

# Real-Time FPGA-Based Testbed for Evaluating Digital Predistortion in Fully Digital MIMO Transmitters

by

Hoda Barkhordar-pour

A thesis  
presented to the University of Waterloo  
in fulfillment of the  
thesis requirement for the degree of  
Master of Applied Science  
in  
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2023

© Hoda Barkhordar-pour 2023

## **Author's Declaration**

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

## Statement of Contributions

This thesis includes material that has appeared in conference proceedings and published in IEEE MTT-S International Microwave Symposium Digest.

### Work presented in Chapter 3:

Hoda Barkhordar-pour selected the digital hardware for the testbed and designed and implemented the digital signal processing section of the testbed, which includes the PC-based server, the datapath of the signal flow on the FPGA and the radio hardware of the FPGA. Jin Gyu Lim designed, fabricated and tested the RF circuits, antenna and mechanical frame of the testbed. Mohammed Almoneer designed and simulated the SISO DPD in hardware descriptive language. Hoda Barkhordar-pour performed the experiment and compiled the results. Mohammed Almoneer drafted the manuscript. Hoda Barkhordar-pour reviewed and edited the manuscript. Prof. Slim Boumaiza and Prof. Patrick Mitran reviewed and edited the manuscript and supervised the project.

*Citation:*

M. Almoneer, H. Barkhordar-pour, P. Mitran and S. Boumaiza, "Hardware-efficient implementation of piece-wise digital predistorters for wideband 5G transmitters," *IEEE MTTs Int. Microw. Symp. Dig.*, 2022, pp. 514-517, doi: [10.1109/IMS37962.2022.9865388](https://doi.org/10.1109/IMS37962.2022.9865388).

### Work presented in Chapter 4:

Mohammed Almoneer designed and simulated the DISO DPD in hardware descriptive language. Hoda Barkhordar-pour integrated the DISO DPD module into the FPGA-based testbed that was developed in Chapter 3. Hoda Barkhordar-pour and Jin Gyu Lim conducted the experiments, compiled the results and drafted the manuscript. Prof. Slim Boumaiza and Prof. Patrick Mitran reviewed and edited the manuscript and supervised the project.

*Citation:*

H. Barkhordar-pour, J. G. Lim, M. Almoneer, P. Mitran and S. Boumaiza, "Real-time FPGA-based implementation of digital predistorters for fully digital MIMO transmitters," *IEEE MTTs Int. Microw. Symp. Dig.*, 2023. [Accepted]

## Abstract

As one of the key enabling technologies of 5G networks, massive multiple-input, multiple-output (MIMO) transmitters use many transmit chains to ensure a very high data rate and acceptable signal quality. Realizing Massive MIMO not only includes increasing antenna count but also requires proportionally more power amplifiers (PAs). Digital predistortion (DPD) is a well-established signal processing method that mitigates the non-linearities of a PA when operated near saturation. Design tradeoffs must be carefully considered to reduce the system's overall power requirements given the high PA count in MIMO systems. This implies DPD power consumption for each transmission chain must be minimized. Apart from this, larger transmission bandwidths in next-generation networks require high hardware clock rates on the order of a few gigahertz. Current hardware can satisfy clock rates of up to hundreds of megahertz. Thus, there is a need for parallelized signal processing methods to meet bandwidth requirements.

This thesis investigates and addresses some challenges for deploying massive MIMO systems by designing and building a reconfigurable digital signal processing (DSP) testbed that allows for the implementation and validation of real-time DSP algorithms including DPD, for fully digital massive MIMO transceivers. This testbed allows transmission of up to 16 fully digital transmission chains at sub-6 GHz frequencies and supports up to 120 MHz of modulation bandwidths. Finally, a low-complexity and parallelized piecewise-linear (PWL) dual-input dual-output (DISO) DPD solution is proposed for linearizing MIMO transmitters. This DPD solution is realized with a commercially available field-programmable-gate-array (FPGA).

The developed 16-chain FPGA-based MIMO testbed was first evaluated for single-chain transmission and later with 4 transmission chains using a 2x2 antenna sub-array and tested with 4 uncorrelated signals with 75 MHz bandwidth inside an anechoic chamber. The proposed DPD algorithm was evaluated using both the FPGA-implemented engine and a PC-based engine. With a starting adjacent channel power ratio (ACPR) of -33.4 dB and root normalized mean square error (RNMSE) of 7.71%, the FPGA-based engine reached -47.1 dB of ACPR and achieved an RNMSE of 1.59% after linearization. The PC-based engine performed similarly, reaching ACPR of -48.3 dB and RNMSE of 1.42%. Resource utilization was reported for the DPD implementation of 4 and 8 chains. An 8-chain reduced-complexity DPD implementation uses up to 89% of the DSP slices on the FPGA.

## Acknowledgements

I would like to thank Professor Slim Boumaiza and Professor Patrick Mitran for guiding me in my research, teaching me critical thinking and providing a healthy research environment.

I thank my mother and father for bearing hardships all these years just so my sister and I would have the best education growing up. Thanks to Sepehr, for his patience and the survival kits on deadline nights. Thank you, Boshra, for the memes and laughs and for keeping me sane. I couldn't have done it without the unconditional support of my loved ones.

A big thank you to my colleagues in the EmRG group, who listened to me complain about the FPGA and gathered for laughs and discussions, about research and life. I will always cherish the memories of the first conference we attended together and I wish them all the best.

# Table of Contents

Author's Declaration	ii
Statement of Contributions	iii
Abstract	iv
Acknowledgements	v
List of Figures	ix
List of Tables	xii
List of Abbreviations	xiii
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Thesis Objectives . . . . .	3
1.3 Thesis Outline . . . . .	4
<b>2 Background and Literature Review of Hardware Implementation of Digital Predistortion</b>	<b>5</b>
2.1 Digital Predistortion of Single Chain Transmitters . . . . .	5
2.1.1 Single Chain Transmitter Architecture . . . . .	5

2.1.2	Power Amplifier Non-Linear Behavior . . . . .	7
2.1.3	Linearity Figures of Merit . . . . .	8
2.1.4	Digital Predistortion for Linearizing Single Chain Transmitters . . . . .	11
2.2	Digital Predistortion of MIMO Transmitters . . . . .	15
2.2.1	MIMO Transmitter Architecture . . . . .	15
2.2.2	Impact of Antenna Crosstalk on Power Amplifier Behavior . . . . .	17
2.2.3	Dual-Input-Single-Output Digital Predistortion for MIMO Transmitters . . . . .	20
2.3	Review of Existing DPD Hardware Implementations for Single Chain and MIMO Transmitters . . . . .	22
2.4	Review of Real-time MIMO Transmitter Testbeds . . . . .	25
2.5	Discussion . . . . .	27
<b>3</b>	<b>Real-time Testbed Development and Implementation for DSP Applications in Fully Digital MIMO Transmitters</b>	<b>28</b>
3.1	Introduction . . . . .	28
3.2	Real-time FPGA-based Testbed Requirements . . . . .	29
3.3	Testbed Implementation Considerations . . . . .	35
3.3.1	Server-based User Interface Design . . . . .	35
3.3.2	Signal Streaming Memory . . . . .	36
3.3.3	Clock Domain Crossing . . . . .	36
3.3.4	Data Converter IP Configurations . . . . .	37
3.4	SISO DPD Engine for Single-Chain Transmitters . . . . .	40
3.4.1	Piecewise SISO DPD Formulation . . . . .	41
3.4.2	Hardware Implementation of SISO DPD . . . . .	42
3.4.3	Validation Setup and Results . . . . .	44
3.5	Complete Testbed Summary . . . . .	48

<b>4</b>	<b>DPD Implementation for MIMO Transmitters and Validation Results</b>	<b>49</b>
4.1	DISO DPD Engine for MIMO Transmitters . . . . .	49
4.1.1	Piecewise DISO DPD Formulation . . . . .	50
4.1.2	System Setup . . . . .	51
4.1.3	Validation Results . . . . .	52
4.2	Testbed Comparison and Capacity Analysis . . . . .	55
4.2.1	Testbed Comparison . . . . .	55
4.2.2	Capacity Analysis . . . . .	55
<b>5</b>	<b>Conclusion and Future Work</b>	<b>61</b>
5.1	Conclusion . . . . .	61
5.2	Future Work . . . . .	62
	<b>References</b>	<b>64</b>



# List of Figures

1.1	Conceptual wireless communication base station in 4G LTE (left) and 5G networks, which deploy MIMO technology (right). . . . .	2
2.1	A single transmitter chain base station architecture with an ideal PA. . . .	6
2.2	A single transmitter chain base station architecture including the DPD engine and the transmitter-observation receiver feedback structure used for training the DPD. . . . .	7
2.3	Linearity-efficiency tradeoff in the PA. [1] . . . . .	8
2.4	PA non-idealities including non-linearity and memory effect, without and with DPD. . . . .	10
2.5	Conceptual diagram showing DPD process in complex baseband equivalent domain. The direct learning process (green) and indirect learning process (blue) are pictured. . . . .	13
2.6	A MIMO transmitter chain architecture. . . . .	16
2.7	Beamforming structures: (a) Digital beamforming (b) Analog RF beamforming (c) Hybrid beamforming. . . . .	18
2.8	Incident and reflected waves in $k$ -chain MIMO system. . . . .	19
2.9	Structure of the DISO DPD proposed in [2], consisting of a DISO DPD engine in each transmission chain and a shared CTMM module. . . . .	21
2.10	Categorized prior work on complexity reduction of the predistorter . . . . .	23
2.11	LuMaMi of Lund University (left) and of KU Leuven (right). . . . .	26
2.12	Argos of Rice University (left) and Southeast University's testbed (right). .	27

3.1	The chosen hardware for the MIMO testbed, Xilinx RFSoc ZCU216 Evaluation Board with the daughterboard and PLL board attached. . . . .	32
3.2	Simplified diagram showing the components of the processing hardware and radio hardware. For simplicity, it is assumed that only one transceiver path is enabled, however, multiple chains are active in a MIMO system. . . . .	33
3.3	EmRG 16-chain fully digital MIMO testbed. . . . .	34
3.4	Block diagram showing the components of the signal processing hardware .	35
3.5	Block diagram showing the components of the radio hardware . . . . .	38
3.6	Different Nyquist zone operations made possible using two different reconstruction waveforms on the Xilinx RFSoc [3]. . . . .	39
3.7	Proposed real-time PWL SISO DPD, implemented in FPGA hardware. . .	40
3.8	Piecewise linear function $f(x; \alpha, \phi)$ . . . . .	42
3.9	DSP48E2, the DSP slices available in Ultrascale RFSoc devices [4]. . . . .	43
3.10	Multiply-add in a (a) tree structure (b) cascade structure which optimizes DSP slice resource usage. . . . .	43
3.11	Distribution and parallelization scheme used in the main DPD module that will allow sample processing at high rates [5]. . . . .	44
3.12	Block diagram of the setup used to validate the single-chain operation of the testbed. The testbed receives the pre-distorted signal $\tilde{z}$ when PC-based DPD is tested (blue) and receives the ideal signal $\tilde{x}$ when FPGA-based DPD is tested (green). . . . .	45
3.13	Power spectral density of the observed signal ( $\tilde{y}$ ) in single-chain operation.	46
3.14	Logarithmic AM/AM or gain distortion of the observed signal ( $\tilde{y}$ ) in single-chain operation. . . . .	47
3.15	AM/PM or phase distortion of the observed signal ( $\tilde{y}$ ) in single-chain operation. . . . .	48
4.1	Proposed real-time PWL DISO DPD, implemented in FPGA hardware. . .	50
4.2	Block diagram of the measurement setup. . . . .	52
4.3	Experimental fully digital massive MIMO transmitter setup including the 4-element sub-array that was used in the experiments. . . . .	53

4.4	Power spectrum comparison of (a) without DPD, (b) PC-based DPD engine, and (c) FPGA-based DPD engine, at an estimated over-the-air boresight receiver. . . . .	54
4.5	Relation between the maximum number of implementable chains and maximum possible linearization bandwidth and its consequent interpolation factor for a given DPD. . . . .	58
4.6	Relation between the maximum number of implementable chains and maximum possible linearization bandwidth and its minimum parallel number of streams ( $P$ ) need to implement a given DPD. . . . .	59
4.7	Trade-off between DPD hardware complexity, linearization bandwidth and the maximum number of implementable chains. . . . .	60

# List of Tables

2.1	Antenna mutual coupling quantified through ACPR levels. . . . .	17
3.1	Real-time MIMO testbed requirements. . . . .	30
3.2	Commercial products considered to serve as the digital baseband domain of the testbed. Cells populated with ‘N/A’ were not declared by the manufacturing company. . . . .	31
3.3	Performance comparison of PC-based and FPGA-based SISO DPD engine. . . . .	46
4.1	Performance comparison of PC-based and FPGA-based DISO DPD engine on the four-chain RF front-end. . . . .	55
4.2	FPGA Resource Utilization of Proposed DPD . . . . .	56
4.3	MIMO testbed comparison of those in literature and the developed testbed in this thesis. . . . .	57

# List of Abbreviations

- 5G** fifth-generation
- ACPR** Adjacent channel power ratio
- ADC** analog-to-digital converter
- ASIC** application-specific integrated circuit
- BRAM** block ram
- CBBE** complex baseband equivalent
- CDC** clock domain crossing
- CRV** complexity reduced Volterra
- CTMM** crosstalk and mismatch model
- DAC** digital-to-analog converter
- DC** direct current
- DDR** dynamic deviation reduction
- DISO** dual-input single-output
- DMA** data memory access
- DPD** digital predistortion
- DSP** digital signal processing

**DUT** device under test

**DVR** decomposed vector rotation

**FLOPS** floating-point operations per sample

**FPGA** field-programmable gate array

**GMP** general memory polynomial

**GSps** Giga Samples per second

**LTE** long-term-evolution

**LUT** look-up table

**MIMO** multiple-input-multiple-output

**MP** memory-polynomial

**MU-MIMO** multi-user MIMO

**NR** new radio

**OOB** out-of-band

**OTA** over-the-air

**PA** power amplifier

**PAPR** peak-to-average power ratio

**PL** programmable logic

**PLL** phase-locked loop

**PS** processing system

**PSD** power spectral density

**PWL** piece-wise linear

**RF** radio frequency

**RNMSE** root normalized mean square error

**SDR** software-defined-radios

**SISO** single-input single-output

**TOR** transmission-observation-receiver

**UI** user interface

# Chapter 1

## Introduction

### 1.1 Motivation

The number of connected devices worldwide has been reported to be around 13.1 billion in 2022 and is forecast to reach 29.4 billion by the year 2030 [6]. Given this explosive growth of connected devices and the limited spectrum available, future-generation networks must allow greater network coverage and capacity while being considerate of energy consumption and overall network efficiency. As bandwidth cannot be increased indefinitely, some techniques are introduced in 4G long-term evolution (long-term-evolution (LTE)) networks and fifth-generation (5G) new radio (NR) to provide higher spectral and energy efficiency, including microcells, beamforming, full duplex communication, and multiple-input-multiple-output (MIMO).

Massive MIMO is one of the key enablers of 5G, which deploys multiple antennas to increase the data throughput and system capacity (Fig. 1.1). Users utilizing MIMO can use the same time or frequency resource simultaneously for transmission. The first wireless standard to specify MIMO use was LTE in 2008 for 2 transmitting and 2 receiving antennas (2x2 MIMO). In 4G LTE networks, 4x4 MIMO became prevalent to allow higher user capacity. 5G NR networks are being deployed in higher frequencies compared to previous generation networks. The shorter wavelengths at these frequencies result in smaller radio frequency component dimensions including smaller antennas. This results in smaller MIMO design form factors and makes a 32 antenna MIMO feasible and realizable [7].

As a result of closely-spaced antennas in MIMO systems, adjacent antenna elements experience inter-antenna cross-coupling. Signals from adjacent chains couple back into



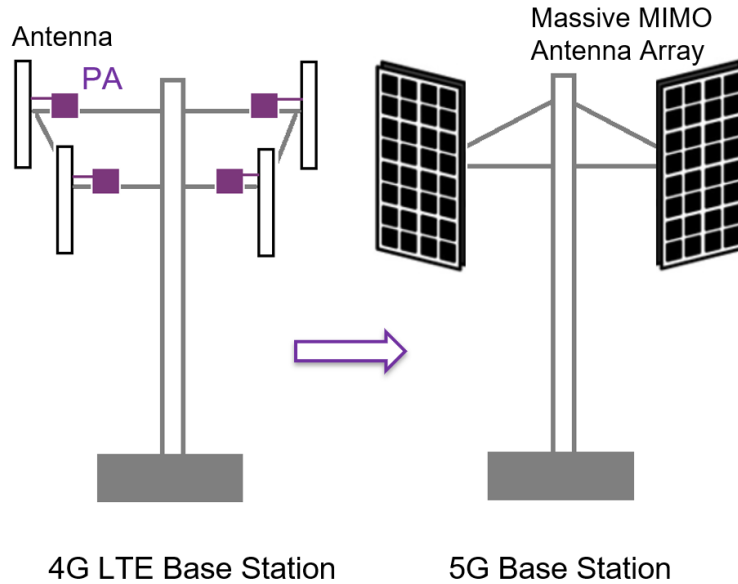


Figure 1.1: Conceptual wireless communication base station in 4G LTE (left) and 5G networks, which deploy MIMO technology (right).

each transmitter chain and affect the operation of the power amplifier (PA), giving rise to additional non-linearity in the PA. Previously, in single-chain transmitters, intrinsic PA non-linearity was able to be addressed using a comparatively simple signal processing method called digital predistortion (DPD). By using a feedback loop, DPD can correct the non-linearity (and consequent spectral regrowth) of power efficient PAs iteratively.

Compared to single-chain transmitters, in MIMO systems, the complexity of the DPD block increases, since DPD will also have to mitigate additional non-linearity due to antenna cross-coupling. Complexity-reduced hardware implementation of single-chain DPD systems has been explored. This includes look-up table (LUT)-based implementations [8, 9, 10, 11, 12, 13, 14] and exploring reduced-complexity DPD basis functions [15, 16, 9, 17, 18]. As complexity scales in MIMO systems, the methods in literature may not be optimal for MIMO DPD methods. To ensure the feasibility of deployment in MIMO systems, MIMO DPD methods must first be implemented in hardware to evaluate their performance.

Realizing massive MIMO not only includes increasing antenna count but also requires a higher number of power amplifiers. Careful consideration must be taken to reduce the power consumption of the whole system to ensure overall system efficiency. This implies

signal processing power consumption, including the DPD power consumption, must be reduced for each transmission chain. Apart from this, larger transmission bandwidths in next-generation networks require high hardware processing rates in the order of at least tens of gigahertz. Current hardware can satisfy up to hundreds of megahertz. One solution to process tens of gigahertz of bandwidth using available digital hardware clock rates is to use parallel-processing schemes [19]. Thus, by developing parallel-processing digital signal processing (DSP) methods, it is possible to meet the bandwidth requirements of next-generation networks.

As MIMO technology prevails in 5G telecommunications networks, signal processing methods must be tested for viability and performance before deployment. Developing a MIMO testbed not only enables research in MIMO DSP methods but also allows exploration of the potential of MIMO technology. Some testbeds have already been developed globally [20, 21, 22, 23] but they lack the necessary bandwidth for research in hardware verification of MIMO DPD methods for MIMO base station transmitters.

## 1.2 Thesis Objectives

This thesis investigates and addresses some of the challenges of deploying massive MIMO systems by:

- Designing and building a re-configurable real-time DSP testbed that allows the implementation and validation of DSP algorithms for massive MIMO transceivers. The platform supports up to 16 chains of fully-digital signal transmission at sub-6 GHz frequencies and supports up to a maximum linearization bandwidth of 1.5 GHz.
- Proposing, implementing and testing a low-complexity and parallel-processing single-chain DPD solution and expanding the DPD to a MIMO DPD solution. The solution is verified using the developed testbed and meets signal quality standards for 5G NR transmission bandwidths.
- Providing analysis on the capacity of the developed MIMO testbed. The limited processing resources available on the testbed dictate a tradeoff between the number of transmitting chains and the DPD complexity per chain. The challenges and benefits of a distributed processor for MIMO transmitters are discussed.

## 1.3 Thesis Outline

This thesis is organized as follows. First, in Chapter 2, background theory is discussed regarding DPD techniques for linearizing power amplifiers in single-chain transmitters and MIMO transmitters respectively. In the same chapter, a literature review is presented on DPD hardware implementation methods that reduce implementation complexity and power consumption of the DPD unit. Finally, a review of MIMO systems that were developed for research purposes is presented.

Chapter 3 presents the process of designing and developing the MIMO testbed. Testbed requirements are clarified and design choices that satisfy these requirements are described. Next, some key design challenges and their respective solutions are presented. Finally, to verify the single-chain transmission performance of the testbed, a low complexity, power efficient and parallel processing DPD engine architecture is proposed for single-chain transmitters. This real-time DPD architecture is implemented on the mentioned reconfigurable testbed. The DPD achieves a linearization bandwidth of 1.2 GHz with a digital hardware clock rate of 300 MHz.

In Chapter 4, the single-chain DPD engine is augmented to compensate antenna crosstalk in MIMO transmitters and is presented as the dual-input-single-output DPD. This DPD engine is implemented on the testbed and tested on a 4-chain MIMO system that was designed and built in the lab. Measurement results are presented. To conclude Chapter 4, a brief analysis of the capacity and DPD complexity tradeoff of the MIMO testbed is illustrated. Finally, a summary and conclusion of this thesis are presented in Chapter 5. Potential future works that can result from the work done in this thesis are also discussed in this chapter.

# Chapter 2

## Background and Literature Review of Hardware Implementation of Digital Predistortion

This chapter reviews the relevant background and literature needed to develop a MIMO testbed and implement a low-complexity wideband DPD solution in hardware for MIMO systems. DPD is a signal processing method that compensates for PA non-linearity such that the cascade of the DPD block with the PA results in a linear power amplifying module. In order to understand the scope of the complexity of a MIMO DPD block, this chapter first reviews single-chain DPD in Section 2.1 before addressing MIMO systems in Section 2.2. Previous literature has proposed a variety of techniques to simplify the DPD engine's structure, albeit for a single transmitter chain. These methods are reviewed in Section 2.3. Finally, this chapter reviews all known developed MIMO testbeds as the baseline that will be used to compare the developed testbed in this thesis.

### 2.1 Digital Predistortion of Single Chain Transmitters

#### 2.1.1 Single Chain Transmitter Architecture

Fig. 2.1 shows the architecture of a single transmitter chain base station with an ideal PA. The modules in this transmitter can be categorized into two sections: the digital baseband

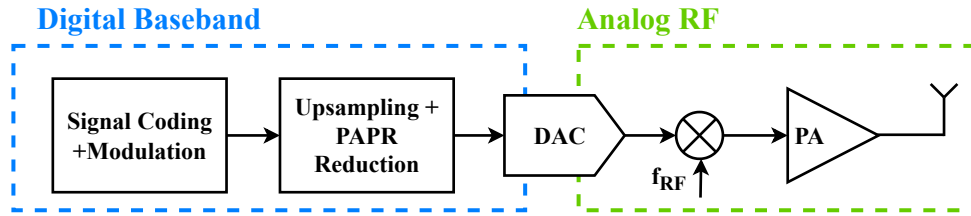


Figure 2.1: A single transmitter chain base station architecture with an ideal PA.

modules that are implemented in digital hardware and the analog radio frequency (RF) modules comprising RF circuits. Data converters, the digital-to-analog converter (DAC) and analog-to-digital converter (ADC), make the transition between these two sections possible. The modules used in the transmitter chain are as follows:

- A signal generator that streams the coded and modulated signal.
- An upsampling and peak-to-average power ratio (PAPR) reduction unit. This unit upsamples the signal to a wider bandwidth in order to accommodate the inevitable spectral regrowth due to non-linearity and memory effects present in the chain, a PAPR reduction module is necessary to avoid PA saturation.
- A DAC that converts the digital signal to a baseband analog signal.
- An upconverter to transfer the baseband analog signal to the desired RF frequency that is within the PA's operational frequency.
- A PA, and if required, a driver amplifier to facilitate the required input power level to the PA.
- A single transmitting antenna.

Due to the non-linear nature of the PA and the memory effects that the RF component line-up will exhibit, a transmission-observation-receiver (TOR) along with a signal processing module, the DPD unit, is needed to mitigate non-idealities of the single transmission chain (see Fig. 2.2). The TOR will sample the RF signal and provide a feedback loop to the DPD unit. The TOR consists of the following modules:

- An RF coupler that will sample the RF signal after the power amplifier and before the antenna.

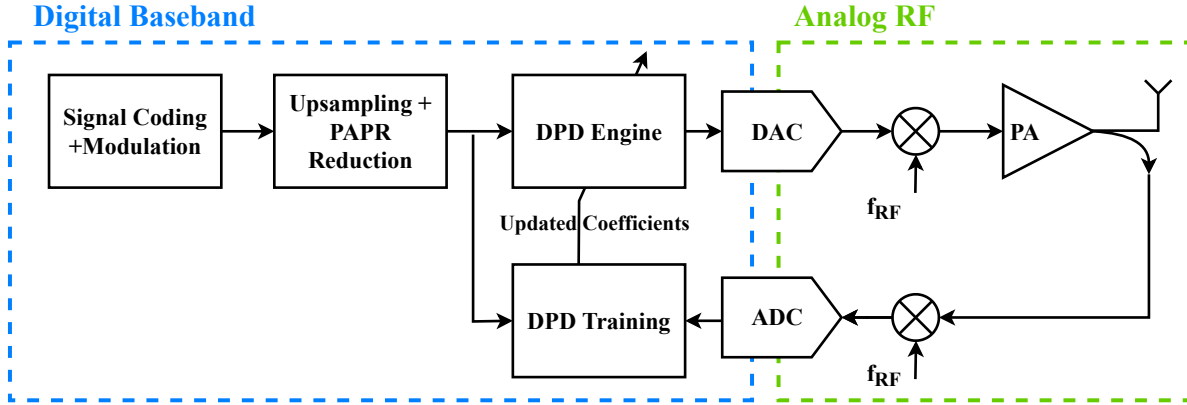


Figure 2.2: A single transmitter chain base station architecture including the DPD engine and the transmitter-observation receiver feedback structure used for training the DPD.

- A downconverter that will bring the PA output signal to the analog baseband and an ADC that will convert the analog baseband signal to the digital baseband.
- A DPD training unit that will use the ideal signal before distortion and the sampled signal after distortion by the PA to generate the parameters (coefficients) needed by the DPD engine. The DPD engine will distort the ideal signal to generate a pre-distorted signal that will be sent to the PA. Further details about the DPD unit are discussed in Section 2.1.4.

## 2.1.2 Power Amplifier Non-Linear Behavior

In a communications transmitter chain, after the data signal is up-converted to the desired RF frequency, the power amplifier provides the required power level to the signal before being radiated by the antenna. As the most power-consuming component in the transmitter, the PA must have high efficiency. One compromise in designing and utilizing PAs is the linearity and efficiency trade-off. PAs exhibit non-linear behaviour when given a high-power signal at their input and eventually saturate as the input power is further increased. This non-linear behaviour leads to spectral regrowth.

Communication standards specify out-of-band (OOB) emission levels [24]. A simple approach to achieve the required linearity to meet the OOB emission levels is to use the PA in power back-off. As a consequence of the power back-off, the PA will be operating with lower efficiency. This trade-off is visualized in Fig. 2.3. Thus, it is desirable to operate

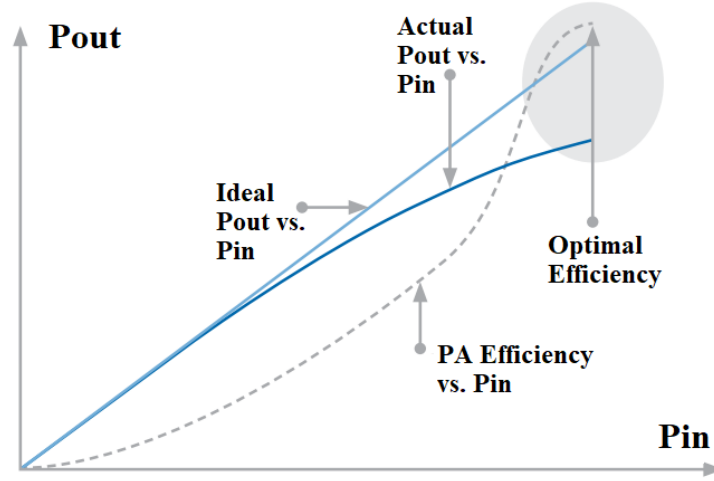


Figure 2.3: Linearity-efficiency tradeoff in the PA. [1]

the PA at high efficiency and linearize the PA using signal processing methods instead of operating at back-off. The signal processing method that is studied in this work to linearize the PA is digital predistortion or DPD.

### 2.1.3 Linearity Figures of Merit

Below, several metrics that will be used to describe signal characteristics and quantify non-idealities of the PA are reviewed.

#### PAPR

Peak-to-Average Power Ratio or PAPR is a measure of the peak power of a signal relative to its average power. In communications systems, the PAPR of a signal is an important factor that affects the performance of the system, i.e., a high PAPR signal can saturate the PA. It is important to back off the input signal of a PA by at least the PAPR of the signal from the saturation power of the PA to avoid significant non-linear distortion. The PAPR of the signal  $x(t)$  is defined as follows:

$$\text{PAPR [dB]} = 10 \log_{10} \frac{\max(|x(t)|^2)}{\text{mean}(|x(t)|^2)}$$

## Linearization Bandwidth

As seen in Fig. 2.4, PA non-linearity gives rise to OOB emissions and spectral regrowth. To linearize this non-linearity with DSP methods, it is necessary to capture and generate a bandwidth wider than that of the modulated signal. This wider bandwidth is called the linearization bandwidth. As a rule of thumb, it is sufficient to capture at least 5 times the bandwidth of the modulated signal. For example, in Fig. 2.4, the modulated signal bandwidth is approximately 80 MHz and the linearization bandwidth is 400 MHz.

## Amplitude and Phase Distortion

AM/AM distortion refers to the distortion of the amplitude of a signal as it is amplified. It is a visual tool that shows the PA's power-dependent behaviour. It is typically plotted as a graph with the input signal level on the x-axis and the output signal level on the y-axis. The AM/AM distortion plot shows how the output signal level changes as the input signal level are varied. Similarly, the AM/PM distortion plot shows how the phase shift of the output signal changes due to distortion, as the input signal level is varied. Fig. 2.4 shows the amplitude and phase distortion of a complex baseband signal at the output of a PA, without and with DPD. This figure also shows how DPD can eradicate spectral regrowth. Note that the complex baseband equivalent (CBBE) signal of the PA output is used to obtain the plots.

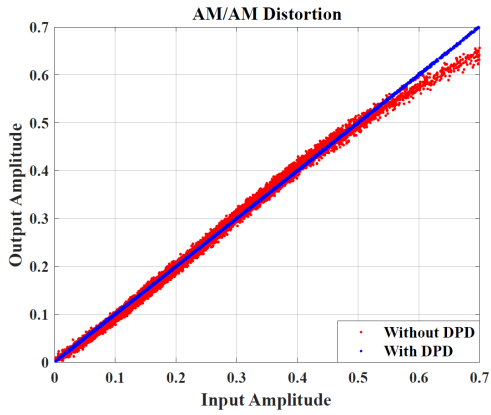
## ACPR

Adjacent channel power ratio (ACPR) is a performance metric that quantifies out-of-band emissions. ACPR is typically measured in decibels (dB) and is expressed as the difference between the power of the signal in an adjacent channel and the power of the signal in the desired channel. For example, the ACPR of the signal in Fig. 2.4 without DPD is approximately  $(-110 \text{ dBm}) - (-80 \text{ dBm}) = -30 \text{ dB}$  and with DPD, ACPR is improved to  $(-135 \text{ dBm}) - (-80 \text{ dBm}) = -55 \text{ dB}$ . The 5G NR standard requires an ACPR of -45 dB or better for a base station [24].

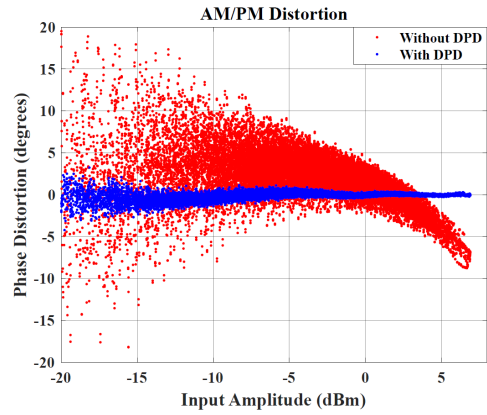
## Root Normalized Mean Square Error (RNMSE)

In this thesis, the root normalized mean square error (RNMSE) is a measure of the difference between the ideal signal  $x(t)$  and the amplified PA output  $y(t)$ , expressed in percentage. The 5G NR standard requires RNMSE of 2.2% or better for a base station [24].

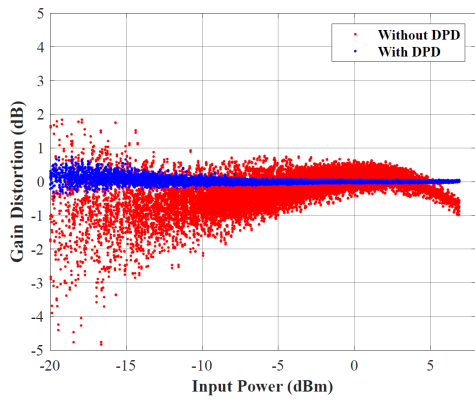




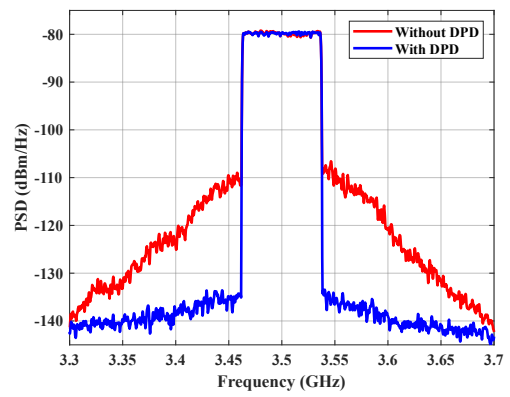
(a) Amplitude distortion or AM/AM



(b) Phase distortion or AM/PM



(c) Gain distortion or logarithmic AM/AM



(d) Power spectral density

Figure 2.4: PA non-idealities including non-linearity and memory effect, without and with DPD.

## 2.1.4 Digital Predistortion for Linearizing Single Chain Transmitters

In essence, single-input single-output (SISO) DPD aims to linearize the PA by applying the inverse of the non-linear transfer function of the PA, i.e., by predistorting the signal input to the PA. By doing so, the cascade of the DPD block and the PA will behave linearly. In the next section, the forward model of the PA is reviewed followed by a section on DPD models and learning methods.

### Forward Modelling the PA

The non-linear characteristics of a PA can be modelled using the Volterra/Wiener representation of non-linear systems, first presented in [25]. DPD model derivation starts with modelling the PA non-linearity with the Volterra expansion representing a non-linear system exhibiting memory effects.

$$y(t) = \sum_{l=1}^{\infty} \int_{\mathbb{R}^l} h_l(\tau_1, \dots, \tau_l) \cdot \prod_{r=1}^l x(t - \tau_r) d\tau_1 \cdots d\tau_l \quad (2.1)$$

In (2.1),  $h(t_1, \dots, t_n)$  is the  $n^{\text{th}}$  order Volterra kernel. Together, the kernels of order  $n = 1, 2, \dots$  describe the non-linear behaviour of the PA. Memory effects are represented by memory values  $\tau_1, \dots, \tau_n$ . The real bandpass signal  $x(t)$ , centered at frequency  $\omega_0 = 2\pi f_0 t$  has the following complex baseband representation:

$$x(t) = \frac{1}{2}(\tilde{x}(t)e^{j\omega_0 t} + \tilde{x}^*(t)e^{-j\omega_0 t}) \quad (2.2)$$

where  $\tilde{x}(t)$  is called the complex envelope or complex baseband equivalent (CBBE) of  $x(t)$ .

Substituting (2.2) in (2.1) and only keeping terms with the center frequency  $\omega_0$ , we have:

$$y(t) = \sum_{n=0}^{\infty} \int_{\mathbb{R}^{2n+1}} \frac{\tilde{h}_n(\tau_1, \dots, \tau_{2n+1})}{2^n} \cdot \prod_{r=1}^n \tilde{x}^*(t - \tau_r) \cdot \prod_{k=n+1}^{2n+1} \tilde{x}(t - \tau_k) d\tau_1, \dots, d\tau_{2n+1} \quad (2.3)$$

The main challenge of using the Volterra model in (2.3) is determining the Volterra kernels. Although the Volterra model is the most general and complete formulation of

modelling a non-linear system, it is not practical. Thus, practical models have been developed that provide acceptable modelling accuracy and have reduced complexity. Prevalent models in the literature include general memory polynomial (GMP) [26], dynamic deviation reduction (DDR) [27], decomposed vector rotation (DVR) [28] and complexity reduced Volterra (CRV) [29].

## Digital Predistortion

Fig. 2.5 shows a baseband equivalent model of a single transmitter chain DPD consisting of a DPD engine, a DPD training unit and the PA. For simplicity, the TOR is not pictured. If a PA is weakly non-linear, it is commonly assumed that the DPD model structure used to linearize the PA has a model formulation identical to that of the PA's model. Generally, the baseband equivalent model generates the CBBE predistorted signal  $\tilde{z}[n]$  as follows:

$$\tilde{z}[n] = \sum_{p=1}^P w_p \tilde{\psi}_p(\tilde{\mathbf{x}}[n]), \quad (2.4)$$

where the DPD model has  $P$  basis functions  $\tilde{\psi}_1(\cdot), \tilde{\psi}_2(\cdot), \dots, \tilde{\psi}_P(\cdot)$  and  $\tilde{\mathbf{x}}[n] = [\tilde{x}[n], \dots, \tilde{x}[n-N+1]]$ . In this thesis, the DPD model used is a modified version of the DDR model seen in [27]. Using this model, the predistorted signal  $\tilde{z}[n]$  is:

$$\begin{aligned} \tilde{z}[n] = & \sum_{p=0}^{P_1} \sum_{m=0}^M w_{1,p,m} \tilde{x}[n-m] |\tilde{x}[n-m]|^p \\ & + \sum_{p=0}^{P_2} \sum_{m=1}^M w_{2,p,m} \tilde{x}[n-m] |\tilde{x}[n]|^p \\ & + \tilde{x}^2[n] \sum_{p=0}^{P_3} \sum_{m=1}^M w_{3,p,m} \tilde{x}^*[n-m] |\tilde{x}[n]|^p \\ & + \tilde{x}^*[n] \sum_{p=0}^{P_4} \sum_{m=1}^M w_{4,p,m} \tilde{x}^2[n-m] |\tilde{x}[n]|^p \end{aligned} \quad (2.5)$$

where  $P_1$  is the maximum static non-linearity order,  $P_2$ ,  $P_3$  and  $P_4$  are secondary non-linearity orders and  $M$  is the memory depth, all obtained empirically.

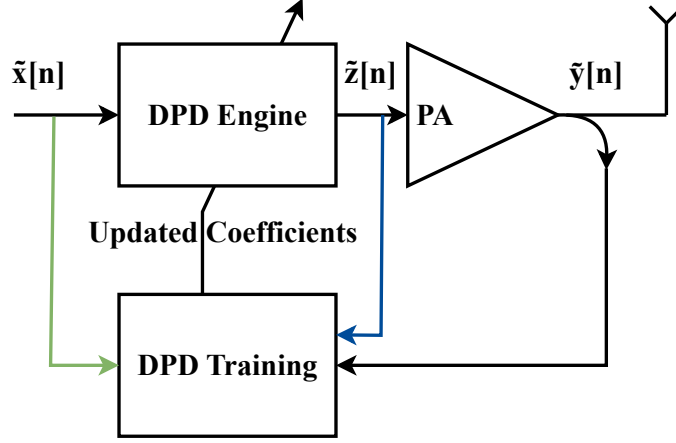


Figure 2.5: Conceptual diagram showing DPD process in complex baseband equivalent domain. The direct learning process (green) and indirect learning process (blue) are pictured.

The goal is to determine the  $w_p$  coefficients in (2.4) that minimize the distortion of the output of the PA. This is commonly found using one of two learning methods, called direct and indirect learning. The former method, first introduced along with the Volterra model for modelling non-linear amplifiers in [25], seeks to minimize an objective function  $J$  that is the mean-square error between the data signal  $\tilde{x}[n]$  and the PA output  $\tilde{y}[n]$ . Specifically,  $\mathbf{w} = [w_1, \dots, w_p]$  is sought that minimizes the following objective function over  $N$  samples:

$$J = \sum_{n=1}^N \left\| \frac{\tilde{y}[n]}{G} - \tilde{x}[n] \right\|^2, \quad (2.6)$$

where  $G$  is the PA gain. In order to calculate the complex coefficients  $\mathbf{w}$ , a gradient descent-based approach is adapted [30, 31]. Specifically, to minimize the real cost function  $J$ , the following iterations are computed:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mu \frac{\partial J}{\partial \mathbf{w}^*}, \quad (2.7)$$

where  $\mu$  is the step size or learning rate. In order to compute  $\frac{\partial J}{\partial \mathbf{w}^*}$  in (2.7), the PA output  $\tilde{y}[n]$  is modeled as:

$$\tilde{y}[n] = h_0 \tilde{x}[n] + \tilde{e}[n], \quad (2.8)$$

where  $\tilde{e}[n]$  is the distortion of the output of the PA and  $h_0$  is the linear gain of the PA. Since the PA is weakly non-linear, by ignoring  $\tilde{e}[n]$  in (2.8),  $\frac{\partial J}{\partial \mathbf{w}^*}$  can be computed from (2.8) and (2.7), resulting in:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu \frac{G}{h_0} (\Psi_{\tilde{x}}^H \Psi_{\tilde{x}})^{-1} \Psi_{\tilde{x}}^H \left( \frac{\tilde{\mathbf{y}}}{G} - \tilde{\mathbf{x}} \right) \quad (2.9)$$

where  $\tilde{\mathbf{x}} = [\tilde{x}[n], \dots, \tilde{x}[n - N + 1]]^T$  and  $\Psi_{\tilde{x}}$  is a  $N \times P$  matrix representing the  $P$  basis functions for  $N$  samples of  $\tilde{x}[n]$ , i.e.,

$$\Psi_{\tilde{x}} = \begin{pmatrix} \tilde{\psi}_1(\tilde{\mathbf{x}}[n]) & \cdots & \tilde{\psi}_P(\tilde{\mathbf{x}}[n]) \\ \tilde{\psi}_1(\tilde{\mathbf{x}}[n-1]) & \cdots & \tilde{\psi}_P(\tilde{\mathbf{x}}[n-1]) \\ \vdots & \vdots & \vdots \\ \tilde{\psi}_1(\tilde{\mathbf{x}}[n-N+1]) & \cdots & \tilde{\psi}_P(\tilde{\mathbf{x}}[n-N+1]) \end{pmatrix} \quad (2.10)$$

Alternatively, indirect learning finds  $\mathbf{w}$  by minimizing the error between a post-distorted output signal  $\frac{\tilde{y}[n]}{G}$  and the predistorted signal  $\tilde{z}[n]$  [32]. To do this, a copy of the predistortion engine is used to apply the DPD model to the scaled PA output  $\frac{\tilde{y}[n]}{G}$ . Then the DPD output is compared against the predistorted signal  $\tilde{z}[n]$  and coefficients  $\mathbf{w}$  are updated to minimize the error between the two as follows:

$$\mathbf{w} = \arg \min_{\mathbf{w}} \sum_{n=1}^N \left\| \sum_{p=0}^P w_p \tilde{\psi}_p \left( \frac{\tilde{y}[n]}{G} \right) - \tilde{z}[n] \right\|^2 \quad (2.11)$$

The optimal solution for the coefficient matrix  $\mathbf{w}$  in (2.11) is given as:

$$\mathbf{w} = (\Psi_{\tilde{y}}^H \Psi_{\tilde{y}})^{-1} \Psi_{\tilde{y}}^H \tilde{\mathbf{z}}, \quad (2.12)$$

where  $\tilde{\mathbf{z}} = [\tilde{z}[n], \dots, \tilde{z}[n - N + 1]]^T$  and  $\Psi_{\tilde{y}}$  is defined similar to (2.10):

$$\Psi_{\tilde{y}} = \begin{pmatrix} \tilde{\psi}_1(\tilde{\mathbf{y}}[n]) & \cdots & \tilde{\psi}_P(\tilde{\mathbf{y}}[n]) \\ \tilde{\psi}_1(\tilde{\mathbf{y}}[n-1]) & \cdots & \tilde{\psi}_P(\tilde{\mathbf{y}}[n-1]) \\ \vdots & \vdots & \vdots \\ \tilde{\psi}_1(\tilde{\mathbf{y}}[n-N+1]) & \cdots & \tilde{\psi}_P(\tilde{\mathbf{y}}[n-N+1]) \end{pmatrix} \quad (2.13)$$

Finally, the coefficients obtained using (2.12) are updated in the original DPD engine. Indeed, the indirect learning method does not use the gradient-descent approach in (2.7) and is quicker in updating DPD coefficients. The caveat to using indirect learning is that after a few update iterations, the signal statistics change and the PA behaviour changes along with it. Thus, in practice, indirect learning is used for the first few iterations and the results are further refined through more direct learning iterations.

## 2.2 Digital Predistortion of MIMO Transmitters

### 2.2.1 MIMO Transmitter Architecture

One of the key enabling technologies of 5G communication networks that will expedite high-speed communications at gigabits-per-second speed is MIMO systems. These systems deploy numerous transmission chains including data converters, PAs and antennas to achieve data rate and signal quality requirements of 5G NR. Increasing the number of antennas at the base station reduces the required overall radiated power due to coherent combining gains, and increases the number of terminals that can be served simultaneously while providing uniformly good service to them. Furthermore, utilizing numerous transmitting antennas results in channel hardening and avoids deep fading and shadow fading due to channel propagation [33]. The challenge in utilizing MIMO systems is to be mindful of size and space, power consumption and antenna loading effects when designing MIMO architectures.

Analogous to a single transmitter architecture, a MIMO transmitter chain is depicted in Fig. 2.6. This architecture consists of duplicates of the single transmitter in Fig. 2.1. In addition to coding and modulation, signals are precoded before being streamed. Precoding is a technology that is used to improve the performance of the system by precoding the transmitted signals to counteract the negative effects of the over-the-air (OTA) channel. It is evident that the complexity of the MIMO transmitter architecture increases significantly when the DPD unit and multiple TORs are added to the architecture in Fig. 2.6.

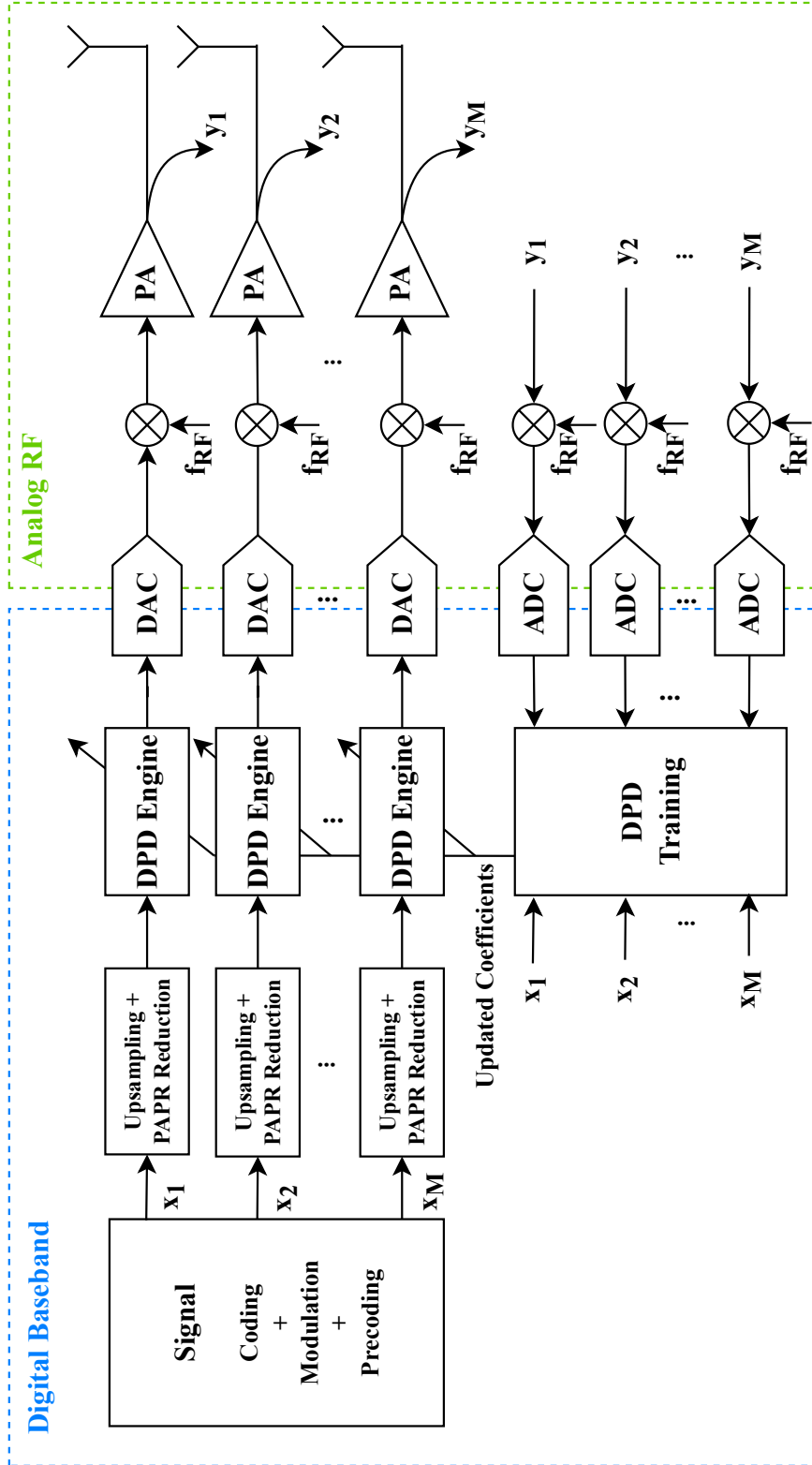


Figure 2.6: A MIMO transmitter chain architecture.

Antenna Separation	Mutual Coupling	ACPR
0.36 $\lambda$	-14.0 dB	-46.4 dB
0.44 $\lambda$	-20.9 dB	-52.3 dB
0.59 $\lambda$	-28.4 dB	-57.4 dB

Table 2.1: Antenna mutual coupling quantified through ACPR levels.

## Beamforming

One method that will unleash the potential of the MIMO structure introduced in this section is beamforming. Essentially, beamforming is the ability to control the direction in which each main antenna beam in the MIMO array is directed towards. There are three types of beamforming, as shown in Fig. 2.7. Digital beamforming controls each and every RF chain with a separate and independent digital chain. On the other hand, analog RF beamforming consists of one digital signal stream that splits into  $N$  streams and the phase shifters together control the angle at which the beam is steered by the array. Although digital beamforming consumes more power than its RF counterpart, it provides more flexibility for each transmitter chain. Hybrid beamforming is a mixture of both digital and analog RF beamforming, providing flexibility and lower power consumption. The beamforming structure that is achieved in this thesis is digital beamforming. This structure provides the flexibility to demonstrate analog beamforming and hybrid beamforming.

### 2.2.2 Impact of Antenna Crosstalk on Power Amplifier Behavior

Due to the high level of integration in MIMO chipsets, antenna elements are very close to each other and under the assumption that the antennas of a MIMO transmitter are emitting at the same frequency, mutual coupling between elements is detrimental to signal quality. An increase in mutual coupling between antenna elements results in increased OOB emissions and spectral regrowth [34]. Table 2.1 quantifies this regrowth through ACPR measurements done by varying the separation between two transmitter chain antennas and consequently, the mutual coupling level [35]. As antenna separation distances are decreased from 0.59  $\lambda$  to 0.44  $\lambda$ , the mutual coupling between antennas increases by approximately 8 dB and ACPR degrades by 5 dB. It is noteworthy to mention that in order to exploit their potential, MIMO systems usually consist of more than two antennas. This means each



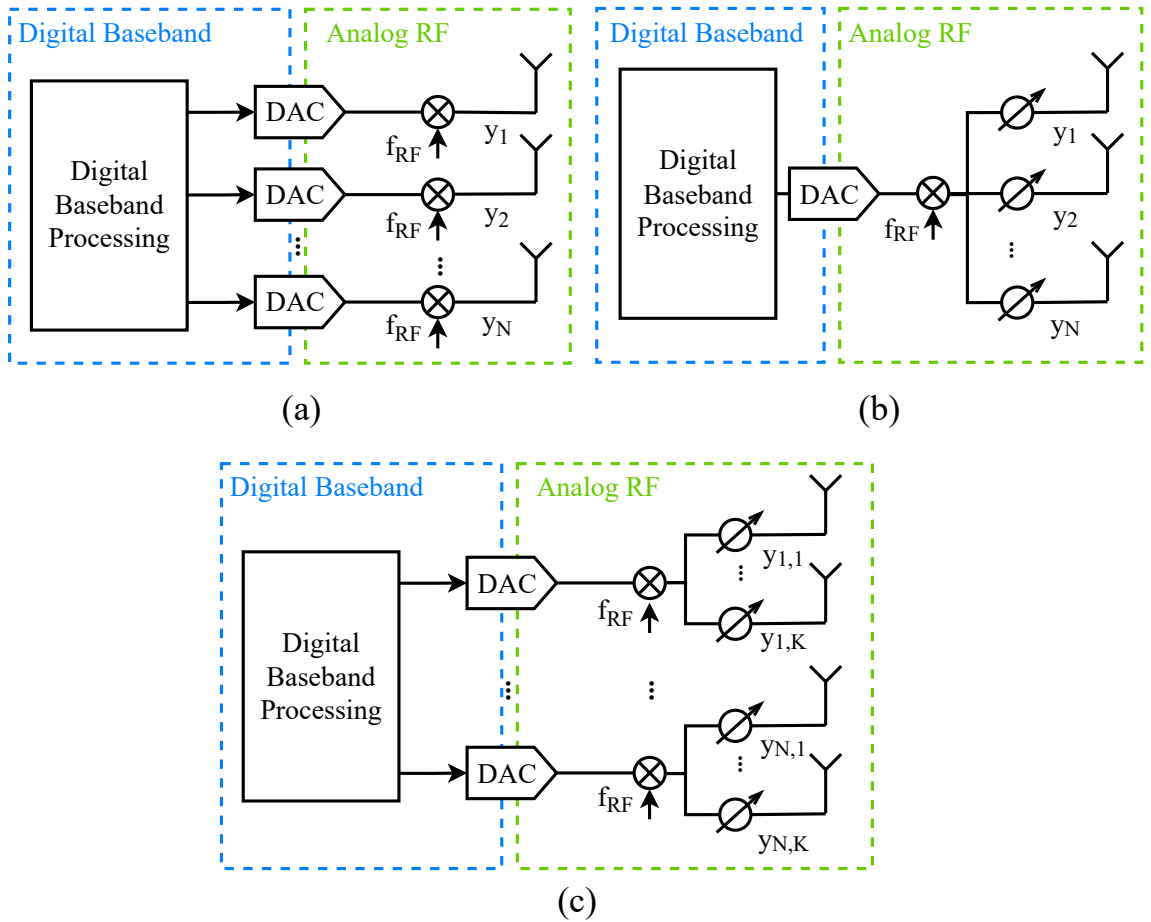


Figure 2.7: Beamforming structures: (a) Digital beamforming (b) Analog RF beamforming (c) Hybrid beamforming.

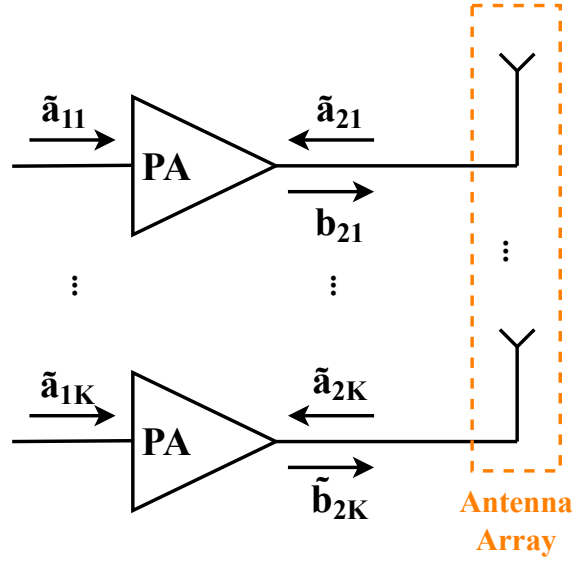


Figure 2.8: Incident and reflected waves in  $k$ -chain MIMO system.

antenna in a MIMO system is surrounded by more than one neighbouring antenna and thus the numbers depicted in Table 2.1 may be exacerbated as antenna count increases.

One solution utilized to eliminate the inter-antenna crosstalk effect in MIMO systems is to place a circulator or isolator between the power amplifier and antenna. These decoupling isolators take up a large area [36], incur additional costs [37], or have narrow operating bandwidths [38], unsuitable for MIMO systems.

As DPD is a signal processing method that is deployed in single-chain transmitters and avoids problems that arise with isolator uses, it is desirable to modify the DPD presented in Section 2.1.4 to compensate for antenna crosstalk-induced non-linearity in addition to intrinsic PA non-linearity. Thus, one solution that circumvents the mentioned shortcomings is a more complex DPD. To minimize complexity, this DPD should not require complete knowledge of all signals to be transmitted by all other antennas. In the extreme case, only a second signal is input to the DPD that summarizes the crosstalk effect of surrounding antennas. Hence it is referred to as the dual-input single-output (DISO) DPD.

### 2.2.3 Dual-Input-Single-Output Digital Predistortion for MIMO Transmitters

The effect of antenna crosstalk in MIMO transmitters is cancelled by utilizing DISO DPD. Similar to deriving SISO DPD in Section 2.1.4, DISO DPD starts with investigating the forward model to represent a MIMO system exhibiting inter-antenna cross-coupling. Consider the incident and reflected waves in each RF chain of a MIMO transmitter, shown in Fig. 2.8. As explored in [39], the mutual coupling of antennas in MIMO systems that affects the reflected wave  $\tilde{a}_{2i}$  not only exacerbates the non-linear behaviour of the PA but also affects the PA's direct current (DC) behaviour and in return, its efficiency.

Given the  $K$ -element antenna array S-parameters around the center frequency as  $\mathbf{S}_{K \times K}$ , the reflected wave  $\tilde{a}_{2i}$  from the antenna at any chain  $i$  can be written in terms of the incident wave  $\tilde{b}_{2i}$  from  $K$  antennas as:

$$\tilde{a}_{2i}[n] = \sum_{k=1}^K s_{ik} \tilde{b}_{2k}[n]. \quad (2.14)$$

The output  $\tilde{b}_{2i}$  of the  $i^{\text{th}}$  PA depends on both its input  $\tilde{a}_{1i}$  and the reflected wave  $\tilde{a}_{2i}$  on its output port [40]. Hence, based on (2.14), the output of each PA in a MIMO system also depends on the PA output of each of its  $K-1$  neighbouring chains. Similar to the derivation of Volterra forward models in Section 2.1.4, forward models for PAs in MIMO systems have been proposed in [41]. Given that the Volterra expansion is not of practical value, pruned models are addressed next. Pruning is a technique for reducing the complexity of a trained DPD, with the goal of decreasing the computational requirements without sacrificing its performance. A common pruning method is to remove the least important terms (i.e., those with the smallest absolute value coefficient) from the expansion that describes the predistorted signal  $\tilde{z}[n]$ .

In the literature, different assumptions have been made regarding the coupled signals and their subsequent effect on the linear and non-linear distortion in a specific transmitter chain. The MIMO DPD models in [42, 43] cancel non-linear crosstalk happening before PAs but do not compensate for non-linear crosstalk occurring after the PA. In [44], the authors propose a DPD model based on the assumption that antenna crosstalk only introduces linear distortion.

In [2], the authors assume both linear and nonlinear crosstalk after the PA. Their proposed DPD model comprises two blocks; a linear crosstalk and mismatch (crosstalk and mismatch model (CTMM)) block that models antenna coupling and mismatch and

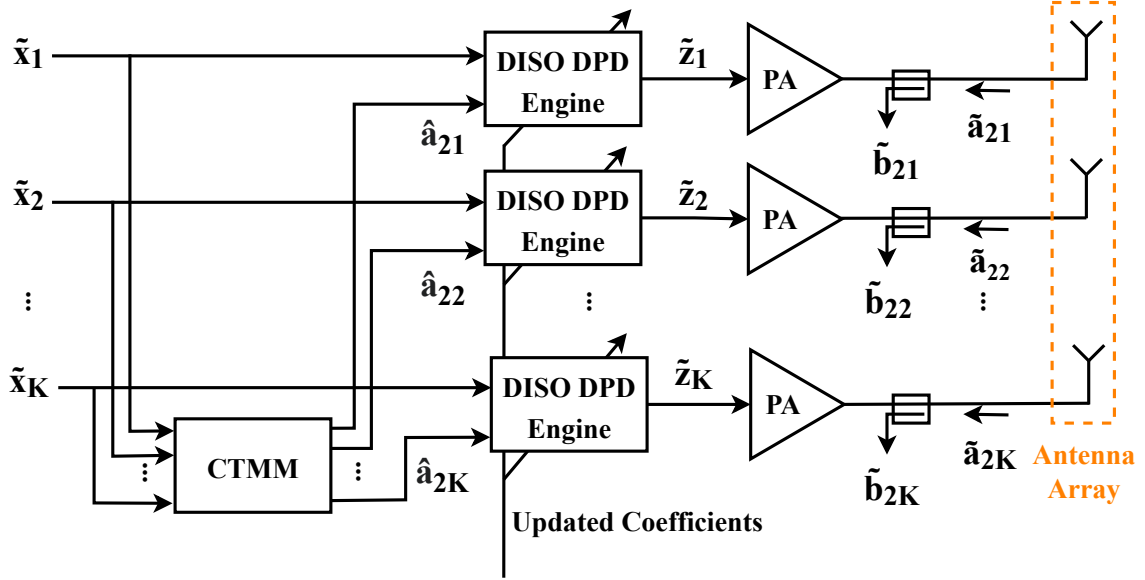


Figure 2.9: Structure of the DISO DPD proposed in [2], consisting of a DISO DPD engine in each transmission chain and a shared CTMM module.

that is shared between all transmitter chains, and a non-linear dual-input DPD unit in each transmitter chain (see Fig. 2.9). The function of the CTMM block is to compute the estimates  $\hat{a}_{2i}$  of  $\tilde{a}_{2i}$  in (2.14). The  $k^{\text{th}}$  dual-input DPD output  $\tilde{z}_k[n]$  is proposed in the following manner:

$$\begin{aligned}
\tilde{z}_k[n] = & \sum_{p=0}^{P_0-1} \sum_{m_0=0}^{M_0} \alpha_{p,m_0} \tilde{x}[n-m_0] |\tilde{x}[n-m_0]|^p \\
& + \sum_{p=0}^{P_1-1} \sum_{m_0=0}^{M_1} \sum_{m_1=0}^{M_2} \beta_{p,m_0,m_1} \hat{a}_{2k}[n-m_0] |\tilde{x}_k[n-m_1]|^p \\
& + \sum_{p=0}^{P_2-3} \sum_{m_0=0}^{M_3} \sum_{m_1=0}^{M_4} \gamma_{p,m_0,m_1} \hat{a}_{2k}^*[n-m_0] \tilde{x}_k^2[n-m_1] |\tilde{x}_k[n-m_1]|^p.
\end{aligned} \tag{2.15}$$

In summary, numerous works exist in the literature to circumvent the adverse effects of antenna crosstalk by augmenting existing DPD methods. The approach presented in [2] is the most general solution as it makes no narrowing assumptions on the type of crosstalk between adjacent antenna. That being said, research into DSP methods for

antenna crosstalk compensation lacks physical model analysis regarding the circuit-level effects of crosstalk in the RF frontend.

## 2.3 Review of Existing DPD Hardware Implementations for Single Chain and MIMO Transmitters

Aside from theoretical work done on digital predistortion systems, the practical implementation of such systems is of high importance. This is specifically true in 5G systems which deploy a high count of base stations compared to previous generations of networks, necessitating power consumption reduction and resource reduction in hardware implementation. As part of the process of deploying signal processing methods in industry, initially, evaluation of DSP methods is done by implementing the methods on an field-programmable gate array (FPGA) and results are assessed in terms of resource utilization, power consumption and performance. Although FPGAs provide a re-programmable platform to verify signal processing algorithms, using them in the field is debatable. Compared to application-specific integrated circuit (ASIC) chips, FPGAs have high unit cost, high power consumption and greater unit size. After evaluating DSP methods' performance, satisfactory methods are hardened on an ASIC and used in deployed systems. Due to the widespread availability of FPGA circuits and simpler design flow compared to ASICs, hardware optimization is explored using FPGA chips in the literature.

Research on reducing complexity and power overhead in predistortion systems can be divided into two categories: DPD model simplification and hardware design methods (see Fig. 2.10). Efforts in DPD model simplification can be divided into three subcategories: (1) basis complexity reduction [15, 16], (2) Lookup table (LUT)-based models [8, 9, 10, 11] and (3) Piecewise-linear models (piece-wise linear (PWL)), also known as magnitude selective affine models [9, 17, 18]. Likewise, hardware design methods can also be divided into three subcategories: (1) realizing any model through LUTs [12, 13, 14], (2) optimizing the datapath [45], and (3) parallel-processing methods [19]. In this section, the work previously done in each category and relevant to this thesis is reviewed.

Many research works in the literature propose a cascade of two DSP units as a DPD solution in order to simplify the complexity of each individual unit. In [15], the authors propose a feedback memory topology that contains linear and non-linear memory effects as the unit cascaded with the memoryless DPD engine. This simplifies the DPD structure, albeit requiring two sets of coefficients and their respective training. In addition, the authors propose a coefficient training algorithm that is  $O(n)$  complex, allowing for the coefficient training to be done online on the FPGA. Similar to [15], cascaded DPD is used

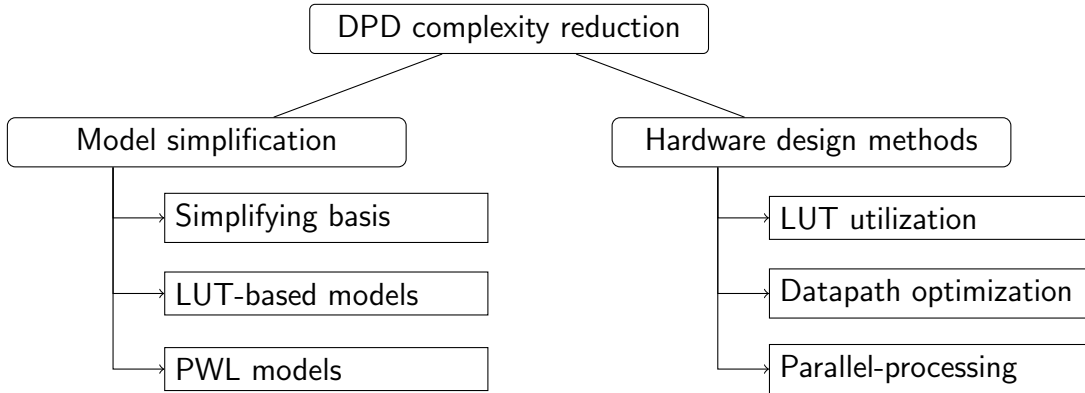


Figure 2.10: Categorized prior work on complexity reduction of the predistorter

in [16]. In [16], DPD is proposed for a directional modulation system comprising multiple transmitter chains and should include a predistorter in each chain. A common DPD block is used for all chains, followed by a separate auxiliary DPD block in cascade per chain to fine-tune the remaining non-linearity. It was claimed that this scheme would be suitable for hardware implementation, however, no results were presented regarding resource utilization to verify this claim.

The essence of LUT-based DPD models is to use a predefined table of values that can be used to compensate for the nonlinear distortions introduced by the PA. In a LUT-based DPD, the input signal is first processed to generate a set of input-output characteristics, which are then used to create the lookup table. The LUT is typically implemented as a memory array that can be quickly accessed in real time, and it contains the predistortion values that correspond to different input signal levels. When the input signal is applied to the power amplifier, the LUT is used to apply the appropriate predistortion values to the signal, so that the output of the amplifier is as close as possible to the desired linear response. The main benefits of LUT-based predistortion is that it can correct for a wide range of nonlinear distortions, it can be implemented in real-time, it has low complexity and it requires less computation power compared to other techniques such as polynomial modelling. However, the LUT-based DPD requires a high-resolution memory table and it is not so adaptable to changes in the PA's characteristics like temperature or aging.

The work presented by the authors in [8, 10] is based on the PA behavioural model introduced in [8]. In [8], the authors implement the DPD engine on an FPGA using a LUT-based implementation, and DPD training is done offline on a co-processor. This is followed by [10], in which the training is transferred to the FPGA, allowing for a full online DPD training and application. Resource utilization is reported in [10], showing a

reduction in FPGA processing slices at the cost of increased block ram (BRAM) count and low clocking frequency (40 MHz). BRAMs make significant contributions to the overall power consumption of a hardware implementation and it is preferred that they be reduced or altogether avoided.

Another LUT-based DPD is that in [11]. This DPD provides a fair trade-off between BRAM, processing slices and clocking frequency (92.16 MHz) and is based on the well-known memory-polynomial (MP) model. First, the MP model is manipulated to achieve a lower-complexity implementation. Then each section of the model is implemented using independent “adaptive” LUT units whilst allowing flexible memory depth. However, in practice, the LUT units are replaced with BRAMs, whose count is subject to increase with an increase in memory depth. Power consumption is not reported, however, an increased BRAM count leads to an increase in power consumption.

The piecewise linear DPD model (PWL) is another prevalent type of model used to correct for nonlinear distortions of the PA. It is a variation of the LUT-based DPD method where the LUT is divided into several regions, each of which has a different predistortion function. In a PWL DPD system, the input signal is first divided into multiple segments using a set of predefined threshold levels. Each segment is then processed using a separate linear predistortion function. The goal is to find the appropriate predistortion function for each segment, such that the output of the PA is as close as possible to the desired linear response.

The work presented in [9] exhibits a DPD formulation that is both a piece-wise model and implementable through LUTs. By providing a piece-wise model, basis functions are significantly simplified in terms of hardware complexity. What distinguishes [9] from the research in [17, 18] on PWL models is that the authors also ensure continuity between intervals. Although this model was not implemented in hardware, it is evident that it would lead to significant resource optimizations on the FPGA, while being independent of the DPD model chosen. Hence, this work’s piece-wise formulations have been adapted in this thesis.

Two other noteworthy works adapting a piecewise DPD model are [17] and [18]. As one of the earliest works in PWL DPD, the authors in [17] determine a partitioning strategy and allow for any Volterra-based DPD model to be implemented in their vector-switched model. Hardware implementation is not available in this work; however, complexity analysis is reported in terms of complex floating-point operations per sample (FLOPS). In [18], the authors implement a magnitude selective affine PWL model to substitute non-linear crossterms. The proposed model was implemented on an FPGA and complexity was reported in terms of the number of coefficients, registers and processing slices used.

Although the model used in [18] is questionable regarding instability on interval limits due to discontinuity, it achieves low complexity and fair FPGA clock frequency (260 MHz).

Several hardware implementation optimization techniques exist that can be used to improve the performance and efficiency of a real-time DPD system. The works in [13], [12], [14], [45] and [19] all use hardware implementation techniques to further optimize resource usage and clocking frequency of the FPGA. The notable work in [14] and [12] both use LUTs in hardware implementation, both works acknowledge using high BRAM and processing resource units and provide no alternative solutions. In [13], a Volterra series-based DPD is implemented in hardware using gain-indexed LUTs with time-shared multiplier usage to decrease resource count. FPGA resource utilization was reported and showed low processing resources but high BRAM count. The authors also mention that while sharing processing resources allows for optimized resource usage, it requires data to be processed in serial, resulting in high processing clock rates.

The authors in [19] provide an elegant solution to meet high bandwidth standards in 5G NR: parallel processing. To further increase the processing speed, this solution uses multiple DPD sub-units in parallel to perform the computations. In other words, this solution takes advantage of the parallel data feeding properties of state-of-the-art data converters by generating the signal that is input to the data converters in parallel streams. Although the resource count for parallel processing increases with parallel stream count, parallel processing is key to meeting high bandwidth requirements within the constraints of the maximum digital clocking rates currently available. In [46, 47], it is demonstrated that SISO DPD can linearize the main beam in analog beamforming MIMO systems. Thus, the parallel-processing method is used to implement SISO DPD in [19] and linearization performance is evaluated using a 64-element analog beamforming array.

At the time of writing this thesis, the author’s investigation of the hardware implementation of DPD solutions for fully-digital MIMO systems was inconclusive.

## 2.4 Review of Real-time MIMO Transmitter Testbeds

Noteworthy experimental platforms have been developed in academia developed to demonstrate the potential of MIMO transceivers. These include LuMaMi (Lund University and University of Bristol)[20], ESAT lab’s product (KU Leuven University) [21], Argos (Rice University)[22] and State Key Lab of Millimeter Waves’ array (Southeast University of China) [23].

The platforms introduced in [20] and [21] are identical and use commercially available software-defined-radios (SDR) as the transceiver unit (Fig. 2.11). These systems can ac-





Figure 2.11: LuMaMi of Lund University (left) and of KU Leuven (right).

commodate up to 128 transmitter chains and operate in sub-6 GHz frequency but only allow for 40 MHz of instantaneous bandwidth. The platforms use up to 50 FPGA processors for coherent signal transmission and synchronization. The commercial SDRs in these testbeds limit the configurability of the RF frontend which in turn limits the applications of this testbed. These platforms have been used in research regarding OTA channel sounding, validating massive MIMO's proof-of-concept, MIMO cloud-RAN and relaying.

On the other hand, the Argos platform from Rice University presented in [22] uses an in-house designed SDR product and is developed for research in multi-user MIMO (MU-MIMO), channel sounding and self-interference reduction. This platform supports up to 64 base station transmitter chains and 15 users and operates at sub-6 GHz frequencies with a maximum bandwidth of 20 MHz. Pairs of SDRs are controlled and synchronized using one FPGA unit. Although the customized SDR provides RF frontend flexibility, the supported bandwidth is insufficient to exploit the full potential of a 5G MIMO system.

Most recently, Southeast University in China presented a 64-chain fully digital testbed targeting 3.4-3.6 GHz [23]. This testbed is claimed to support 200 MHz of signal bandwidth in each chain using one processing FPGA for every few chains. Although this testbed is promising in characteristics, no further measurements or results have been reported other than those in [23].

At the time of writing this thesis, only the aforementioned experimental platforms were known to the author.

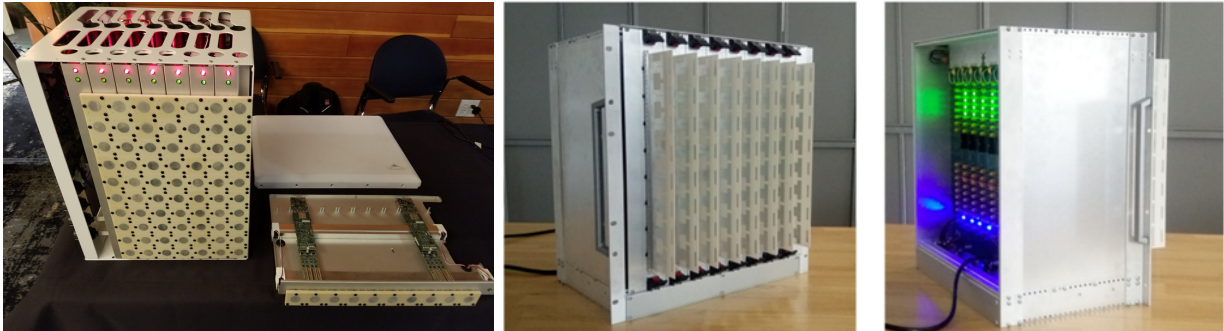


Figure 2.12: Argos of Rice University (left) and Southeast University’s testbed (right).

## 2.5 Discussion

In this chapter, MIMO systems were introduced as enabling technology in fifth-generation wireless networks. Furthermore, DPD was instigated as a signal processing method that mitigates non-linearities of the RF chain. However, migrating to MIMO systems requires more complex and resource-demanding DPD units due to inter-antenna coupling. An expansive review of DPD hardware implementations was presented. Experimental platforms that are available globally in academia for experimenting on MIMO capabilities were examined. However, there is a noticeable gap between the MIMO technology that is to be deployed in 5G networks and the current MIMO systems, as promised data rates in 5G systems have yet to be exploited. Furthermore, there is no reconfigurable platform that provides flexibility in the RF front-end and allows for exploring linearization of MIMO systems in a real-time system that is similar to what is to be deployed in 5G base stations.

The research that is presented in the following chapters of this thesis aims to provide the central baseband processing unit for a reconfigurable real-time testbed that will rapidly evaluate DPD and signal processing algorithms for the feasibility and efficiency of their design under real conditions of a MIMO system. The testbed that is developed as part of the contribution of this thesis is capable of providing up to 16 chains of RF-modulated signals with up to 1.5 GHz of linearization bandwidth for any custom-designed RF front-end (PA and antenna). Using a combination of hardware implementation techniques that were reviewed in this section, an optimized DPD algorithm is proposed for MIMO systems. The developed testbed provides the experimental platform on which the DPD’s performance is evaluated.

# Chapter 3

## Real-time Testbed Development and Implementation for DSP Applications in Fully Digital MIMO Transmitters

### 3.1 Introduction

Fifth-generation telecommunication networks aim to achieve data rates in the order of gigabits-per-second, almost 20 times that of LTE networks. In order to reach these data rates, modulated signals, with hundreds of megahertz of bandwidth are to be used in conjunction with MIMO technology. Massive MIMO is envisaged to deploy hundreds of parallel transmitter chains in the base station that operate simultaneously, facilitating high data rates. MIMO also ensures improved signal quality as it is known to provide transmit diversity to overcome multi-path fading in wireless channels.

Each transmitter chain in the base station is equipped with a PA to provide the necessary output power for the transmitted signal in that chain. As an inseparable part of PA operation, real-time DPD is deployed in the digital domain of every chain to linearize the transmitted signal. In order to capture the OOB emissions of the PAs, a linearization bandwidth of around 5 times that of the modulated signal bandwidth is required. The DPD modules are implemented in hardware to allow real-time operation. However, the maximum fabric clock of a commercially-available FPGA cannot accommodate the required linearization bandwidths of 5G systems.

Digital chips are also limited in computational resources. The number of digital transmitter chains implemented on each digital chip is restricted by the total available processing and reprocessing resources needed for each DPD module. Therefore it is critical that in the development stage, proposed MIMO DPD methods are evaluated in terms of resource consumption. This calls for a reconfigurable transmitter testbed that allows multi-chain transmission with a flexible digital back-end that can evaluate signal processing methods developed for MIMO systems, including DPD.

It can be argued that multiple digital chips, specifically FPGAs, be used, where each FPGA houses a subset of the total chains, such that the computational burden of MIMO DPD is distributed among them. This poses additional complexities. The multiple FPGAs must be synchronized using a reference clock and common trigger. Furthermore, MIMO DPD may require knowledge of signals of adjacent chains. Thus, a high-speed data transfer protocol must be in place between FPGA chips.

In this chapter, the steps for developing the MIMO testbed are described, discussing requirements along with system design considerations. Next, the challenges of developing the mentioned testbed on an off-the-shelf FPGA platform are presented along with measures taken to overcome these challenges. A complexity-reduced DPD for single-chain transmitters is then presented, along with a parallelization scheme that allows wide linearization bandwidths while meeting hardware timing constraints. To conclude this chapter, the proposed DPD is realized on the MIMO testbed to verify the single-chain operation of the testbed.

## 3.2 Real-time FPGA-based Testbed Requirements

The first step in developing the MIMO testbed is to specify the requirements of the testbed. The requirements, outlined in Table 3.1, are defined such that the transmitted signal meets signal standards of 5G base stations, i.e., as specified in [24], and that the testbed allows for scalability of the number of MIMO chains and supports modular insertion of DSP blocks, which in this work is the DPD engine.

The testbed comprises two main parts: the digital processing hardware and the radio hardware. The digital processing hardware is the digital backbone of the testbed that includes signal generation, streaming memory blocks and DSP modules. The signal is then passed on to the radio hardware, which houses the data converters that convert the digital signal into an analog signal.

The digital processing hardware is more or less independent of the frequency and bandwidth requirements. The processing hardware needs only to be reconfigurable in order

Number of Chains	16
Frequency	3.3 - 3.8 GHz
Modulated Signal Bandwidth	$\geq 100$ MHz
Processing Hardware	FPGA

Table 3.1: Real-time MIMO testbed requirements.

to be used as an experimental testbed for different DSP methods, have a fabric clock of hundreds of megahertz and have the capability to potentially connect to a PC through an ethernet port. Establishing a connection to the PC is advantageous since it provides convenient development tools and interfaces. This connection also enables the option of using a PC as an additional processing block in the preliminary development of DSP algorithms with the testbed operated as an arbitrary waveform transmitter. In this case, the PC generates the final signal to be transmitted, uploads the waveform to the testbed and the testbed converts the signal to an analog waveform in the desired RF frequency.

On the other hand, the radio hardware is limited by the number of chains, the frequency band of interest (that is the N78 band from 3.3 to 3.8 GHz), and modulated signal bandwidth requirements. Inter-chains synchronization and coherence are critical when choosing radio hardware as well as verifying that there exists a master synchronization procedure that would allow the expansion of the platform to include a higher number of coherent chains. The chosen radio hardware must contain data converters with sampling rates in the order of a few Giga Samples per second (GSps) to allow for the transmission of a linearization bandwidth approximately 5 times that of the modulated signal bandwidth. Moreover, the data converters should allow internal mixing of the baseband signals to the targeted N78 band frequency range.

Table 3.2 shows a comparison of available commercial products that qualify for serving as the digital baseband domain of the MIMO testbed. After careful comparison between commercially available products that could serve as the digital baseband platform of the testbed, the Xilinx Ultrascale+ RFSoc Evaluation Board was chosen. This product is especially attractive in terms of price per chain, given the fact that multiple of these boards can be synchronized to provide a scaled MIMO system with a higher number of chains. It is thus deemed a valuable testbed platform. The caveat is its long development time, requiring the developer to control the smallest of details in order to develop the testbed. Meanwhile, other off-the-shelf products are much more expensive but provide a plug-and-play option for the user.

	Keysight	Analog Devices	National Instruments	Xilinx RFSoc
	M5000 Series	QUAD-MxFE	USRP X410	ZCU111 ZCU216
DAC	Num. Chains Bit Resolution Sampling Rate	16 16 12 GSps	4 14 N/A	8 14 6.5 GSps 9.85 GSps
ADC	Num. Chains Bit Resolution Sampling Rate	16 12 4 GSps	4 12 N/A	8 12 4 GSps 2.5 GSps
FPGA	Model DSP Slices Memory	Xilinx VCU118 6.8K 2 x 4 GB	Xilinx ZU28DR 4.2K 3 x 4 GB	Xilinx ZU28DR 4.2K 2 x 4 GB
Sync	Ref In Ref Out Trigger IO	Yes Possible Possible	Yes N/A Yes	Yes Possible Possible
	Max. Operating Frequency Capable Bandwidth Max. Output Power Price per chain (\$CAD)	16 GHz 2 GHz N/A 7 dBm 4.5K	7.5 GHz N/A 400 MHz <23 dBm 8.1K	8 GHz 4 GHz 6 GHz N/A 0 dBm 1.35K 1.65K

Table 3.2: Commercial products considered to serve as the digital baseband domain of the testbed. Cells populated with ‘N/A’ were not declared by the manufacturing company.

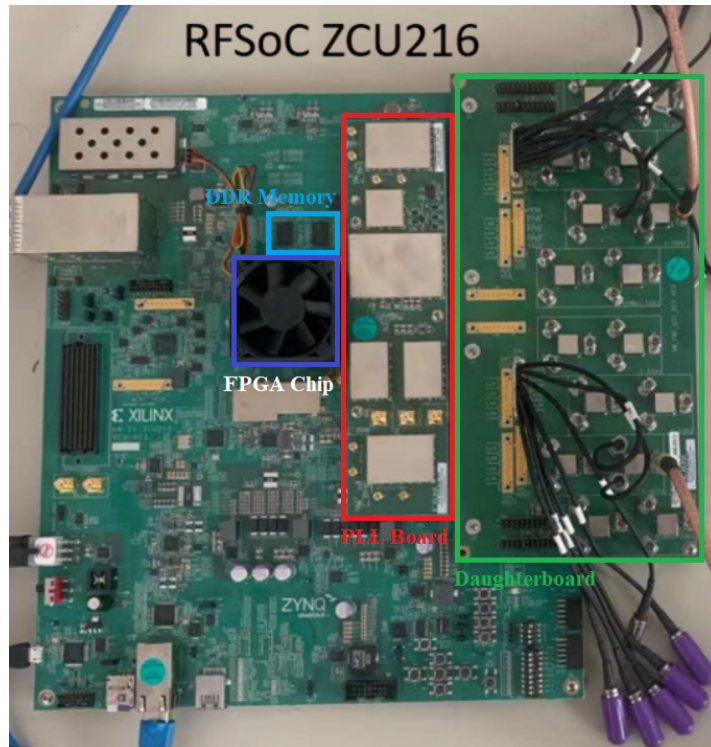


Figure 3.1: The chosen hardware for the MIMO testbed, Xilinx RFSoc ZCU216 Evaluation Board with the daughterboard and PLL board attached.

The characteristics of two Xilinx RFSoc evaluation boards can be seen in the last two columns of Table 3.2. In order to verify the feasibility of MIMO testbed design, the less-costly ZCU111 board was first used as a prototype. This evaluation board could not meet the requirements in Table 3.1 due to limited chain count and limited sampling rate but did meet performance requirements for lower operation frequency and signal bandwidth.

The final testbed was implemented on the ZCU216 board. This evaluation board is shown in Fig. 3.1 and consists of the RFSoc FPGA chip, a phase-locked loop (PLL) board that generates the clock rates for the data converters and a daughter board. The daughterboard conditions the analog voltage output of the integrated data converters by passing the signal through a balun (differential to the single-ended converter) and low-pass filter for all 16 chains.

A block diagram of the MIMO testbed after implementation is shown in Fig. 3.2. The signal processing hardware of the testbed consists of two hardware platforms: a PC and an FPGA evaluation board. The PC is used to generate the signal, perform addition

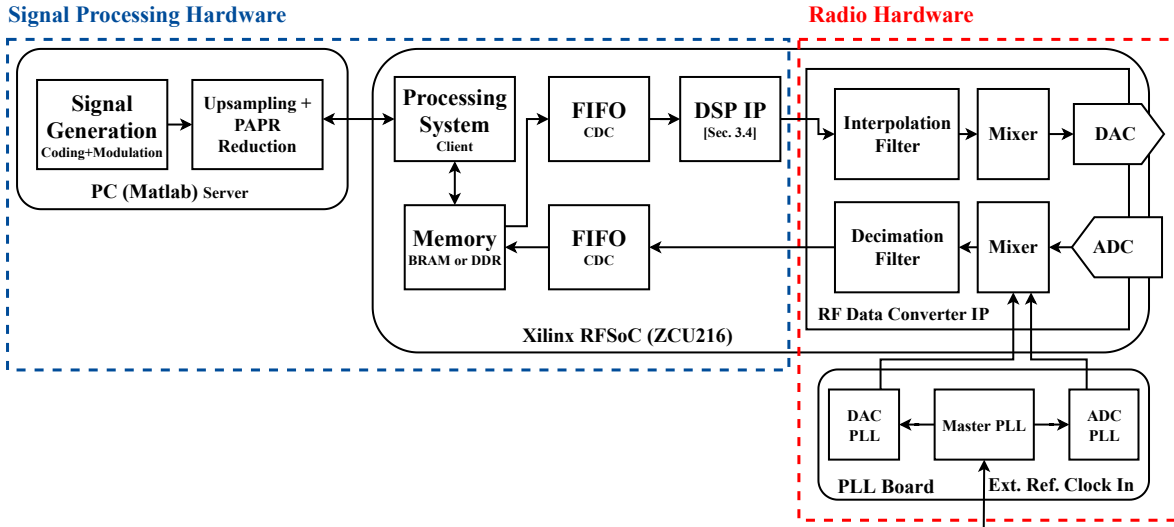


Figure 3.2: Simplified diagram showing the components of the processing hardware and radio hardware. For simplicity, it is assumed that only one transceiver path is enabled, however, multiple chains are active in a MIMO system.

precoding (if needed) and over-sample the modulated signal to occupy the linearization bandwidth. Next, the signal is transferred using the UDP protocol through ethernet where it is received by the ARM processor on the evaluation board. The signal is stored on FPGA memory to allow looped streaming of the signal. When the user commands signal transmission from the PC, the signal is streamed from the memory to the radio hardware.

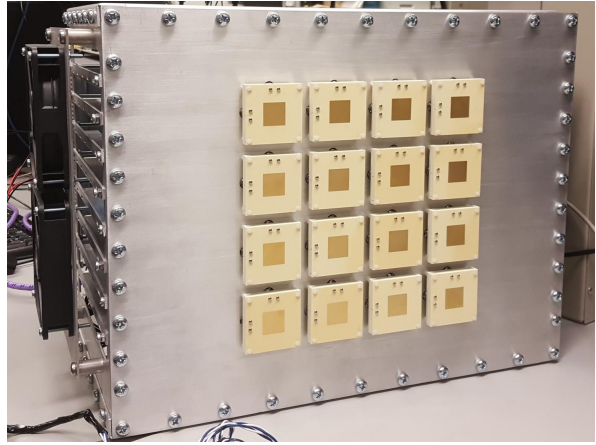
While the design of the signal processing flow is platform-agnostic, the radio hardware configurations and capabilities are platform-specific. The radio hardware of the Xilinx RFSoc comprises the RF Data Converter IP and the PLLs. The configurations of the data converters must be set jointly with the PLLs. Details regarding the radio hardware settings are presented in Section 3.3.4.

Finally, the completed real-time testbed for fully digital MIMO is pictured in Fig. 3.3. The PC (not pictured) is connected to the RFSoc evaluation board which is then connected to the RF front end. The front end consists of 16 PAs and 16 antenna elements, placed inside a mechanical frame.

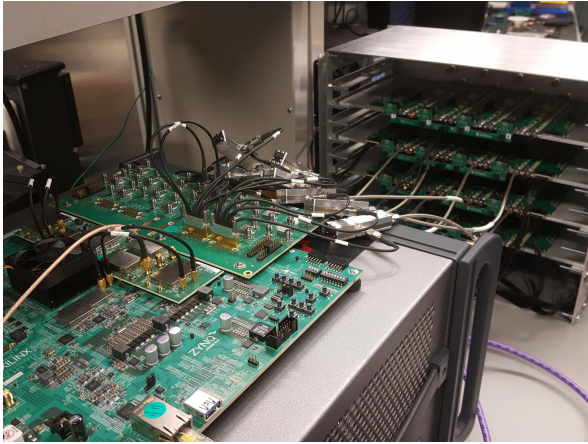




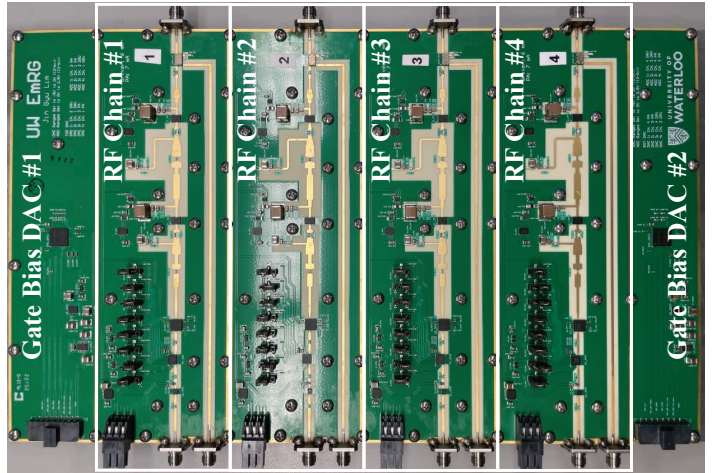
(a) RF front-end showing 4 blades, each housing 4 two-stage PAs.



(b) 16-element antenna array placed in the mechanical frame.



(c) FPGA board.



(d) Instance of one fabricated RF front-end panel comprised of four chains.

Figure 3.3: EmRG 16-chain fully digital MIMO testbed.

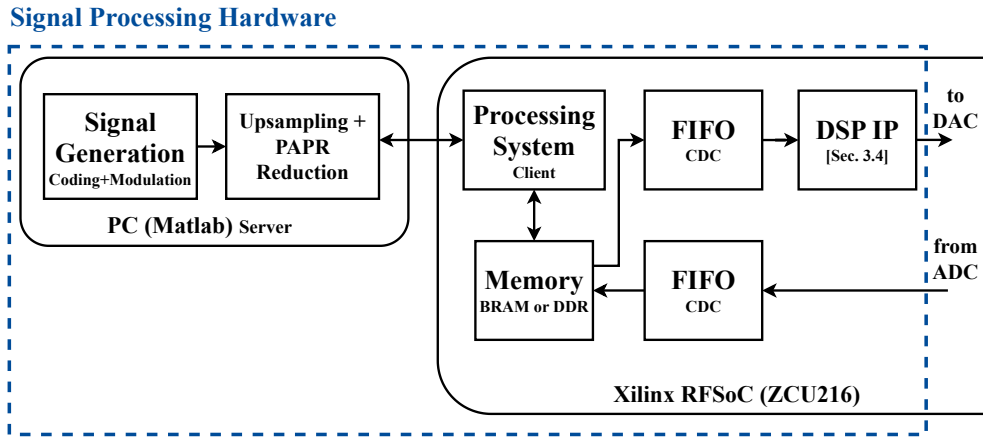


Figure 3.4: Block diagram showing the components of the signal processing hardware

### 3.3 Testbed Implementation Considerations

The implementation of the MIMO testbed is complex with design challenges ranging from user interface (UI) design for ease of control to adjusting data converter configuration in order to achieve desired bandwidth and center frequency. In this section, the challenges are organized based on their implementation platform and the design solution for implementation is discussed.

#### 3.3.1 Server-based User Interface Design

To provide ease of access to any user using the testbed, regardless of their knowledge of the digital design in the processing hardware, the UI is implemented in Matlab. Previously, for commercial equipment available in the lab, there existed a UI that allowed the user to choose the transmitting instrument and the instrument that will be used to receive the signal. The MIMO platform control functions are written so as to perform similar to commercial instruments available in the lab. This allows the user to mix and match their transmitting and receiving instruments between the MIMO platform and other available transceiver equipment in the lab.

When called on, the platform is communicated with, through a server/client connection with Matlab. The communication protocol used is TCP, in order to ensure intact and continuous signal transmission from the PC to the platform hardware. Known as the control hardware on the FPGA evaluation board, the ARM processing system (PS) contains

the server that receives the command to be carried out and asserts the necessary signals on the programmable logic (PL) datapath based on the received command. This server also includes the ability to upload and download signals of arbitrary length to and from the memory in the FPGA PL. Although the UI design is mostly independent of the hardware chosen for the MIMO testbed, it is important to adjust the memory address and signal sampling rate based on what is implemented in the hardware.

### 3.3.2 Signal Streaming Memory

There are two types of memory available when developing an FPGA-based platform for signal streaming: PL-integrated memory (i.e., BRAM) and off-chip DDR memory (i.e., DRAM). With BRAM memory, the memory is realized on the FPGA logic, thus access latency is low and data bandwidth is high. Nevertheless, space is limited and on-chip routing may be compromised due to extra routing to BRAMs. However, DRAMs are separate hardware from the FPGA and therefore access latency is comparably high. DRAMs operate with an independent clock which limits their data bandwidth considerably. The benefit of using DRAM memory for data storage and streaming is the ample 4 GB storage space without compromising routing. In both cases, data movement is managed through the data memory access (DMA) module. To avoid increasing the number of clock domains in the datapath and the problems arising from multiple clock domains, when designing with DRAM memory, the DRAM clock is used as the PL datapath clock. In this work, two separate datapaths were designed, one including BRAM as the streaming memory and the other using DRAM, allowing the option of low latency or access to larger memory, depending on the application and desired data rate.

### 3.3.3 Clock Domain Crossing

The processing hardware is designed to operate regardless of the radio hardware chosen. The basis of this design is that the processing hardware and radio hardware operate on different clocks that are generated using a common reference clock. Thus, data in the datapath design experiences clock domain crossing (CDC). CDCs are notorious for causing metastability. This happens when the signal transferring to the destination clock domain transitions very close to the active edge of the destination clock, leading to setup or hold violations in the destination domain. The signal may then oscillate for an unknown amount of time, leading to an unstable signal when the next active edge arrives. If unaddressed, metastability may lead to unknown signal states, high current flow and

even FPGA burnout. In the MIMO testbed design, metastability is avoided by adding two stages of registers before crossing clock domains. This is a widespread practice in CDC datapath design.

### 3.3.4 Data Converter IP Configurations

The UI and processing hardware design is mostly independent of the commercial brand of the chosen radio hardware. The radio hardware configurations are what dictate the PL data rate and data stream quality fed to the data converters. These configurations include the data converter sampling rate, parallel stream count, interpolation factor and Nyquist zone operation. In the following, the most effective configurations and their effects on the data rate are discussed.

#### Clock Generation with PLL Board

The DACs on the ZCU216 RFSoc board are capable of operating at sampling rates up to 9.85 GSps. Generating high sampling rates requires a PLL with high-quality clock output. The ZCU216 board contains a dedicated clock generating board that contains one master PLL that uses a reference clock (external or internally generated) to produce an intermediate clock. This intermediate clock is then passed to dedicated PLLs for both DAC and ADC. These PLLs contain programmable registers that can be adjusted to provide any arbitrary clock rate. Programming these registers is done during the board start-up routine and clock generation, thus sampling rate adjustment can only be done by reprogramming the FPGA when all clock operations have halted and are being rebooted.

#### High-Speed Data Configuration

Current maximum FPGA clock rates range up to hundreds of MHz whereas the data converter sampling rates required for wide-band signal transmission are in the order of GHz. In order to make this possible, data converters are equipped with interpolation filters and parallelized processing. Interpolation filters interpolate the low-rate signal received to a higher-rate signal without changing the modulated signal bandwidth. Data converters also achieve high sampling rates by receiving data at lower rates in  $P$  parallel streams, registering them in FIFOs and processing  $P$  samples per clock cycle.

Adjusting these parameters comes with their respective costs. Given a targeted data converter sampling rate, a high interpolation factor means a lower modulated signal band-

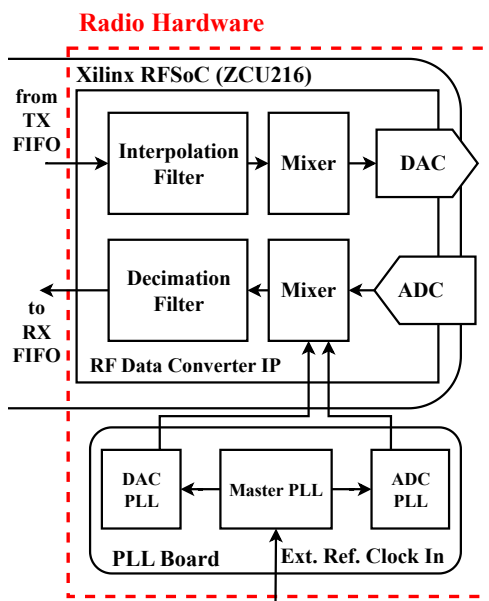


Figure 3.5: Block diagram showing the components of the radio hardware

width. In the same way, while more parallel streams mean achieving higher sampling rates, it results in increased usage of processing resources in the DSP block since each parallel stream needs its own dedicated processing resource. This tradeoff is further discussed in Section 4.2.

Finally, higher data converter sampling rates pose strict data stream requirements on the stream they receive, as they cannot afford to receive data in bursts or split streams. This may require extra registers to build a continuous data stream in the processing hardware datapath.

### Nyquist Zone Operation

The Nyquist theorem states that a signal can be reconstructed only if it is sampled at least twice that of its highest apparent frequency. For a data converter, the region between the DC frequency and half of the sampling rate is called the first Nyquist zone, followed by higher Nyquist zones at every integer multiple of half the sampling rate. Given a sampling rate of  $F_s$  Samples per second, the  $n$ -th Nyquist zone is then located in  $(n - 1) \cdot \frac{F_s}{2} < f < n \cdot \frac{F_s}{2}$ . Although the signal to be transmitted can be located within any of these zones, its images will appear in all other zones. It is to note that the mixer block in Fig. 3.5 can only

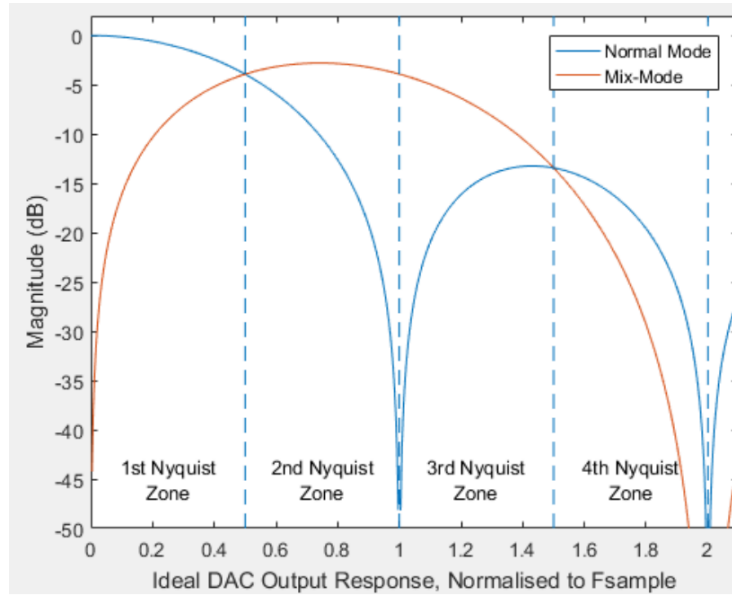


Figure 3.6: Different Nyquist zone operations made possible using two different reconstruction waveforms on the Xilinx RFSoc [3].

up-convert the signal to frequencies within the range of 80% that of the set sampling rate. For example, if the DAC sampling rate is set to 7.5 GSps, then the mixer can up-convert the signal to any frequency up to 6 GHz.

In the transmitter chain, the DAC can use different reconstruction waveforms to produce the analog waveform from its input digital samples. Depending on the type of reconstruction waveform used, the signal in one of the Nyquist zones is transmitted and all other image signals are attenuated significantly in the DAC output. Fig. 3.6 shows the two different modes of operation on the Xilinx RFSoc board [3]. The normal mode waveform is used to transmit signals in the first Nyquist, with the second Nyquist image attenuated drastically while the mix-mode waveform is used to transmit signals in the second Nyquist zone with its odd Nyquist image attenuated, but less so than that in normal mode. In order to transmit a signal in the N78 band as per the requirements in Table 3.1, the sampling rate can be set in two ways: either a lower sampling rate placing the target frequency band in the second Nyquist zone or a higher sampling rate placing the targeted band in the first Nyquist zone, each at their own cost. In practice, using mix mode on the RFSoc shows a significantly high-power image signal in the first Nyquist. To prevent the image signal from reaching the rest of the transmitter RF chain, additional external band-pass filters on the DAC output are required, which is costly and attenuates the output signal. Ultimately,

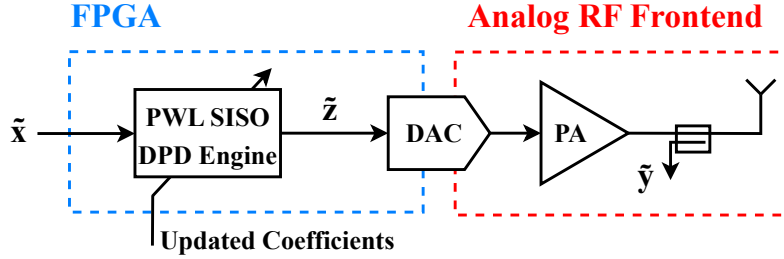


Figure 3.7: Proposed real-time PWL SISO DPD, implemented in FPGA hardware.

the RFSoc DAC is used at a high sampling rate that would put the N78 band in the first Nyquist. In this case, the second Nyquist image is attenuated by the normal-mode operation and additional low-pass filters are placed at the output of the DAC. Low-pass filters are less costly than band-pass filters and more versatile, as they can be used in other applications geared towards other frequency bands.

### 3.4 SISO DPD Engine for Single-Chain Transmitters

It is desirable that a single-chain transmitter of the developed testbed is tested before expanding it to a 16-chain transmitter. In the following section, the developed testbed is evaluated under modulated signals. This evaluation was done in two stages. First, the testbed is tested under the absence of the DPD IP block in Fig. 3.2. This intermediate step is part of the development process to ensure that the generated signal in the Matlab server is correctly transferred through the datapath and converted to an analog signal on all enabled transmitter chains. Next, after being independently validated in a hardware language test-bench simulation, the DPD IP block is inserted into the previously validated datapath and its performance is evaluated against a PC-based DPD engine. In this section, a piecewise linear (PWL) SISO DPD engine is described and tested on a single chain of the developed testbed.

The proposed real-time DPD for a single-chain transmitter is presented in Fig. 3.7. In this figure, all signals are CBBE. Specifically, the input to the PA before DPD is given by  $\tilde{x}$ , the predistorted signal fed to the PA is  $\tilde{z}$ , and the output of the PA is  $\tilde{y}$ .

In the section that follows, the theory of the SISO PWL DPD model is described in detail. The model is then converted into  $P$  parallel streams to relax timing constraints and allow for wide linearization bandwidths and optimized for less processing resource usage.

### 3.4.1 Piecewise SISO DPD Formulation

The DPD model used in this thesis uses a modified PWL model with the squared magnitude of the input signal  $x[n]$  to eliminate the need for the square-root function and reduce hardware complexity [5]. Specifically, let  $\boldsymbol{\alpha} = (\alpha_0, \dots, \alpha_N)$  be a real-valued vector and  $\boldsymbol{\phi} = (\phi_0, \dots, \phi_N)$  be a complex-valued vector, both of length  $N + 1$ . Let  $f(x; \boldsymbol{\alpha}, \boldsymbol{\phi})$  denote the piecewise linear function that interpolates linearly between  $(\alpha_j, \phi_j)$  for  $j = 0, \dots, N$ , i.e.,

$$f(x; \boldsymbol{\alpha}, \boldsymbol{\phi}) = \phi_j + \left( \frac{\phi_{j+1} - \phi_j}{\alpha_{j+1} - \alpha_j} \right) (x - \alpha_j) = A_j x + B_j, \quad \alpha_j \leq x \leq \alpha_{j+1} \quad (3.1)$$

where  $A_j = \left( \frac{\phi_{j+1} - \phi_j}{\alpha_{j+1} - \alpha_j} \right)$  and  $B_j = \phi_j - \alpha_j A_j$  are coefficients to be determined. Fig. 3.8 shows the piece-wise linear function  $f(x; \boldsymbol{\alpha}, \boldsymbol{\phi})$  which linearly interpolates  $(\alpha_j, \phi_j)$  and  $(\alpha_{j+1}, \phi_{j+1})$  over the interval  $x \in [\alpha_j, \alpha_{j+1}]$ . The choice of  $\boldsymbol{\alpha}$ , which divides the region over which  $f(x; \boldsymbol{\alpha}, \boldsymbol{\phi})$  is defined on, is arbitrary. In the piece-linear model adapted in (3.2), the intervals equivalent to  $\boldsymbol{\alpha}$  is chosen to be equal segments of the interval  $|x[n]|^2$ , i.e.  $\alpha_j = \frac{j}{N} \max(|x[n]|^2)$ . This results in unequal segments with respect to  $x[n]$  that is  $\sqrt{\frac{j}{N}} \max(|x[n]|)$ . Using the power of  $x$  ( $|x[n]|^2$ ) instead of the magnitude of  $x$  ( $|x[n]|$ ) alleviates the need to use the square-root function in hardware implementation, thus reducing the number of processing hardware needed.

The DPD model previously introduced in (2.5) is adapted as the basis of a piece-wise linear model. The piece-wise SISO model is then given by:

$$\begin{aligned} \tilde{z}[n] &= \sum_{m=0}^M \tilde{x}[n-m] f(|\tilde{x}[n-m]|^2; \boldsymbol{\alpha}, \boldsymbol{\beta}^m) \\ &+ \sum_{m=1}^M \tilde{x}[n-m] f(|\tilde{x}[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\gamma}^m) \\ &+ \tilde{x}^2[n] \sum_{m=1}^M \tilde{x}^*[n-m] f(|\tilde{x}[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\zeta}^m) \\ &+ \tilde{x}^*[n] \sum_{m=1}^M \tilde{x}^2[n-m] f(|\tilde{x}[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\xi}^m), \end{aligned} \quad (3.2)$$

where  $\boldsymbol{\beta}^m, \boldsymbol{\gamma}^m, \boldsymbol{\zeta}^m$  and  $\boldsymbol{\xi}^m$  for  $m = 0, \dots, M$ , are vectors of coefficients to be fit. In the next section, details of the hardware implementation and the parallelization method are described.



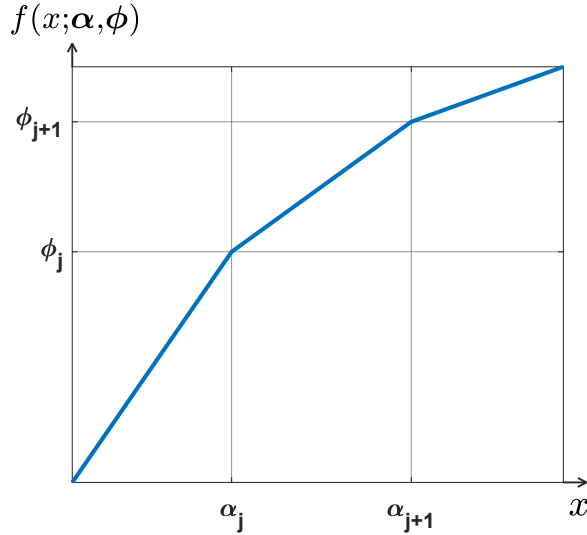


Figure 3.8: Piecewise linear function  $f(x; \alpha, \phi)$

### 3.4.2 Hardware Implementation of SISO DPD

The first step in approaching the hardware implementation of the DPD engine is to understand the hardware building blocks available on the RFSoc FPGA. This allows for platform-specific optimization in resource usage. Since the RFSoc is from the Ultrascale line of products, it uses the Ultrascale DSP slice architecture, DSP48E2 [4], shown in Fig. 3.9. This DSP slice can multiply 27 bits by 18 bits and outputs 48 bits of data in its final stage. The choice of the location of the fixed point is optimized to provide the best signal quality and DPD result.

Furthermore, from Fig. 3.9 it is deduced that this DSP slice structure is optimized for multiply-accumulate operations. Thus, when facing addition operations after a multiplication operation, instead of adding products in a tree structure, a cascade-add minimizes DSP slice utilization. Fig. 3.10 illustrates this resource optimization when implementing multiply-add operations.

The hardware implementation of the proposed SISO DPD engine comprises two modules: the SISO engine and a coefficient update module. Both modules are implemented such that their highest system-level wrapper is compliant with the Xilinx data bus interface and the modules' most important parameters can be set through this wrapper, shielding the detail-intensive code from the user. In fact, when migrating to a different hardware platform, the only adjustment that might need to be made to the DPD IP is in the high-

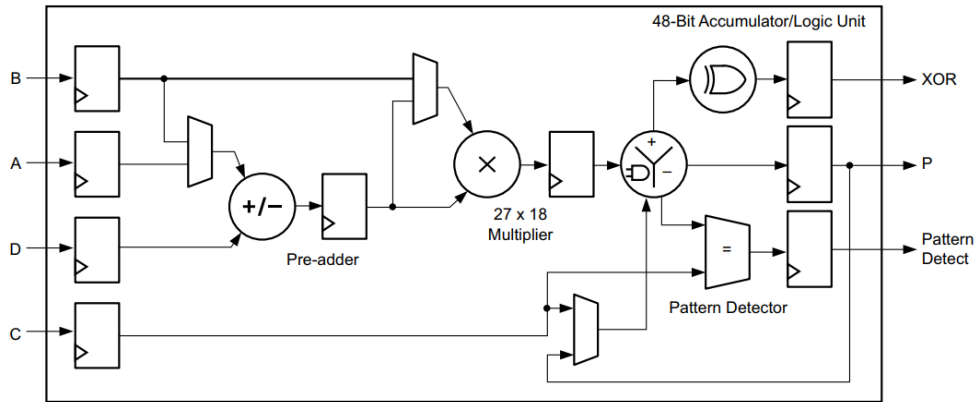


Figure 3.9: DSP48E2, the DSP slices available in Ultrascale RFSoc devices [4].

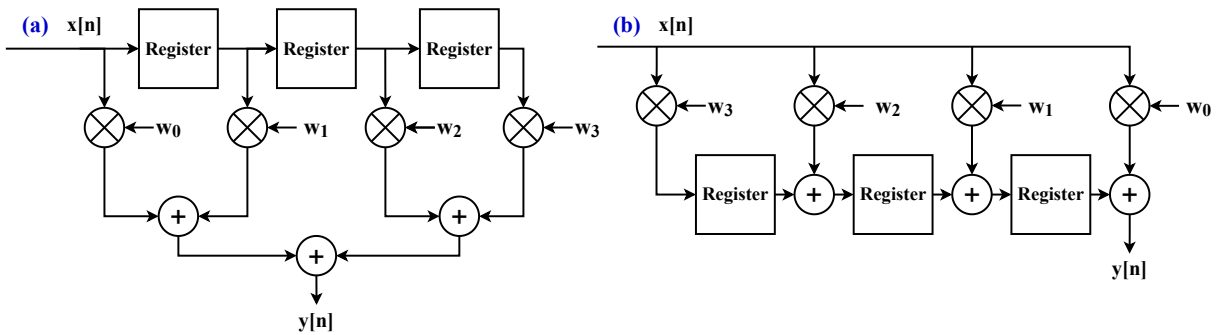


Figure 3.10: Multiply-add in a (a) tree structure (b) cascade structure which optimizes DSP slice resource usage.

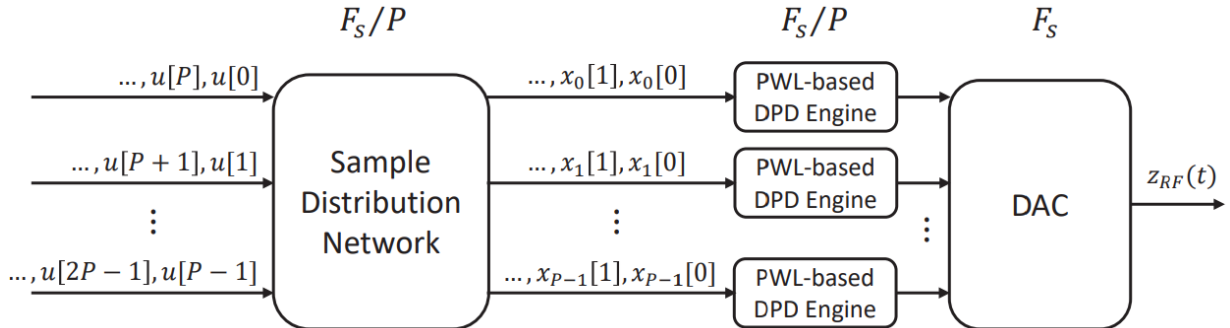


Figure 3.11: Distribution and parallelization scheme used in the main DPD module that will allow sample processing at high rates [5].

level wrapper. The coefficient register is a shift register that is updated when updated coefficients are received by the server. The DPD engine receives updated coefficients from the registers.

Finally, the implemented DPD engine is adjusted to include parallelization. This is necessary to allow the system to accommodate wideband modulated signals in the order of hundreds of megahertz. With parallelization, the input data stream to the DPD engine,  $u[n]$ , is split into  $P$  streams while assuming a sampling rate of  $F_s$ . In this case, the DPD engine consists of  $P$  sub-engines, each running at  $F_s/P$ . The split data stream is redistributed to these sub-engines, based on the required sample in each sub-engine to produce the pre-distorted signal  $z[n]$ . Fig. 3.11 shows the parallelization scheme adapted here [5].

### 3.4.3 Validation Setup and Results

#### Validation Setup

A 75 MHz OFDM signals with 8 dB PAPR is generated using MATLAB and uploaded to the testbed through an ethernet connection. The data is then transferred from the FPGA memory to the 14-bit integrated DAC data converters of the FPGA. The enabled DAC operates at 9.6 GSps with an interpolation factor of 8, allowing for 1.2 GHz of linearization bandwidth. Operating at this high sampling rate permits up-conversion to a center frequency of 3.5 GHz using the integrated mixer on the data converter IP. However, the limited bandwidth of the PA under-test restricts the linearization experiments to mod-

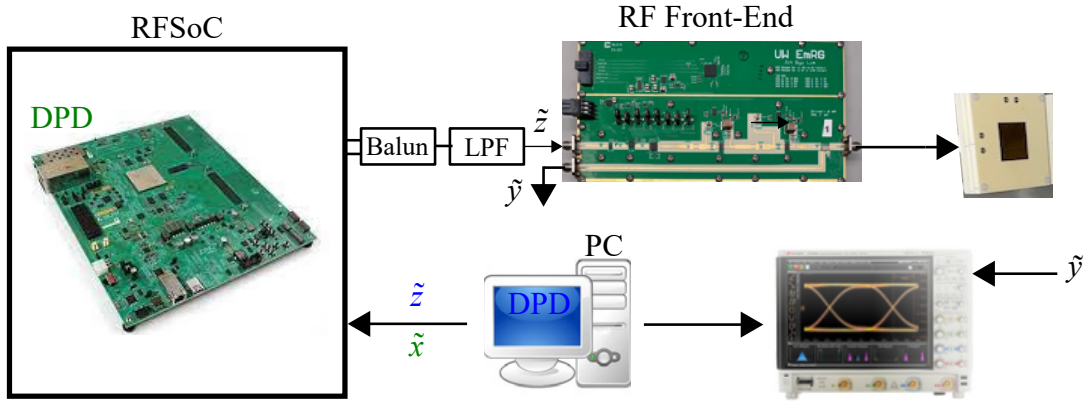


Figure 3.12: Block diagram of the setup used to validate the single-chain operation of the testbed. The testbed receives the pre-distorted signal  $\tilde{z}$  when PC-based DPD is tested (blue) and receives the ideal signal  $\tilde{x}$  when FPGA-based DPD is tested (green).

ulated signals with 75 MHz modulation bandwidth and consequently, to a linearization bandwidth of about 375 MHz.

The path between the FPGA board and the RF front-end chain includes a balun (Anaren BD1631J50100AHF) and a low-pass filter (Mini-Circuits VLF-5500+). This low-pass filter eliminates any image signal in higher Nyquist zones, generated by the DAC. The RF chain consists of a variable gain stage and a two-stage PA, so no external drivers are needed between the FPGA and RF chains. The variable gain stage consists of two pre-drivers (Qorvo QPA9127 & QPA9842) and a digitally controlled attenuator (Analog Devices HMC1122), providing 23-35.5 dB of gain over 2.7-3.8 GHz. The two-stage PA is designed using a 6W GaN transistor (Wolfspeed CGHV1F006S) to allow for the small signal gain of 25-28 dB, peak drain efficiency of  $\approx 50\%$ , and  $P_{\text{sat}} = 38$  dBm over 3.2-3.8 GHz. At the output stage of the two-stage PA, a 30 dB directional coupler (Anaren X4C30F1-30S) is mounted to function as the TOR for training DPD. The output port of the RF front-end connects to a custom-built antenna and the signal from the coupled port is fed into an oscilloscope (Keysight DSOS404A). The integrated ADC on the FPGA board is not used as the TOR due to its limited sampling rate and inability to satisfy Nyquist rate, shown in Table 3.2. The captured signal is then sent to a PC to perform DPD training. In the case of validating the FPGA-based SISO DPD, the trained coefficients are uploaded to the FPGA and sent to the DPD modules to be updated.

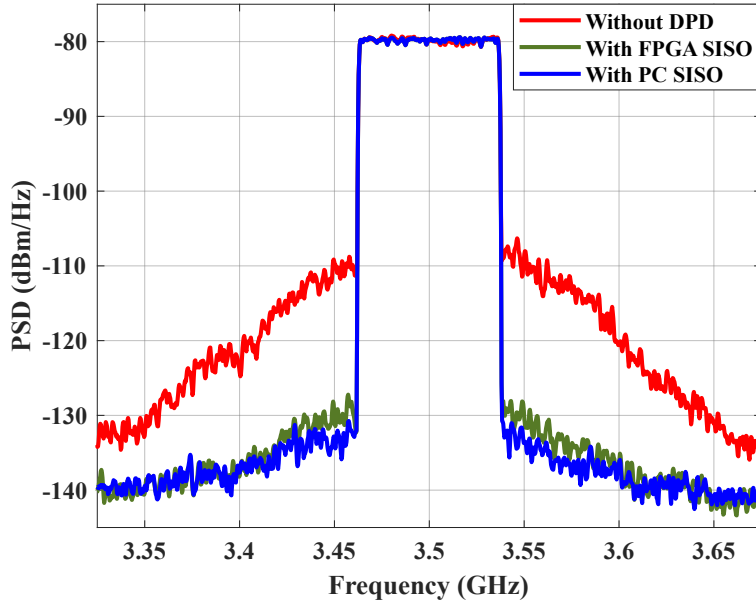


Figure 3.13: Power spectral density of the observed signal ( $\tilde{y}$ ) in single-chain operation.

### Validation Results

The first step to validate the design of the testbed is to use it as a signal generator that receives a signal  $\tilde{x}$  from the user, stores it in the memory and cyclically streams the memory to its DACs (see Fig. 3.12). Subsequently, the signal  $\tilde{x}$  is transmitted to the device under test (DUT) and the output of the DUT,  $\tilde{y}$ , is received by the scope. The signal quality is then measured (labelled "Without DPD"). Next, the PC trains and applies the SISO DPD, proposed in Sec. 3.4, to the signal  $\tilde{x}$  before uploading the pre-distorted signal  $\tilde{z}$  to

	RNMSE (%)	ACPR (dB)	$P_{\text{avg}}$ (dBm)
Without DPD	5.8	-32.8	32.1
With PC SISO DPD	0.4	-53.8	-
With FPGA SISO DPD	0.5	-52.6	-

Table 3.3: Performance comparison of PC-based and FPGA-based SISO DPD engine.

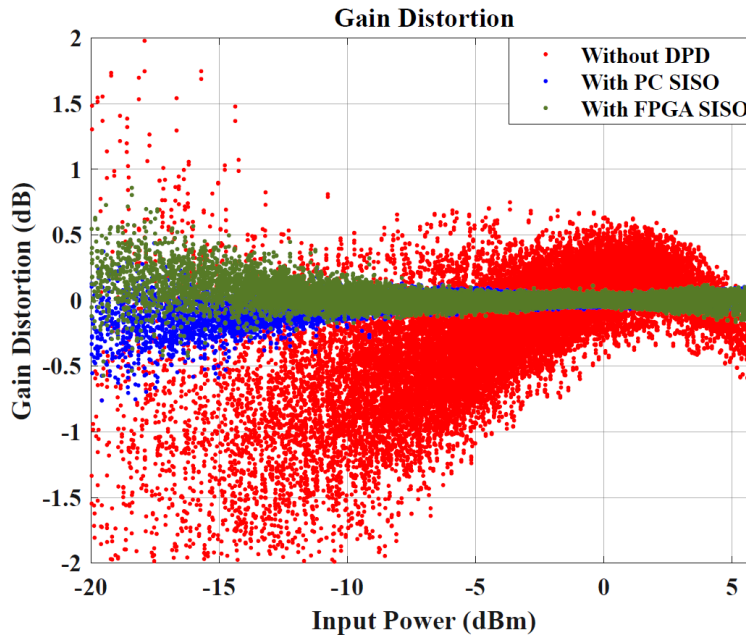


Figure 3.14: Logarithmic AM/AM or gain distortion of the observed signal ( $\tilde{y}$ ) in single-chain operation.

the FPGA. The signal quality after performing DPD is measured by ACPR and RNMSE (labelled "With PC SISO"). After confirming the testbed operation as a signal generator, the proposed SISO DPD is implemented on the FPGA. In this step, the PC initially uploads the signal  $\tilde{x}$ , periodically performs DPD training and uploads DPD coefficients to the FPGA. The FPGA SISO DPD engine applies the coefficients to the uploaded signal (labelled "With FPGA SISO"), resulting in the pre-distorted signal  $\tilde{z}$ . The pre-distorted signal is transmitted and observed in the same way as the PC-based evaluation.

Finally, the performance of the FPGA-based DPD engine is compared against the PC-based engine that was evaluated in the first validation step. This comparison is illustrated by the power spectral density (PSD) (Fig. 3.13), logarithmic AM/AM (Fig. 3.14), the AM/PM merit (Fig. 3.15) and numerically in Table 3.3. The SISO PC-based and FPGA-based DPD perform within less than 1 dB ACPR of each other with almost identical RNMSE performance. Thus the single-chain operation of the testbed is validated. This paves the way to expand the testbed to multiple-chain operation.

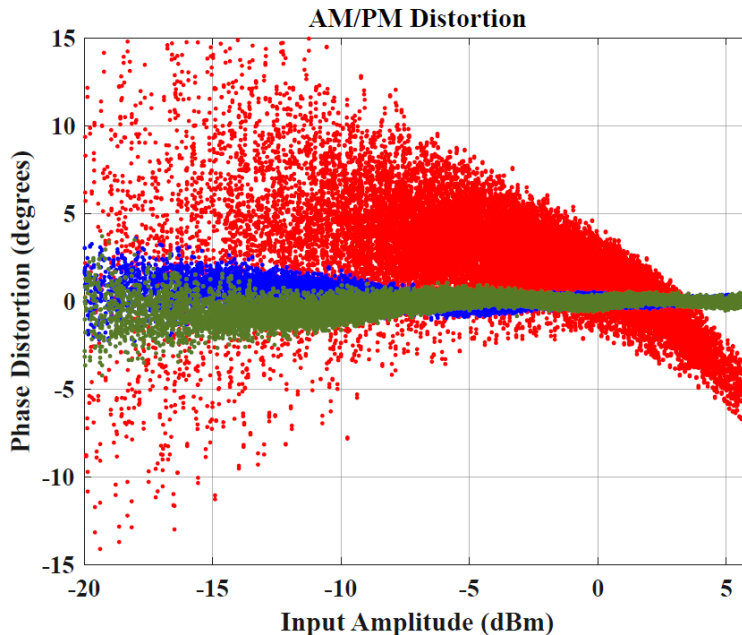


Figure 3.15: AM/PM or phase distortion of the observed signal ( $\tilde{y}$ ) in single-chain operation.

### 3.5 Complete Testbed Summary

In this chapter, the requirements and design of the real-time MIMO testbed were presented. Challenges arising from hardware implementation of a real-time system with wide signal bandwidths were investigated and design considerations were explained. The digital platform of the system was realized with a commercial Xilinx RFSoc evaluation board, which included integrated high-speed data converters as part of the radio hardware.

The built testbed can operate in two modes: signal generation mode and real-time DSP application mode. In signal generation mode, the DSP IP in Fig. 3.2 is bypassed and the processing hardware transmits the signal generated by the server, serving up to 1.2 GHz of bandwidth. In signal generation mode, it is still possible to perform additional DSP algorithms, e.g., DPD, on the generated signal in the server. Alternately, a real-time DSP application mode is developed to test the performance of the hardware implementation of the DSP module in real-time. Considering a DPD module, the testbed can accommodate up to 1.2 GHz of linearization bandwidth per chain. Using a single RFSoc board, the testbed can support 16 fully-digital chains simultaneously. The multiple-chain operation of the testbed is validated in the next chapter.

# Chapter 4

## DPD Implementation for MIMO Transmitters and Validation Results

In the following chapter, the SISO DPD presented in Sec. 3.4 is adapted to a DISO DPD model to linearize MIMO transmitters. The DISO DPD is then implemented in hardware and its performance is evaluated against the PC-based DISO DPD engine. To conclude, the capacity of the developed MIMO testbed is analyzed with regard to linearization bandwidth, DPD complexity and the number of transmitter chains.

### 4.1 DISO DPD Engine for MIMO Transmitters

Spectral regrowth and OOB emissions are an inevitable result of the nonlinearity of power amplifiers in the transmitter chain. These can be compensated by a DPD block before the PA. As the number of transmitter chains increases in MIMO systems, low-complexity DPD engines are sought after to allow implementation on resource-limited processing hardware. The goal of this work is to research a DPD algorithm which reduces resource usage to implement the most number of DPD engines on the developed testbed of Section 3.3.

The most important phenomenon that arises in MIMO transmitters is the antenna crosstalk discussed in Section 2.2.2. Since antennas are near each other, each transmitter chain is affected by its neighbouring chains' radiation. This crosstalk effect gives rise to additional non-linearity in the PA and calls for a DPD algorithm that considers antenna crosstalk, i.e., the DISO DPD introduced in Section 2.2.3. The basis of the DPD proposed in this section is the same as (2.15), however, adjustments were made to reduce the hardware implementation complexity (see Sec. 4.1.1).



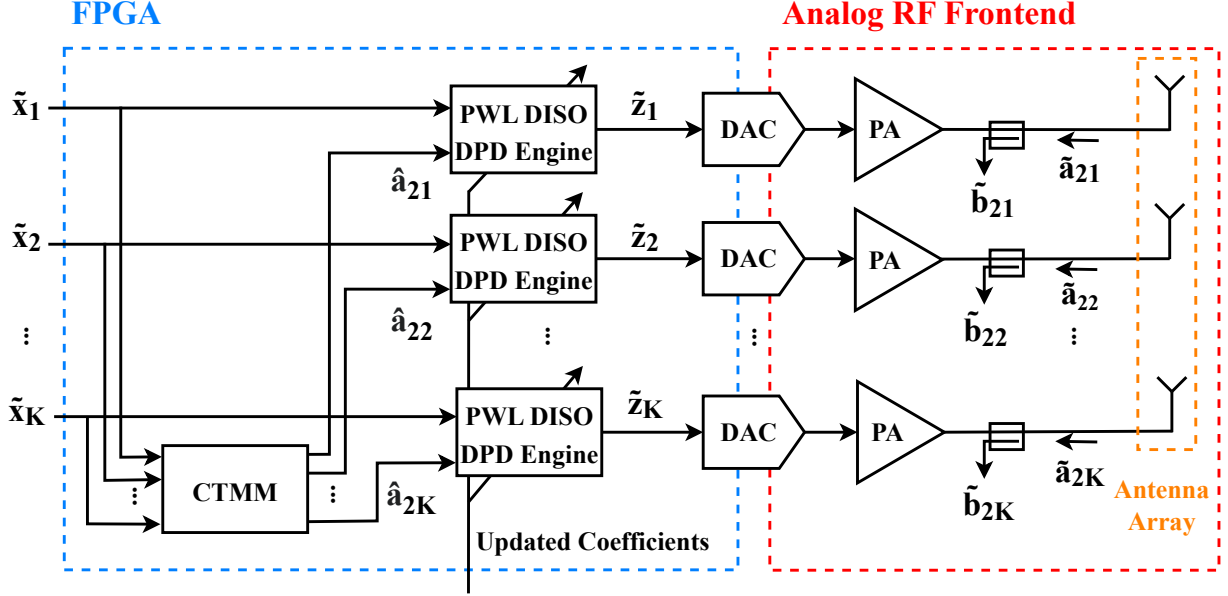


Figure 4.1: Proposed real-time PWL DISO DPD, implemented in FPGA hardware.

The proposed real-time DPD for a MIMO transmitter with  $N$  chains is presented in Fig. 4.1. In this figure, all signals are CBBE. Specifically, the inputs to the PAs before DPD are given by  $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_N$ , the predistorted signals fed to the PAs are  $\tilde{z}_1, \tilde{z}_2, \dots, \tilde{z}_N$ , and the outputs of the PAs are  $\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_N$ . The CTMM module estimates the reflected waves  $\tilde{a}_{21}, \tilde{a}_{22}, \dots, \tilde{a}_{2N}$  from the inputs  $\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_N$ , yielding estimates  $\hat{a}_{21}, \hat{a}_{22}, \dots, \hat{a}_{2N}$ . The  $n$ th DISO DPD engine has as inputs  $\tilde{x}_n$  and the estimate  $\hat{a}_{2n}$ , from which the output  $\tilde{z}_n$  is to be computed.

In the section that follows, the employed DISO PWL DPD model is described in detail, along with the CTMM module implementation. The model is then parallelized into  $P$  streams to relax timing constraints and allow for wide linearization bandwidths.

#### 4.1.1 Piecewise DISO DPD Formulation

The SISO DPD model in Section 3.4 is adjusted to include terms that would mitigate antenna cross-coupling. To address antenna cross-coupling in fully digital MIMO transmitters, a CTMM block is used to estimate the  $K$  reflected waves  $a_{2k}$  from the input signals

$\tilde{\mathbf{x}} = (\tilde{x}_1[n], \dots, \tilde{x}_K[n])$  according to:

$$\hat{a}_{2k}[n] = \sum_{i \neq k} \sum_{q=0}^{Q-1} \lambda_{ki}[q] \tilde{x}_i[n-q], \quad (4.1)$$

where  $Q$  denotes the CTMM memory depth and  $\lambda_{ki}[0], \dots, \lambda_{ki}[Q-1]$  are the coefficients that characterize the coupling from the  $i$ th antenna to the  $k$ th antenna. These coefficients are extracted from the antenna S-parameters within the MIMO system's frequency band of operation [2].

The proposed PWL DISO model is obtained by extending the SISO model in (3.2) to include the estimated reflected waves  $\hat{a}_{2k}$ . The predistorter signal  $\tilde{z}_k[n]$  that is the output of the  $k$ th DPD engine is then described as:

$$\begin{aligned} \tilde{z}_k[n] = & \sum_{m=0}^M \tilde{x}_k[n-m] f(|\tilde{x}_k[n-m]|^2; \boldsymbol{\alpha}, \boldsymbol{\beta}^m) \\ & + \sum_{m=1}^M \tilde{x}_k[n-m] f(|\tilde{x}_k[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\gamma}^m) \\ & + \tilde{x}_k^2[n] \sum_{m=1}^M \tilde{x}_k^*[n-m] f(|\tilde{x}_k[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\zeta}^m) \\ & + \tilde{x}_k^*[n] \sum_{m=1}^M \tilde{x}_k^2[n-m] f(|\tilde{x}_k[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\xi}^m) \\ & + \sum_{m=0}^L \hat{a}_{2k}[n-m] f(|\tilde{x}_k[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\nu}^m) \\ & + \sum_{m=0}^L \hat{a}_{2k}^*[n-m] \tilde{x}_k^2[n-m] f(|\tilde{x}_k[n]|^2; \boldsymbol{\alpha}, \boldsymbol{\tau}^m), \end{aligned} \quad (4.2)$$

where  $L$  is the memory depth of the new terms that include the estimated reflected waves. The PWL DISO DPD model is then parameterized by the coefficient vectors  $\boldsymbol{\beta}^m, \boldsymbol{\gamma}^m, \boldsymbol{\zeta}^m$  and  $\boldsymbol{\xi}^m, \boldsymbol{\nu}^m$  and  $\boldsymbol{\tau}^m$ , for  $m = 0, \dots, M$  and  $\boldsymbol{\nu}^m$  and  $\boldsymbol{\tau}^m$ , for  $m = 0, \dots, L$ . This concludes the DPD model that is to be implemented. Similar to Section 3.4.2, hardware optimization and parallel processing are applied to the hardware implementation.

### 4.1.2 System Setup

The block diagram of the measurement setup used to validate the proposed MIMO FPGA implementation in the developed real-time MIMO testbed is depicted in Fig. 4.2 along with the equipment setup in Fig. 4.3. Four uncorrelated 75 MHz OFDM signals with 8 dB PAPR are generated using MATLAB and uploaded to the setup. Similar to the data flow of the single-chain operation of the testbed, the data is transferred from the FPGA

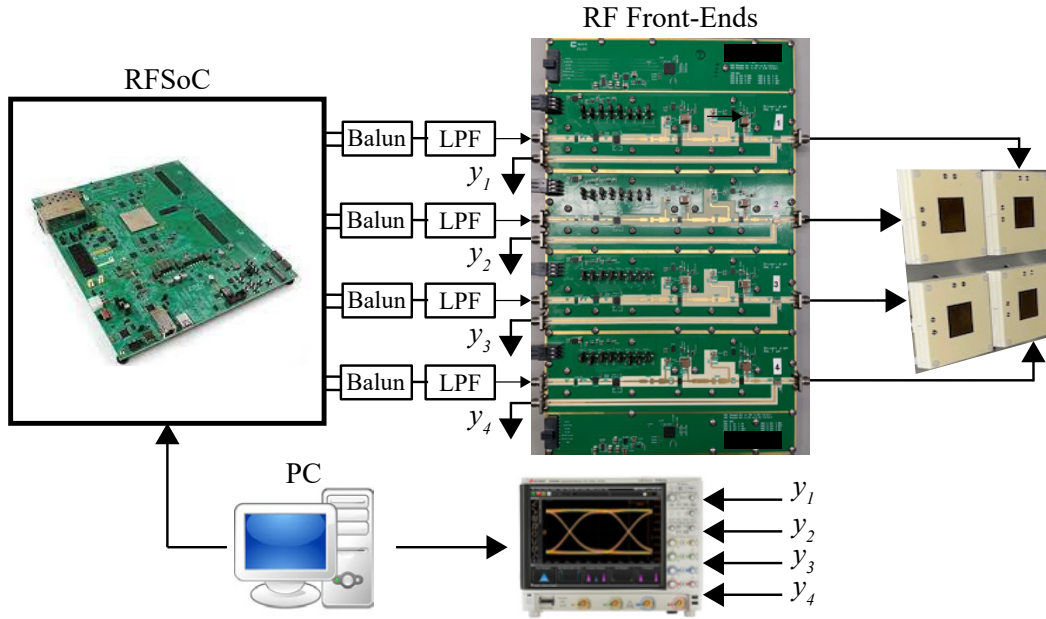


Figure 4.2: Block diagram of the measurement setup.

memory, through the implemented PWL DISO DPD engine to the DAC data converters of the FPGA. Each of the four enabled DACs operates at 9.6 GSps and allows for 1.2 GHz of linearization bandwidth. The center frequency is set to 3.5 GHz.

The RF front-end and DUT consist of the same components described in Sec. 3.4. However, in this setup, the output ports of the RF front-ends connect to a custom-built  $2 \times 2$  antenna sub-array and the signals from the coupled ports are fed into the four-channel oscilloscope. The captured signals are then sent to a PC to perform DPD training. In the case of validating the FPGA-based DISO DPD, the trained coefficients are uploaded to the FPGA and sent to the DPD modules to be updated.

### 4.1.3 Validation Results

To validate the implementation of the DISO PWL DPD on FPGA, the coefficients were trained using a PC and applied using i) a PC, and ii) the FPGA implementation. The performance of the PC and FPGA-based DPD engines were evaluated by RNMSE and ACPR (see Section 2.1.3). For estimating over-the-air (OTA) measurements at boresight, the vector-summed signal of all four chains is used to calculate the performance metrics.

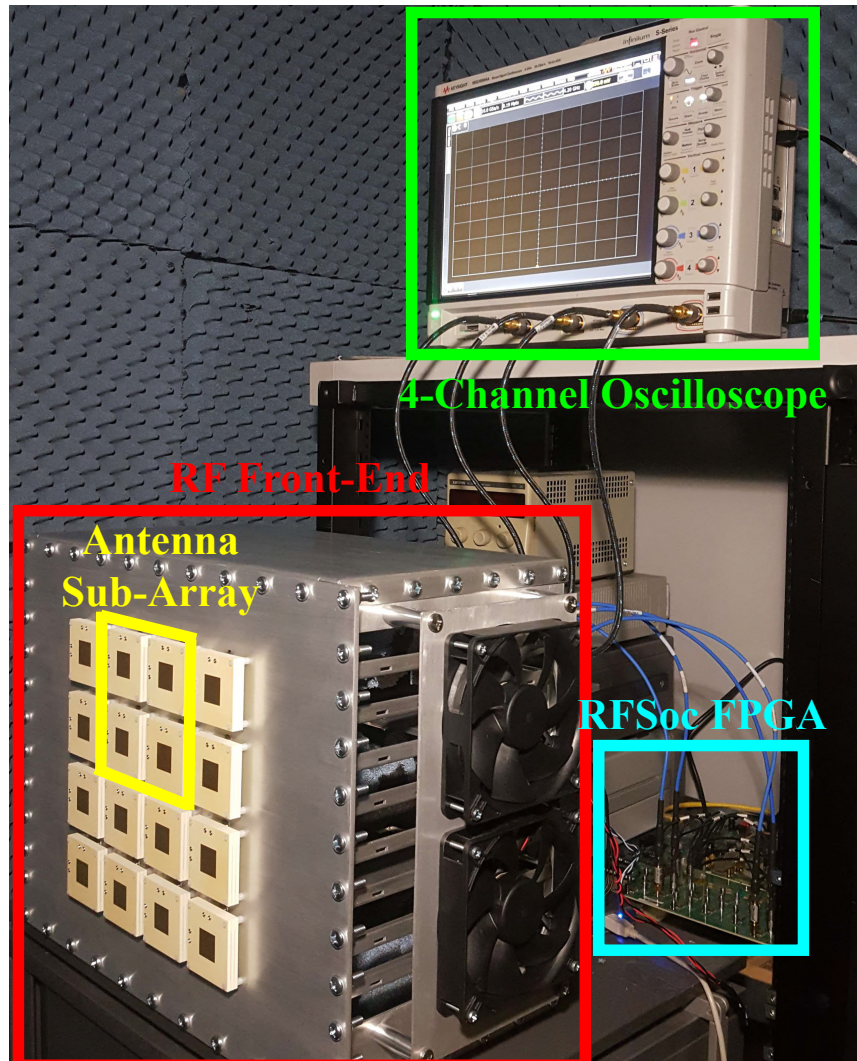


Figure 4.3: Experimental fully digital massive MIMO transmitter setup including the 4-element sub-array that was used in the experiments.

Table 4.1 shows RNMSE and ACPR results for without and with DPD and Fig. 4.4 shows the estimated OTA signal spectrum (a) without DPD, (b) with DPD engine on PC, and (c) with DPD engine on FPGA. Due to the complications of OTA measurements at sub-6 GHz frequencies which require controlled OTA channels and reverberation chambers, the spectrum estimation is obtained by summing the signals received at every chain's TOR. It is noted that the PC-based DPD outperformed the FPGA-based DISO by 1 dB ACPR

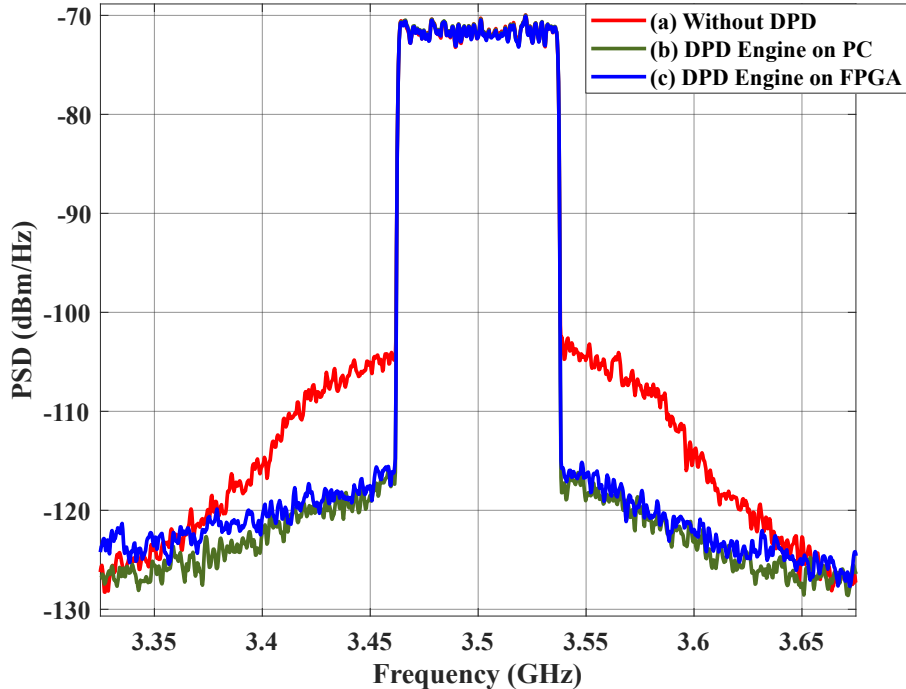


Figure 4.4: Power spectrum comparison of (a) without DPD, (b) PC-based DPD engine, and (c) FPGA-based DPD engine, at an estimated over-the-air boresight receiver.

due to the higher precision of double-precision floating-point-based arithmetic on the PC compared to fixed-point on the FPGA.

Table 4.2 reports the resource utilization of the FPGA-based DPD for single-chain,  $K = 4$  and  $K = 8$  chains. Although the proposed DPD model in this work is optimized to use the least number of multipliers to minimize DSP slices usage, around 89% of the DSP resources on the RFSoc are used when the number of chains is increased to  $K = 8$ . As this DPD engine implementation is not LUT-based (e.g., [14, 15]), BRAM use is avoided as it is unnecessary and power consuming. For this DPD implementation, the power consumption per DPD chain amounts to 3.15 W while the power consumption of a single RF chain used in the measurements is 7.5 W for the output power reported in Table 4.1. On the other hand, the PWL SISO DPD engine consumes 1.6 W of power. When the chain count doubles from  $K = 4$  to  $K = 8$ , the processing resources are found to approximately double as well. Meeting timing requirements when resource utilization is above 70% is challenging.

	RNMSE (%)	ACPR (dB)	$P_{\text{avg}}$ (dBm)
Without DPD	7.71	-33.4	34.3
With PC DISO DPD	1.42	-48.3	-
With FPGA DISO DPD	1.59	-47.1	-

Table 4.1: Performance comparison of PC-based and FPGA-based DISO DPD engine on the four-chain RF front-end.

It goes without mentioning that one state-of-the-art RFSoc FPGA does not have sufficient resources to implement a higher chain count assuming the proposed complexity-reduced DPD model is used with the current DAC configurations and parallelization factor.

## 4.2 Testbed Comparison and Capacity Analysis

### 4.2.1 Testbed Comparison

The real-time reconfigurable testbed developed in this thesis is compared to the testbeds reviewed in Sec. 2.2. Table 4.3 shows a comparison between the characteristics of the MIMO testbed developed in this work and that in literature. One prominent property of this testbed is its very wide bandwidth capability compared to most other testbeds. Although the testbed in this work only comprises 16 chains, it is scalable in design and the possibility exists to expand to a higher number of coherent chains. This testbed is distinguished in providing an experimental platform for testing DPD methods for MIMO transmitters.

### 4.2.2 Capacity Analysis

While the real-time MIMO transmitter testbed developed in this thesis offers attractive properties that will advance research in MIMO transmitters, it is crucial to explore the limitations of the testbed and the capacity it provides in DPD implementation. The limitation explored in this section is the maximum number of implementable chains and the maximum possible linearization bandwidth and they are induced by the following elements:

Number of Chains	1	4	8
DSP Slice Count (per stream)	63	119	119
DSP Slice Count (total)	250	1904	3808
DSP Slice (%)	5.8	44.5	89.1
Slice LUTs	12269	107464	216060
Slice LUTs (%)	3.5	25.2	50.8
Slice Regs	14145	110337	220673
Slice Regs (%)	1.6	12.9	25.9
BRAMs	0	0	0
Power (W)	1.6	12.6	23.6

Table 4.2: FPGA Resource Utilization of Proposed DPD

- The maximum sampling rate of the integrated DACs on the ZCU216 RFSoc evaluation board (9.85 GSps).
- The maximum clock rate the FPGA can operate at with the implemented hardware design. Equivalently, this is the FPGA clock rate at which the hardware design can be successfully implemented, placed and routed. Empirically, this clock rate is found to be in the interval [250, 320] MHz.
- The total number of DSP slices available on the ZCU216 RFSoc (4272 total DSP slices).
- The discrete interpolation factors available in the configuration of the radio hardware of the testbed (see Sec. 3.3.4, see [3] for values).

The following analysis is done by assuming the operating center frequency of the testbed to be in the interval  $f_c \in [2.4, 4.8]$  GHz. This limits the sampling frequency choices to the interval  $F_s \in [8.4, 9.85]$  GSps (see Sec. 3.3.4).

The first part of the analysis is to investigate the trade-off between the maximum number of implementable chains and the maximum linearization bandwidth. This trade-off

MIMO Testbed Comparison					
	Num. Chains	Frequency	Signal Bandwidth	Radio Hardware	Purpose
<b>LuMaMi</b> [20]	100	1.2-6 GHz	20 MHz	National Instruments SDR	Channel sounding, algorithm evaluation
<b>KU Leuven ESAT</b> [21]	34	$\leq 4.4$ GHz	40 MHz	National Instruments SDR	Cloud-RAN, relaying
<b>Argos</b> [22]	160	$\leq 3.8$ GHz	56 MHz	In-house designed SDR	Channel sounding
<b>Southeast University</b> [23]	64	3.4-3.6 GHz	200 MHz	Analog Devices DAC/ADC	Not declared
<b>This work</b>	<b>16</b>	$\leq 4.8$ GHz	<b>120 MHz</b>	<b>Xilinx RF-SoC</b>	<b>Base station emulation, channel sounding</b>

Table 4.3: MIMO testbed comparison of those in literature and the developed testbed in this thesis.

assumes the MIMO DPD implementation DSP slice usage to be that in Table 4.2. To increase the linearization bandwidth, the number of parallel streams ( $P$ ) used in implementation is increased. On the other hand, the maximum number of implemented chains is limited by the total number of available DSP slices. Thus, the more chains implemented on the FPGA, the smaller the linearization bandwidth of each chain will be. Thus, when implementing more chains, larger interpolation factors are used to interpolate smaller bandwidth to high  $F_s$  rates. The following equations quantify these trade-offs:

$$\text{Max. Chains} \times P = \frac{\text{DSP slices (total)}}{\text{DSP Slices (per stream)}}, \quad (4.3)$$

and

$$\text{Lin. Bandwidth} = \text{Clock Rate} \times P = \frac{F_s}{\text{Interpolation Factor}}, \quad (4.4)$$



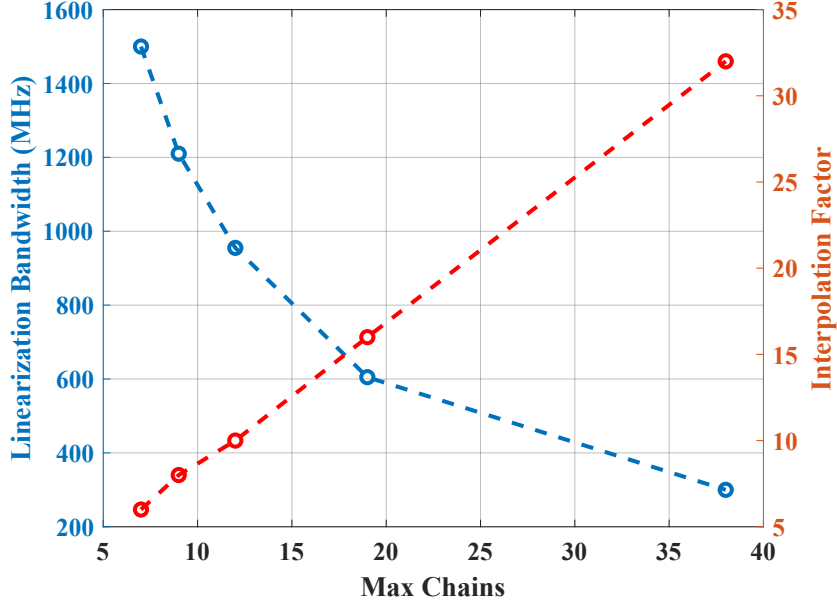


Figure 4.5: Relation between the maximum number of implementable chains and maximum possible linearization bandwidth and its consequent interpolation factor for a given DPD.

where  $P$  denotes the number of parallel streams per chain and  $F_s$  denotes the sampling frequency of each DAC. Both sides of (4.3) represent the total number of streams in the parallelized hardware implementation.

By applying the testbed constraints to (4.3) and (4.4), Fig. 4.5 and Fig. 4.6 are obtained. These two plots visualize the trade-off between the maximum linearization bandwidth and the maximum number of chains that are implementable on the developed testbed. Fig.4.5 shows the interpolation factor that is used to achieve each bandwidth and Fig.4.6 shows the parallelization factor ( $P$ ) used to achieve each bandwidth, given that more parallel streams are used to achieve wider linearization bandwidths.

Finally, the second part of the analysis explores the potentials of lower complexity DPD implementations on the developed MIMO testbed in Fig. 4.7. This plot illustrates how the full capacity of the testbed, 16-chain transmission, can be accomplished for a wide bandwidth given the DPD complexity is reduced. As an example, the complexity of the proposed SISO DPD given in Table 4.2 amounts to approximately 63 DSP slices per parallel stream. Based on Fig. 4.7, this SISO DPD implementation can be used to achieve a linearization bandwidth of 1.2 GHz for 16 chains of transmission. On the other hand,

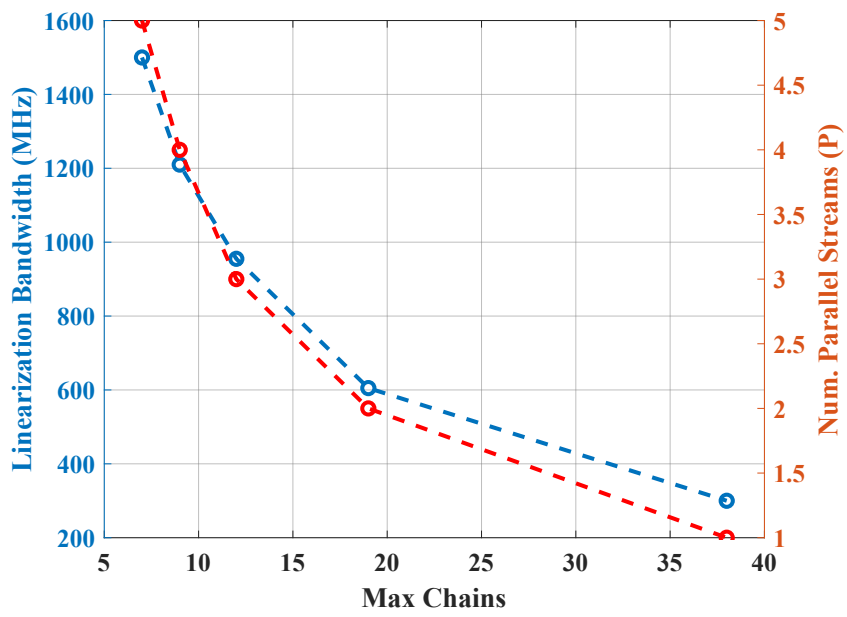


Figure 4.6: Relation between the maximum number of implementable chains and maximum possible linearization bandwidth and its minimum parallel number of streams ( $P$ ) need to implement a given DPD.

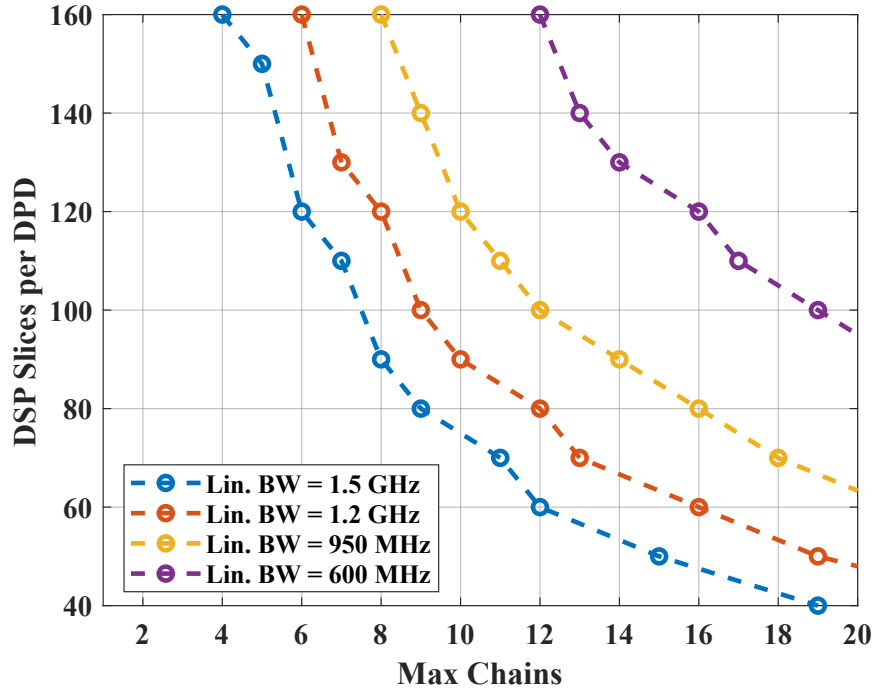


Figure 4.7: Trade-off between DPD hardware complexity, linearization bandwidth and the maximum number of implementable chains.

the DISO DPD implementation uses approximately 120 DSP slices per parallel stream per chain (see Table 4.2). Based on Fig. 4.7, to achieve a linearization bandwidth of 1.2 GHz using the proposed DISO DPD, only 8 chains can be activated. By decreasing the linearization bandwidth to 600 MHz, 16 chains can be linearized using the proposed DISO DPD.

# Chapter 5

## Conclusion and Future Work

### 5.1 Conclusion

In this thesis, a real-time reconfigurable FPGA-based MIMO testbed was developed. This testbed allows for testing MIMO systems up to 16 chains for modulated signals as wide as 240 MHz. The FPGA hardware facilitates quick evaluation of the hardware implementation of DSP techniques including DPD. To demonstrate the testbed reconfigurability, a reduced-complexity SISO DPD was proposed, evaluated and finally augmented for a MIMO system that experiences antenna crosstalk. The proposed DISO DPD was implemented on the FPGA for 4 and 8 chains and verified against its corresponding PC-based DPD. Before concluding this thesis, an overview of each chapter is presented.

In Chapter 2, background theory was presented on transmitter chains for single-chain and MIMO systems. In this chapter, digital predistortion was introduced as a method to mitigate PA non-linearity. DISO DPD was considered as one of the solutions proposed in the literature to overcome additional non-idealities that arise with MIMO setups due to antenna crosstalk. On the other hand, the necessity of developing a MIMO testbed to evaluate MIMO systems and DSP solutions in a timely manner was discussed. Developed platforms reported in the literature and their capabilities were reviewed. Finally, techniques to reduce the hardware implementation complexity of DPD reported in the literature were investigated to apply the best techniques to the proposed DPD engine in this thesis.

In Chapter 3, the design process of the MIMO testbed was described. First, testbed requirements were outlined and hardware was chosen to satisfy the requirements. Next, key design characteristics were presented which covered server implementation to allow the

user to communicate with the testbed, high-speed clock generation using PLL boards and data converter flexibility and options. The final testbed product has up to 16 transmitter chains with DACs operating at 9.6 GSps and parallel data streams, running at 300 MHz per stream. This allows for 1.2 GHz of signal bandwidth in signal generation mode. In the second part of this chapter, a low-complexity SISO DPD solution was proposed. The DPD was implemented in hardware and parallelized to allow integration with the testbed. The hardware implementation was also optimized to minimize DSP slice usage on the Xilinx RFSoc FPGA. Measurement results were presented for a single-chain transmitter that demonstrates testbed capabilities before extending its operation to multiple chains.

In Chapter 4, the DISO DPD is formulated and verified. The verification setup consisted of a PC that is used as the training device for the DPD coefficients and also controls the testbed through an established TCP connection. The PC connects to the digital FPGA-based hardware unit housing hardware-implemented DPD engines and comprising high-speed data converters. Finally, the digital processing unit is connected to the in-lab designed RF transmitter front-end which includes pre-driver components, PAs and the 16-chain antenna array.

The setup was tested with 4 uncorrelated 16-QAM OFDM signals with 8 dB of PAPR inside an anechoic chamber. The testbed was evaluated for 4 chains of transmission using a 2x2 antenna sub-array. The proposed DPD algorithm was evaluated using both the FPGA-implemented engine and a PC-based engine. With a starting ACPR of -33.4 dB and RNMSE of 7.71%, the FPGA-based engine was able to reach -47.1 dB of ACPR and achieved an RNMSE of 1.59% after linearization. The PC-based engine performed similarly, reaching ACPR of -48.3 dB and RNMSE of 1.42%. Resource utilization was reported for DPD implementation of single-chain, 4 and 8 chains. An 8-chain reduced-complexity DPD implementation uses up to 89% of the DSP slices on the RFSoc FPGA, motivating the investigation of alternate DPD solutions for MIMO systems.

## 5.2 Future Work

The goal of this work was to develop the digital processing unit of a MIMO testbed that would allow quick evaluation of MIMO DSP solutions. Although this goal was achieved during the time of this study, there are many improvements to be made to the testbed and more than that, many MIMO explorations are to be done. This includes the following avenues:

## Circuit-aware Crosstalk Compensation

As mentioned in Section 2.2.3, DSP crosstalk compensation methods that exist in the literature do not incorporate circuit-level analysis when developing methods to circumvent antenna crosstalk. By studying the physical model of the PA, a better understanding can be achieved as to how crosstalk increases signal distortion. This could lead to more effective and optimized approaches to eliminating crosstalk through DSP methods.

## Neural Network-Based DPD

The availability of a MIMO testbed opens the door for exploring different DPD implementations. Previously, research has been done in both areas of neural network-based DPD ([48, 49, 50, 51]) and low-complexity hardware implementation of neural networks for DPD applications ([52, 53, 54]). These neural networks usually consist of two layers, each comprising 10-50 neurons. Some of the hardware implementation challenges include reducing the processing complexity of back-propagation parameter updates and meeting latency requirements. This testbed provides an agile framework to experiment with low-complexity hardware implementation of neural network-based DPD.

## Channel Reciprocity Calibration

Channel reciprocity simply states that the channel characteristics are the same when signals are transmitted from the transmitter to the receiver and when they are transmitted from the receiver to the transmitter. However, in MIMO transmitters, the transmitted signal is affected by antenna cross-coupling. Thus, to achieve channel reciprocity, the coupling between chains should be calibrated. In addition to evaluating the calibration using the developed testbed, the linearizability of MIMO transmitters can be re-assessed after calibration.

# References

- [1] N. Instruments, “White paper: Testing PAs under digital predistortion and dynamic power supply conditions,” [https://download.ni.com/evaluation/coretest/RFIC%20White%20Paper%20Series\\_Part%202.pdf](https://download.ni.com/evaluation/coretest/RFIC%20White%20Paper%20Series_Part%202.pdf).
- [2] K. Hausmair, P. N. Landin, U. Gustavsson, C. Fager, and T. Eriksson, “Digital predistortion for multi-antenna transmitters affected by antenna crosstalk,” *IEEE Trans. Microw. Theory Tech.*, vol. 66, no. 3, pp. 1524–1535, 2018.
- [3] *Zynq UltraScale+ RFSoc RF Data Converter Product Guide*, Xilinx, 8 2021, v2.5.
- [4] *UltraScale Architecture DSP Slice User Guide*, Xilinx, 8 2021, rev. 1.11.
- [5] M. Almoner, H. Barkhordar-pour, P. Mitran, and S. Boumaiza, “Hardware-efficient implementation of piece-wise digital predistorters for wideband 5G transmitters,” in *IEEE MTTTS Int. Microw. Symp. Dig.*, 2022, pp. 514–517.
- [6] Statista, “Number of internet of things (IoT) connected devices worldwide from 2019 to 2021, with forecasts from 2022 to 2030,” <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>.
- [7] Avnet, “Understanding massive MIMO technology,” <https://www.avnet.com/wps/portal/abacus/solutions/markets/communications/5g-solutions/understanding-massive-mimo-technology/>.
- [8] P. L. Gilabert, A. Cesari, G. Montoro, E. Bertran, and J.-M. Dilhac, “Multi-lookup table FPGA implementation of an adaptive digital predistorter for linearizing RF power amplifiers with memory effects,” *IEEE Trans. Microw. Theory Tech.*, vol. 56, no. 2, pp. 372–384, 2008.

- [9] A. Molina, K. Rajamani, and K. Azadet, "Digital predistortion using lookup tables with linear interpolation and extrapolation: Direct least squares coefficient adaptation," *IEEE Trans. Microw. Theory Tech.*, vol. 65, no. 3, pp. 980–987, 2017.
- [10] P. L. Gilabert, G. Montoro, and E. Bertran, "FPGA implementation of a real-time NARMA-based digital adaptive predistorter," *IEEE Trans. Circuits Syst. II*, vol. 58, no. 7, pp. 402–406, 2011.
- [11] Y. Ma, Y. Yamao, Y. Akaiwa, and C. Yu, "FPGA implementation of adaptive digital predistorter with fast convergence rate and low complexity for multi-channel transmitters," *IEEE Trans. Microw. Theory Tech.*, vol. 61, no. 11, pp. 3961–3973, 2013.
- [12] C. D. Presti, D. F. Kimball, and P. M. Asbeck, "Closed-loop digital predistortion system with fast real-time adaptation applied to a handset WCDMA PA module," *IEEE Trans. Microw. Theory Tech.*, vol. 60, no. 3, pp. 604–618, 2012.
- [13] L. Guan and A. Zhu, "Low-cost FPGA implementation of Volterra series-based digital predistorter for RF power amplifiers," *IEEE Trans. Microw. Theory Tech.*, vol. 58, no. 4, pp. 866–872, 2010.
- [14] A. Kwan, F. Ghannouchi, O. Hammi, M. Helaoui, and M. Smith, "Look-up table-based digital predistorter implementation for field programmable gate arrays using long-term evolution signals with 60 MHz bandwidth," *IET Sci. Meas. Technol.*, vol. 6, no. 3, p. 181, 2012.
- [15] C.-F. Cheang, P.-I. Mak, and R. P. Martins, "A hardware-efficient feedback polynomial topology for DPD linearization of power amplifiers: Theory and FPGA validation," *IEEE Trans. Circuits Syst. I*, vol. 65, no. 9, pp. 2889–2902, 2018.
- [16] L. Chen, W. Chen, Y.-J. Liu, Y. He, X. Liu, T. Cao, F. M. Ghannouchi, and Z. Feng, "Linearization of a directional modulation transmitter using low-complexity cascaded digital predistortion," *IEEE Trans. Microw. Theory Tech.*, vol. 67, no. 11, pp. 4467–4478, 2019.
- [17] S. Afsardoost, T. Eriksson, and C. Fager, "Digital predistortion using a vector-switched model," *IEEE Trans. Microw. Theory Tech.*, vol. 60, no. 4, pp. 1166–1174, 2012.
- [18] Y. Li, W. Cao, and A. Zhu, "Instantaneous sample indexed magnitude-selective affine function-based behavioral model for digital predistortion of RF power amplifiers," *IEEE Trans. Microw. Theory Tech.*, vol. 66, no. 11, pp. 5000–5010, 2018.



- [19] H. Huang, J. Xia, and S. Boumaiza, “Novel parallel-processing-based hardware implementation of baseband digital predistorters for linearizing wideband 5G transmitters,” *IEEE Trans. Microw. Theory Tech.*, vol. 68, no. 9, pp. 4066–4076, 2020.
- [20] S. Malkowsky, J. Vieira, L. Liu, P. Harris, K. Nieman, N. Kundargi, I. C. Wong, F. Tufvesson, V. Öwall, and O. Edfors, “The world’s first real-time testbed for massive MIMO: Design, implementation, and validation,” *IEEE Access*, vol. 5, pp. 9073–9088, 2017.
- [21] E. Bjornson, L. Van der Perre, S. Buzzi, and E. G. Larsson, “Massive MIMO in sub-6 GHz and mmwave: Physical, practical, and use-case differences,” *IEEE Wirel.*, vol. 26, no. 2, pp. 100–108, 2019.
- [22] C. Shepard, H. Yu, N. Anand, E. Li, T. Marzetta, R. Yang, and L. Zhong, “Argos: Practical many-antenna base stations,” *Proc. annu. Int. Conf. Mob.*, p. 53–64, 2012. [Online]. Available: <https://doi.org/10.1145/2348543.2348553>
- [23] C. Guo, W. Hong, L. Tian, Z. H. Jiang, J. Zhou, J. Chen, H. Wang, Z. Yu, C. Yu, J. Zhai, N. Zhang, D. Hou, G. Yang, Z. Chen, F. Wu, Y. Yu, C. Shu, F. Huang, X.-W. Zhu, and Z.-C. Hao, “Design and implementation of a full-digital beamforming array with nonreciprocal tx/rx beam patterns,” *IEEE Antennas Wirel. Propag. Lett.*, vol. 19, no. 11, pp. 1978–1982, 2020.
- [24] ETSI, “5G NR base station (BS) radio transmission and reception,” [https://www.etsi.org/deliver/etsi\\_ts/138100\\_138199/13810101/15.02.00\\_60/ts\\_13810101v150200p.pdf](https://www.etsi.org/deliver/etsi_ts/138100_138199/13810101/15.02.00_60/ts_13810101v150200p.pdf).
- [25] M. Schetzen, “Nonlinear system modeling based on the Wiener theory,” *Proceedings of the IEEE*, vol. 69, no. 12, pp. 1557–1573, 1981.
- [26] D. Morgan, Z. Ma, J. Kim, M. Zierdt, and J. Pastalan, “A generalized memory polynomial model for digital predistortion of RF power amplifiers,” *IEEE Trans. Sig. Process.*, vol. 54, no. 10, pp. 3852–3860, 2006.
- [27] A. Zhu, J. C. Pedro, and T. J. Brazil, “Dynamic deviation reduction-based Volterra behavioral modeling of RF power amplifiers,” *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 12, pp. 4323–4332, 2006.
- [28] A. Zhu, “Decomposed vector rotation-based behavioral modeling for digital predistortion of RF power amplifiers,” *IEEE Trans. Microw. Theory Tech.*, vol. 63, no. 2, pp. 737–744, 2015.

- [29] F. Mkadem, M. C. Fares, S. Boumaiza, and J. Wood, “Complexity-reduced Volterra series model for power amplifier digital predistortion,” *Analog Integr. Circ. Sig. Process.*, vol. 79, no. 2, p. 331–343, 2014. [Online]. Available: <https://link.springer.com/article/10.1007/s10470-014-0266-4#citeas>
- [30] L. Ding, F. Mujica, and Z. Yang, “Digital predistortion using direct learning with reduced bandwidth feedback,” *IEEE MTTTS Int. Microw. Symp. Dig.*, pp. 1–3, 2013.
- [31] R. N. Braithwaite, “Memory correction of a Doherty power amplifier with a WCDMA input using digital predistortion,” *IEEE MTTTS Int. Microw. Symp. Dig.*, pp. 1526–1529, 2006.
- [32] C. Eun and E. Powers, “A new Volterra predistorter based on the indirect learning architecture,” *IEEE Trans. Sig. Process.*, vol. 45, no. 1, pp. 223–227, 1997.
- [33] T. L. Marzetta, E. G. Larsson, H. Yang, and H. Q. Ngo, *Fundamentals of Massive MIMO*. Cambridge, United Kingdom: Cambridge University Press, 2016.
- [34] X. Chen, S. Zhang, and Q. Li, “A review of mutual coupling in MIMO systems,” *IEEE Access*, vol. 6, pp. 24 706–24 719, 2018.
- [35] C. Fager, X. Bland, K. Hausmair, J. Chani Cahuana, and T. Eriksson, “Prediction of smart antenna transmitter characteristics using a new behavioral modeling approach,” in *IEEE MTTTS Int. Microw. Symp. Dig.*, 2014, pp. 1–4.
- [36] F. Yang and Y. Rahmat-Samii, “Microstrip antennas integrated with electromagnetic band-gap (EBG) structures: a low mutual coupling design for array applications,” *IEEE Trans. Antennas Propag.*, vol. 51, no. 10, pp. 2936–2946, 2003.
- [37] S. Chen, Y. Wang, and S. Chung, “A decoupling technique for increasing the port isolation between two strongly coupled antennas,” *IEEE Trans. Antennas Propag.*, vol. 56, no. 12, pp. 3650–3658, 2008.
- [38] A. C. K. Mak, C. R. Rowell, and R. D. Murch, “Isolation enhancement between two closely packed antennas,” *IEEE Trans. Antennas Propag.*, vol. 56, no. 11, pp. 3411–3419, 2008.
- [39] F. M. Barradas, P. M. Tomé, J. M. Gomes, T. R. Cunha, P. M. Cabral, and J. C. Pedro, “Power, linearity, and efficiency prediction for MIMO arrays with antenna coupling,” *IEEE Trans. Microw. Theory Tech.*, vol. 65, no. 12, pp. 5284–5297, 2017.

- [40] S. Cripps, *RF Power Amplifiers for Wireless Communications, Second Edition*. Artech House, 2006.
- [41] K. Hausmair, S. Gustafsson, C. Sánchez-Pérez, P. N. Landin, U. Gustavsson, T. Eriksson, and C. Fager, “Prediction of nonlinear distortion in wideband active antenna arrays,” *IEEE Trans. Microw. Theory Tech.*, vol. 65, no. 11, pp. 4550–4563, 2017.
- [42] S. A. Bassam, M. Helaoui, and F. M. Ghannouchi, “Crossover digital predistorter for the compensation of crosstalk and nonlinearity in MIMO transmitters,” *IEEE Trans. Microw. Theory Tech.*, vol. 57, no. 5, pp. 1119–1128, 2009.
- [43] M. V. Amiri, S. A. Bassam, M. Helaoui, and F. M. Ghannouchi, “Matrix-based orthogonal polynomials for MIMO transmitter linearization,” in *IEEE CAMAD*, 2010, pp. 57–60.
- [44] P. M. Suryasarman and A. Springer, “A comparative analysis of adaptive digital predistortion algorithms for multiple antenna transmitters,” *IEEE Trans. Circuits Syst. I*, vol. 62, no. 5, pp. 1412–1420, 2015.
- [45] B. Fehri and S. Boumaiza, “Automated symbolic optimization and high level synthesis of single- and multi- band digital pre-distortion hardware in an FPGA,” in *IEEE MTTTS Int. Microw. Symp. Dig.*, 2015, pp. 1–3.
- [46] X. Liu, Q. Zhang, W. Chen, H. Feng, L. Chen, F. M. Ghannouchi, and Z. Feng, “Beam-oriented digital predistortion for 5g massive MIMO hybrid beamforming transmitters,” *IEEE Trans. Microw. Theory Tech.*, vol. 66, no. 7, pp. 3419–3432, 2018.
- [47] E. Ng, Y. Beltagy, G. Scarlato, A. Ben Ayed, P. Mitran, and S. Boumaiza, “Digital predistortion of millimeter-wave rf beamforming arrays using low number of steering angle-dependent coefficient sets,” *IEEE Trans. Microw. Theory Tech.*, vol. 67, no. 11, pp. 4479–4492, 2019.
- [48] A. Brihuega, L. Anttila, and M. Valkama, “Neural-network-based digital predistortion for active antenna arrays under load modulation,” *IEEE Microw. Wirel. Compon. Lett.*, vol. 30, no. 8, pp. 843–846, 2020.
- [49] F. Mkadem and S. Boumaiza, “Physically inspired neural network model for RF power amplifier behavioral modeling and digital predistortion,” *IEEE Trans. Microw. Theory Tech.*, vol. 59, no. 4, pp. 913–923, 2011.

- [50] Y. Zhang, Y. Li, F. Liu, and A. Zhu, "Vector decomposition based time-delay neural network behavioral model for digital predistortion of RF power amplifiers," *IEEE Access*, vol. 7, pp. 91 559–91 568, 2019.
- [51] T. Liu, S. Boumaiza, and F. Ghannouchi, "Dynamic behavioral modeling of 3G power amplifiers using real-valued time-delay neural networks," *IEEE Trans. Microw. Theory Tech.*, vol. 52, no. 3, pp. 1025–1033, 2004.
- [52] S. Yesil, C. Sen, and A. O. Yilmaz, "Experimental analysis and FPGA implementation of the real-valued time delay neural network based digital predistortion," in *Proc. 26th IEEE Int. Conf. Electron. Circuits Syst. (ICECS)*, 2019, pp. 614–617.
- [53] M. Tanio, N. Ishii, and N. Kamiya, "A sparse neural network-based power adaptive DPD design and its hardware implementation," *IEEE Access*, vol. 10, pp. 114 673–114 682, 2022.
- [54] C. Tarver, A. Balatsoukas-Stimming, and J. R. Cavallaro, "Design and implementation of a neural network based predistorter for enhanced mobile broadband," in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*, 2019, pp. 296–301.