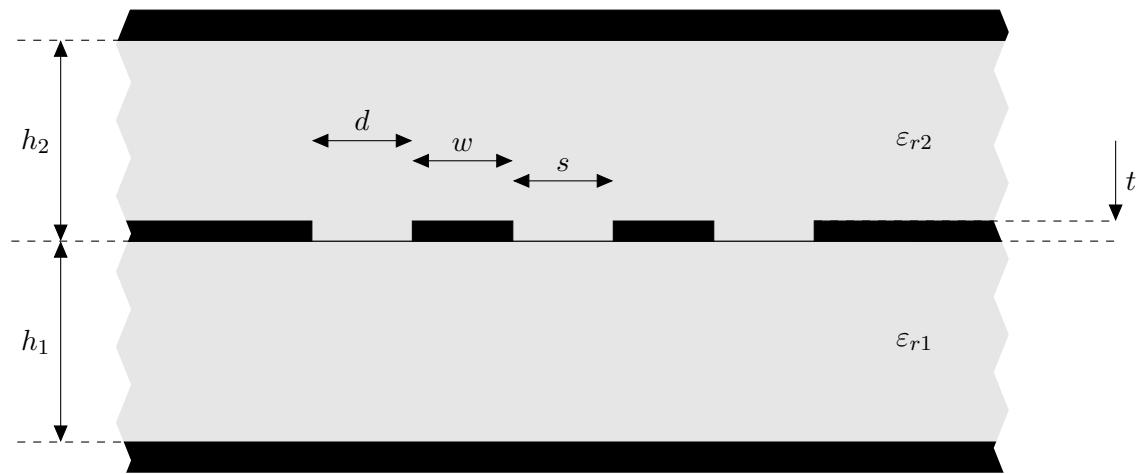


Figure 4.20.: Differential Offset Coplanar Waveguide

4.3. Production

5. Selecting and Analyzing the Back-End Readout Card

The back-end readout card for the system under development, the Zynq UltraScale+ RFSoC ZCU216 Evaluation Card, was chosen taking into consideration the points described in section 3.4. In this section, the overall architecture and features of the card are presented. A possibility for evaluation of the card is also demonstrated. At last, a design for the read-out firmware is proposed.

5.1. Xilinx Zynq UltraScale+ RFSoC ZCU216 Evaluation Card

Zynq UltraScale+ RFSoCs: Combine RF data converter subsystem and forward error correction with industry-leading programmable logic and heterogeneous processing capability. Integrated RF-ADCs, RF-DACs, and soft decision FECs (SD-FEC) provide the key subsystems for multiband, multi-mode cellular radios and cable infrastructure

With the data converters integrated directly into the FPGA using parallel interfaces, they don't require the prohibitively high-pin-count external connections needed for discrete parallel interface converters, allowing more converter

- Sixteen 14-bit, 2.5GSPS RF-ADC
- Sixteen 14-bit, 10GSPS RF-DAC
- I/O expansion options – FPGA Mezzanine Card (FMC+) interfaces, RFMC 2.0 interfaces, and Pmod connections

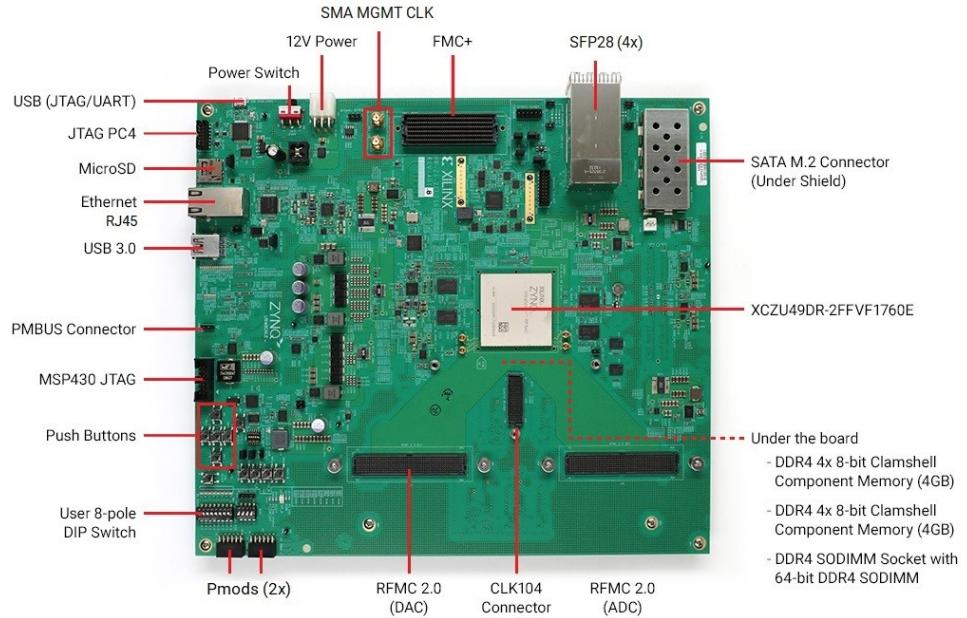


Figure 5.1.: ZCU216 evaluation board

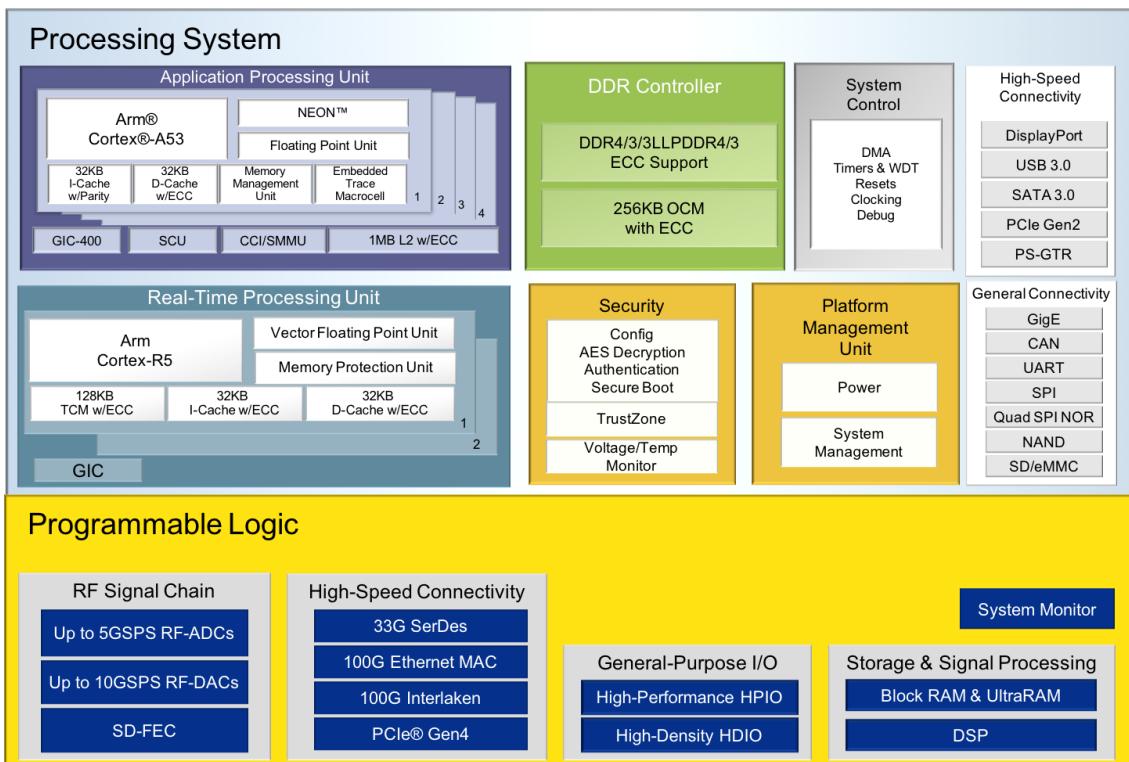


Figure 5.2.: RFSoC block diagram

5.2. Features

5.3. Evaluation Tool

5.4. Firmware

5.4.1. RF Data Converter

5.4.2. SoC

5.4.3. RDMA over Converged Ethernet (RoCE)

5.4.4. System Integration

Interleaving

The necessary step size for the delay chips, when using 16 ADC@2 GS/s in time-interleaving mode, is: $\frac{2 \text{ GS/s}}{16} = 31 \text{ ps}$ However, providing individual clocks to the ADCs is not possible on the ZCU216 card. ADCs are grouped together into tiles, each tile containing four converters. One single reference clock signal is propagated to all tiles. Sampling clock is adjusted at each tile individually, however this clocking signal is the same for all of the four converters in the tile.

Clock to THA: 500 MHz

Total Hold time: 1 ns

→ Step size for delay:

$$\frac{1 \text{ ns}}{16 \text{ channels}} = 62.5 \text{ ps} \quad (5.1)$$

6. Results

7. Conclusion and Outlook

7.1. Measurements with the developed front-end card

Card is in production. Measurements will be done, but not in the scope of the thesis.

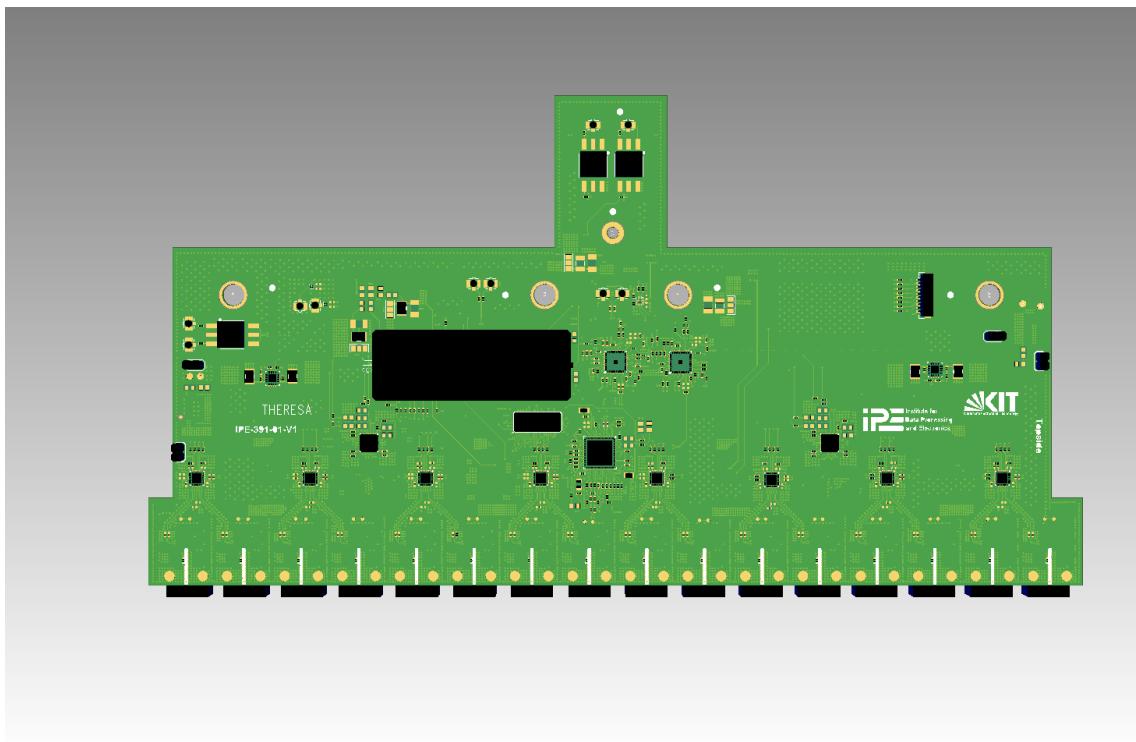


Figure 7.1.: wow

Acknowledgments

Appendix

A. Characteristic Impedance Of Coplanar Waveguides

Edge-Coupled Coplanar Waveguide

Characteristic impedance[?, p197-198]:

$$Z_{0,o} = \frac{\eta_0}{\sqrt{\epsilon_{eff,o}}} \left(\frac{1.0}{2.0 \frac{K(k_o)}{K'(k_o)} + \frac{K(\beta_1)}{K'(\beta_1)}} \right) \quad (\text{A.1})$$

$$Z_{0,e} = \frac{\eta_0}{\sqrt{\epsilon_{eff,e}}} \left(\frac{1.0}{2.0 \frac{K(k_e)}{K'(k_e)} + \frac{K(\beta_1 k_1)}{K'(\beta_1 k_1)}} \right) \quad (\text{A.2})$$

$$\epsilon_{eff,o} = \frac{2.0 \epsilon_r \frac{K(k_o)}{K'(k_o)} + \frac{K(\beta_1)}{K'(\beta_1)}}{2.0 \frac{K(k_o)}{K'(k_o)} + \frac{K(\beta_1)}{K'(\beta_1)}} \quad (\text{A.3})$$

$$\epsilon_{eff,e} = \frac{2.0 \epsilon_r \frac{K(k_e)}{K'(k_e)} + \frac{K(\beta_1 k_1)}{K'(\beta_1 k_1)}}{2.0 \frac{K(k_e)}{K'(k_e)} + \frac{K(\beta_1 k_1)}{K'(\beta_1 k_1)}} \quad (\text{A.4})$$

Where

$$k_o = \Lambda \frac{-\sqrt{\Lambda^2 - t_c^2} + \sqrt{\Lambda^2 - t_B^2}}{t_B \sqrt{\Lambda^2 - t_c^2} + t_c \sqrt{\Lambda^2 - t_B^2}} \quad (\text{A.5})$$

$$k_e = \Lambda' \frac{-\sqrt{\Lambda'^2 - t_c'^2} + \sqrt{\Lambda'^2 - t_B'^2}}{t_B' \sqrt{\Lambda'^2 - t_c'^2} + t_c' \sqrt{\Lambda'^2 - t_B'^2}} \quad (\text{A.6})$$

$$\Lambda = \frac{\sinh^2 \left(\frac{\pi(s/2.0+w+d)}{2.0h} \right)}{2} \quad (\text{A.7})$$

$$t_c = \sinh^2 \left(\frac{\pi(s/2.0+w)}{2.0h} \right) - \Lambda \quad (\text{A.8})$$

$$t_B = \sinh^2 \left(\frac{\pi s}{4.0h} \right) - \Lambda \quad (\text{A.9})$$

$$\Lambda' = \frac{\cosh^2 \left(\frac{\pi(s/2.0+w+d)}{2.0h} \right)}{2} \quad (\text{A.10})$$

$$t'_c = \sinh^2 \left(\frac{\pi(s/2.0 + w)}{2.0h} \right) - \Lambda' + 1.0 \quad (\text{A.11})$$

$$t'_B = \sinh^2 \left(\frac{\pi s}{4.0h} \right) - \Lambda + 1.0 \quad (\text{A.12})$$

The parameters have to be chosen according to

$$s + 2.0w + 2.0d \leq h \quad (\text{A.13})$$

to guarantee coplanar propagation. [?]

Surface Coplanar Waveguide with Ground

The characteristic impedance of a coplanar waveguide is given as follows [?]:

$$Z_0 = \frac{60.0\pi}{\sqrt{\epsilon_{eff}}} \frac{1.0}{\frac{K(k)}{K(k')} + \frac{K(k_1)}{K(k'_1)}} \quad (\text{A.14})$$

It comprises of the following components, with $K(k)$ being an elliptical integral of the first kind (see also [?, p. 430]):

$$k = a/b \quad (\text{A.15})$$

$$k' = \sqrt{1.0 - k^2} \quad (\text{A.16})$$

$$k'_1 = \sqrt{1.0 - k_1^2} \quad (\text{A.17})$$

$$k_1 = \frac{\tanh(\frac{\pi a}{4.0h})}{\tanh(\frac{\pi b}{4.0h})} \quad (\text{A.18})$$

$$\epsilon_{eff} = \frac{1.0 + \epsilon_r \frac{K(k')}{K(k)} \frac{K(k_1)}{K(k'_1)}}{1.0 + \frac{K(k')}{K(k)} \frac{K(k_1)}{K(k'_1)}} \quad (\text{A.19})$$

B. QuickStart Guide for Evaluation of ZCU216 Board

C. 3D model of front-end card

D. Code

```
'timescale 1ns / 1ps

module SDI_Delay_NB6L295(
    input [10:0]           In_1, In_2, In_3, In_4, In_5, In_6, In_7, In_8, // 
                           data for respective delay chips
    input                  Clk,
    input                  Reset,
    output reg [7:0]        EN, // enable signal for delay chips, active LOW
    output reg              SDIN, // configuration data
    output reg              SLOAD, // signals delay chip to load previously sent
                           data
    output                 SCLK // clock for serial communication with delay chips
);

reg                      start_clk;
assign SCLK = start_clk & (!Clk);
```

```

reg [21:0] In_1_reg, In_2_reg, In_3_reg, In_4_reg, In_5_reg,
In_6_reg, In_7_reg, In_8_reg; // registers to intermediately store the
inputs

reg [7:0] select; // register used by Priority Encoder to detect
which input changed

parameter DATA_SHIFT_WIDTH = 11; // number of bits to be shifted
during transmission, 1 Data word = 11 bits
reg [4:0] clk_cnt;

reg [DATA_SHIFT_WIDTH-1:0] Data_reg; // register for storing data for
state machine

reg start; // signal for state machine to start sending
reg data
reg dataSent; // flags if transmission for one delay chip
is finished

parameter dly = 1; // delay control

reg delayReady;

always @ (posedge Clk)
begin
    if (select == 'd0) delayReady <= #dly 'b1;
    else delayReady <= #dly 'b0;
end

// Priority Encoder
// Check if any input has changed, select which data should be sent
accordingly
always @ (posedge Clk)
begin
    if (Reset)
        begin
            In_1_reg <= #dly 'd0;
            In_2_reg <= #dly 'd0;
            In_3_reg <= #dly 'd0;
            In_4_reg <= #dly 'd0;
            In_5_reg <= #dly 'd0;
            In_6_reg <= #dly 'd0;
            In_7_reg <= #dly 'd0;
            In_8_reg <= #dly 'd0;
            Data_reg <= #dly 'd0;

            select <= #dly 'd0;

            start <= #dly 1'b0;;
        end
    else
        begin
            if (~start & delayReady)
                begin
                    select[7] <= #dly In_1_reg != In_1;
                    select[6] <= #dly In_2_reg != In_2;
                    select[5] <= #dly In_3_reg != In_3;
                    select[4] <= #dly In_4_reg != In_4;
                    select[3] <= #dly In_5_reg != In_5;
                    select[2] <= #dly In_6_reg != In_6;
                    select[1] <= #dly In_7_reg != In_7;
                    select[0] <= #dly In_8_reg != In_8;
                end
        end
    end

```

```

        end
    else
        begin
            if (clk_cnt == 4'd12 & ~start_clk) // = end of
                sequence
                    start          <= #dly 1'b0;
            else
                start          <= #dly 1'b1;
        end

    casex (select)
        8'b1???????: begin
            if (~dataSent)
                begin
                    In_1_reg      <= #dly In_1;
                    Data_reg       <= #dly In_1;
                    EN             <= #dly
                        8'b01111111;
                    start          <= #dly 1'b1;
                end
            else
                begin
                    start          <= #dly 1'b0;
                    select [7]     <= #dly 1'b0;
                end
        end
        8'b01???????: begin
            if (~dataSent)
                begin
                    In_2_reg      <= #dly In_2;
                    Data_reg       <= #dly In_2;
                    EN             <= #dly
                        8'b10111111;
                    start          <= #dly 1'b1;
                end
            else
                begin
                    select [6]     <= #dly 1'b0;
                    start          <= #dly 1'b0;
                end
        end
        8'b001?????: begin
            if (~dataSent)
                begin
                    In_3_reg      <= #dly In_3;
                    Data_reg       <= #dly In_3;
                    EN             <= #dly
                        8'b11011111;
                    start          <= #dly 1'b1;
                end
            else
                begin
                    select [5]     <= #dly 1'b0;
                    start          <= #dly 1'b0;
                end
        end
        8'b0001????: begin
            if (~dataSent)
                begin
                    In_4_reg      <= #dly In_4;

```

```

Data_reg          <= #dly In_4;
EN               <= #dly
8'b11101111;    <= #dly 1'b1;
start            <= #dly 1'b1;
end

else
begin
  select [4]      <= #dly 1'b0;
  start           <= #dly 1'b0;
end
end
8'b00001???: begin
if (~dataSent)
begin
  In_5_reg        <= #dly In_5;
  Data_reg         <= #dly In_5;
  EN               <= #dly
  8'b11110111;    <= #dly 1'b1;
  start            <= #dly 1'b1;
end

else
begin
  select [3]      <= #dly 1'b0;
  start           <= #dly 1'b0;
end
end
8'b000001???: begin
if (~dataSent)
begin
  In_6_reg        <= #dly In_6;
  Data_reg         <= #dly In_6;
  EN               <= #dly
  8'b11111011;    <= #dly 1'b1;
  start            <= #dly 1'b1;
end

else
begin
  select [2]      <= #dly 1'b0;
  start           <= #dly 1'b0;
end
end
8'b0000001?: begin
if (~dataSent)
begin
  In_7_reg        <= #dly In_7;
  Data_reg         <= #dly In_7;
  EN               <= #dly
  8'b11111101;    <= #dly 1'b1;
  start            <= #dly 1'b1;
end

else
begin
  select [1]      <= #dly 1'b0;
  start           <= #dly 1'b0;
end
end
8'b00000001?: begin

```

```

        if (~dataSent)
        begin
            In_8_reg           <= #dly In_8;
            Data_reg           <= #dly In_8;
            EN                 <= #dly
                                8'b11111110;
            start              <= #dly 1'b1;
        end
        else
            begin
                select [0]      <= #dly 1'b0;
                start             <= #dly 1'b0;
            end
        end
    default:
        begin
            EN                 <= #dly
                                8'b11111111;
            start              <= #dly 1'b0;
        end
    endcase
end
/*
// State Machine for Sending Configuration Data to Delay Chip NB6L295
/*
   State          Description
   -----
RESET           Resetting all parameters and registers ->
               if (reset): stay; else: to IDLE
IDLE            Waiting for start signal from priority
               encoder -> if (start): to LOAD_P0; else: stay
LOAD_P0          Load first half of Delay_X - which
               corresponds to data for Delay PD0 on delay chip - into
               temporary register -> to LOAD_P1
LOAD_P1          Load second half of Delay_X - which
               corresponds to data for Delay PD1 on delay chip - into
               temporary register -> to SHIFT
SHIFT            Shift bits for sending serial bitstream to
               SDIN, assert SLOAD -> to END
END              End transmission, deassert SLOAD, inform
               priority encoder about end of transmission -> to IDLE
*/
parameter RESET      = 3'd0;
parameter IDLE       = 3'd1;
parameter LOAD        = 3'd2;
parameter SHIFT       = 3'd3;
parameter END         = 3'd4;
reg [2:0] STATE;
reg [DATA_SHIFT_WIDTH-1:0]     tmp;

always @ (posedge Clk)
begin
    if (Reset)
        begin
            STATE      <= #dly RESET;
            tmp        <= #dly 'd0;
            dataSent   <= #dly 1'b0;
            start_clk  <= #dly 1'b0;
            SLOAD      <= #dly 1'b0;
            clk_cnt    <= #dly 1'b0;
        end
end

```

```

else
begin
  case (STATE)
    RESET:
      begin
        if (Reset)
          STATE    <= #dly RESET;
        else
          STATE    <= #dly IDLE;
      end // RESET
    IDLE:
      begin
        SDIN      <= #dly 1'b0;
        clk_cnt   <= #dly 5'd0;
        dataSent  <= #dly 1'b0;
        SLOAD     <= #dly 1'b0;

        if (start & ~dataSent)
          STATE    <= #dly LOAD;
        else
          STATE    <= #dly IDLE;
      end // IDLE
    LOAD:
      begin
        tmp       <= #dly Data_reg;
        STATE    <= #dly SHIFT;
      end // LOAD_W1
    SHIFT:
      begin
        if (clk_cnt < 4'd12) // number of bits to be
        shifted //
        begin
          start_clk    <= #dly 1'b1;
          clk_cnt      <= #dly clk_cnt +1;
          tmp         <= #dly
                      {tmp[DATA_SHIFT_WIDTH-2:0], 1'b0};
          SDIN        <= #dly
                      tmp[DATA_SHIFT_WIDTH-1];
        end
        else
          begin
            SLOAD      <= #dly 1'b1;
            clk_cnt    <= #dly
                        clk_cnt;
            start_clk  <= #dly 1'b0;
            STATE      <= #dly END;
            SDIN        <= #dly 1'b0;
          end
      end // SHIFT
    END:
      begin
        SLOAD      <= #dly 1'b0;
        start_clk  <= #dly 1'b0;
        dataSent   <= #dly 1'b1;
        clk_cnt    <= #dly clk_cnt;
        SDIN        <= #dly 1'b0;
        STATE      <= #dly IDLE;
      end // END
    default:
      STATE    <= #dly RESET;
  endcase
end

```

endmodule