

# Comparing HLS with VHDL at the example of an FM Radio Receiver

Michael Wurm



## MASTERARBEIT

eingereicht am  
Fachhochschul-Masterstudiengang

Embedded Systems Design

in Hagenberg

im Juli 2021

Advisor:

DI (FH) Dr. Florian Eibensteiner

© Copyright 2021 Michael Wurm

This work is published under the conditions of the Creative Commons License *Attribution-NonCommercial-NoDerivatives 4.0 International* (CC BY-NC-ND 4.0)—see <https://creativecommons.org/licenses/by-nc-nd/4.0/>.

# Declaration

I hereby declare and confirm that this thesis is entirely the result of my own original work. Where other sources of information have been used, they have been indicated as such and properly acknowledged. I further declare that this or similar work has not been submitted for credit elsewhere. This printed copy is identical to the submitted electronic version.

Hagenberg, July 15, 2021

Michael Wurm

# Contents

<b>Declaration</b>	<b>iv</b>
<b>Preface</b>	<b>viii</b>
<b>Abstract</b>	<b>ix</b>
<b>Kurzfassung</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Broadcast Radio . . . . .	1
1.3 Goal . . . . .	1
<b>2 Signal Processing Theory</b>	<b>3</b>
2.1 Modulation from Baseband to RF . . . . .	3
2.2 Modulation from RF to Baseband . . . . .	3
2.3 Oversampling / Sample Rate Reduction . . . . .	3
2.4 Frequency Modulation in Broadcasting . . . . .	3
2.4.1 Mathematical description . . . . .	3
2.4.2 Frequency Band . . . . .	4
2.4.3 Frequency Spectrum of a Broadcast FM Channel . . . . .	5
2.5 Algorithms for Digital FM Demodulation . . . . .	6
2.5.1 Frequency Discriminator . . . . .	7
2.5.2 Phase-Locked Loop . . . . .	9
2.6 Demodulation of Broadcast FM . . . . .	9
2.6.1 Mono Audio . . . . .	9
2.6.2 Stereo Audio . . . . .	9
<b>3 HLS</b>	<b>11</b>
3.1 Introduction / State of the Art . . . . .	11
3.2 Functionality (transform high-level code to HDL) . . . . .	11
3.3 Language Support . . . . .	11
3.3.1 C++ . . . . .	11
3.3.2 SystemC . . . . .	11
3.4 Coding . . . . .	11
3.4.1 Compiler Directives (#pragma's) . . . . .	11

3.4.2	Data types . . . . .	11
3.4.3	Functions . . . . .	11
3.4.4	Loops . . . . .	11
3.4.5	Conditional statements . . . . .	11
3.5	Advantages / Disadvantages . . . . .	11
<b>4</b>	<b>System Architecture/Concept</b>	<b>12</b>
4.1	Block Diagram (with details) . . . . .	12
4.1.1	describe blocks . . . . .	12
4.2	Which parts in HLS and VHDL . . . . .	12
4.3	Test Environment . . . . .	12
<b>5</b>	<b>Implementation</b>	<b>13</b>
5.1	Matlab/Python Model . . . . .	13
5.2	VHDL (no shortcut) . . . . .	13
5.2.1	Channel Selection (IF to Channel-BB) . . . . .	13
5.2.2	Phase Detector . . . . .	13
5.2.3	other Elements . . . . .	13
5.3	High-Level Synthesis . . . . .	13
5.3.1	Channel Selection (IF to Channel-BB) . . . . .	13
5.3.2	Phase Detector . . . . .	13
5.3.3	other Elements . . . . .	13
5.4	Common Testbench . . . . .	13
5.4.1	Architecture (same tb for VHDL and HLS-generated HDL) . . . . .	13
5.4.2	Framework cocotb, with ghdl compiler . . . . .	13
5.5	Develop HLS compiler independent code (optional) . . . . .	14
5.5.1	Challenges . . . . .	14
<b>6</b>	<b>Deployment on Hardware</b>	<b>15</b>
6.1	Hardware Platform . . . . .	15
6.1.1	RTL2832u Dongle . . . . .	15
6.1.2	ZedBoard . . . . .	15
<b>7</b>	<b>Test Results</b>	<b>16</b>
7.1	Functionality . . . . .	16
7.1.1	Implementation effort/time . . . . .	16
7.1.2	Hardware Utilization . . . . .	16
7.1.3	Others . . . . .	16
<b>8</b>	<b>Using Literature</b>	<b>17</b>
<b>A</b>	<b>Technical Details</b>	<b>18</b>
<b>B</b>	<b>Supplementary Materials</b>	<b>19</b>
B.1	PDF Files . . . . .	19
B.2	Media Files . . . . .	19
B.3	Online Sources (PDF Captures) . . . . .	19

Contents	vii
<b>C Questionnaire</b>	<b>20</b>
<b>D LaTeX Source Code</b>	<b>21</b>
<b>References</b>	<b>22</b>
Literature . . . . .	22
Online sources . . . . .	22

# Preface



# Abstract

This should be a 1-page (maximum) summary of your work in English.

# Kurzfassung

An dieser Stelle steht eine Zusammenfassung der Arbeit, Umfang max. 1 Seite. ...

# Chapter 1

## Introduction

### 1.1 Motivation

FM is a common RF signal that's available everywhere.

The demodulated signal is an audio signal, that can be listened to. This is subjectively more attractive than a generic data stream.

### 1.2 Broadcast Radio

Evolution from AM to FM because of certain advantages, etc.

Explain FM usage/existence nowadays (geographical; frequencies; devices; new standard DAB, etc)

### 1.3 Goal

put all knowledge that was accumulated during studies (HSD, ESD, mention subjects such as DSP) together in a project.

practical usage of DSP in an FPGA

develop an FM receiver and explain it, so that it can be followed in a tutorial with an affordable budget in hardware.

show multiple ways of how to implement the same thing, in different levels of abstraction.

Weigh each ways' field of application, efficiency and applicability.

Thesis Design Decisions:

Main focus on system design.

How to achieve the same result in 3 (4?) different levels of abstraction?

1. GnuRadio (?)
2. Matlab/Python
3. HLS

Implementation targeting Xilinx' ZedBoard.

Optionally:

Intel as a comparison (synth only, no HW) - “How platform-independent is HLS?”

4. VHDL

Compare: - Implementation effort (difficulty) - Implementation time - Usefulness on Target Systems (uC, SoC, RasPi, etc) - others?

## Chapter 2

# Signal Processing Theory

### 2.1 Modulation from Baseband to RF

### 2.2 Modulation from RF to Baseband

### 2.3 Oversampling / Sample Rate Reduction

Example:

In case the multiplex signal needs to be stored in a file with 44.1 kHz samplerate, in order to avoid aliasing, a lowpass with a cutoff frequency of 15 kHz and sufficient attenuation before 19 kHz needs to be inserted before the decimation and file output.

### 2.4 Frequency Modulation in Broadcasting

Frequency modulation (FM) is a widely used standard to transmit data streams. The probably best known usecase therefor is commercial broadcast radio, where an audio stream is transmitted. Devices to receive these streams are available for low prices to the public.

This chapter describes the main properties of broadcast FM, such as the mathematical description, frequency bands that are used, or the specific frequency parts within a channel spectrum.

#### 2.4.1 Mathematical description

The mathematical description of an FM baseband signal can be described with the formulas presented in this section. FM encodes the information content in its instantaneous frequency. This means that the measured frequency at any moment in time represents a specific value of a transmitted message. For a general classification, FM belongs to the group of angle, or phase modulated signals (PM). The simple reason therefor is, that a frequency has a direct relationship to an angle, if the signal is seen on a unit circle.

The instantaneous frequency  $f_i$  of an FM signal can be described as

$$f_i = f_c + \Delta f \cdot m(t) \tag{2.1}$$

where  $f_c$  is the carrier- or center frequency,  $\Delta f$  is the maximum frequency deviation and  $m(t)$  is the information or message signal that is to be transmitted. Simply looking at this equation, the frequency deviation, which is sometimes also called the swing, varies in the range between  $f_c \pm \Delta f$ .

Considering the relationship between frequency and angle, as described above, and after some simple substitutions which will not be described into detail here, the equation for a generic frequency modulated wave is the following

$$s(t) = A_c \cos\left(2\pi f_c t + 2\pi k_f \int m(t) dt\right) \quad (2.2)$$

where  $A_c$  is the amplitude of the resulting FM signal and  $k_f$  is the frequency sensitivity factor. This sensitivity factor has a direct relationship with the modulation index  $\beta$ .

$$\beta = \frac{\Delta f}{f_m} = \frac{k_f A_m}{f_m} \quad (2.3)$$

where  $f_m$  is the highest existing frequency, or the bandwidth of the information signal.

Formula (2.2) describes a frequency modulated signal with a generic message signal  $m(t)$ . A widely used example application of FM is broadcast radio, where audio streams are transmitted. An audio stream can be described as a cosine wave. Therefore, the information signal that is to be transmitted in broadcast radio can be represented as a cosine wave in the form of

$$m(t) = A_m \cos(2\pi f_m t). \quad (2.4)$$

By inserting this message signal into the generic FM formula (2.2), the final equation for an FM signal transforms into the following form.

$$y(t) = A_c \cos\left(2\pi f_c t + \beta \sin(2\pi f_m t)\right) \quad (2.5)$$

Several formulas, as well as their derivations are described according to [ref\_FM\_Maths\_Info\_1][ref\_F

### 2.4.2 Frequency Band

The frequency band that is used for FM broadcasting is defined worldwide and spans from 87.5 to 108 MHz. However, some countries only partially use the band [ref\_itu\_regulations]. The range is located within the so-called Very-High-Frequency (VHF) band, which is open to the public in terms of usage. This means, that any transmitter may use the specified frequency range freely. Austria allowed the legal usage in 2006 [ref\_austria\_rundfunkgesetz\_2006]. However, the transmission power needs to be limited, so that neighboring receivers are not disturbed in receiving any existing channels. The European Commission specifies the power limit as 50 nW of effective radiated power (ERP) [ref\_eu\_commission\_radio\_spectrum]. Another limitation is, that a transmitter must be capable to transmit on multiple center frequencies within the broadcasting band. It must not be fixed to a single center frequency. Through this specification, a transmitter is capable of switching its transmission center frequency, in order to avoid interference with another transmitters signal.

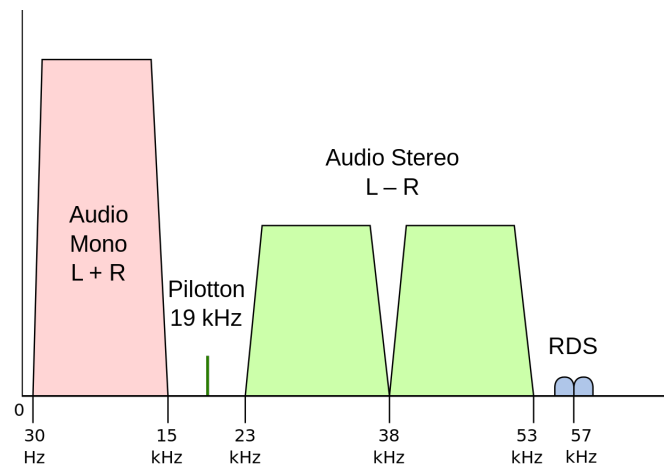
In a practical usecase this means that a transmitter needs to select a channel or frequency range, that is not already occupied by an official transmitter. If a free range is found, the transmitter may start sending an FM signal with the mentioned power limitations [ref\_ebu\_fm\_modulators\_in\_europe\_power\_regulations].

### 2.4.3 Frequency Spectrum of a Broadcast FM Channel

The entire FM broadcasting band from 87.5 to 108 MHz divides into multiple channels which can be used. In Europe, the European Telecommunications Standards Institute (ETSI) sets the respective standards for the usage of these frequencies. The main specifications for a single FM broadcasting channel are explained in this section.

Each channel may allocate a bandwidth of 200 kHz. Within one of these channels, the maximum deviation from the center frequency shall not exceed 75 kHz. This leaves a guard band of 25 kHz on either side, to minimize interference with adjacent channels. Center frequencies may be allocated on multiples of 100 kHz. However, neighboring channels need to be considered therefore, since this may result in an overlap in the spectrum.

Fig.2.1 shows the frequency spectrum of a single channel, located at its carrier- or center frequency. For a better overview, only the positive spectrum part is shown. The negative part is an exact mirror, because the transmitted and received signals from the antenna are real signals - they do not have an imaginary part. The spectrum consists of multiple parts. Therefore, it is also referred to as ‘multiplex signal’ in the literature. Several parts are described in the following paragraphs.



**Figure 2.1:** Allocation of frequencies in an FM channel [ref\_fig\_channel\_freqs].

#### Mono Audio Part

The mono audio stream is located between 30 Hz and 15 kHz. This signal is built by the sum of the left and right audio channels. The upper limit at 15 kHz is chosen to maintain a sufficient spacing to the first subcarrier.

For audio streaming, as in FM broadcasting, this is not a limitation, since it is unlikely to have frequencies higher than 15 kHz in an audio stream. Also, this already reaches the upper limit of the human ears' bandwidth, so higher spectral parts will not be heard anyway.

### Pilot Tone

The first subcarrier is allocated at an offset of 19 kHz from the center frequency. This subcarrier is also called the 'pilot tone', since it is a continuous signal. It is independent of any varying message signal content. The pilot tone is used for stereo audio demodulation. To regenerate a stereo audio signal, the left and right audio channels need to be recovered correctly.

### Stereo Audio Difference

A signal that is constructed by the difference between the left and right audio channel is modulated on a 38 kHz subcarrier. This subcarrier is an integer multiple of the pilot tone at 19 kHz, for practical reasons. However, the 38 kHz carrier is suppressed and thus not visible in the received spectrum. The modulation technique that is used for this spectral part is called dual-sideband suppressed-carrier (DSB-SC). Even though the carrier is suppressed, it can still be recovered, because it is phase coherent with the 19 kHz pilot tone per definition.

The bandwidth for this difference-signal spans 15 kHz on either side of the subcarrier. It is used to generate a stereo audio signal, in combination with the mono signal. This process is explained into more detail in chapter ??.

### Additional Services

Considering all these parts in the spectrum, there is still free bandwidth available to use up to the maximum channel bandwidth of 100 kHz in one sideband. Because of that, additional services were added to the pure audio transmission, to provide additional data services and information.

Services that were implemented are the Data Radio Channel (DARC), which is mostly used in Japan and the USA, the Subsidiary Communication Authorization (SCA) and the Radio Data System (RDS) [ref\_rohde\_u\_schwarz]. Out of these, RDS is the most significant service in Europe. It is used to transmit additional information about the channel, such as the radio stations' name, the currently playing songs' title or traffic information.

The information about the frequency spectrum was taken from [ref\_fm\_broadcast\_tutorial\_and\_bas

## 2.5 Algorithms for Digital FM Demodulation

An FM signal that is received, for example from an antenna, needs to be demodulated in order to decode its actual data content. In general, an FM demodulator serves the purpose to transform a frequency modulated signal (FM) into an amplitude modulated signal (AM), so that AM signal processing techniques can be applied afterwards.

The radio frequency antenna signal usually needs to be amplified, bandpass-filtered, and down-converted to baseband, using subsampling and quadrature-mixing, or similar strategies. This part of the signal processing chain is assumed to be working correctly and is out of the scope of this chapter. The FM signal that is evaluated here is assumed

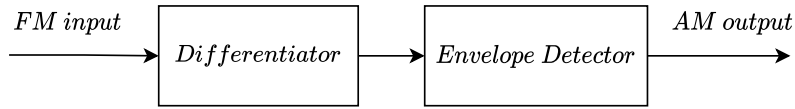


to be a quadrature-mixed signal in baseband, which means that inphase and quadrature (I/Q) signals are available.

In the following sections, two digital FM demodulator variants are described.

### 2.5.1 Frequency Discriminator

A frequency discriminator generates an output that is directly proportional to the frequency of the input FM signal. In other words, it converts FM to AM. The discriminator consists of two main parts - a differentiator and an envelope detector. Different strategies can be implemented for both parts. A block diagram for a frequency discriminator with its two main parts is shown in Fig.2.2.



**Figure 2.2:** Frequency discriminator block diagram [ref\_fig\_freq\_discriminator].

#### Differentiator

The differentiator already converts the FM signal into an AM signal, but still leaves an FM content. The transformation that happens here can be expressed in a formula, by simply differentiating Eq.(2.2). This operation, with some additional simple algebraic transformations, results in

$$\frac{ds(t)}{dt} = A_c 2\pi \left[ f_c + k_f m(t) \right] \sin \left( \omega_c t + 2\pi k_f \int m(t) dt - \pi \right) \quad (2.6)$$

This formula shows the AM and an FM contents clearly. The FM part is described by the sine function, with its argument. The more important part is the AM part, which resembles the message signal. It is described by the term within the square brackets.

A time-domain example for this signal is shown in the top left diagram in Fig.2.3.

An important requirement for the differentiator is, that the input signal is of a constant amplitude [ref\_schnyder\_haller]. The transmitted signal is subject to random additive noise on the transmission channel. The differentiator would deliver wrong results because of this noise, when subtracting two consecutive samples, as described in Eq.(2.9). To achieve a constant amplitude, the received complex baseband signal  $s(t)$  can be normalized to a value of one.

$$s_{norm} = \frac{s}{|s|} = \frac{A(n) e^{j\phi_{FM}(n)}}{|A(n) e^{j\phi_{FM}(n)}|} = e^{j\phi_{FM}(n)} \quad (2.7)$$

$$e^{j\phi_{FM}(n)} = |1| \quad (2.8)$$

In a hardware implementation, a differentiation is simply performed by subtracting two consecutive samples, like

$$\frac{ds(t)}{dt} = s(t) - s(t - \Delta t) \quad (2.9)$$

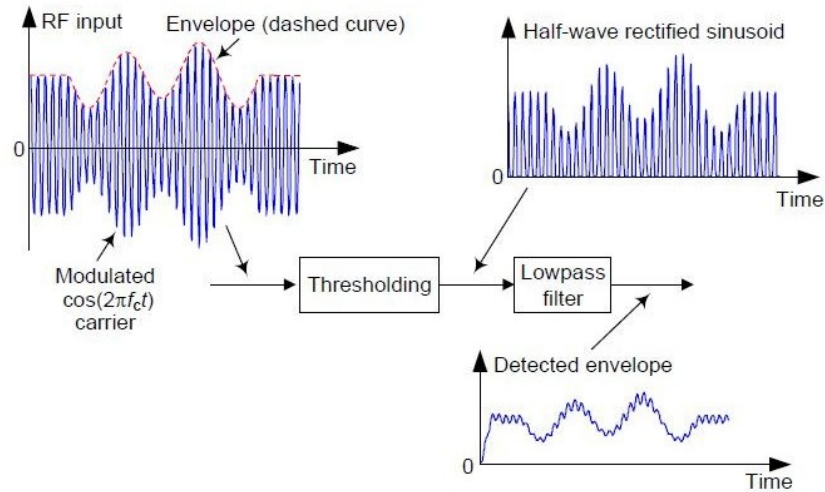
where  $\Delta t$  is the inverse of the sample frequency.

### Envelope Detector

The differentiated signal still consists of a FM part, as explained above, using Eq.(2.6). In order to remove this high-frequency part and extract the low-frequency envelope, the signal needs to be fed into an envelope detector. A block diagram is shown in Fig.2.2.

The method implemented in Fig.2.3 is one of the most simple ones and is usually referred to as ‘Asynchronous Half-Wave Envelope Detector’[[ref\\_envelope\\_detector](#)]. It consists of a thresholding unit to remove negative values and a conventional lowpass filter. In analog implementations, the thresholding unit is a diode. The cutoff frequency of the lowpass needs to be adapted to the envelope signals’ maximum frequency. It needs to be able to follow the signal, but also sufficiently smooth out the rectified signal.

Different methods may be implemented for the envelope detector. For example, an ‘Asynchronous Full-Wave Envelope Detector’ can be used. Therefore, the previous method only needs to swap the thresholding unit with an unit that calculates the absolute value. In that architecture, a full-wave rectification is performed on the signal, which significantly improves the envelope detection accuracy [[ref\\_envelope\\_detector](#)]. The envelope is followed more exactly, because of the higher power that is available in the signal, since the entire signal is taken into account, and not only the positive half, as in the previous half-wave method.



**Figure 2.3:** Time-domain signals in envelope detection [[ref\\_roppel](#)].

### 2.5.2 Phase-Locked Loop

A Phase-Locked Loop (PLL) can also be used to transform an FM signal to an AM signal. A PLL is a feedback loop that is usually used to generate an output signal that has a fixed phase reference to its input. In the case of FM demodulation, the loop filter in the feedback branch is configured to be able to follow the frequency variations on the input, which is directly related to the encoded information. The output of the PLL directly delivers an AM signal, which corresponds to the transmitted information. [ref\_schnyder\_haller]

FM demodulation with a PLL is not described into more detail, since this thesis is focussing on the concept of an entire system architecture.

## 2.6 Demodulation of Broadcast FM

This chapter describes the demodulation of the information content of an FM broadcast channel, as it is used in commercial FM radio broadcasting. The assumption here is, that the actual FM demodulation, as described in chapter 2.5 is already done correctly. A frequency spectrum of the multiplex signal of an FM channel is shown in Fig.2.1.

### 2.6.1 Mono Audio

The first part of the channel frequency spectrum is the mono audio signal. It is the summation signal of the left and right audio channel. Thus, it contains enough information to replay an audio stream in mono quality. A mono receiver can thus simply replay the so-called multiplex signal that is produced by the FM demodulator, without any further filtering or processing. The 19 kHz pilot tone, and any higher frequency components, will not be audible by most people, because it is outside of the range of a human ear. Besides that, any used speaker may also be unable to produce a frequency in this range.

Consequently, in order to demodulate this spectral part, a lowpass filter is required. The filter needs to have a cutoff frequency of 15 kHz, to allow the entire mono audio spectrum to pass. Sufficient attenuation must be available before 19 kHz, in order to suppress the pilot tone.

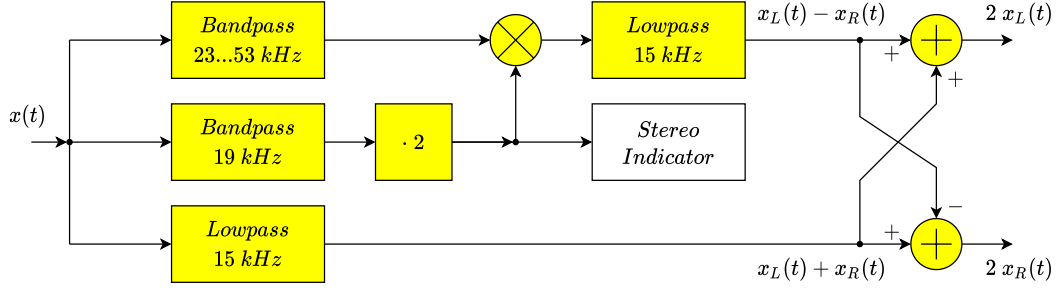
### 2.6.2 Stereo Audio

The demodulation of an FM channel to recover a stereo audio signal, requires a more sophisticated approach and multiple steps in signal processing. Two parts need to be combined in order to achieve stereo. The sum and the difference of the left and right audio channel signals - the mono and the stereo difference part, respectively. The following equations need to be applied, to compute a left and a right channel.

$$\begin{aligned}(L + R) + (L - R) &= (2)L \\ (L + R) - (L - R) &= (2)R\end{aligned}\tag{2.10}$$

where  $(L + R)$  represents the mono part and  $(L - R)$  the stereo difference.

The block diagram in Fig.2.4 illustrates the signal processing chain for stereo audio. In the block diagram,  $x(t)$  represents the multiplex signal from the FM demodulator. The signals  $x_L(t)$  and  $x_R(t)$  describe the left and right audio signal, respectively.



**Figure 2.4:** Block diagram of FM stereo audio demodulation [ref\_roppel].

The central branch selects the pilot tone at 19 kHz with a bandpass filter. This bandpass needs to have a frequency response that is sharp enough to have a sufficient attenuation at  $\pm 4$  kHz, since this is where the mono and difference signals' frequency spectra are allocated. Afterwards, the pilot tone frequency is doubled to achieve a 38 kHz subcarrier. This guarantees a phase-coherency between pilot and subcarrier, which is required to correctly shift the difference signal down to baseband.

The frequency doubling can be implemented with multiple methods. The pilot tone can be modulated by simply multiplying it with itself. Another option is to use a PLL and configure it to generate the double frequency at its output. The PLL lock indicator can then be re-used as an indicator, whether a pilot tone is existent.

In the upper branch, a bandpass filter selects the difference signal with ranges from 23 to 53 kHz. It is then modulated to baseband by the previously generated 38 kHz subcarrier. Subsequently, a lowpass limits signal bandwidth to 15 kHz, which removes the generated modulation artifacts.

The lower branch lowpass-filters the summation, or mono signal, as described in paragraph 2.6.1.

The rightmost part in the diagram performs the combination of the mono and the difference signal, according to Eq.(2.10).

## Chapter 3

# High-Level Synthesis

### 3.1 Introduction / State of the Art

### 3.2 Functionality (transform high-level code to HDL)

### 3.3 Language Support

#### 3.3.1 C++

#### 3.3.2 SystemC

### 3.4 Coding

#### 3.4.1 Compiler Directives (#pragma's)

#### 3.4.2 Data types

#### 3.4.3 Functions

#### 3.4.4 Loops

#### 3.4.5 Conditional statements

### 3.5 Advantages / Disadvantages

easy and much faster to get complex HDL code, such as image/video processing, through libraries like OpenCV. However, it is less optimized, and a developer still needs to have a strong hardware background, to be able to understand what is being generated in hardware from x lines of code in HLS/Cpp.

## Chapter 4

# System Architecture/Concept

### 4.1 Block Diagram (with details)

#### 4.1.1 describe blocks

### 4.2 Which parts in HLS and VHDL

### 4.3 Test Environment

## Chapter 5

# Implementation

### 5.1 Matlab/Python Model

in fixed point, close to hardware level algorithm

### 5.2 VHDL (no shortcut)

#### 5.2.1 Channel Selection (IF to Channel-BB)

#### 5.2.2 Phase Detector

#### 5.2.3 other Elements

### 5.3 High-Level Synthesis

#### 5.3.1 Channel Selection (IF to Channel-BB)

#### 5.3.2 Phase Detector

#### 5.3.3 other Elements

### 5.4 Common Testbench

#### 5.4.1 Architecture (same tb for VHDL and HLS-generated HDL)

#### 5.4.2 Framework cocotb, with ghdl compiler

Instantiate both HDL models in the testbench.

Display a direct comparison of outputs in graphs. This is practical, since the cocotb framework runs in python and graphs can be generated using Python's matplotlib.

## 5.5 Develop HLS compiler independent code (optional)

### 5.5.1 Challenges

Different compilers use different `#pragmas`, etc.

The `#pragmas` need to be within the code, at the appropriate positions, which makes it difficult to write compiler independent code....



## Chapter 6

# Deployment on Hardware

### 6.1 Hardware Platform

#### 6.1.1 RTL2832u Dongle

#### 6.1.2 ZedBoard

## Chapter 7

# Test Results

### 7.1 Functionality

#### 7.1.1 Implementation effort/time

#### 7.1.2 Hardware Utilization

#### 7.1.3 Others

## Chapter 8

# Using Literature and other Resources

[1]

## Appendix A

### Technical Details

## Appendix B

# Supplementary Materials

List of supplementary data submitted to the degree-granting institution for archival storage (in ZIP format).

### B.1 PDF Files

Path: /

thesis.pdf . . . . . Master/Bachelor thesis (complete document)

### B.2 Media Files

Path: /media

\*.ai, \*.pdf . . . . . Adobe Illustrator files  
\*.jpg, \*.png . . . . . raster images  
\*.mp3 . . . . . audio files  
\*.mp4 . . . . . video files

### B.3 Online Sources (PDF Captures)

Path: /online-sources

Reliquienschrein-Wikipedia.pdf [2]

Appendix C

Questionnaire

Appendix D

LaTeX Source Code

# References

## Literature

- [1] Hubert M. Drake, Milton D. McLaughlin, and Harold R. Goodman. *Results obtained during accelerated transonic tests of the Bell XS-1 airplane in flights to a MACH number of 0.92*. Tech. rep. NACA-RM-L8A05A. Edwards, CA: NASA Dryden Flight Research Center, Jan. 1948. URL: [https://www.nasa.gov/centers/dryden/pdf/87528main\\_RM-L8A05A.pdf](https://www.nasa.gov/centers/dryden/pdf/87528main_RM-L8A05A.pdf) (cit. on p. 17).

## Online sources

- [2] *Reliquienschrein*. Sept. 2018. URL: <https://de.wikipedia.org/wiki/Reliquienschrein> (visited on 02/28/2019).



# Check Final Print Size

— Check final print size! —



— Remove this page after printing! —