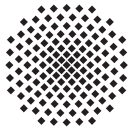


University of Stuttgart
Germany

Institute of
Robust Power
Semiconductor Systems

Master's Thesis

Digital Self-Interference Cancellation using FPGA for In-Band Full-Duplex Radios

Swapnil Sunil Roge

Supervisor: Laura Manoliu, M. Sc.
Prof. Dr.-Ing. Ingmar Kallfass

Period: 14.11.2022 – 14.05.2023

Stuttgart, 14.05.2023

Postanschrift: Institut für Robuste Leistungshalbleitersysteme
Pfaffenwaldring 47

D-70569 Stuttgart

Tel.: +49 (0) 711 685 68700
Fax.: +49 (0) 711 685 58700
E-Mail: sekretariat@ilh.uni-stuttgart.de
Web: www.ilh.uni-stuttgart.de

Declaration

I hereby declare that this thesis is my own work and effort and follows the regulations related to good scientific practice of the University of Stuttgart in its latest form. All sources cited or quoted are indicated and acknowledged by means of a comprehensive list of references.

Stuttgart, 14.05.2023

Swapnil Sunil Roge

Executive Abstract

Conventionally, the transmission and the reception of signals in a particular wireless communication system is performed using the half-duplex method, wherein the transmitter and the receiver signals are either time-multiplexed or frequency-multiplexed. However, in case of an in-band full-duplex system, the bidirectional communication of signals is performed simultaneously in the same frequency band, which improves the spectral efficiency of these systems by a factor of two as compared to the traditional half-duplex systems. Therefore, the in-band full-duplex communication systems can double the data rate provided by the half-duplex communication systems, thereby making the former a matter of interest across the wireless research community.

However, the in-band full-duplex systems have their own set of disadvantages. The major challenge is the self-interference imposed by the high-power transmitter signal on the incoming low-power receiver signal, which further degrades the latter and negatively impacts its estimation. Out of the various methodologies to mitigate the self-interference from the receiver signal, this work focuses on the digital self-interference cancellation techniques.

In this thesis, the effects of the self-interference signal on the receiver signal are examined. Furthermore, the different digital self-interference cancellation methods employed for suppressing the self-interference are comparatively analysed. Finally, the field-programmable gate array based implementation of the various digital self-interference cancellation algorithms and their respective performance results are presented as well.

Zusammenfassung

Konventionell erfolgt das Senden und Empfangen von Signalen in einem bestimmten drahtlosen Kommunikationssystem im Halbduplexverfahren, wobei die Sender- und Empfängersignale entweder zeit- oder frequenzmultiplexiert werden. Bei einem In-Band-Vollduplex-System wird die bidirektionale Kommunikation von Signalen jedoch gleichzeitig im selben Frequenzband durchgeführt, was die spektrale Effizienz dieser Systeme im Vergleich zu den herkömmlichen Halbduplex-Systemen um den Faktor zwei verbessert. Daher können die In-Band-Vollduplex-Kommunikationssysteme die von den Halbduplex-Kommunikationssystemen bereitgestellte Datenrate verdoppeln, was erstere zu einer Angelegenheit von Interesse für die gesamte Mobilfunkforschungsgemeinschaft macht.

Die In-Band-Vollduplex-Systeme haben jedoch eine Reihe von Nachteilen. Die größte Herausforderung ist die Selbstinterferenz, die durch das Sendersignal mit hoher Leistung auf das eingehende Empfängersignal mit niedriger Leistung ausgeübt wird, was letzteres weiter verschlechtert und sich negativ auf seine Schätzung auswirkt. Von den verschiedenen Methoden zur Abschwächung der Selbststörung des Empfängersignals konzentriert sich diese Arbeit auf die digitalen Techniken zur Unterdrückung der Selbststörung.

In dieser Arbeit werden die Auswirkungen des Selbststörsignals auf das Empfangssignal untersucht. Darüber hinaus werden die verschiedenen digitalen Methoden zur Unterdrückung der Selbststörung vergleichend analysiert. Schließlich werden die auf einem feldprogrammierbaren Gate-Array basierende Implementierung der verschiedenen digitalen Algorithmen zur Unterdrückung der Selbststörung und ihre jeweiligen Leistungsergebnisse vorgestellt.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

The communication industry faces tremendous pressure to meet the increasing demand of higher data rates in this digital era. Improving the spectral efficiency of the wireless network is one of the ways of increasing the data transmission speed. Conventional wireless communication technologies utilise a half-duplex (HD) system which involves multiplexing of the bidirectional signals over frequency or time creating a limitation on the efficient utilisation of the available spectrum. Even the state-of-the-art techniques such as multiple-input and multiple-output (MIMO) [16, 31] and orthogonal frequency-division multiplexing (OFDM) [15] cannot effectively meet the required demand for high data rates because the underlying systems using these technologies employ the HD communication [30].

The wireless research communities are therefore motivated to develop in-band full-duplex (IBFD) systems which allow simultaneous bidirectional signal communication over the same frequency channel. Thus, IBFD systems can theoretically improve the spectrum efficiency by a factor of two as compared to the conventional HD systems. Development of IBFD communication networks can have significant impact on the long-term evolution (LTE) industry, for example, which conventionally utilise two separate frequency bands for the uplink and the downlink channels for achieving the full-duplex operations. IBFD can allow the LTE networks to achieve the same quality of service by utilising only a single channel or a frequency band, and halve their current spectrum needs [6].

1.2 Challenges

1.2.1 Self-Interference Signal

Although IBFD has proven to be meritorious over the conventional HD systems, there is a significant disadvantage preventing the practical IBFD systems to achieve the doubling of the spectral efficiency. This major downside is caused by the large power difference between the self-interference (SI) imposed by the transmitter (TX) signal and the receiver (RX) signal. The figures 1.1 and 1.2 illustrate the effect of SI on the RX signal [22].

The SI signal generally consists of two components – the direct (or linear) and the reflected (or non-linear) SI component. When the bidirectional communication occurs at the same frequency, the TX signal introduces a co-channel interference on the RX signal, corresponding to the direct SI

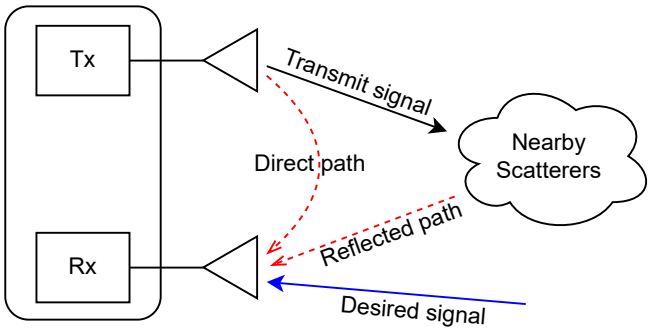


Figure 1.1: The effect of SI in a typical separate-antenna IBFD system.

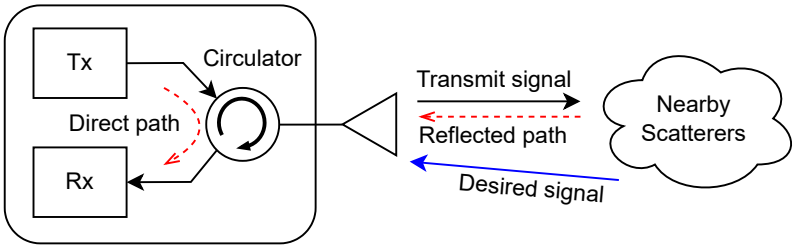


Figure 1.2: The effect of SI in a typical shared-antenna IBFD system.

component. In general, the magnitude of this component is greater than the actual RX signal. Since the direct SI component originates from the TX signal itself, it is already known to the transceiver. Therefore, this component can be estimated and suppressed from the RX signal. However, the second component, that is the reflected SI, occurs due to the reflection from the propagation medium which cannot be known in advance since the medium/channel is dynamic in nature. Therefore, the cancellation of the reflected SI component is comparatively more challenging [19].

As an example (refer figure 1.4), the different SI components induced by the TX signal in a typical Wi-Fi radio with a bandwidth (BW) of 80 MHz, the RX noise floor of -90 dBm and the TX power of 20 dBm are as follows [6,30]:

- Linear/main component of 20 dBm strength (110 dB above the noise floor).
- Non-linear component of -10 dBm strength (80 dB above the noise floor).
- Transmitter noise of -40 dBm strength (50 dB above the noise floor).

The linear component arises from the signal attenuations and reflections from the environment. Secondly, the non-linear component is contributed by circuit power leakage, non-flat hardware frequency response and higher order signal harmonics. Finally, the transmitter noise is induced by the imperfect designs of the TX power amplifier (PA) and local oscillator phase noise [6].

1.2.2 Suppression of Self-Interference Signal

Accurate decoding and/or estimation of the signal of interest requires effective cancellation of all the components of the SI induced by the TX signal. Therefore, suitable SI cancellation techniques must

be incorporated which can successfully predict and thereby model the distortions created by the SI which may help provide a sufficient compensation to the signal of interest [11] by suppressing the SI power to the same level as the RX noise floor.

The different types of SI cancellation techniques are classified as shown in the figure 1.3 [30].

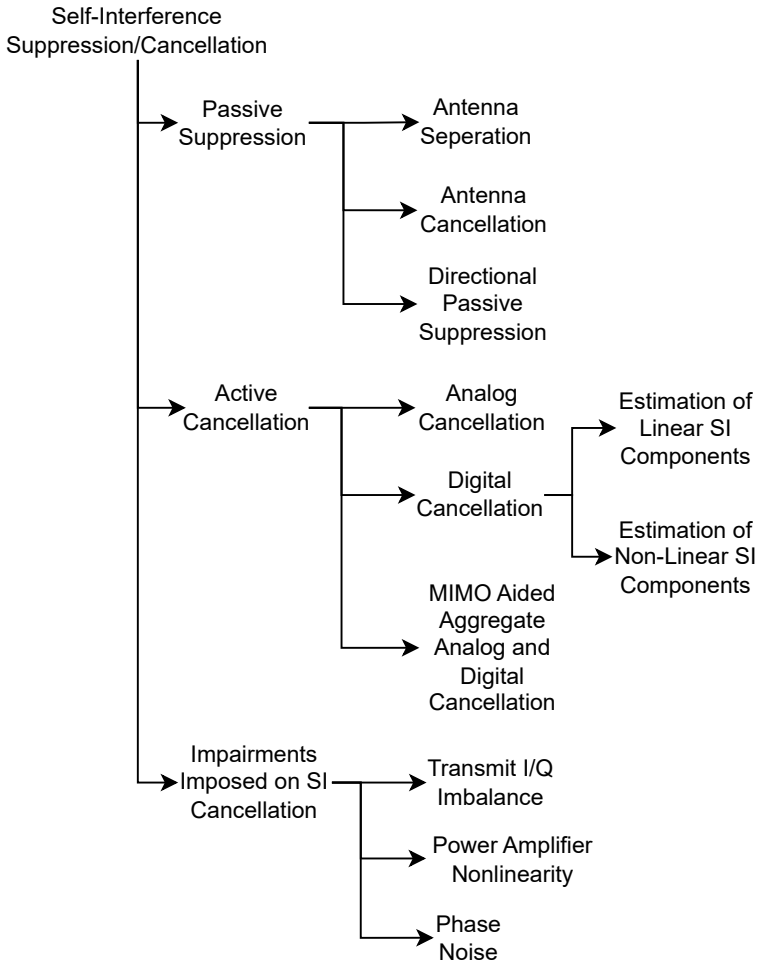


Figure 1.3: The classification of different SI cancellation/suppression techniques.

The figure 1.4 shows the different components of the SI induced by the TX signal signal for a typical Wi-Fi radio previously mentioned in the section 1.2.1 [6]. The RX signal demands an analog and a digital SI cancellation of 60 dB and 50 dB respectively for suppressing the SI to the RX noise floor as shown. Under certain situations, such as hardware imperfections, the analog and digital cancellations might not be enough. In such cases, passive suppression techniques are utilised along with the analog and digital SI suppression techniques. The passive suppression uses the path-loss effect between the TA and the RA to cancel out the SI [7].

Apart from negatively impacting the RX signal estimation, the SI signal also limits the spectral efficiency which might cause serious problems such as destabilisation of the communication system [30] and oscillations in the transceivers [23]. Therefore, the implementation of SI cancellation measures is an important aspect of the IBFD communication systems [1,2,20].

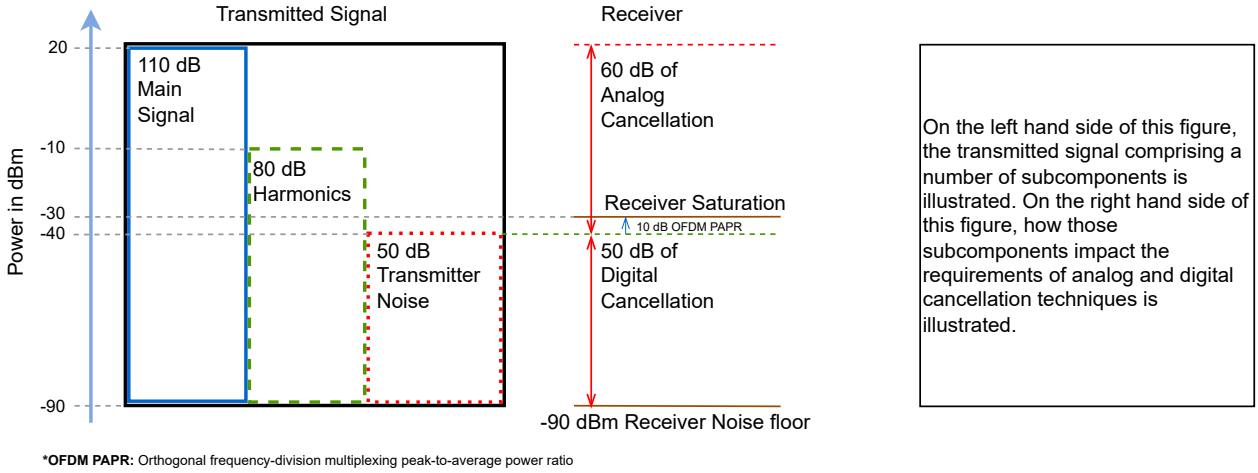


Figure 1.4: The different SI components induced by the TX signal in a typical Wi-Fi radio and the corresponding SI suppression methods required in the RX signal.

Nevertheless, every SI cancellation method plays its own role and cannot be used as a substitute for the other SI suppression methods. Although this thesis focuses only on the digital SI cancellation technique, it is assumed that other SI cancellation approaches as shown in the figure 1.3 have already been implemented as required in the transceiver system, because the digital SI cancellation alone cannot suppress the SI induced in the RX signal.

2 State-of-the-art Self-Interference Cancellation Techniques

This chapter deals with the various sources of SI occurring in the IBFD transceivers along with their respective state-of-the-art cancellation/suppression techniques. Although this thesis particularly deals with the digital SI cancellation techniques, the other cancellation techniques are still given a brief treatment.

The figure 2.1 [30] shows the various SI cancellation techniques which have to be applied chronologically across the different signal paths in a typical IBFD RX system.

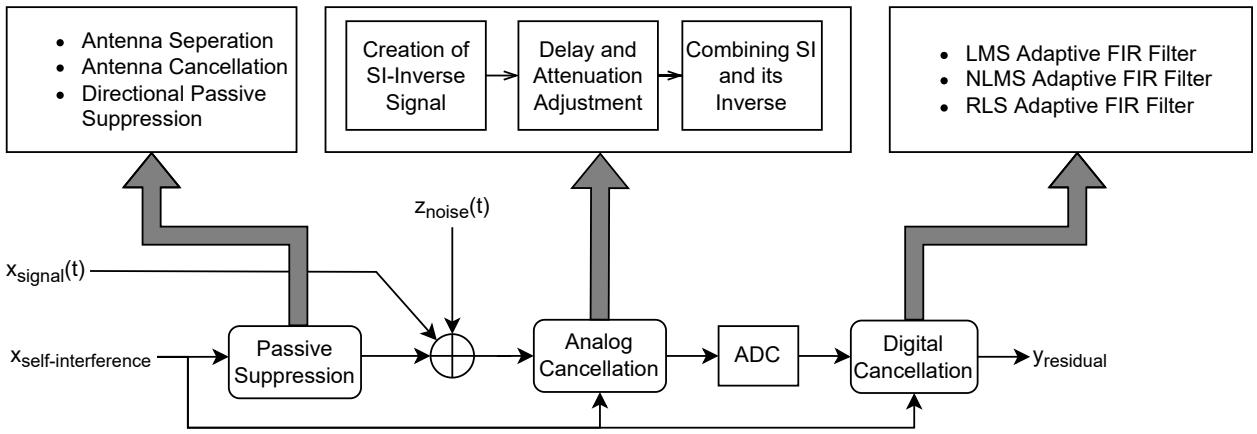


Figure 2.1: The various SI cancellation/suppression techniques employed across the different signal paths in a typical IBFD RX system.

Based on the chronological order of execution of the various SI cancellation methods inside any IBFD transceiver system, the SI cancellation techniques to be discussed in this chapter are as follows:

- Passive Suppression
- Analog Cancellation
- Digital Cancellation

2.1 Passive Suppression

The passive suppression techniques deal with the impact of SI which occurs even before any signal impinges upon the RA, as described in the figure 1.1. The passive suppression techniques, which

are considered to be the first line of defence against the SI signal, rely on achieving a high amount of TA-RA isolation for SI cancellation. The basic principle of such techniques revolve around accomplishing the TA-RA electromagnetic decoupling, which attenuates the SI signal even before it reaches the RA. Based on the methodology of attaining such kind of separation/isolation, the passive suppression techniques are further classified as follows [30]:

- Antenna Separation (AS)-Based Passive Suppression
- Antenna Cancellation (AC)-Based Passive Suppression
- Directional Passive Suppression

2.1.1 Antenna Separation-Based Passive Suppression

The AS methodology achieves the SI suppression by exploiting the effect of TA-RA path-loss. In this case, the SI component in the RX signal is effectively reduced by simply increasing the physical TA-RA separation within a typical IBFD separate-antenna system shown in the figure 1.1.

For instance, in a typical full-duplex (FD) compact-relay antenna system operating at 2.6 GHz centre frequency in a multipath environment initially achieved a TA-RA isolation of 48 dB. This isolation further improved to 70 dB upon increasing the TA-RA separation to 5 m, while ensuring the best possible TA-RA antenna orientation. Although an isolation of 70 dB is not enough for a typical FD communication, employing an interference canceller has the potential to increase the operational feasibility [10, 30].

2.1.2 Antenna Cancellation-Based Passive Suppression

In this technique, the concept of destructive interference is brought to play. As shown in the figure 2.2 [7], the AC-based SI suppression involves the use of two TAs and a single RA such that the distance of both the TAs from the RA differ by an odd multiple of $\frac{\lambda}{2}$, where λ is the wavelength of the TX signal. Such kind of arrangement ensures that a destructive interference of the TA-induced SI occurs at the RA, thereby reducing the SI-impact in the RX signal. However, the most effective SI cancellation occurs only when both the TAs are transmitting with an identical signal power.

The system described in the figure 2.2 employs the use of two power splitters as shown, which introduce a 6 dB reduction in the signal. This makes the TA1 power 6 dB lower than the TA2 power. Therefore, the values of d and λ are chosen such that effectively, the powers corresponding to TA1 and TA2 are matched and the necessity of an additional attenuator for the power-matching is avoided. In practice, the AC-based passive SI suppression can potentially attenuate the SI signal within the RX systems by about 30 dB [7].

2.1.3 Directional Passive Suppression

In this method, the SI suppression is achieved by minimising the intersection between the main radiation lobes of the TA and the RA. This is usually achieved by adjusting the TA-RA angle in a typ-

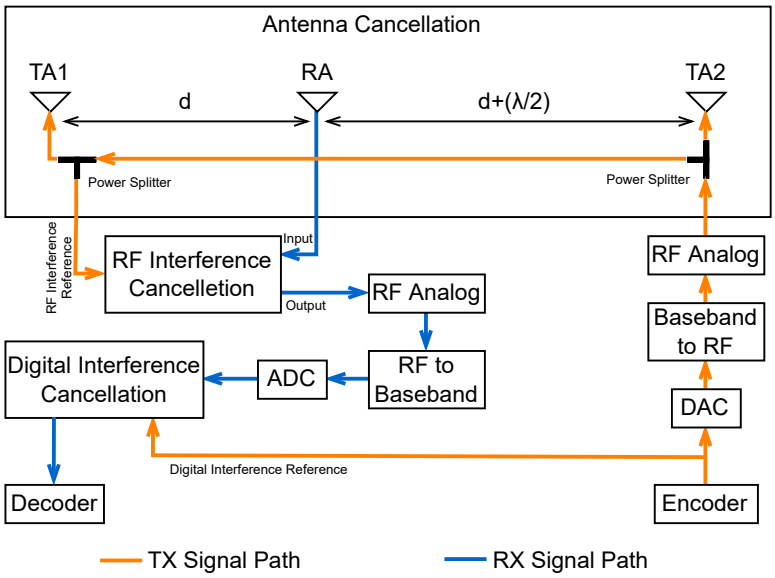


Figure 2.2: The block diagram of a wireless IBFD node employing an AC-based passive SI suppression method.

ical FD node. In [9], a research evaluating this type of passive SI suppression technique employed in a typical cellular base-station was performed, wherein the experimental results showed that the directional antennas perform better than the respective omnidirectional counterparts in providing significant SI suppression. Moreover, the combined incorporation of the digital SI cancellation techniques along with the directional passive suppression effectively contributes in achieving the FD double data-rate in the base-station, when the TA-RA angle is maintained within 90° to 120° [9,30].

2.2 Analog Cancellation

The Wi-Fi example discussed in the section 1.2.1 and the figure 1.4 recommends a 110 dB suppression of the SI component in the RX signal for a successful IBFD operation. However, none of the passive suppression techniques discussed in the section 2.1 provide enough SI cancellation for the same. It is therefore necessary to incorporate the analog SI cancellation methods.

The basic principle of this cancellation technique involves the creation of a reference signal corresponding to the exact copy of the SI signal, at all possible instances. This reference signal is then subtracted from the contaminated RX signal in order to attenuate the SI power to an acceptable level [12]. Although the majority of the analog SI cancellation methodologies operate at the radio frequency (RF) level [7,8,12], some of the techniques can be performed at the base-band frequency level [21] as well.

A typical balun (balanced/unbalanced) transformer-based analog-domain SI cancellation method [12], as shown in the figure 2.3, involves the following steps [30]:

- Creation of Self-Interference Inverse Signal (or RF Reference Signal).

- Delay and Attenuation Adjustment.
- Cancellation of Self-Interference Signal.

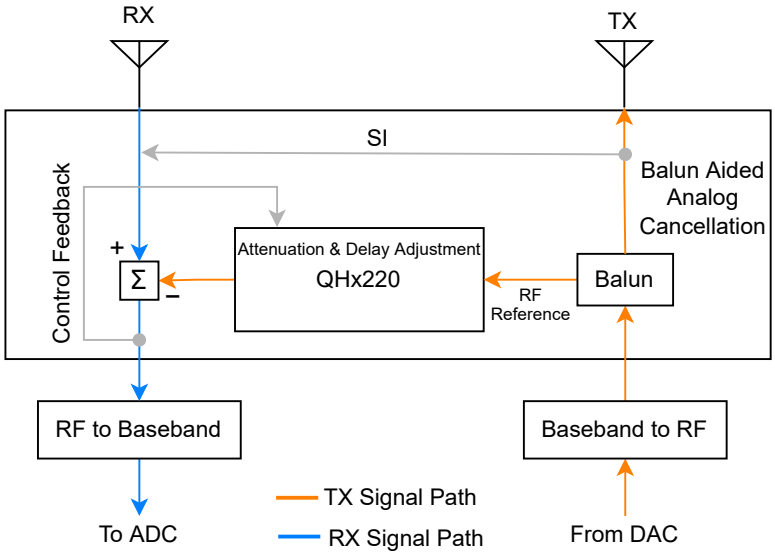


Figure 2.3: The block diagram of a typical balun-aided analog SI cancellation method.

2.2.1 Creation of Self-Interference Inverse Signal

The SI signal is inverted simply by performing phase-inversion. An FD radio can be specifically designed to accomplish this operation. Although this method provides feasible SI cancellation only at the central frequency, practical wide BW applications often rely on balun transformers [12] for generating the SI-inverse signal, as shown in the figure 2.3.

2.2.2 Delay and Attenuation Adjustment

In practice, the RX signal undergoes attenuation and delay as it propagates through the air or any other medium. Therefore, if an identical attenuation and delay is applied to the RF reference (or inverted-SI) signal output from the balun transformer, there is a possibility of a significant SI suppression. For example, the QHx220 RF SI cancellation chip [17] can provide such kind of an adaptive and controllable phase and amplitude adjustment for the aforementioned systems with a frequency range of 300 MHz to 3 GHz.

2.2.3 Cancellation of Self-Interference Signal

The SI-inverse signal output from the delay and attenuation adjustment stage is combined at the RA for nullifying the SI effect, as described in the figure 2.3. In practice, the residual SI power is non-zero due to hardware imperfections due to reasons including power-leakage and/or non-flat frequency response of the balun transformer. Although perfect-SI cancellation is not possible in

practice, various self-tuning algorithms are available [12], which minimise the residual SI power to the lowest possible level by adaptively adjusting the phase and amplitude in the delay and attenuation adjustment stage.

The aforementioned technique has undergone various experimental evaluations. For instance, in a research study involving a typical 900 MHz band network, the RF-level analog SI cancellation circuit incorporating a QHx220 chip [17] alone provided a cancellation of about 20 dB, which improved to approximately 50 dB, when the antenna-based cancellation techniques were employed as well [7, 18]. Another example consists of a typical 40 MHz BW network, which achieved an SI cancellation of about 45 dB using the SI-inverse technique without employing any other cancellation methods [12].

Nevertheless, the analog SI cancellation method described in this section is unable to completely cancel out the SI power due to reasons such as the hardware imperfections caused by the RF power-leakage, as well as the non-flat frequency response of the balun transformer [30]. Therefore, the digital SI cancellation techniques must be incorporated to deal with the residual SI power, as shown in the figure 2.1.

2.2.4 Avoiding Analog Self-Interference Cancellation

This section discusses the possibility of skipping the aforementioned analog cancellation technique and undergoing the digital SI cancellation process directly. However, this explanation requires a certain degree of background information about the internal structure of a typical FD RX node (refer figure 2.4) and therefore, it is discussed first, followed by the explanation concerning the avoidance of the analog SI cancellation.

The RX signal obtained from the passive SI suppression stages undergo amplification by the automatic gain control (AGC) before it is down-converted to the baseband frequency as shown in the figure 2.4. The AGC performs the normalization of the RX signal to the range of [-1, 1], which makes the signal suitable for the filtering and the analog-to-digital converter (ADC) sampling collectively known as the digitization process. The digital SI cancellation techniques can then be employed after the digitization stage, if needed.

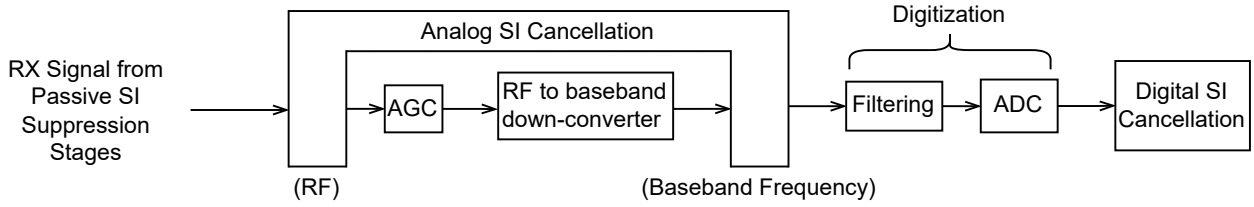


Figure 2.4: The block diagram of the internal structure of a typical FD RX node.

However, the SI component in the RX signal is very strong as compared to the desired signal component, in general. This saturates the AGC and therefore, the desired RX signal component occupies a relatively smaller share of the normalized AGC output range ([-1, 1]). This prevents an

efficient usage of the ADC dynamic range, since it captures a relatively smaller proportion of the desired part of the incoming signal. Moreover, a comparatively stronger SI power component in the RX signal increases the quantization noise at the ADC output, resulting into an overall negative signal-to-interference-plus-noise ratio (SINR). This further complicates the SI suppression using the digital SI cancellation methods [12, 13]. Therefore, attenuating the SI power component in the RX signal using the analog cancellation methods is crucial, and it must be performed before the digitization stage.

To summarise, the analog cancellation methods greatly improve the resolution of the desired signal component, thereby allowing efficient digital SI cancellation in the upcoming stages [7].

2.3 Digital Cancellation

The digital self-interference cancellation (DSIC) techniques act as the last line of defence against the residual SI power, that escaped the influence of the passive and the analog SI cancellation methods discussed in the sections 2.1 and 2.2 respectively. As shown in the figure 2.4, the DSIC techniques come into picture after the digitization process is performed. The working principle of the DSIC techniques involve two fundamental steps - firstly, estimating the SI channel and secondly, subtracting the known TX signal from the RX signal after passing the former through the estimated channel obtained in the first step [30].

The RX signal, which is obtained after the analog cancellation stage, still contains a residual SI power comprising of the linear and the non-linear component. While the linear component has static/time-invariant properties, the non-linear component is dynamic/time-variant in nature, since the latter is contributed by the reflections from the continuously changing surroundings, as mentioned in the section 1.2.1. Therefore, it is important to estimate not only the linear-component, but also the non-linear component of the residual SI channel, to ensure a greater degree of the digital SI suppression [6, 30].

2.3.1 Linear Self-Interference Channel Estimation

In this case, the SI component channel is modelled as a non-causal linear function of the known transmitted signal $x[n]$ [6, 30]. The received signal $y[n]$ at any instant n can then represented by the following equation:

$$y[n] = \sum_{z=1-k}^{k} x[n-z] \cdot h[z] + w[n] \tag{2.1}$$

where $h[n]$ denotes the SI channel attenuation, $w[n]$ represents the additive noise component and $k > 0$ denotes the memory length of the estimated linear-SI channel.

By defining $y = [\,y[0], ..., y[n]\,]^T$, where x^T denotes the transpose of the vector x, the estimated SI channel vector $\hat{h}$ is given by:

$$\hat{h} = (A^H A)^{-1} A^H y \tag{2.2}$$

where A is the training matrix defined by:

$$A = \begin{bmatrix} x[-k] & \cdots & x[0] & \cdots & x[k-1] \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x[n-k] & \cdots & x[n] & \cdots & x[n+k-1] \end{bmatrix} \tag{2.3}$$

and A^H denotes the Hermitian transpose of the matrix A. Pre-computing the value of A can substantially reduce the computational complexity of the above algorithm.

2.3.2 Non-Linear Self-Interference Channel Estimation

The non-linear SI channel estimation allows further suppression of the residual SI power; however, this estimation is so challenging that an approximate model based on a Taylor series expansion is generally incorporated in the digital domain [6, 30]. The aforementioned model is represented by the following equation:

$$y[n] = \sum_{\substack{m \in odd\ terms \\ n=-k,\ldots,k}} x[n] \cdot (|x[n]|)^{m-1} \cdot h_m[n] \tag{2.4}$$

In the equation 2.4, the first term corresponds to the linear SI component, which comprises of the majority of the residual SI power. However, this component has already been dealt with in the section 2.3.1. Moreover, as per [6], only the odd-ordered terms of the equation 2.4 actually contribute to the non-linear SI component. Nevertheless, the higher odd-ordered terms in the equation 2.4 contribute to a smaller proportion of the non-linear SI power component, since they involve the multiplication of fractional values which exponentially decrease with the increase in the odd-ordered index. This property can be exploited by allowing only a limited number of terms of the equation 2.4 to estimate the non-linear SI channel in practical applications.

This finishes the discussion of the state-of-the-art SI cancellation techniques. The upcoming chapters are oriented towards the detailed discussion, development and analysis of the various DSIC techniques.

3 Self-Interference Channel Estimation using Adaptive Finite Impulse Response Filters

This chapter involves the theoretical discussion of the DSIC methodology based on the adaptive finite impulse response (FIR) filters, which are used for predicting/estimating the unknown SI channel in the digital domain. The basic principle of this technique can be explained using the figure 3.1, which also serves as a model for describing the DSIC method based on the aforementioned SI channel estimation concept.

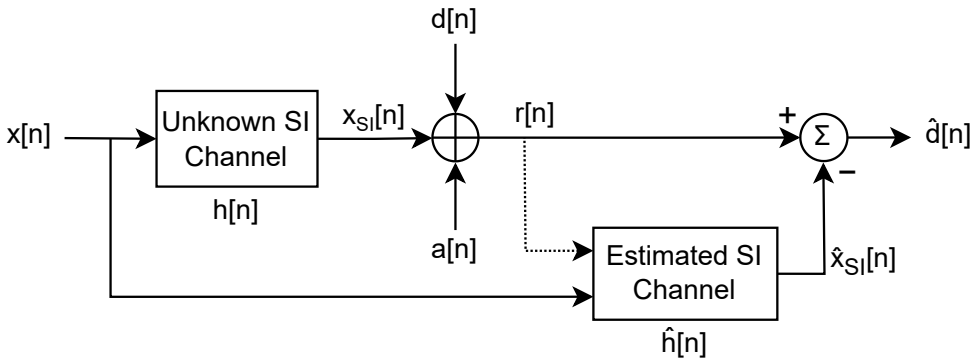


Figure 3.1: The block diagram of the adaptive FIR filter-based digital SI cancellation model.

The following list gives the information about all the digital domain terms/variables that are used in this thesis:

- **n** : The current time index.
- **x[n]** : The original TX signal.
- **h[n]** : The impulse response of the unknown SI channel.
- $\mathbf{x_{SI}[n]}$: The TX-induced SI signal (to be suppressed).
- **d[n]** : The desired RX signal.
- **a[n]** : The additive white Gaussian noise (AWGN) signal.
- **r[n]** : The observed RX signal, which is contaminated with $x_{SI}[n]$ and $a[n]$.
- $\hat{\mathbf{h}}\mathbf{[n]}$: The impulse response of the estimated SI channel (an estimate of $h[n]$).
- $\hat{\mathbf{x}}_{\mathbf{SI}}\mathbf{[n]}$: The estimated SI signal (an estimate of $x_{SI}[n]$).
- $\hat{\mathbf{d}}\mathbf{[n]}$: The estimated RX signal obtained after the SI signal suppression (an estimate of $d[n]$).
- **L** : The length of the FIR filter.

- $\mathbf{w_0[n]}, ..., \mathbf{w_{L-1}[n]}$: The weights (or coefficients) of the FIR filter.
- μ : The step size of the least mean squares and the normalised least mean squares filters (a constant value).
- λ : The forgetting factor of the recursive least squares filter (a constant value).

Referring to the figure 3.1, all the aforementioned terms can be expressed using the following equations:

$$x_{SI}[n] = x[n] * h[n] \tag{3.1}$$

$$r[n] = x_{SI}[n] + d[n] + a[n] \tag{3.2}$$

$$\hat{x}_{SI}[n] = x[n] * \hat{h}[n] \tag{3.3}$$

$$\hat{d}[n] = r[n] - \hat{x}_{SI}[n] \tag{3.4}$$

where $*$ denotes the convolution operation.

The major task in the aforementioned methodology is in determining $\hat{x}_{SI}[n]$, which is also the focus of this discussion. Once that is calculated, the DSIC operation can simply be performed using the equation 3.4.

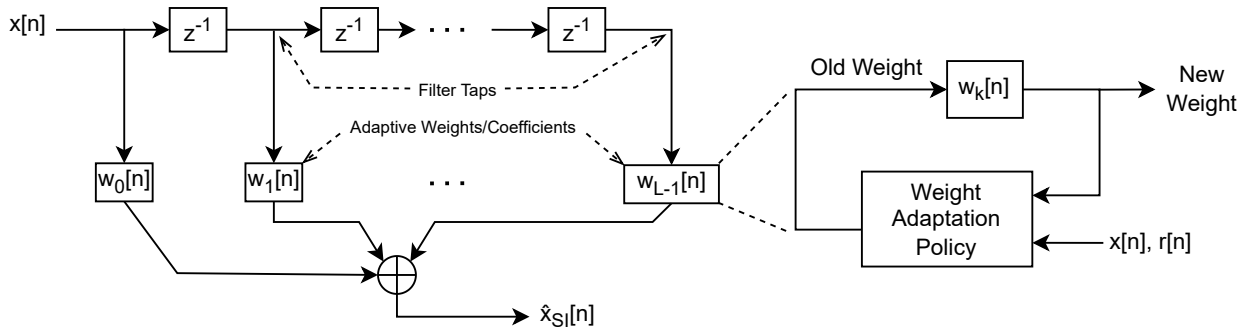


Figure 3.2: The block diagram of the estimated SI channel.

The figure 3.2 gives a detailed description about the internals of the estimated SI channel, which basically consists of an adaptive FIR filter of length L. The term z^{-1} denotes a delay element and the terms $w_0[n], ..., w_{L-1}[n]$ denote the time-varying filter weights (or coefficients), which must be iteratively modified to ensure the maximum possible cancellation of the residual SI power. Owing to the adaptive nature of these weights, the corresponding filters are therefore called as the adaptive FIR filters. Based on the adaptation policy used for modifying the weights, the various adaptive FIR filters to be discussed in this work are as follows:

- Least Mean Squares Filter
- Normalised Least Mean Squares Filter
- Recursive Least Squares Filter

Nevertheless, the aforementioned techniques require the following two conditions to be fulfilled for a successful operation:

- The IBFD RX system must have a priori knowledge of $x[n]$.
- The $x[n]$ and $d[n]$ signal must be different (they must have different symbol sequences). Refer section 6.1 for further explanation about this point.

3.1 Least Mean Squares Filter

The figure 3.3 shows the block diagram of the least mean squares (LMS) adaptive FIR filter.

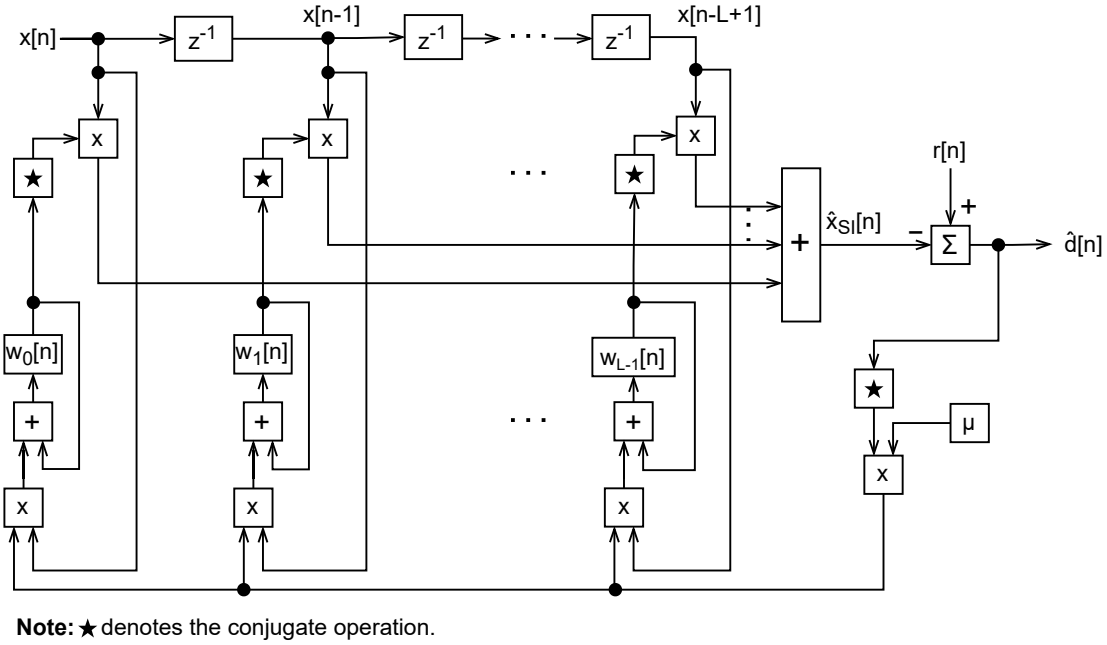


Figure 3.3: The block diagram of the LMS adaptive FIR filter.

This filter can be explained using the following equations:

$$\hat{x}_{SI}[n] = \sum_{k=0}^{L-1} w_k^*[n] \cdot x[n-k] \tag{3.5}$$

$$\hat{d}[n] = r[n] - \hat{x}_{SI}[n] \tag{3.6}$$

$$w_k[n+1] = w_k[n] + \mu \cdot \hat{d}^*[n] \cdot x[n-k]\,; \quad k \in \{0,...,L-1\} \tag{3.7}$$

where $w_k^*[n]$ and $\hat{d}^*[n]$ denote the conjugates of $w_k[n]$ and $\hat{d}[n]$ respectively.

In this filter, the weights are adapted such that the mean square error between $r[n]$ and $\hat{x}_{SI}[n]$ is minimised. The equation 3.5 provides the estimated SI signal $\hat{x}_{SI}[n]$. The term $x[n-k]$ denotes a delay in $x[n]$ by k samples, due to the k^{th} delay element (z^{-1}) as shown in the figure 3.3. Next, the equation 3.6 represents the DSIC operation, which is used to obtain the estimated version of the desired RX signal $\hat{d}[n]$. Finally, the equation 3.7 performs the update of the k^{th} weight for the next iteration, by using the current values of $\hat{d}^*[n]$ and $x[n-k]$. Generally, all the initial weight values

can be assumed as 0. As the LMS filter iterates further, each of the weights eventually converge to their respective optimal values, which contribute in providing the best possible estimate of the SI signal. Moreover, the term μ in the equation 3.7 denotes the step size, which is a constant value that determines the convergence rate of the LMS filter. The value of μ must be in the following range, for a successful convergence:

$$0 < \mu < 2 \tag{3.8}$$

Although a larger μ increases the filter convergence time, it may cause the weights to converge to non-optimal values, which creates a roadblock in the pursuit of the best possible SI suppression. However, if μ is very small, the weights can reach their optimal values at the cost of increased convergence time [27]. Therefore, it is important to perform a correct empirical determination (or tuning) of the value of μ.

3.2 Normalised Least Mean Squares Filter

The LMS adaptive filter discussed in the section 3.1 has a major drawback, which makes it very sensitive to the scaling of $x[n]$. To elaborate, this filter becomes unstable in presence of an amplitude fluctuation in $x[n]$. Therefore, the empirical determination of μ for a stable operation of the LMS filter becomes difficult in this case [27].

As a solution to this demerit, the normalised least mean squares (NLMS) adaptive filter is developed (figure 3.4), wherein the problem caused due to the amplitude scaling of $x[n]$ is solved by normalising this signal with its own power, as expressed in the following equation:

$$w_k[n+1] = w_k[n] + \mu \cdot \hat{d}^*[n] \cdot \frac{x[n-k]}{c + \sum_{k=0}^{L-1} |x[n-k]|^2} \; ; \quad k \in \{0, ..., L-1\} \tag{3.9}$$

where c is a safety factor, which is a small positive constant added to prevent the denominator from becoming 0. Theoretically, the only difference between the NLMS and the LMS filter is the weight-update equation 3.9, which involves the normalisation of $x[n]$. Apart from this difference, the rest of the equations and the filter operation remains the same as described in the section 3.1.

3.3 Recursive Least Squares Filter

The recursive least squares (RLS) adaptive FIR filter follows a recursive weight adaptation policy, with a goal of minimising the weighted linear least squares error [29] between $x[n]$ and $r[n]$. Although this filter provides a very high convergence speed, it comes with a higher level of complexity and computational costs amongst the other adaptive filters discussed in the sections 3.1 and 3.2 [19, 28]. The RLS filter can be mathematically expressed using the following equations:

$$g_k[n+1] = \frac{\lambda^{-1} \cdot P_k[n] \cdot x[n-k]}{1 + \lambda^{-1} \cdot P_k[n] \cdot |x[n-k]|^2} \; ; \quad k \in \{0, ..., L-1\} \tag{3.10}$$

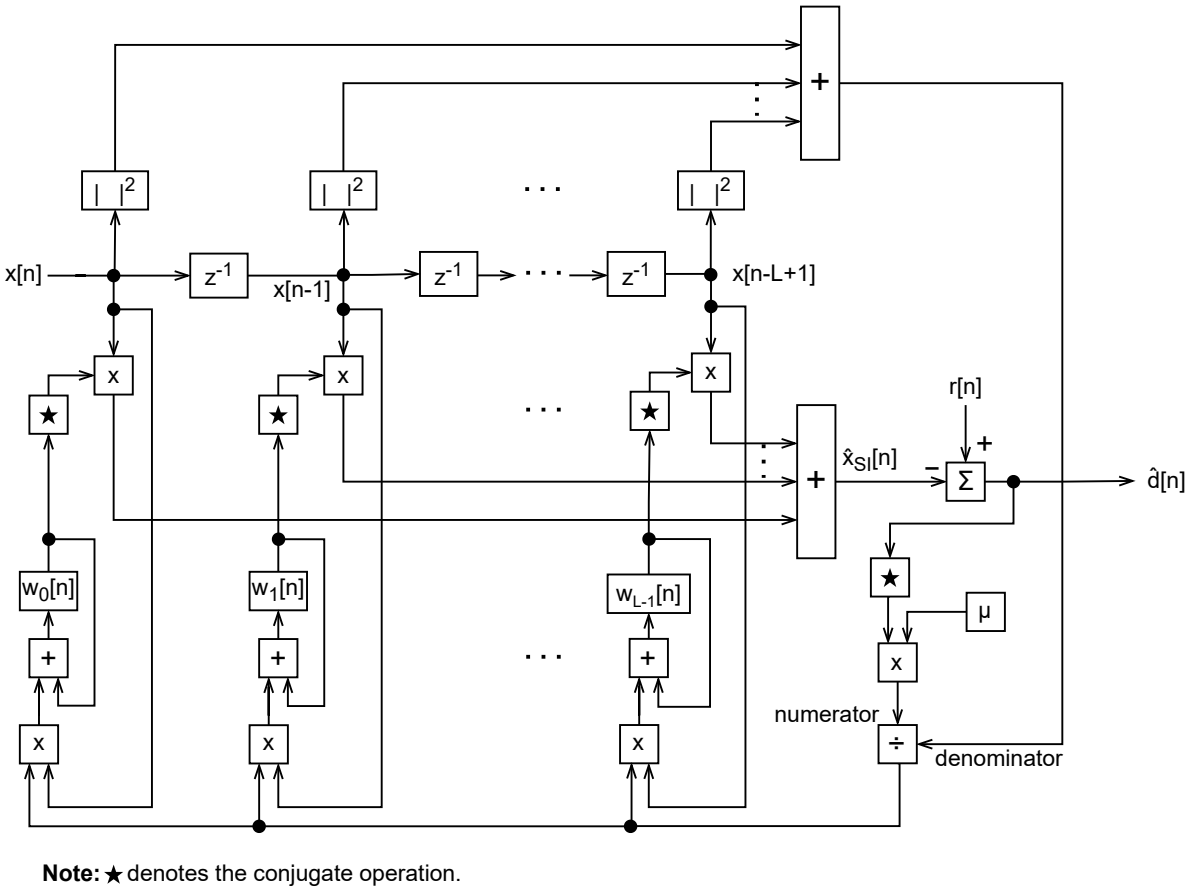


Figure 3.4: The block diagram of the NLMS adaptive FIR filter.

$$\hat{x}_{SI}[n] = \sum_{k=0}^{L-1} w_k[n] \cdot x[n-k] \tag{3.11}$$

$$\hat{d}[n] = r[n] - \hat{x}_{SI}[n] \tag{3.12}$$

$$w_k[n+1] = w_k[n] + \hat{d}[n] \cdot g_k^*[n]\,; \quad k \in \{0,...,L-1\} \tag{3.13}$$

$$P_k[n+1] = \lambda^{-1} \cdot P_k[n]\,(1 - g_k[n] \cdot x^*[n-k])\,; \quad k \in \{0,...,L-1\} \tag{3.14}$$

where $y^*[n]$ denotes the conjugate of $y[n]$. As compared to the sections 3.1 and 3.2, the following new terms are introduced in this filter:

- $\mathbf{g_k[n]}$: The gain variable corresponding to the k^{th} tap.
- $\mathbf{P_k[n]}$: The inverse-covariance variable corresponding to the k^{th} tap.

The terms $g_k[n]$ and $P_k[n]$ calculated using the equations 3.10 and 3.14 respectively can be considered as intermediate variables used for iteratively updating the filter weights in the equation 3.13. Similar to the LMS counterpart, all the filter weights can be initialised to 0. However, using the equation 3.13, the weights are adapted such that the weighted linear least squares error [29] between $x[n]$ and $r[n]$ is minimised. Moving on, the estimated SI signal $\hat{x}_{SI}[n]$ can be calculated using the

equation 3.11. Finally, the equation 3.12 represents the DSIC operation, which is used to obtain the estimated version of the desired RX signal $\hat{d}[n]$.

The term λ used in the equations 3.10 and 3.14 is known as the forgetting factor, which is a constant value typically belonging to the following range [25]:

$$0 < \left(1 - \frac{1}{2L}\right) < \lambda < 1 \tag{3.15}$$

As the value of λ gets closer to 1, the number of previous samples that the RLS filter remembers increases. Thus, setting the λ as 1 signifies an infinite filter memory [25]. Moreover, the value of λ determines the convergence rate of the filter. Therefore, it is important to perform a correct empirical determination of the value of λ for achieving the best possible SI signal suppression. Next, the initial value of the inverse-covariance variable $P_k[n]$ is given by:

$$P_k[0] = \frac{1}{\sigma^2}\ ; \quad k \in \{0, ..., L-1\} \tag{3.16}$$

where σ^2 denotes the initial input variance estimate, which must be correctly tuned for achieving the best possible convergence time and the SI signal suppression.

3.4 Determination of Filter Length

The adaptive FIR filter theory described in this chapter involves a variable known as the filter length L. In this section, an attempt is made to determine the value of L, for striking the best possible balance between the filter performance and the complexity. In this case, the contributions from Nadh et al. [14] are taken into consideration, wherein a Taylor series approximation of the SI channel is achieved for the FD radios and an attempt is made to reduce the dimensionality of the parameter space of this unknown SI channel to 2 or 3.

3.4.1 Taylor Series Approximation Model for SI Signal

The figure 3.5 shows the manner in which the TX signal $y(t)$ interferes with the RX chain in a typical FD transceiver system. The terms a_k and τ_k shown in the figure 3.5 denote the gain and the delay, respectively of the different components of the interference signal $I(t)$ mentioned in terms of $y(t)$. These different components of $I(t)$ cover all the paths of the SI signal, such as the leakage-path via the TA-RA coupling, the leakage-path through the printed circuit board (PCB) (in case of board-based systems) or through the substrate (for integrated circuit-based systems) and also the SI-path due to the external environment-based reflections of the TX signal detected by the RA. Likewise, it is assumed that there are M different SI-leakage paths from the TX to the RX system. Therefore,

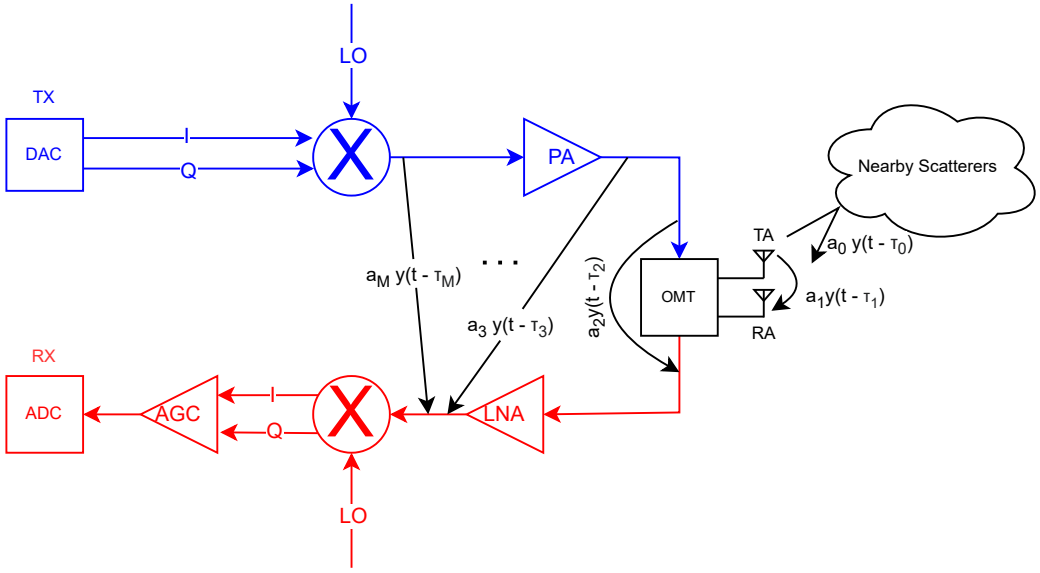


Figure 3.5: The block diagram depicting the leakage of the TX signal $y(t)$ within a typical FD transceiver.

the signal $I(t)$ is represented as:

$$\begin{aligned} I(t) &= \sum_{k=1}^{M} a_k \cdot y(t-\tau_k) \\ &= \sum_{k=1}^{M} a_k \cdot \mathrm{Re}[x(t-\tau_k) e^{j2\pi f_c (t-\tau_k)}] \end{aligned} \tag{3.17}$$

where $x(t)$ and f_c denote the complex baseband TX signal and the TX frequency, respectively. In the model shown in the figure 3.5, the variables M, a_k and τ_k are unknown, which makes the SI modelling complicated with $2M$ unknown values (M unknown values of a_k and τ_k each). Therefore, in [14], the Taylor series approximation is applied to $x(t)$ for simplifying the SI-model. Using this approximation, $x(t-\tau_k)$ is given by:

$$x(t-\tau_k) = x(t) - x'(t) \cdot \tau_k + E(\tau_k, t) \tag{3.18}$$

where $E(\tau_k, t)$ denotes the residual error (ignored for a smaller τ_k value) and $x'(t) = \frac{dx(t)}{dt}$. Substituting the equation 3.18 in 3.17 results into:

$$I(t) = \mathrm{Re}[x(t) \cdot e^{j2\pi f_c t} \cdot C_0 - x'(t) \cdot e^{j2\pi f_c t} \cdot C_1] + E_2(t) \tag{3.19}$$

where

$$C_0 = \sum_{k=1}^{M} a_k \cdot e^{-j2\pi f_c \tau_k} \tag{3.20}$$

and

$$C_1 = \sum_{k=1}^{M} a_k \cdot \tau_k \cdot e^{-j2\pi f_c \tau_k} \tag{3.21}$$

and

$$E_2(t) = \sum_{k=1}^{M} a_k \cdot \mathrm{Re}[E(\tau_k, t) \cdot e^{j2\pi f_c (t-\tau_k)}] \tag{3.22}$$

In the aforementioned equations, C_0 and C_1 are unknown constants, which must be determined for a given geometry of backscatters. It can be observed that, when the Taylor series approximation of $I(t)$ is expanded up to the $x'(t)$ term (equation 3.19), the number of unknowns are reduced to 2 (i.e. C_0 and C_1) from $2M$ (equation 3.17). For an even better approximation of $I(t)$, generally the Taylor series expansion of the same up to the term $x''(t)$ (where $x''(t) = \frac{d^2x(t)}{dt^2}$) is considered. As a result:

$$I(t) = \mathrm{Re}[C_0 x(t) e^{j2\pi f_c t}] - \mathrm{Re}[C_1 x'(t) e^{j2\pi f_c t}] + \mathrm{Re}[C_2 \cdot x''(t) \cdot e^{j2\pi f_c t}] + E_3(t) \tag{3.23}$$

where

$$C_2 = \sum_{k=1}^{M} \frac{a_k {\tau_k}^2}{2} \cdot e^{-j2\pi f_c \tau_k} \tag{3.24}$$

In case of the Taylor series approximation of $I(t)$ up to the $x''(t)$ term (equation 3.23), there are three unknowns obtained, namely C_0, C_1 and C_2 given by the equations 3.20, 3.21 and 3.24 respectively. As far as the term $E_3(t)$ (from the equation 3.23) is considered, this represents the residual error, whose value decreases with decreasing τ_k. If the linear-SI component of $I(t)$ is taken into consideration, $E_3(t)$ can be safely ignored, because the linear-SI component is majorly contributed by the SI occurring inside the FD transceiver (corresponding to a small τ_k value). Moreover, the non-linear SI-component with a higher τ_k occurs outside the FD transceiver, mostly due to the SI contributed by the environmental reflections of the TX signal detected by the RA. However, this type of the SI component has a very high attenuation, due to which the effective contribution of the non-linear SI components to the $E_3(t)$ term can be neglected. Therefore, the term $E_3(t)$ can be generally ignored for all the practical scenarios, in order to further reduce the complexity of this approximation. Furthermore, the Taylor series approximation can be expanded to incorporate even the terms with the order higher than $x''(t)$. However, this further improves the approximation of $I(t)$ at the cost of higher complexity.

The results of [14] stated that the Taylor series approximation of $I(t)$ up to the $x''(t)$ term (equation 3.23) provides a good approximation to model an unknown SI signal, leading to only three unknown terms, namely C_0, C_1 and C_2. This is a huge reduction in the unknown parameter space, when compared to the equation 3.17, with $2M$ unknown terms (for M different values of τ_k and a_k).

3.4.2 Application of Taylor Series Approximation Model

The aforementioned concept can be used as a basis to model an unknown SI channel for this thesis. Since the number of unknown variables involved in the best possible approximation of the SI signal

$I(t)$ amounted to only 3 (namely C_0, C_1 and C_2), this logic can be extrapolated to the fact that any unknown SI channel can be effectively cancelled using an adaptive FIR filter of length $L = 3$. In this case, the three unknown filter weight values, namely w_0, w_1 and w_2, are considered analogous to the terms C_0, C_1 and C_2 as mentioned in the equations 3.20, 3.21 and 3.24 respectively. Consequently, all the adaptive FIR filters are simulated and implemented with the filter length $L = 3$ and all the unknown weight values, namely w_0, w_1 and w_2, are iteratively determined using the different weight adaptation policies defined for the LMS, the NLMS and the RLS filters.

4 Simulation and Analysis of Digital Cancellation Techniques

This chapter discusses the following:

- **Simulation & Analysis of Self-Interference Signal:** Simulating and analysing the impact of TX-induced SI signal on the RX signal.
- **Analysis of Simulated Adaptive Finite Impulse Response Filters:** Analysing the DSIC performance of the simulated version of the adaptive FIR filters discussed in the chapter 3.

The MATLAB and Simulink software is used for the aforementioned purposes.

4.1 Simulation & Analysis of Self-Interference Signal

4.1.1 Simulation of Self-Interference Environment

The figure 4.1 describes the block diagram of the simulation environment used for studying the impact of SI signal on the RX signal.

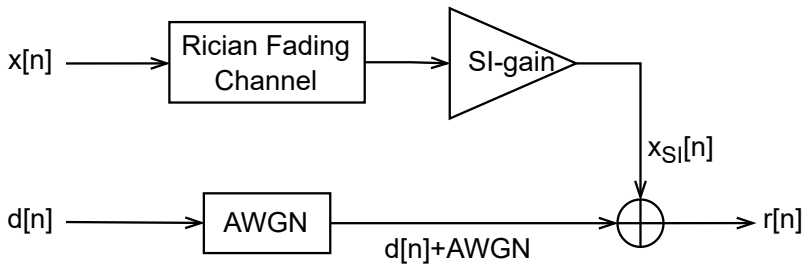


Figure 4.1: The block diagram of the simulation environment used for studying the impact of SI signal on the RX signal.

The signals $x[n]$ and $d[n]$ are generated using the quadrature phase-shift keying (QPSK) modulation technique. $x_{SI}[n]$ is generated by passing $x[n]$ through a Rician fading channel, followed by the SI-gain block. While the Rician fading channel is used to effectively mimic the direct and the reflected propagation of $x_{SI}[n]$ in a typical FD environment [3], the SI-gain block is employed to control the gain (in dB) of $x_{SI}[n]$. The Rician fading channel is characterised by the Rician factor k given by:

$$k = \frac{P_{x_{SI}[n],linear}}{P_{x_{SI}[n],non-linear}} \tag{4.1}$$

where $P_{x_{SI}[n],linear}$ and $P_{x_{SI}[n],non-linear}$ denote the signal powers of the linear (or direct) and the non-linear (or reflected) components of $x_{SI}[n]$ respectively. Finally, $r[n]$ is obtained by passing $d[n]$ through an AWGN channel and adding $x_{SI}[n]$ to the same. This AWGN channel is characterised by the signal-to-noise ratio (SNR) (dB).

The figure 4.2 compares the constellation diagrams of $d[n]$ and $r[n]$ for the different values of the Rician factor k, AWGN SNR (dB) and SI-gain (dB), as mentioned in the table 4.1.

Case	Rician factor k	AWGN SNR (dB)	SI-gain (dB)
1	10	20	0, -5, -10
2	10	15, 20, 25	0
3	10, 20, 30	20	0

Table 4.1: The different test-cases to be used for understanding the impact of SI on $r[n]$.

4.1.2 Analysis of Self-Interference Environment

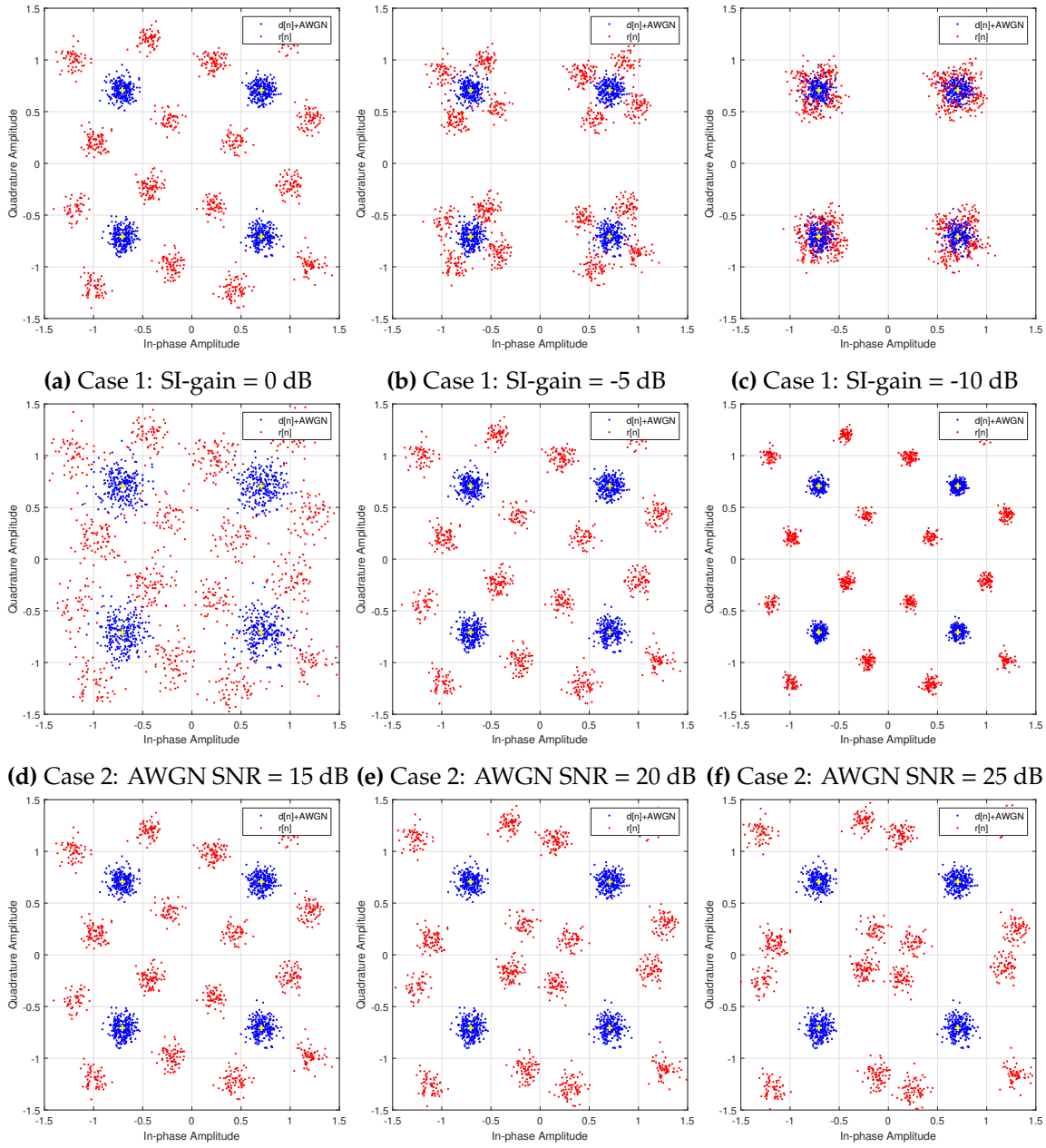
For analysing the performance of the SI environment, it is important to evaluate the quality of the all the involved signals. In this work, the signal quality is quantified using the root mean square (RMS) error vector magnitude (EVM) (%) [26] value due to the following reasons:

- The RMS EVM value is proportional to the distance of the received m-ary phase-shift keying (PSK) symbol from the ideal constellation point within a constellation diagram.
- Therefore, a lower RMS EVM value signifies closeness to the ideal constellation point, which means the signal quality is better. On the contrary, a higher RMS EVM value indicates a poor signal quality, since the probability that the received symbol gets incorrectly decoded increases as it deviates from its ideal location.

Moving on, the figure 4.3 along with the table 4.1 describe the impact of the Rician factor k, AWGN SNR (dB) and SI-gain (dB) on the RMS EVM (%) of r[n].

By analysing the figures 4.2 and 4.3, it can be understood that, for $r[n]$:

- A higher AWGN SNR value cannot guarantee a smaller (and better) value of RMS EVM. As observed from the figures 4.2d to 4.2f and 4.3b, even though initially the RMS EVM is very sensitive to the AWGN SNR, as the AWGN SNR increases beyond 20 dB, the RMS EVM values get saturated.
- Keeping the total SI power constant, if the linear-SI power-component is made stronger in proportion, it does not create much impact on the variation of the RMS EVM value. For instance, from the figures 4.2g to 4.2i and 4.3c, the Rician factor k does not significantly increase the RMS EVM for $k > 50$.
- However, the RMS EVM values show a significant jump in the two aforementioned cases with the increase in SI-gain. Moreover, it is evident from the figure 4.3a that the RMS EVM value increases almost exponentially with the SI-gain. Therefore, the SI-gain value, which is



(a) Case 1: SI-gain = 0 dB **(b)** Case 1: SI-gain = -5 dB **(c)** Case 1: SI-gain = -10 dB

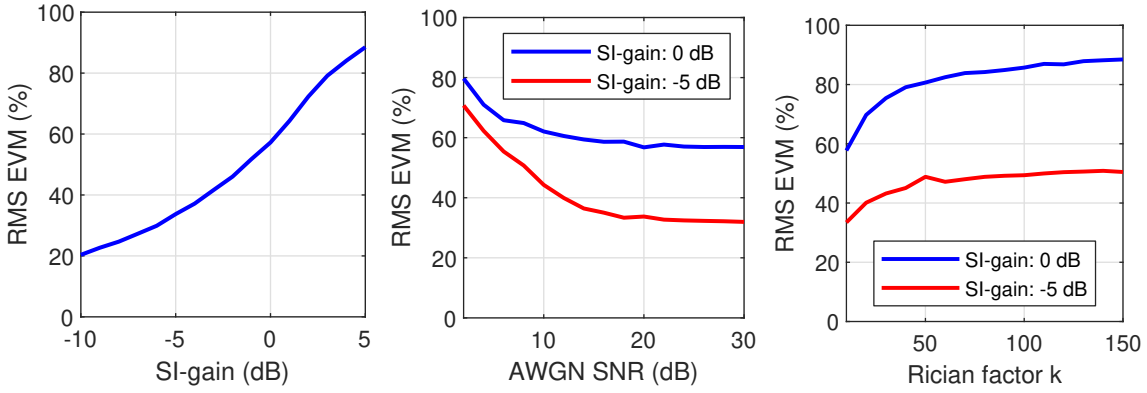
(d) Case 2: AWGN SNR = 15 dB **(e)** Case 2: AWGN SNR = 20 dB **(f)** Case 2: AWGN SNR = 25 dB

(g) Case 3: Rician factor $k = 10$ **(h)** Case 3: Rician factor $k = 20$ **(i)** Case 3: Rician factor $k = 30$

Figure 4.2: The various constellation diagrams of $d[n]$ and $r[n]$ plotted for the different test-cases mentioned in the table 4.1.

proportional to the total SI-power, strongly determines the course of the RMS EVM value.

Based on the aforementioned explanations, the SI-gain parameter is the most important out of the other parameters (AWGN SNR and Rician factor k) for evaluating the DSIC performance of the adaptive FIR filters.



(a) Case 1: Rician factor k=10, AWGN SNR=20 dB **(b)** Case 2: Rician factor k=10 **(c)** Case 3: AWGN SNR=20 dB

Figure 4.3: The variations in the RMS EVM (%) value of $r[n]$ plotted for the different test-cases mentioned in the table 4.1.

4.2 Analysis of Simulated Adaptive Finite Impulse Response Filters

This section involves the analysis of the DSIC performance of all the simulated-versions of the adaptive FIR filters discussed in the chapter 3. The figure 4.4 shows the block diagram of the adaptive FIR filter block used for the DSIC operation.

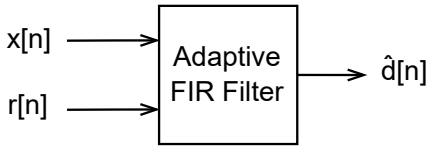


Figure 4.4: The block diagram of the adaptive FIR filter block used for the DSIC operation.

The filter shown in the figure 4.4 takes $x[n]$ and $r[n]$ as its inputs and provides $\hat{d}[n]$ as its output. It is nevertheless important to mention that the signals $x[n]$ and $r[n]$ are always known to the filter. However, the variables discussed in the section 4.1 such as the Rician factor k (dB) of the SI channel, the AWGN SNR (dB) of the RX channel and SI-gain (dB) of the SI channel are unknown to the adaptive filter. Therefore, the performance of the adaptive FIR filters must be tested against these unknown variables by taking into consideration the various filter parameters such as the filter type (LMS, NLMS or RLS filter) and the value of μ (for LMS and NLMS filter) or λ (for RLS filter).

The figure 4.5 (henceforth, consider d-hat[n]= $\hat{d}[n]$ for all the relevant plots in this thesis) shows the convergence characteristics of the different adaptive FIR filters (with filter length $L = 3$) subjected to a typical SI environment with Rician factor $k = 10$, AWGN SNR = 20 dB and initial SI-gain = -5 dB, which is later increased to 0 dB. The convergence characteristics help to understand the number of weight-update iterations needed by a particular filter to converge to the steady-state RMS EVM value, which can be considered as a good performance indicator for comparing the different adaptive FIR filters. Likewise, in the upcoming discussion, the performance analyses of various filters, in terms of their convergence characteristics, are described for the different SI

environments and filter parameters.

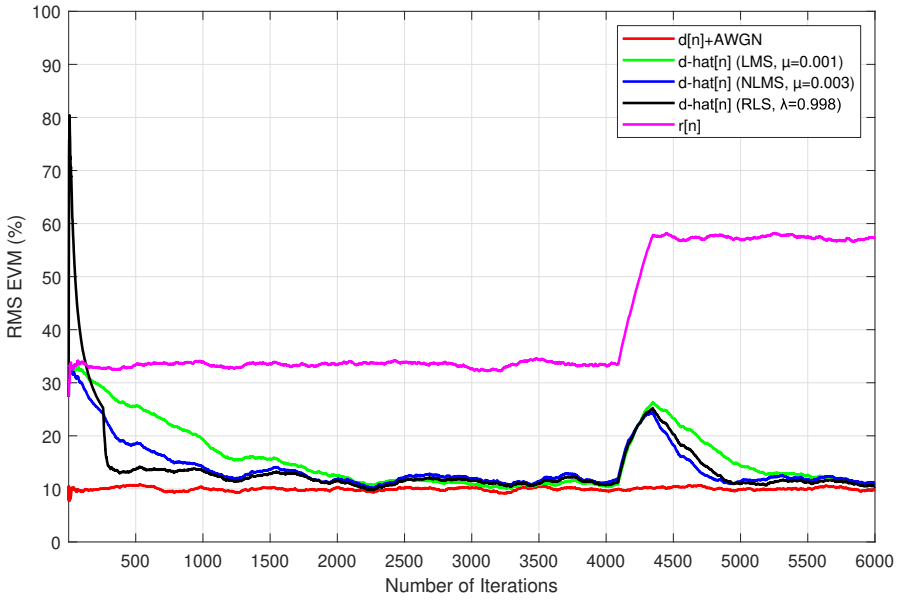


Figure 4.5: The convergence characteristics of the different adaptive FIR filters.

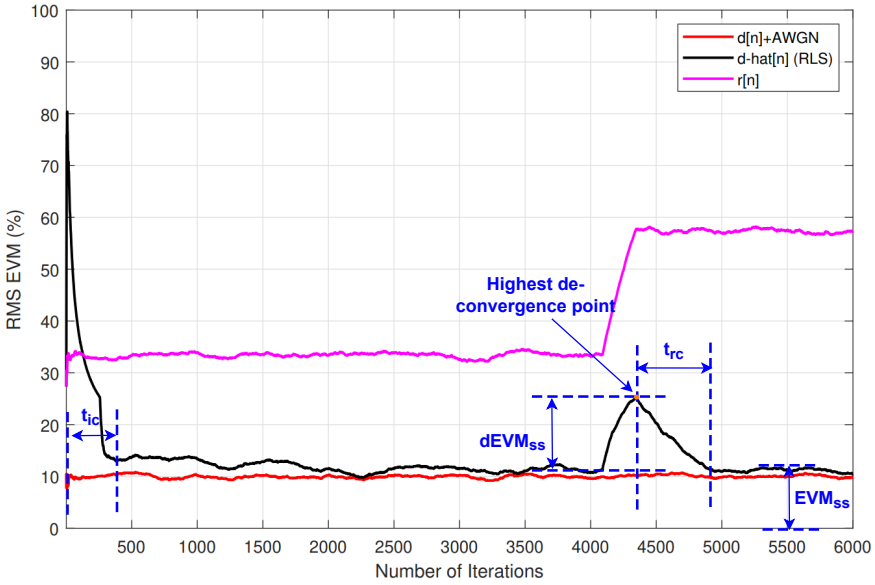


Figure 4.6: The reference plot used for explaining the different terms that are defined to analyse the adaptive FIR filters.

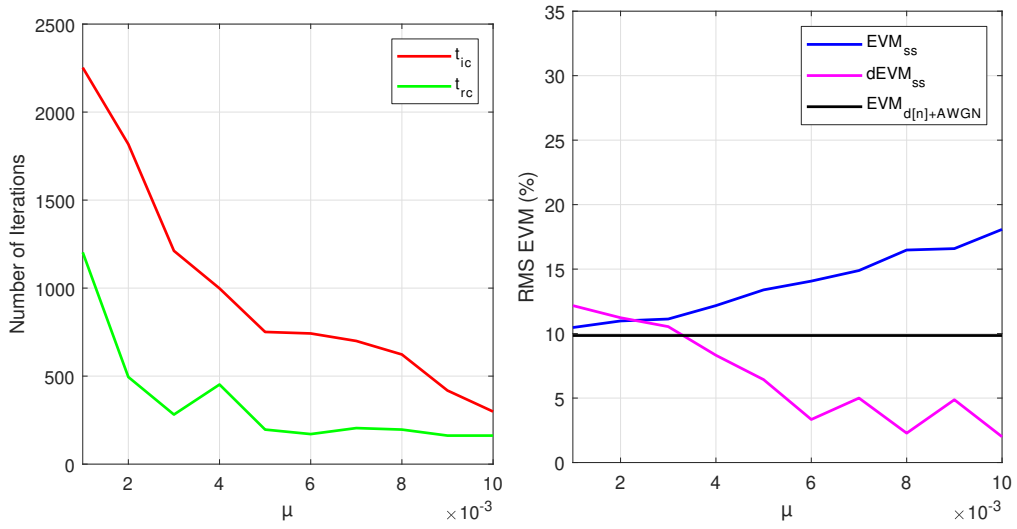
4.2.1 Terminologies for Analysing Filter Convergence Performance

Before proceeding ahead, it is important to introduce certain new terminologies (used throughout this thesis), which can help understand the filter performance analyses. Referring to the figure 4.6, these new terms are as defined as follows:

- **EVM at steady-state (EVM_{ss}):** It is defined as the EVM value of $\hat{d}[n]$ after the adaptive FIR filter is fully converged.
- **Relative EVM overshoot due to de-convergence at steady state ($dEVM_{ss}$):** At the steady-state (when the adaptive filter is fully converged), if $r[n]$ is modified, then the filter comes out of convergence till it reaches the highest de-convergence point. The $dEVM_{ss}$ is therefore defined as the difference between EVM_{ss} and the EVM of $\hat{d}[n]$ at the highest de-convergence point.
- **Initial convergence time (t_{ic}):** It is defined as the number of weight-update iterations (or the amount of time) needed by the adaptive FIR filter to converge to EVM_{ss} starting from the initialised weight values.
- **Re-convergence time (t_{rc}):** It is defined as the number of weight-update iterations (or the amount of time) needed by the adaptive FIR filter to re-converge to EVM_{ss} starting from the highest de-convergence point.

4.2.2 Analysis of Filter Convergence Performance

The figures 4.7, 4.8 and 4.9 analyse the performance of the different adaptive FIR filters (with filter length $L = 3$) based on the aforementioned terms such as EVM_{ss}, $dEVM_{ss}$, t_{ic} and t_{rc}, when subjected to a typical SI environment with Rician factor $k = 10$, AWGN SNR = 20 dB and initial SI-gain = −5 dB (which is later increased to 0 dB).

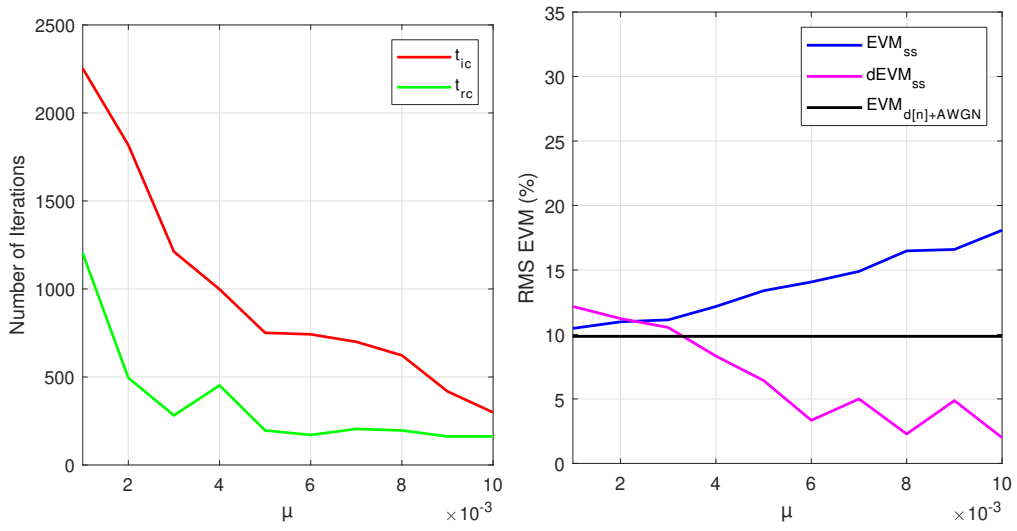


(a) Plot 1 **(b)** Plot 2

Figure 4.7: The variation of t_{ic}, t_{rc}, EVM_{ss} and $dEVM_{ss}$ of $\hat{d}[n]$ with μ for the LMS filter.

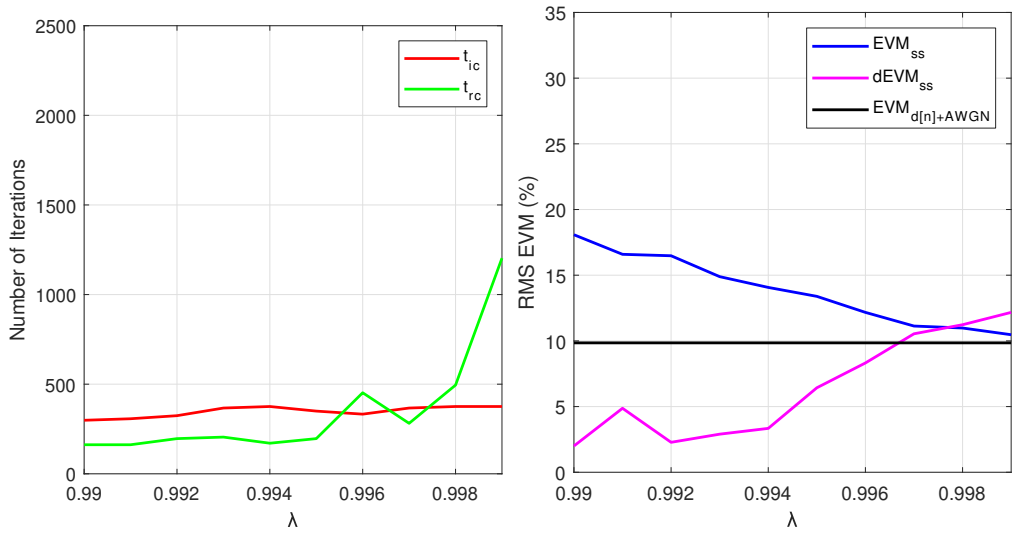
By analysing the figures 4.7, 4.8 and 4.9, it can be understood that:

- The LMS and the NLMS adaptive FIR filters show identical performance, as per the figures 4.7 and 4.8. This is expected because the NLMS filter by definition provides stability against the possible fluctuations in $x[n]$ (refer section 3.2). However, in the current SI en-



(a) Plot 1 **(b)** Plot 2

Figure 4.8: The variation of t_{ic}, t_{rc}, EVM_{ss} and $dEVM_{ss}$ of $\hat{d}[n]$ with μ for the NLMS filter.



(a) Plot 1 **(b)** Plot 2

Figure 4.9: The variation of t_{ic}, t_{rc}, EVM_{ss} and $dEVM_{ss}$ of $\hat{d}[n]$ with λ for the RLS filter.

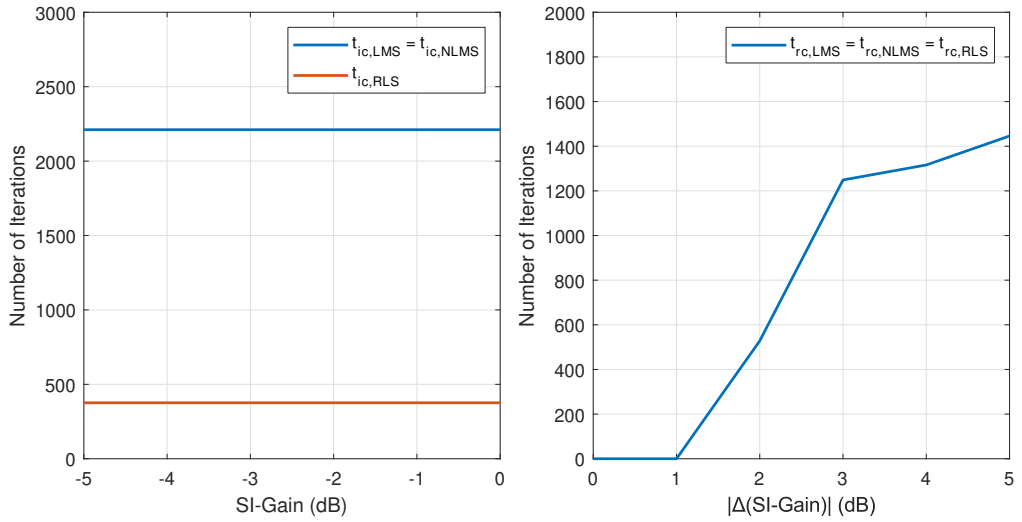
vironment, there are no fluctuations in $x[n]$, since it consists of pure QPSK symbols to be transmitted, that are already known to the IBFD RX system (refer chapter 3). As a result, the performance of both the filters remain the same.

- Moreover, for all the filters, aiming for a smaller (or better) value of EVM_{ss} comes at the cost of a higher t_{ic} and t_{rc} value (refer figures 4.7 to 4.9). However, if a smaller t_{ic} and t_{rc} value is needed, one has to compromise with a higher (or poorer) value of EVM_{ss}, but for an added advantage of a smaller (or better) $dEVM_{ss}$. Finally, the best option to go for in this case depends on the application requirements.
- Furthermore, the DSIC operation based on the adaptive FIR filters can never provide a $\hat{d}[n]$

with an RMS EVM that is better than the RMS EVM of $d[n]$+AWGN. In other words, the adaptive filters cannot remove the AWGN from $\hat{d}[n]$, as evident from the figures 4.7 to 4.9.

- Finally, it can be seen from the figures 4.7 to 4.9 that the RLS filter is clearly outperforming the others, with t_{ic} and t_{rc} values below 500 iterations for a significant range of λ values, as shown in the figure 4.9a.

Moving on, the figure 4.10 shows the variation of t_{ic} and t_{rc} with SI-gain (dB) and $|\Delta(SI-gain)|$ (dB), respectively for the different adaptive filter types with filter length $L = 3$, step size $\mu = 0.001$ (for LMS and NLMS) and forgetting factor $\lambda = 0.999$ (for RLS), when subjected to a typical SI environment with Rician factor $k = 10$ and AWGN SNR = 20 dB. The term $|\Delta(SI-gain)|$ denotes the magnitude of change in the value of SI-gain (in dB).



(a) Plot 1 **(b)** Plot 2

Figure 4.10: The variation of t_{ic} and t_{rc} with SI-gain (dB) and $|\Delta(SI-gain)|$ (dB) respectively for the different adaptive filter types.

From the figure 4.10, it can be understood that:

- For any of the adaptive filters, t_{ic} remains the same irrespective of the initial value of the SI-gain, as shown in the figure 4.10a. In other words, the t_{ic} value can be pre-determined, since it is independent of the SI environment. This statement is only valid if the SI-gain value is constant for the entire duration of t_{ic}. However, in practical-scenarios, a constant SI-gain value is hardly possible because the propagation medium (environment) is dynamic in nature.
- In contrast to the above observation, t_{rc} is directly proportional to the absolute value of the change in the SI-gain. In other words, higher the value of relative change in the SI-gain, higher is the t_{rc}, for all the adaptive filters as observed in the figure 4.10b. As a result, the t_{rc} value cannot be pre-determined.

To sum-up, as far as the simulation-based comparative performance analysis is concerned, the RLS filter performs better than the other adaptive FIR filters. However, it is quite uncommon to expect

similar performance results in the hardware implementations of these filters as well.

5 Hardware Implementation and Analysis of Digital Cancellation Techniques

This chapter deals with the hardware implementation, verification and analysis of the adaptive FIR filters discussed in the chapter 2.

In order to implement the adaptive FIR filters on the hardware, the Thales Alenia Space (TAS) multiMIND (refer figure 5.1) multi-mission payload processing system [24] is employed. It houses a field-programmable gate array (FPGA) device belonging to the Xilinx Zynq Ultrascale+ multi-processor system-on-chip (MPSoC) family. This FPGA board is specially designed as a high performance payload processing core for deploying in applications including nano and micro satellites. Moreover, the processing system (PS) of this FPGA comprises of the Quad-core ARM Cortex-A53 processor up to 1.5 GHz along with Dual-core ARM Cortex-R5 processor up to 600 MHz. The table 5.1 gives the information about the amount of the programmable logic (PL) resources available in this FPGA.

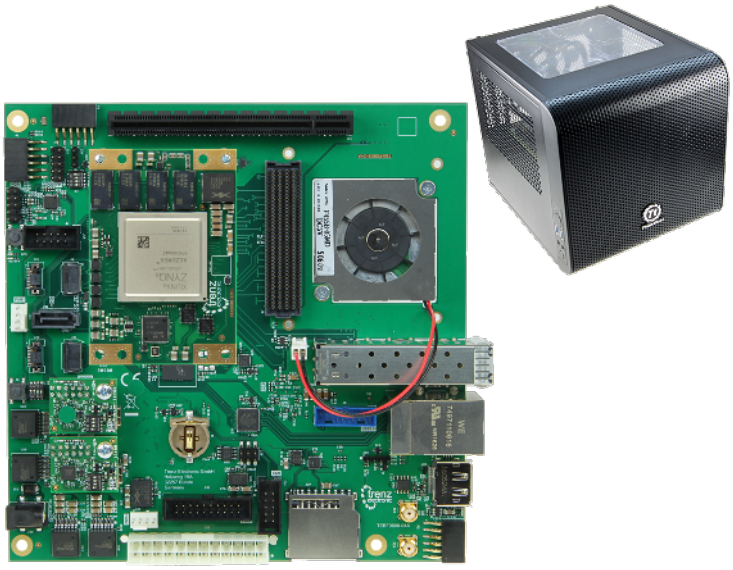


Figure 5.1: The TAS multiMIND multi-mission payload processing system.

Resource	Available
LUT	274080
LUTRAM	144000
FF	548160
BRAM	912
DSP	2520
IO	204
BUFG	404
MMCM	4

Table 5.1: The amount of the PL resources available in the TAS multiMIND FPGA board based on the Xilinx Zynq Ultrascale+ XCZU9EG MPSoC.

5.1 Development of Intellectual Property Cores

Before beginning with the detailed description of the final hardware implementation and the test-setup of the complete DSIC system, it is noteworthy to first discuss the development, implementation and analysis of the intellectual property (IP) cores of the LMS, the NLMS and the RLS adaptive FIR filters. The adaptive FIR filters discussed in the chapter 3 are implemented on the FPGA using the advanced eXtensible Interface (AXI)-4-Stream IP cores. These IP cores are developed using the MATLAB & Simulink hardware description language (HDL) coder software.

5.1.1 General Overview

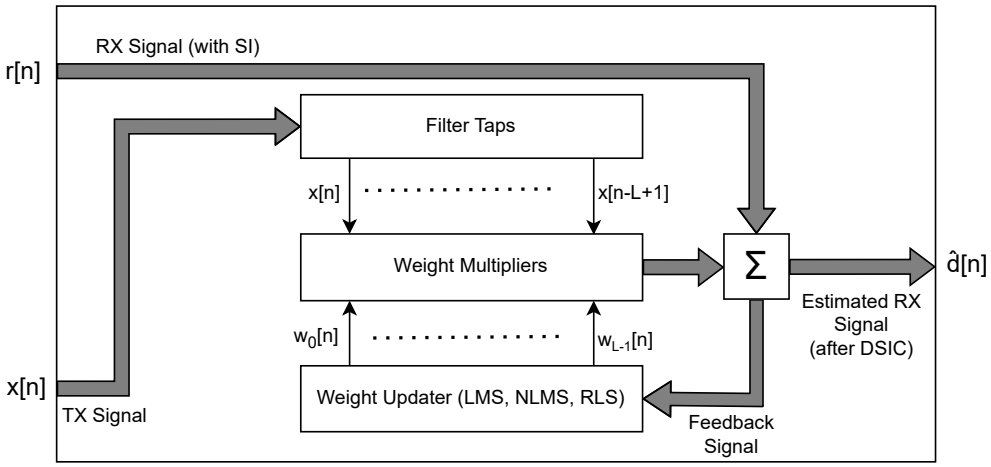


Figure 5.2: The block diagram depicting the basic structure of the adaptive FIR filter IP core.

The figure 5.2 describes the basic structure of the adaptive FIR filter IP core, which remains the same irrespective of the type of the filter (LMS, NLMS or RLS). As shown in this figure, the adaptive FIR filter IP core consists of the following sub-blocks:

- **Filter taps:** This sub-block comprises of $L-1$ delay elements (z^{-1}) through which the TX signal $x[n]$ is passed, as shown in the figures 3.3 and 3.4, where L denotes the filter length.

- **Weight multiplier:** This consists of L multipliers, as shown in the figures 3.3 and 3.4, which multiply each of the filter weights $w_0[n]$, $w_1[n]$, ..., $w_{L-1}[n]$ with the TX signal $x[n]$, $x[n-1]$, ..., $x[n-L+1]$ respectively.
- **Weight updater:** This sub-block is responsible to update the filter weights as per the different weight-adaptation policies discussed in the chapter 3.

Out of all the aforementioned sub-blocks, the first two remain same irrespective of the filter type. However, the last block is different for the various types of adaptive FIR filters (LMS, NLMS or RLS), owing to their respective weight-adaptation policies. Moreover, the adaptive FIR filter IP core mentioned in the figure 5.2 is characterised by the value of step-size μ for the LMS and the NLMS filters and by the value of forgetting factor λ for the RLS filter. Although the figure 5.2 shows a general overview of the adaptive FIR filters for any L, as far as the IP core development is concerned, the value of L is fixed at 3 due to the reasons mentioned in the section 3.4.

5.1.2 Signal Characteristics and Data Format

Before understanding the internal structure and working of the adaptive FIR filter IP cores, it is important to first discuss the characteristics and data format of the input and output signals of the filter IP cores. To begin with, all the signals ($x[n]$, $r[n]$ and $\hat{d}[n]$) associated with the IP core are QPSK modulated, as discussed in the section 4.1. Thus, all these signals have an in-phase (I) and a quadrature (Q) component, represented together as a complex number $I + jQ$. This means that the IP cores basically deal with complex numbers. However, for handling such type of numbers in the FPGA, it is necessary that each of the I-Q components are pre-defined in the HDL using the signed fixed point format, so that the amount of hardware resources utilized in the FPGA can be effectively quantified and limited.

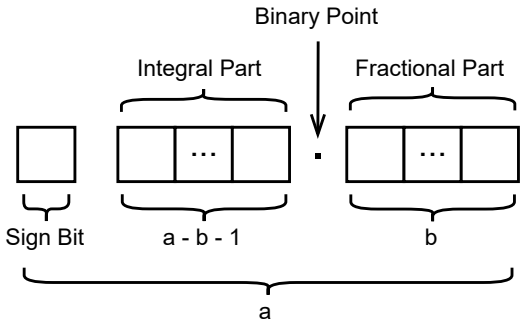


Figure 5.3: The description of the signed fixed point format $sfi(a,b)$.

The figure 5.3 describes the signed fixed point format denoted by $sfi(a,b)$, where a represents the total word length (in bits) and b represents the length of the fractional part (in bits). Since the most significant bit (MSB) is the sign-bit, the length of the integral part is $a-b-1$ bits. If c is a fixed point number represented by $sfi(a,b)$, then the range of c is given by:

$$c \in [-2^{a-b-1},\ 2^{a-b-1} - 2^{-b}) \tag{5.1}$$

From the equation 5.1, it can be deduced that - if the number of bits in the fractional part (b) is very high, the upper and the lower limits of c depend on the number of bits in the integral part ($a-b-1$). Moreover, higher the value of b, higher is the resolution of the fractional part and therefore, finer is the variation in the value of c. Irrespective of the filter type, each of the I-Q components for all the signals ($x[n]$, $r[n]$ and $\hat{d}[n]$) are represented in the IP cores using the $sfi(16,14)$ format respectively, because of the following reasons:

- Firstly, the $sfi(16,14)$ format ensures that both the I and the Q components can respectively vary within a range of [−2, 2) (from the equation 5.1). This permits a large coverage-area of constellation points as shown in the figure 5.4.

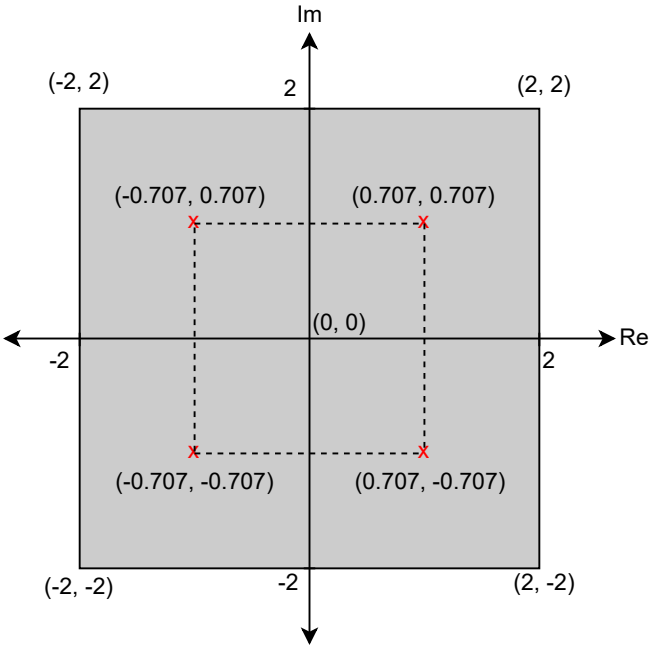


Figure 5.4: The constellation-space (shaded in gray) for signals with the $sfi(16,14)$ format.

- Secondly, the aforementioned format provides a 14-bit resolution to the fractional part of each of the I-Q components, which allows a sufficient level of resolution to the I-Q values.
- The third reason is that the total number of bits used to represent each of the I-Q values cannot be any random integer, rather, it must be a multiple of 8. This permits compatible communication of all the relevant signals of the adaptive FIR filter IP cores with the other IP cores used in the PL.
- Finally, since each of the I-Q components are 16-bit wide, the signals $x[n]$, $r[n]$ and $\hat{d}[n]$ are therefore 32-bit wide respectively. Instead of $sfi(16,14)$, if the $sfi(32,30)$ format is used for representing the I-Q components, then this makes each of the signals $x[n]$, $r[n]$ and $\hat{d}[n]$ to be 64-bit wide. Assigning such a high value of bit-length to the signals can use a significant proportion of the FPGA hardware resources, without offering much benefits. As a result, it is decided to use the $sfi(16,14)$ format for representing the I-Q components.

As far as the values of μ and λ are concerned, they are all represented in the $sfi(32,30)$ format. The valid ranges of the values of μ and λ can be obtained from the equations 3.8 and 3.15 respectively. When the adaptive FIR filter IP core is instantiated in the Vivado software's block diagram, the developer has to specify a valid value of μ or λ in the $sfi(32,30)$ format.

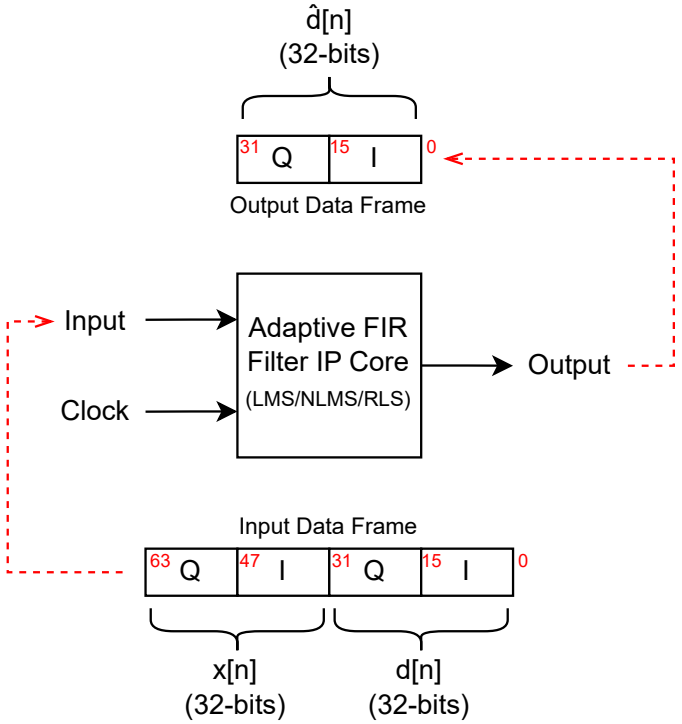


Figure 5.5: The signal data packet format for the adaptive FIR filter IP core.

Moving on, the figure 5.5 provides the data packet format of all the signals associated with each of the adaptive FIR filter IP cores. It can be seen that instead of having two separate 32-bit inputs for signals $x[n]$ and $r[n]$ respectively, a single 64-bit input is used, which samples all the relevant input signals in a combined form. This ensures that there is no relative delay between the input signal sequences $x[n]$ and $r[n]$, because they are sampled simultaneously.

5.1.3 Internal Structure

The figures 5.6 to 5.10 show the internal structure of the IP cores for each of the adaptive FIR filters. The filter theory provided in the chapter 3 form the basis of these structures. Although the functions of most of the components can be determined by their respective operator symbols, the following list gives the information about some of the special components used in the figures 5.6 to 5.9:

- **Delay element** ($\mathbf{z^{-1}}$) : A delay element (z^{-1}), as shown in the figures 5.6 to 5.9a, is represented in the HDL code using a single delay flip-flop. These delay elements are used as filter taps for delaying $x[n]$. Moreover, they are also used as memory elements for storing the current values of the filter weights (for all types of filters) and other intermediate variables such as $P_k[n]$ (for the RLS filter).

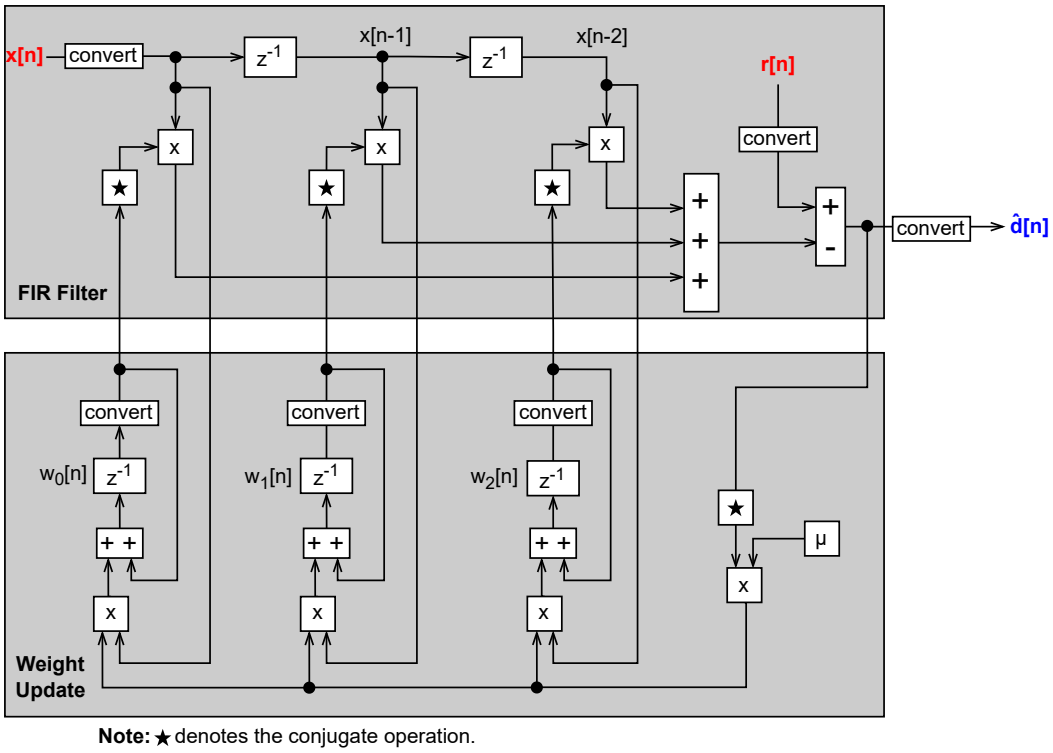


Figure 5.6: The internal structure of the LMS IP core.

- **Convert:** The convert component signifies the sfi data format. This component is used for the signals $x[n]$, $r[n]$ and $\hat{d}[n]$ to specify that they are represented using the $sfi(16,14)$ format. Moreover, the convert component is also used for signifying the $sfi(32,30)$ format for the values of the filter weights and $P_k[n]$ (for the RLS filter).
- **Complex divide:** The MATLAB and Simulink HDL coder does not permit the division of numbers involving complex numerator and/or denominator. However, the NLMS and the RLS filters involve division operations with a complex numerator and a real denominator as shown in the figures 5.7 and 5.9b. In order to carry out these operations, the complex divide component (refer figure 5.10) is developed, which breaks the input complex signal into the real and the imaginary parts and then each of the real and imaginary part is divided by the common real denominator. The end result corresponding to the real and the imaginary part is again recombined back to a complex number, which is made available at the output of the complex divide component.
- **Clock:** All the delay elements (z^{-1}) shown in the figures 5.6 to 5.9a are associated with a common synchronous clock such that the values stored in these delay flip-flops/elements are only updated on the rising edge of the clock signal. In other words, the clock represents the rate at which the filter weights are updated or adapted. Moreover, it also represents the rate at which the input signals ($x[n]$, $r[n]$) and the output signal ($\hat{d}[n]$) are sampled and transmitted-out from the IP cores respectively. Due to the space constraints, the complete routing of the clock signal is not shown in the figures 5.6 to 5.9a.

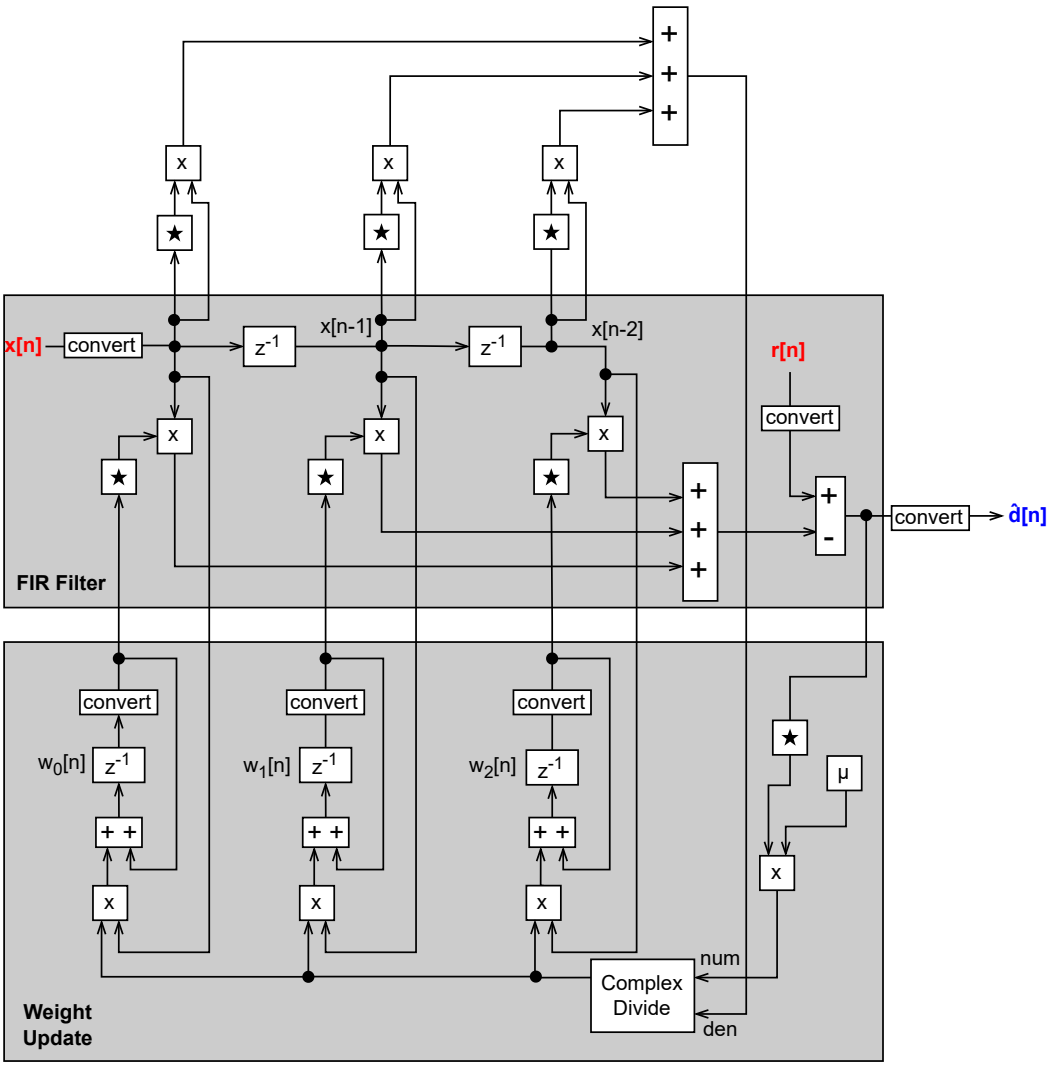


Figure 5.7: The internal structure of the NLMS IP core.

5.2 Pre-Verification Analysis of Intellectual Property Cores

Pre-verification analysis is defined as the analysis of the developed filter IP cores without actually verifying/testing them using the input signals. This type of analysis involves the discussion and comparisons of the computational complexity and FPGA resource utilization for each of the IP cores.

5.2.1 Computational Complexity

This section involves the comparative analysis of the computational complexities of all the implemented adaptive FIR filter IP cores described in the section 5.1.3.

The table 5.2 provides the number of mathematical operational elements utilized in the filter IP

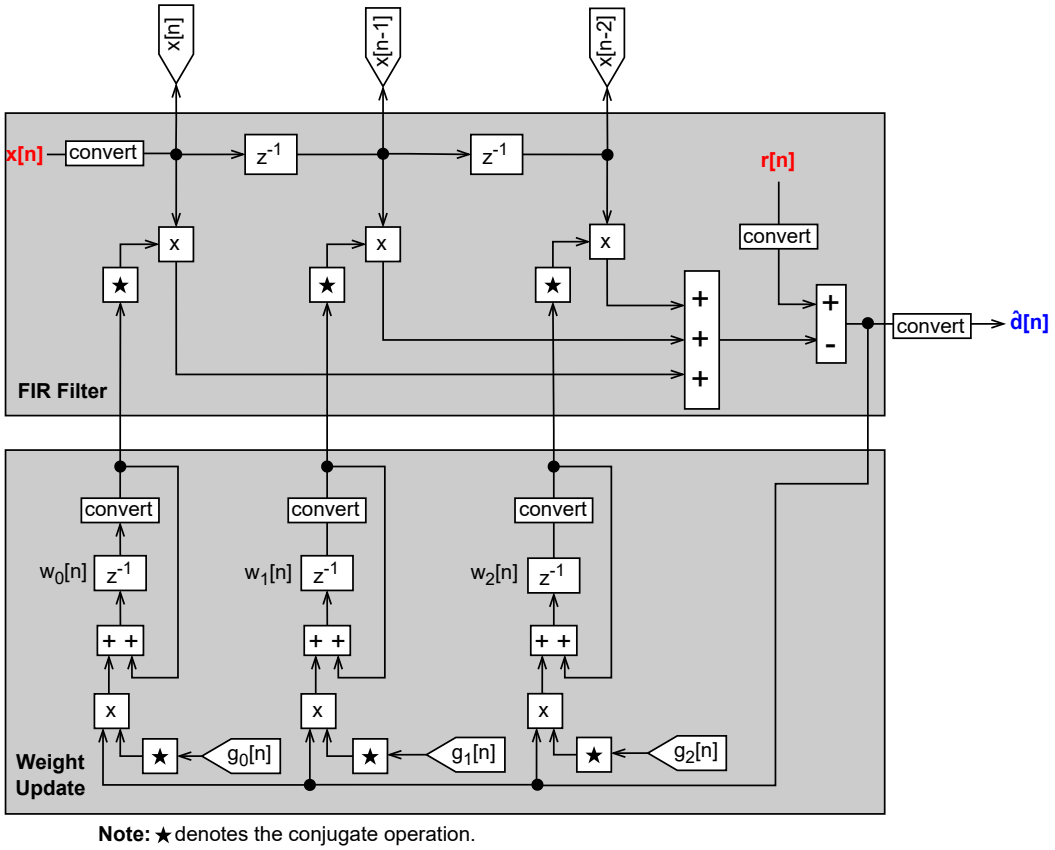


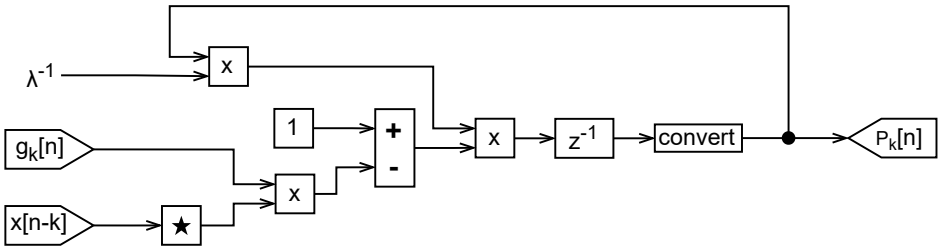
Figure 5.8: The internal structure of the RLS IP core.

cores with length L. While calculating these values, the total number of elements required for the I and the Q components are counted separately.

Operational Elements	Filter Type		
	LMS	NLMS	RLS
Multipliers	$8L+2$	$12L+2$	$28L$
Adders	$8L$	$11L-1$	$18L$
Dividers	0	2	$2L$
Memory Elements	$4L-2$	$4L-2$	$6L-2$

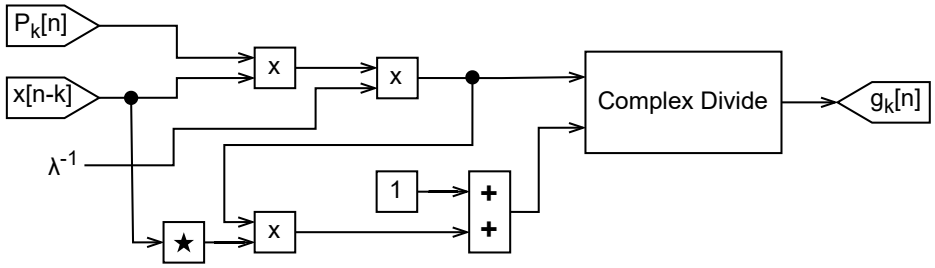
Table 5.2: The number of mathematical operational elements utilized in the different adaptive FIR filters IP cores with the filter length L. In this table, the number of elements required for the I and the Q components are counted separately.

In the final implementation for all the adaptive FIR filters, the value of L is fixed at 3 due to the reasons mentioned in the section 3.4. Therefore, the table 5.3 provides the number of mathematical operations for all the implemented adaptive FIR filter IP cores for $L = 3$. From the tables 5.2 and 5.3, it can be concluded that the RLS adaptive FIR filter has the highest computational complexity out of all the other adaptive filters, since the number of operational elements consumed by the RLS filter is the highest out of the other two adaptive filters.



Note: ★ denotes the conjugate operation.

(a) $P_k[n]$ updater defined in the equation 3.14.



Note: ★ denotes the conjugate operation.

(b) $g_k[n]$ updater defined in the equation 3.10.

Figure 5.9: The internal structures of $P_k[n]$ and $g_k[n]$ blocks of the RLS filter.

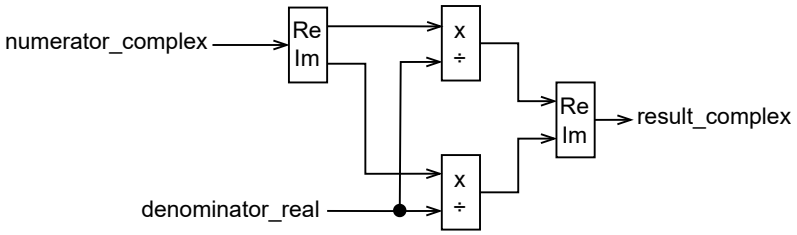


Figure 5.10: The internal structure of the complex-divide block.

Operational Elements	Filter Type		
	LMS	**NLMS**	**RLS**
Multipliers	26	38	84
Adders	24	32	54
Dividers	0	2	6
Memory Elements	10	10	16

Table 5.3: The number of mathematical operational elements utilized in the different adaptive FIR filters IP cores with the filter length $L = 3$. In this table, the number of elements required for the I and the Q components are counted separately.

5.2.2 Results of Post-Implementation Resource Utilization and Timing

As mentioned at the beginning of this chapter, all the adaptive FIR filter IP cores discussed in the section 5.1.3 are synthesized and implemented on the TAS multiMIND FPGA board, which is

based on the Xilinx Zynq Ultrascale+ XCZU9EG MPSoC. The implementation process is performed using the Xilinx Vivado 2017.4 software. The table 5.4 gives the information about the amount of PL resources utilised for implementing the various IP cores on the aforementioned FPGA.

Resource	Available	Utilization		
		LMS	NLMS	RLS
LUT	274080	703	2308	6801
FF	548160	194	226	421
DSP	2520	40	58	234

Table 5.4: The amount of PL resources utilized for implementing the different adaptive FIR filter IP cores in the TAS multiMIND FPGA (Xilinx Zynq Ultrascale+ XCZU9EG).

The table 5.5 gives the information about the maximum possible operational clock frequency value for each of the implemented filter IP cores.

Filter IP Core	Maximum Clock Frequency (MHz)
LMS	60
NLMS	20
RLS	15

Table 5.5: The maximum values of the supported clock frequencies for each of the developed adaptive FIR filter IP cores.

From the tables 5.3, 5.4 and 5.5, it can be concluded that the maximum operational frequency of each of the adaptive FIR filters is inversely proportional to their respective computational complexities and hardware resource utilizations. This statement can be explained using the figure 5.11, which describes the weight-update mechanism of any adaptive FIR filter. As the complexity of the weight-adaptation policy of any filter increases, the time required for estimating the next weight value increases. As a result, the maximum operational frequency decreases.

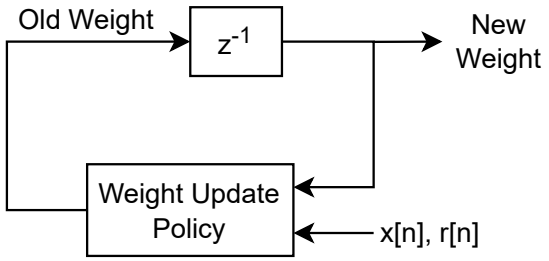


Figure 5.11: The block diagram of the weight-update mechanism.

5.3 Post-Verification Analysis of Intellectual Property Cores

This section deals with the verification of the IP cores described in the section 5.1.3 using the actual TX signal $x[n]$ and the corrupted RX signal $r[n]$. Moreover, the hardware setup and the verification

environment used for testing the IP cores are discussed as well, followed by the comparisons of the verification/test results with the simulation results for each of the filter types.

5.3.1 Test-Setup for Verification

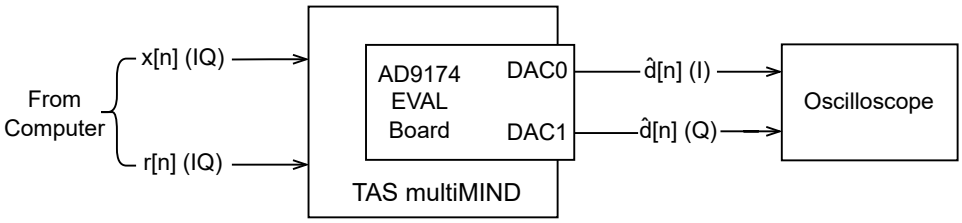


Figure 5.12: The block diagram of the test-setup for the post verification analysis of the filter IP cores.

The figure 5.12 describes the DSIC test-setup, involving the TAS multiMIND board and the Analog Devices AD9174-FMC-EBZ evaluation board. The multiMIND board is introduced at the beginning of this chapter and therefore, it does not need any re-introduction. However, it is important to describe the Analog Devices AD9174-FMC-EBZ evaluation board for understanding all the aspects of the DSIC test-setup in detail.

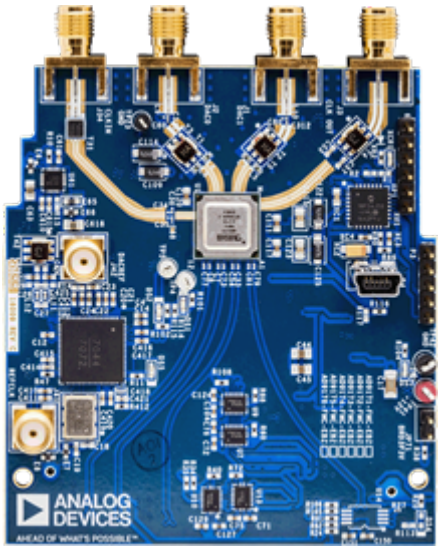


Figure 5.13: The top-view of the AD9174-FMC-EBZ evaluation board.

The Analog Devices AD9174-FMC-EBZ evaluation board (refer figure 5.13) houses a dual, 12-bit RF DAC & direct digital synthesizer [4]. One of the key features of this board is that it can replay any arbitrary waveform at the GHz range through its DAC0 and DAC1 SubMiniature version A (SMA) connectors. This application of the evaluation board is exploited to output the estimated RX signal $\hat{d}[n]$ obtained after the DSIC process. The Analog Devices AD9174-FMC-EBZ evaluation board is connected to the TAS multiMIND system via an FPGA Mezzanine Card (FMC) connector. One of the major shortcomings of this board is that it is not equipped with ADCs. As a result, it cannot

be used to perform the quasi-realtime DSIC operation. Therefore, the input signals for the filter IP core to be verified must be transferred to the underlying FPGA from the computer, as shown in the figure 5.12. The table 5.6 lists all the relevant specifications of the AD9174 DAC.

Parameter	Minimum	Maximum	Unit
DAC Resolution	-	12	bit
DAC0 and DAC1 Full-Scale Output Current (R_{SET} = 5 kΩ)	14.2	28.8	mA
DAC Update Rate	2.91	12.6	GSPS

Table 5.6: The relevant specifications of the AD9174 RF DAC.

5.3.2 Implementation of Verification Environment

The adaptive FIR filter IP cores described in the section 5.1.3 cannot run directly and therefore require a supporting environment for their operation. In order to operate and thereafter, test/verify the filter IP cores on the TAS multiMIND FPGA, a verification environment is designed and implemented using the Vivado 2017.4 software. The figure 5.14 shows the simplified block diagram of this implemented verification environment. In this case, all the input signals associated with the adaptive filter IP cores are sent from the computer to the PS via the universal asynchronous receiver-transmitter (UART) protocol.

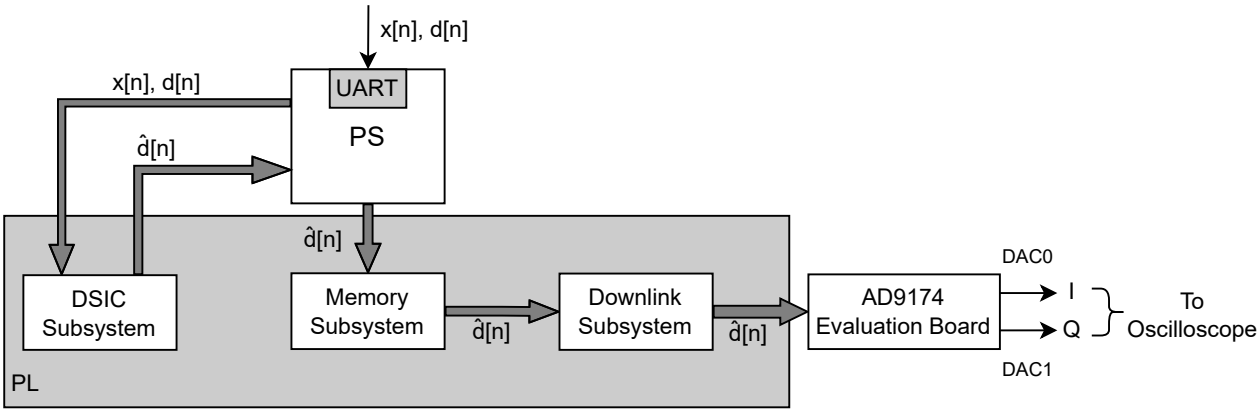


Figure 5.14: The simplified block diagram of the implemented verification environment.

The following list explains the working of the relevant components, modules and IP cores mentioned in the figure 5.14:

- **DSIC subsystem:** This consists of the adaptive FIR filter IP core and a dedicated direct memory access (DMA) buffer. The samples of input signals ($x[n]$, $r[n]$) received via UART for the DSIC operation are first stored in the DMA buffer, before they are transmitted to the filter IP core. After performing the DSIC operation, the filter IP core stores the samples of the estimated RX signal ($\hat{d}[n]$) within the same DMA buffer.

- **Memory subsystem:** For replaying the $\hat{d}[n]$ signal generated by the DSIC subsystem through the AD9174 evaluation board, the samples of the $\hat{d}[n]$ must be stored in a dedicated BRAM provided by the memory subsystem. For this, the $\hat{d}[n]$ samples obtained from the DSIC subsystem are routed via the PS to the BRAM of the memory subsystem. Finally, the memory subsystem transmits the different segments of the samples stored in the BRAM to the downlink subsystem for further processing.
- **Downlink subsystem:** The segments of samples transmitted by the memory subsystem are broken down and aligned into individual frames inside the downlink subsystem. Moreover, the subsystem handles the clocking and synchronization of the sample frames, followed by the initiation of the protocol needed to replay the $\hat{d}[n]$ samples frame-by-frame at GHz frequency through the AD9174 DAC0 and DAC1 SMA connectors.

Out of all the aforementioned subsystems, this thesis deals with only the DSIC subsystem. The downlink and the memory subsystems contribute towards interfacing the AD9174 evaluation board, which is deployed just for replaying $\hat{d}[n]$ obtained after the DSIC process. As a result, the latter two subsystems are explained in an abstract manner as an attempt to avoid deviating from this thesis. Moreover, the block diagram shown in the figure 5.14 depicts only those components that are relevant to the DSIC operation. The discussion of the rest of the hidden components/modules, although necessary for operating the FPGA, is outside the scope of this work.

5.3.3 Results of Post-Implementation Resource Utilization and Timing

Resource	Available	Utilization		
		LMS	NLMS	RLS
LUT	274080	22511	24194	28751
LUTRAM	144000	1349	1350	1349
FF	548160	26501	26533	26744
BRAM	912	79.50	79.50	79.50
DSP	2520	40	58	234
IO	204	16	16	16
BUFG	404	23	23	23
MMCM	4	1	1	1

Table 5.7: The amount of PL resources utilized for implementing the verification environment for the different adaptive FIR filter IP cores in the TAS multiMIND FPGA (Xilinx Zynq Ultrascale+ XCZU9EG).

The table 5.7 provides the amount of the PL resources utilized for implementing the verification environment described in the figure 5.14. Moreover, the table 5.8 gives the information about the clock frequencies at which each of the filter IP cores operate in the verification environment described in the figure 5.14. The following points provides the comparisons between the tables 5.4, 5.5, 5.7 and 5.8:

- **Tables 5.4 and 5.7:** The table 5.4 provides the PL resources required for implementing only the

Filter IP Core	Implementation Clock Frequency (MHz)
LMS	25
NLMS	5
RLS	5

Table 5.8: The clock frequencies for each of the adaptive FIR filter IP cores implemented in the verification environment.

respective adaptive FIR filter IP core, without the verification environment (refer figure 5.14). However, the table 5.7, provides the number of PL resources utilized in implementing the verification environment together with the adaptive FIR filter IP core. As a result, it can be observed that the number of utilized PL resources have significantly increased for each of the adaptive filters respectively, when the simulation environment is taken into consideration.

- **Tables 5.5 and 5.8:** It can be seen that the clock frequencies shown in the table 5.5 are higher than the ones described in the table 5.8, for each of the adaptive filters respectively. This is observed because in the latter case, the simulation environment uses up majority of the PL resources. Therefore, there are proportionately less number of PL resources available for the adaptive filters, which prevents the FPGA to effectively parallelize the filter operation and this causes a reduction in the maximum possible operational clock frequency.

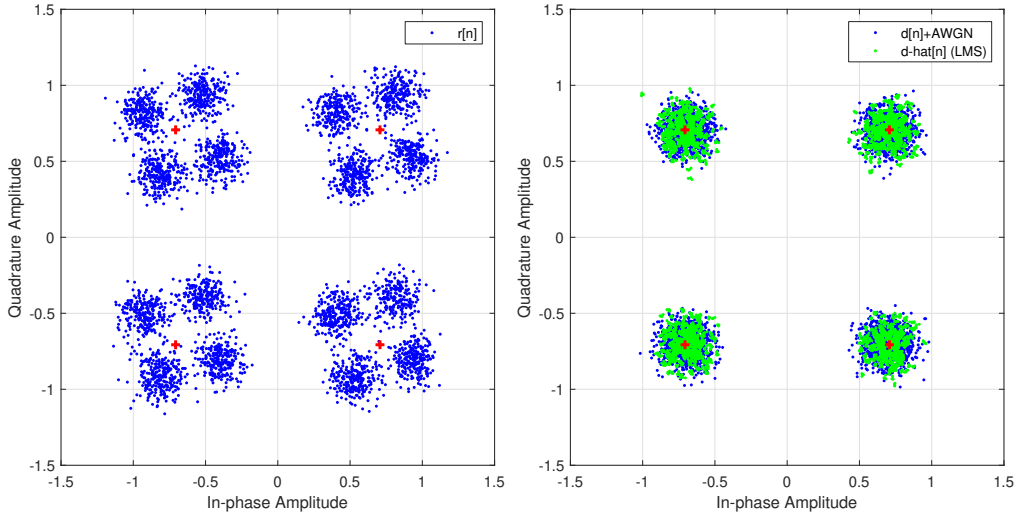
5.3.4 Verification Strategy

This section deals with the strategy used for testing the filter IP cores using the verification environment described in the section 5.3.2. Due to the limitation of the AD9174 evaluation board mentioned in the section 5.3.1, it is decided to evaluate the filter IP cores using the simulated input signals from the section 4.1. The verification of the DSIC operation of the IP core is performed in the following manner:

- In the first step, $x[n]$ and $r[n]$ are sent from the computer to the filter IP core through the UART protocol.
- In the next step, the IP core performs the DSIC operation as soon as it receives all the required input signals ($x[n]$ and $r[n]$) and then generates $\hat{d}[n]$ at the IP core output, whose samples are then stored in a dedicated memory.
- The stored $\hat{d}[n]$ samples are then routed via the PS to the AD9174 evaluation board, as explained in the section 5.3.2.
- Finally, the I and the Q samples of the $\hat{d}[n]$ signal are then replayed at GHz frequency range on the DAC0 and DAC1 SMA connectors respectively of the AD9174 evaluation board.
- The DAC0 and DAC1 SMA connectors are then connected to a digital oscilloscope for observing the QPSK constellations of the $\hat{d}[n]$ signal.

5.3.5 Verification Results

For the verification of the filter IP cores, $x[n]$ and $r[n]$ simulated in a typical SI environment (refer section 4.1), with SI-gain = -5 dB, Rician factor k =10 and AWGN SNR = 20 dB are supplied to each of the adaptive FIR filter IP cores, using the procedure defined in the section 5.3.4. While the figure 5.15a shows the constellation of $r[n]$ supplied to the TAS multiMIND board, the figure 5.15b shows the constellations of $d[n]$+AWGN as well as $\hat{d}[n]$, the latter is generated from the LMS filter IP core. This $\hat{d}[n]$ is sampled from the AD9174 evaluation board output, which is configured to replay the same at 3 GHz. It can be seen that both $d[n]$+AWGN and $\hat{d}[n]$ in the figure 5.15b overlap to a significant proportion, which demonstrates the successful verification of the adaptive LMS FIR filter IP core on the TAS multiMIND FPGA. Furthermore, the RMS EVM values for the $r[n]$, $d[n]$+AWGN and $\hat{d}[n]$ constellations shown in the figure 5.15 are 32.95 %, 11.828 % and 12.751 % respectively. This proves that the adaptive LMS FIR filter IP core is successful to suppress most of the SI in $r[n]$, but is unable to remove the AWGN noise from the same. Finally, observations similar to the figure 5.15b are found in the NLMS and the RLS filter-based FPGA implementations as well.

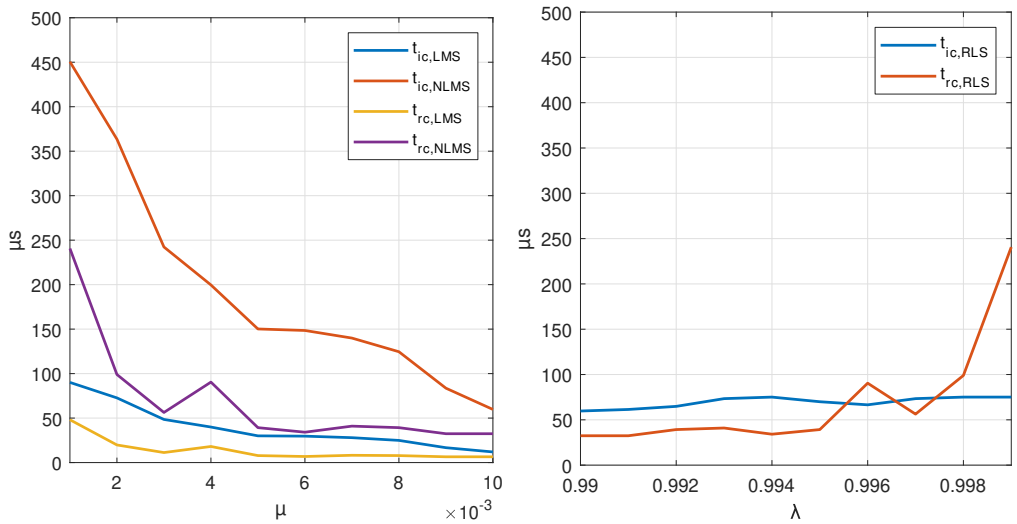


(a) Plot 1 **(b)** Plot 2

Figure 5.15: The constellation diagrams of the signals involved in the verification of the adaptive LMS FIR filter IP core.

5.4 Performance Comparisons of Simulation-based and FPGA-based Filters

In this section, the t_{ic}-t_{rc} performance characteristics of the simulation-based adaptive filters (section 4.2.2) are compared with their respective FPGA-based counterparts. The figure 5.16 compares the variation of t_{ic} and t_{rc} with the value of μ or λ for the different adaptive filters implemented on the TAS multiMIND FPGA.



(a) The variation of t_{ic} and t_{rc} with the value of μ for the LMS and the NLMS filters.

(b) The variation of t_{ic} and t_{rc} with the value of λ for the RLS filter.

Figure 5.16: The variation of t_{ic} and t_{rc} with the value of μ or λ for the different adaptive filters implemented on the TAS multiMIND FPGA.

5.4.1 Comparison of LMS and NLMS Filters

For comparing the t_{ic}-t_{rc} characteristics of the simulation-based LMS and the NLMS filters with their respective FPGA-based counterparts, refer the figures 4.7a, 4.8a and 5.16a. The observations are listed as follows:

- The t_{ic}-t_{rc} characteristics of the simulation-based LMS and the NLMS filters exactly matched each other (refer figures 4.7a and 4.8a) due to the reasons mentioned in the section 4.2.2.
- However, in the FPGA-based implementations of these filters, a contrasting behaviour is observed. The FPGA-based LMS filter outperforms its NLMS counterpart in terms of t_{ic} and t_{rc}, for all the values of μ shown in the figure 5.16a.
- The factors responsible for the aforementioned observation are the computational complexity and the resource utilization. As the computational complexity increases, the filter weights take more time to update.
- As evident from the tables 5.3 and 5.4, the LMS filter is less computational intensive and utilizes less resources as compared to its NLMS counterpart. Therefore, the former performs better than the latter, in terms of t_{ic}-t_{rc} characteristics.

5.4.2 Comparison of LMS and RLS Filters

For comparing the t_{ic}-t_{rc} characteristics of the simulation-based LMS and the RLS filters with their respective FPGA-based counterparts, refer the figures 4.7a, 4.9a, 5.16a and 5.16b. The observations are listed as follows:

- For the simulation-based filters, $t_{ic,LMS}$ is found to be greater than $t_{ic,RLS}$ for a significant range

of μ, except for $\mu > 0.0095$, as evident from the figures 4.7a and 4.9a.

- In contrast to the aforementioned observation, for the FPGA-based filters, it is observed that $t_{ic,LMS}$ is less than $t_{ic,RLS}$ for a significant range of μ, except for $\mu < 0.0025$, as shown in the figures 5.16a and 5.16b.
- Moreover, from the figures 5.16a and 5.16b, it can also be observed that $t_{rc,LMS}$ is less than $t_{rc,RLS}$ for a significant range of μ, except for $\mu < 0.0015$, for FPGA-based LMS and the RLS filters.
- From the aforementioned points, it can be concluded that the FPGA-based LMS filter performs better than the RLS counterpart, for $\mu > 0.0025$.
- As evident from the tables 5.3 and 5.4, the LMS filter is less computational intensive and utilizes less resources as compared to the RLS counterpart, which makes the former faster to operate. Despite the higher convergence characteristics of the RLS filter, it performs slower (worse) than the LMS filter for a significant range of λ, owing to the formers high computational complexity.

The chapter 4 concluded with the fact that the RLS filter performed better than any other filter type, as far as the simulation is concerned. However, when the adaptive filters are implemented in an FPGA, a contrasting behaviour is observed, with the LMS filter outperforming the RLS filter for a significant range of μ.

6 Limitations, Conclusion and Future Work

This chapter involves the description of the limitations of the DSIC system implemented in this thesis. This is followed by the conclusion, along with the recommendations towards future development of the DSIC system.

6.1 Restrictions on Full-Duplex Signals

The DSIC system based on the adaptive FIR filters place certain restrictions on the associated FD signals. For analysing the DSIC system using a typical FD setup, it is generally preferred to loop the TX1 and TX2 signals as shown in the figure 6.1, due to the lack of output ports in the AWG used for supplying the required signals to the FD test-setup. However, the adaptive FIR filters developed in this thesis do not support such type of looping for the TX1 and the TX2 signals.

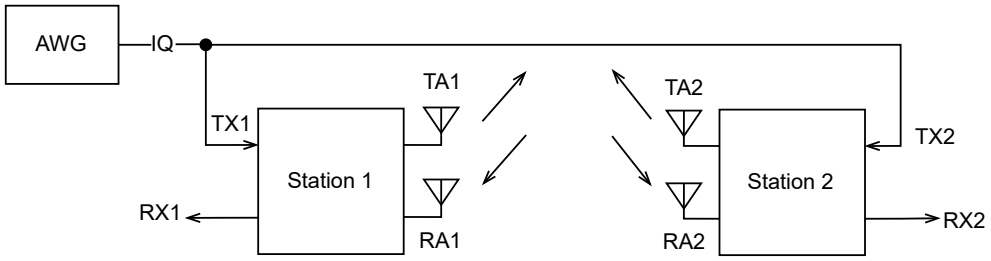
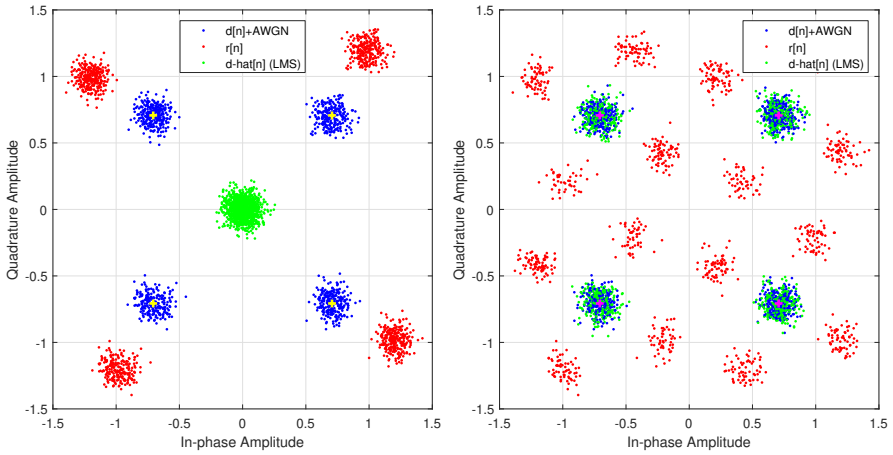


Figure 6.1: The block diagram depicting the case where the two FD stations are connected to the same TX signal.

If the TX1 and the TX2 signals are shorted, then the symbol sequences corresponding to the TX and the RX signals overlap ($x[n] = d[n]$). The figure 6.2 describes the constellation diagrams of the cases when $x[n] = d[n]$ and $x[n] \neq d[n]$ for a typical LMS filter with $L = 3$ and $\mu = 0.001$ in an SI environment with Rician factor $k = 10$, AWGN SNR = 20 dB and SI-gain = 0 dB. The figure 6.2a highlights the fact that looping the TX1 and the TX2 signals ($x[n] = d[n]$) attenuates or amplifies $r[n]$, instead of creating a proper interference constellation pattern in the same, as shown in the figure 6.2b. Moreover, the constellation of $\hat{d}[n]$ does not overlap the standard QPSK symbol locations for the case when $x[n] = d[n]$, as evident from the figure 6.2a. Therefore, the sequences $x[n]$ and $d[n]$ must be different for the proper functioning of the DSIC operation, which can be ensured by generating the TX1 and the TX2 signals using different symbol sequences/patterns.



(a) $TX1 = TX2$ $(x[n] = d[n])$ **(b)** $TX1 \neq TX2$ $(x[n] \neq d[n])$

Figure 6.2: The comparison of constellation diagrams illustrating the impact of connecting the same TX signal across the two FD stations.

6.2 Limitations due to Signed Fixed Point Format

As discussed in the section 5.1.2, all the signals input and output to the adaptive FIR filter IP cores have an $sfi(16,14)$ data format which limits the constellation-space of all the input/output signals to an area highlighted in the figure 5.4. The $sfi(16,14)$ format saturates any constellation point trying to bypass this highlighted region (figure 5.4). This statement can be explained using the figure 6.3, which shows the constellation of $r[n]$ and $\hat{d}[n]$ for a typical SI environment (refer section 4.1) with the Rician factor $k = 10$, AWGN SNR = 20 dB and SI-gain = 10 dB. In this case, $\hat{d}[n]$ is obtained from an LMS filter IP core with $\mu = 0.001$. As evident from the figure 6.3, the SI-gain is so high that some of the constellation points of $r[n]$ get saturated due to the constellation-space limitation placed by the $sfi(16,14)$ format. The saturation caused by the $sfi(16,14)$ format creates a distortion in $r[n]$, which also gets translated to the constellation of $\hat{d}[n]$, as shown in the figure 6.3.

The figure 6.4 shows the variation in the maximum self-interference gain (MSIG) (dB) with the Rician factor k for a typical SI-environment with AWGN SNR = 20 dB. The term MSIG denotes the maximum value of SI-gain, up to which there is no observable distortion in $r[n]$ due to constellation-space limitation imposed by the $sfi(16,14)$ format. From the figure 6.4, it can be understood that:

- The MSIG value is comparatively higher for $k \in [2,4]$. The Rician factor k, as defined in the section 4.1, is inversely proportional to the non-linear SI-power content in any SI environment. This signifies that any of the implemented adaptive FIR filter IP core can provide a better DSIC performance in a non-linear SI environment as compared to the linear counterpart.
- The MSIG-limit can be relatively increased if, for instance, $sfi(32,30)$ format is used for the IP-core input/output signals, instead of the $sfi(16,14)$ format. However, this comes at the cost of an increased FPGA resource utilization.

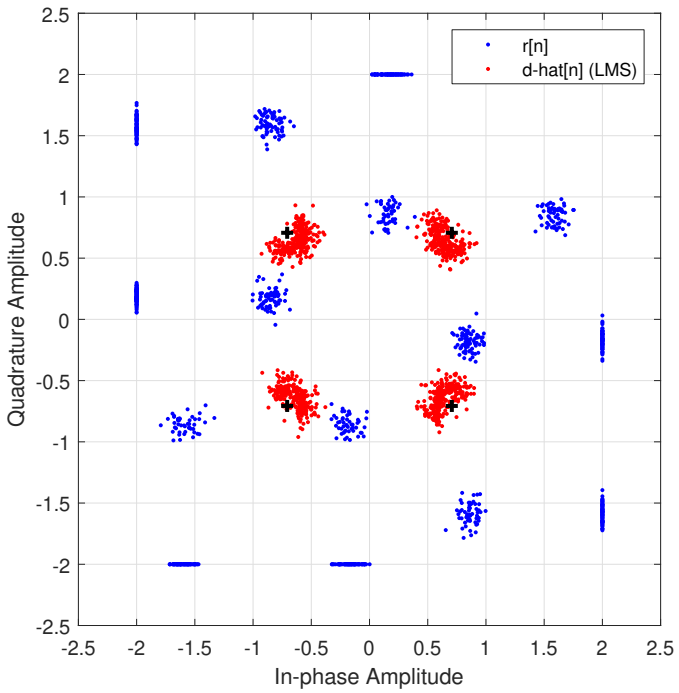


Figure 6.3: The constellations of $r[n]$ and $\hat{d}[n]$ depicting the distortions caused by the constellation-space limitation of the $sfi(16,14)$ format.

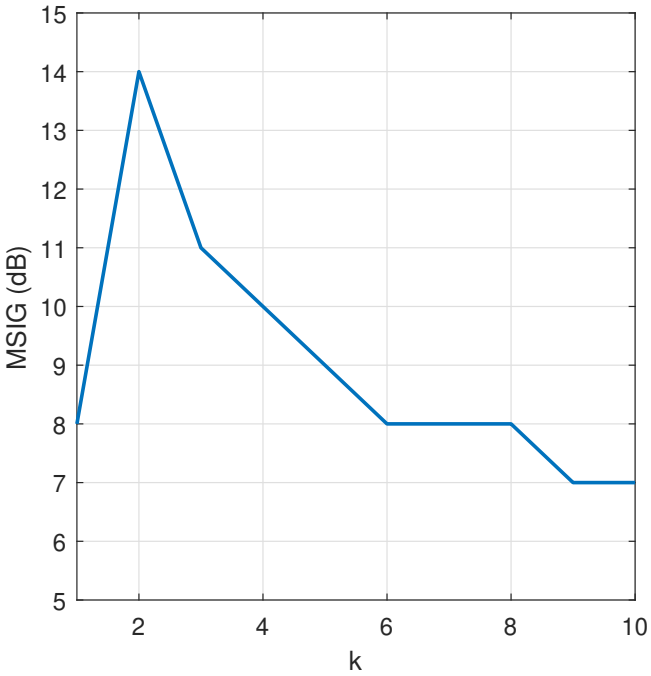


Figure 6.4: The variation in the MSIG (dB) with the Rician factor k.

6.3 Conclusion

6.3.1 Summary

In this thesis, the effects of the SI signal on the RX signal for the IBFD radio are simulated and analysed. This is followed by a successful demonstration of the adaptive FIR filter based DSIC techniques, which are deployed for suppressing the aforementioned SI component in the RX signal.

For analysing the impact of the SI on the RX signal, a simulated SI channel is developed using the Simulink software. The results of this analysis reveal that the RMS EVM value of the RX signal, which is corrupted by the SI signal, is relatively more sensitive towards the gain (dB) of the latter, as compared to the other parameters such as the AWGN SNR (dB) and the Rician factor k of the simulated SI channel.

For performing the DSIC operation on the RX signal which is corrupted by the SI, both the simulation and hardware-based adaptive FIR filters are developed. Each of these filters are based on the LMS, the NLMS and the RLS adaptive algorithms. All the developed filters are able to suppress the distortion caused by the SI signal on the RX signal to a certain extent. The performance of each of these filters are evaluated from their respective convergence characteristics. The results of the convergence characteristics reveal that each of the filters are able to fully converge from the zero-weights state (known as the initial convergence), which signifies their success towards the suppression of the SI in the RX signal. Moreover, each of the filters are also able to re-converge, in case the gain of the SI signal instantaneously vary, which proves their ability to adapt to the dynamic nature of the simulated SI channel. The values of the step size μ (for the LMS and the NLMS filters) and the forgetting factor λ (for the RLS filters) decide the time taken to converge from the zero-weights state (t_{ic}) as well as the time taken to re-converge (t_{rc}) in case of the instantaneous SI-gain fluctuations. In this regard, it is observed for each of the adaptive filters, that a higher t_{ic} and t_{rc} value guarantees a better convergence (i.e. a smaller RMS EVM value of the estimated RX signal). In contrast, a smaller t_{ic} and t_{rc} guarantees a smaller deviation in the RMS EVM value of the estimated RX signal, in case of instantaneous SI-gain fluctuations. Furthermore, it is also observed that none of the adaptive FIR filters are able to eliminate the AWGN from the corrupted RX signal. Finally, it is observed that the simulated RLS filter takes less number of weight-update iterations for both initial-convergence and re-convergence, when compared to the simulated versions of the other filter types.

As far as the hardware implementation is concerned, each of the aforementioned adaptive FIR filters implemented in the TAS multiMIND FPGA are successfully able to reduce the distortion created in the RX signal due to the SI signal, but are unable to eliminate the AWGN noise from the same, as is the case for the simulated filters. However, unlike the latter, in case of the hardware-based filters, it is observed that the LMS adaptive filter performs faster, in general, as compared to the other filter types, owing to the former's low computational complexity.

As far as the limitations are concerned, in case of the hardware-based adaptive FIR filters, the limit imposed by the signed fixed point format on the input and the output signals of the filter IP

core creates a distortion in the estimated RX signal. This causes each of the hardware-implemented filters to provide non-uniform DSIC performance for the different values of the Rician factor k of the simulated SI channel. This results into the implemented filters being more efficient for attenuating the SI with a higher proportion of the non-linear/reflected SI component. Another limitation for the adaptive FIR filters is that, these filters failed to perform the DSIC operation when the TX and the RX signal symbol sequences exactly match each other. Finally, the implemented DSIC system lacks the ability to perform the real-time DSIC operation and it currently requires the input signals to be supplied to the FPGA from the computer via the UART protocol.

6.3.2 Concluding Statement

To conclude, the implemented DSIC system based on the adaptive FIR filters is successfully able to reduce the distortion in the RX signal, which is caused by the SI signal, irrespective of the filter type. As far as the hardware-implementation is concerned, the computational and the hardware complexity of the weight adaption policy of any implemented filter plays a critical role in determining the speed of operation. Moreover, if the filter can converge faster in theory, it does not necessarily mean that it can perform better in the hardware implementation as well. Based on the analysis and the observations performed in this thesis, the LMS filter turns out to be the best from the application perspective, owing to its higher speed of operation and lower resource utilization as compared to the other filter types. Despite the limitations mentioned in the aforementioned sections, the work described in this thesis can be considered as a starting point towards the development of a real-time DSIC system, as described in the upcoming sections.

6.4 Proposed Future Work

This section deals with the future-outlook in terms of the possible development opportunities towards employing a fully-functional FD system.

6.4.1 Quasi-Realtime Full-Duplex Base-station

As described in the chapter 5, the hardware used for the implementation of the adaptive FIR filters consists of the TAS multiMIND FPGA, along with the Analog Devices AD9174-FMC-EBZ evaluation board (refer section 5.3). The main limitation of the latter is that it only consists of the DACs, but not ADCs. As a result, the input signals for the adaptive FIR filter IP cores are provided from the computer via the UART protocol, as described in the section 5.3.4.

One proposed alternative to the AD9174-FMC-EBZ evaluation board is the Analog Devices AD9082-FMCA-EBZ evaluation board, as shown in the figure 6.5. The latter houses an AD9082 chip, which consists of quad mixed-signal front end RF DACs along with dual RF ADCs [5]. The rest of the specifications of this board are mentioned in the table 6.1.

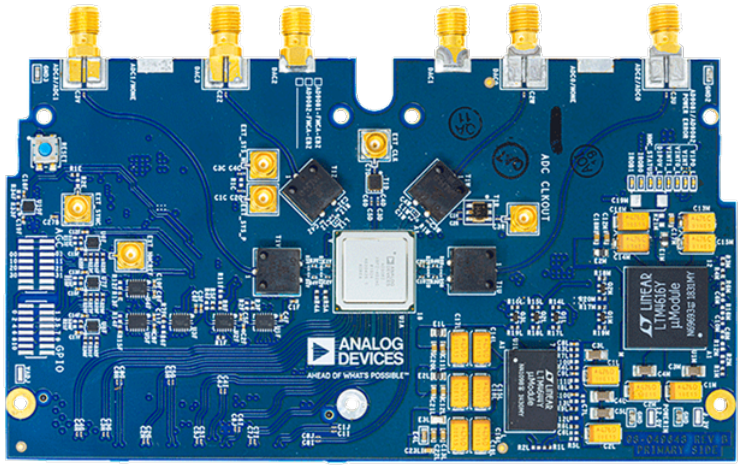


Figure 6.5: The top-view of the Analog Devices AD9082-FMCA-EBZ evaluation board.

Parameter	Minimum	Maximum	Unit
Number of DACs	-	4	-
Number of ADCs	-	2	-
DAC Resolution	-	16	bit
ADC Resolution	-	12	bit
DAC Sample Rate	2.9	12	GSPS
ADC Sample Rate	1.45	6	GSPS

Table 6.1: The relevant specifications of the AD9082 RF DAC and ADC.

Since this board consists of quad DACs and dual ADCs, it can be used for quasi-realtime DSIC operation in the future work. As shown in the figure 6.6, all the DACs and the ADCs can be employed to set-up a FD base station, which not only samples $r[n]$ (at the GHz frequency range) for performing the DSIC operation, but also generates x[n] at the GHz frequency range.

6.4.2 Full-Duplex Data Packet Structure

This section describes an approach that can be utilized for developing a possible FD data-packet structure taking into account the DSIC mechanism developed in this thesis. The basic principle of the FD data-packet revolves around the fact that whenever a data packet is exchanged amongst two communicating FD base-stations (refer figure 6.7), the training data-samples can be added before the actual data-samples. The role of these training samples is to ensure a successful convergence of the adaptive filter-based DSIC system, before the actual samples within the packet are communicated, without being affected by the self-interference. The length and the content of the training samples within an FD data-packet can be mutually agreed amongst the involved base-stations, based on the following two cases:

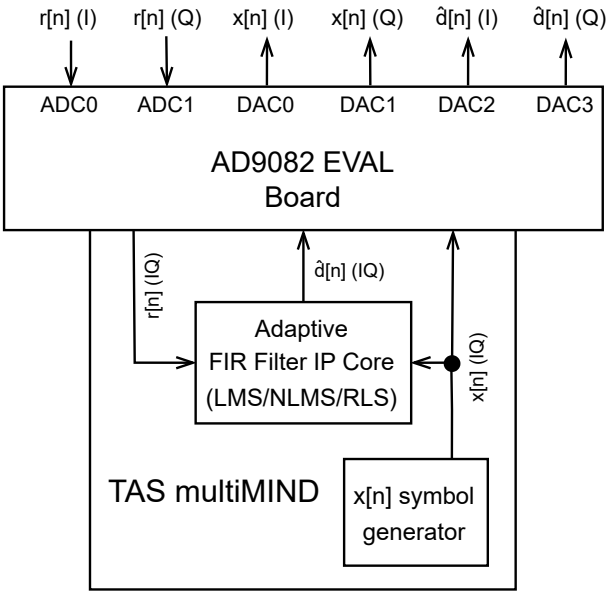


Figure 6.6: The block diagram depicting the future-concept of a quasi-realtime DSIC-equipped FD base-station housing the TAS multiMIND system together with the Analog Devices AD9082-FMCA-EBZ evaluation board.

- **Initial case:** This corresponds to the case when the data-packet is communicated between the base-stations for the first time. This means that the adaptive FIR filter starts its convergence from the zero-weight state. As a result, the number of training samples in the data-packet for the initial case (denoted by N_{ti}) is directly proportional to t_{ic}.
- **Steady state case:** This represents the case wherein a new packet is to be communicated when a data-packet has already been sent before. This means that the adaptive FIR filter has already converged before (filter is already in its steady state) and therefore, the number of training samples (N_{ts}) in this case is directly proportional to t_{rc}.

The observations from the t_{ic}-t_{rc} characteristics described in the section 4.2.2 can be used as a reference towards determining N_{ti} and N_{ts}. However, the detailed research and development in this regard is left for the future-work.

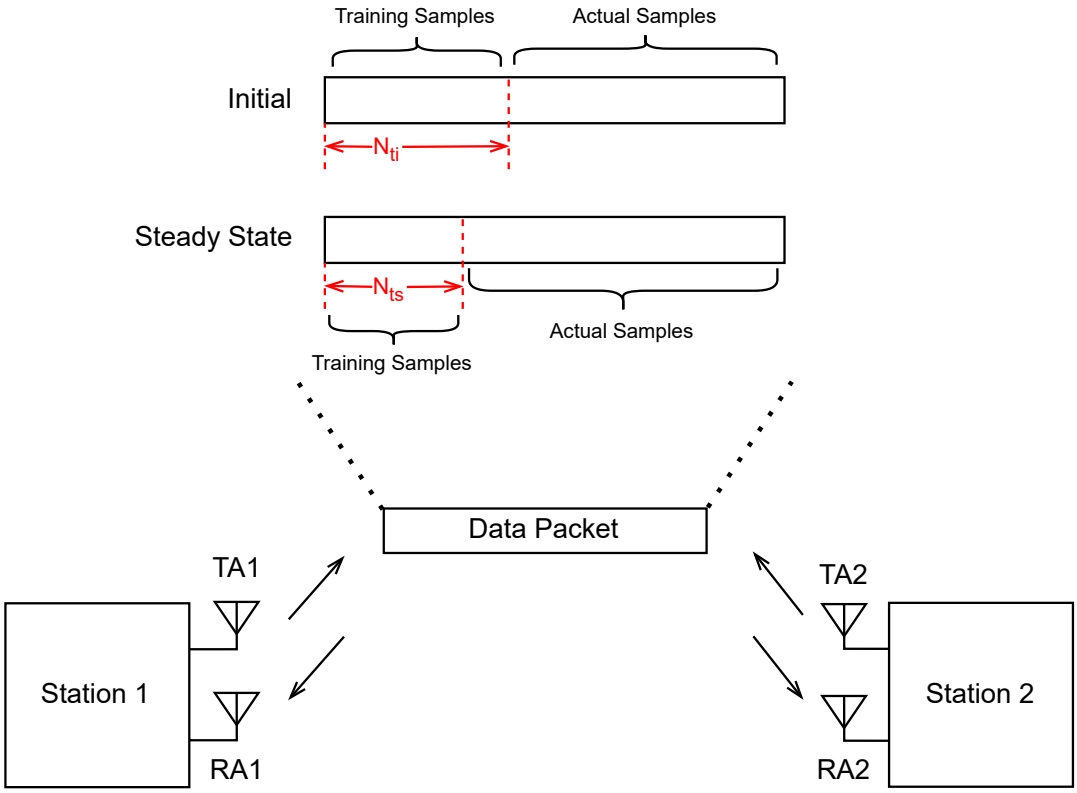


Figure 6.7: The block diagram depicting the future-concept of an FD data-packet structure.

Bibliography

[1] 3GPP: *Full duplex configuration of Un and Uu subframes for type I relay*. 3GPP TSG RAN WG1 R1-100139, Tech. Rep., 2010

[2] 3GPP: *Text proposal on inband full duplex relay for TR 36.814*. 3GPP TSG RAN WG1 R1-101659, Tech. Rep., 2010

[3] Ahmed, Elsayed ; Eltawil, Ahmed M.: All-Digital Self-Interference Cancellation Technique for Full-Duplex Systems. In: *IEEE Transactions on Wireless Communications* 14 (2015), Nr. 7, S. 3519–3532. http://dx.doi.org/10.1109/TWC.2015.2407876. – DOI 10.1109/TWC.2015.2407876

[4] Analog Devices, Inc.: *AD917x-FMC-EBZ Evaluation Board User Guide*. https://wiki.analog.com/resources/eval/dpg/ad917x-fmc-ebz, 2022. – [Online; accessed 02-May-2023]

[5] Analog Devices, Inc.: *AD9081/AD9082 System Development User Guide*. https://www.analog.com/media/en/technical-documentation/user-guides/ad9081-ad9082-ug-1578.pdf, 2023. – [Online; accessed 10-May-2023]

[6] Bharadia, Dinesh ; McMilin, Emily ; Katti, Sachin: Full Duplex Radios. 43 (2013), Nr. 4. http://dx.doi.org/10.1145/2534169.2486033. – DOI 10.1145/2534169.2486033. – ISSN 0146–4833

[7] Choi, Jung I. ; Jain, Mayank ; Srinivasan, Kannan ; Levis, Phil ; Katti, Sachin: Achieving Single Channel, Full Duplex Wireless Communication. In: *Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking*. New York, NY, USA : Association for Computing Machinery, 2010 (MobiCom '10). – ISBN 9781450301817, 1–12

[8] Duarte, Melissa ; Sabharwal, Ashutosh: Full-duplex wireless communications using off-the-shelf radios: Feasibility and first results. In: *2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers*, 2010, S. 1558–1562

[9] Everett, Evan ; Duarte, Melissa ; Dick, Chris ; Sabharwal, Ashutosh: Empowering full-duplex wireless communication by exploiting directional diversity. In: *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, 2011, S. 2002–2006

[10] Haneda, Katsuyuki ; Kahra, Eino ; Wyne, Shurjeel ; Icheln, Clemens ; Vainikainen, Pertti: Measurement of loop-back interference channels for outdoor-to-indoor full-duplex radio relays. In: *Proceedings of the Fourth European Conference on Antennas and Propagation*, 2010, S. 1–5

[11] Hong, Steven ; Brand, Joel ; Choi, Jung I. ; Jain, Mayank ; Mehlman, Jeff ; Katti, Sachin

; Levis, Philip: Applications of self-interference cancellation in 5G and beyond. In: *IEEE Communications Magazine* 52 (2014), Nr. 2, S. 114–121. http://dx.doi.org/10.1109/MCOM.2014.6736751. – DOI 10.1109/MCOM.2014.6736751

[12] Jain, Mayank ; Choi, Jung I. ; Kim, Taemin ; Bharadia, Dinesh ; Seth, Siddharth ; Srinivasan, Kannan ; Levis, Philip ; Katti, Sachin ; Sinha, Prasun: Practical, Real-Time, Full Duplex Wireless. In: *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*. New York, NY, USA : Association for Computing Machinery, 2011 (MobiCom '11). – ISBN 9781450304924, 301–312

[13] Khojastepour, Mohammad A. ; Rangarajan, Sampath: Wideband digital cancellation for full-duplex communications. In: *2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, 2012, S. 1300–1304

[14] Nadh, Arjun ; Samuel, Joseph ; Sharma, Ankit ; Aniruddhan, Sankaran ; Ganti, Radha K.: A Taylor Series Approximation of Self-Interference Channel in Full-Duplex Radios. In: *IEEE Transactions on Wireless Communications* 16 (2017), Nr. 7, S. 4304–4316. http://dx.doi.org/10.1109/TWC.2017.2696938. – DOI 10.1109/TWC.2017.2696938

[15] Nee, Richard v. ; Prasad, Ramjee: *OFDM for Wireless Multimedia Communications*. 1st. USA : Artech House, Inc., 2000. – ISBN 0890065306

[16] PAULRAJ, A.J. ; GORE, D.A. ; NABAR, R.U. ; BOLCSKEI, H.: An overview of MIMO communications - a key to gigabit wireless. In: *Proceedings of the IEEE* 92 (2004), Nr. 2, S. 198–218. http://dx.doi.org/10.1109/JPROC.2003.821915. – DOI 10.1109/JPROC.2003.821915

[17] QHx220: *Active Isolation Enhancer and Interference Canceller*. Datasheet FN6986.0, Renesas Electronics, 2009

[18] Radunovic, Bozidar ; Gunawardena, Dinan ; Key, Peter ; Proutiere, Alexandre ; Singh, Nikhil ; Balan, Horia V. ; DeJean, Gerald: Rethinking Indoor Wireless: Low Power, Low Frequency, Full-duplex. Version: July 2009. https://www.microsoft.com/en-us/research/publication/rethinking-indoor-wireless-low-power-low-frequency-full-duplex/. Microsoft Research, July 2009 (MSR-TR-2009-148). – Forschungsbericht

[19] Rani, Ritu ; Moullec, Yannick L.: Digital Self-Interference Cancellation: Algorithmic Performance Analysis and Software Defined Radio Based Realization. In: *2020 17th Biennial Baltic Electronics Conference (BEC)*, 2020, S. 1–6

[20] Riihonen, Taneli ; Werner, Stefan ; Wichman, Risto: Hybrid Full-Duplex/Half-Duplex Relaying with Transmit Power Adaptation. In: *IEEE Transactions on Wireless Communications* 10 (2011), Nr. 9, S. 3074–3085. http://dx.doi.org/10.1109/TWC.2011.071411.102266. – DOI 10.1109/TWC.2011.071411.102266

[21] Sahai, Achaleshwar ; Patel, Gaurav ; Dick, Chris ; Sabharwal, Ashutosh: On the Impact of Phase Noise on Active Cancelation in Wireless Full-Duplex. In: *IEEE Transactions on Vehicular Technology* 62 (2013), Nr. 9, S. 4494–4510. http://dx.doi.org/10.1109/TVT.2013.2266359. – DOI 10.1109/TVT.2013.2266359

[22] Sharma, Shree K. ; Bogale, Tadilo E. ; Le, Long B. ; Chatzinotas, Symeon ; Wang, Xianbin ; Ottersten, Björn: Dynamic Spectrum Sharing in 5G Wireless Networks With Full-Duplex Technology: Recent Advances and Research Challenges. In: *IEEE Communications Surveys & Tutorials* 20 (2018), Nr. 1, S. 674–707. http://dx.doi.org/10.1109/COMST.2017.2773628. – DOI 10.1109/COMST.2017.2773628

[23] Sung, Yoondong ; Ahn, Junil ; Nguyen, Binh V. ; Kim, Kiseon: Loop-interference suppression strategies using antenna selection in full-duplex MIMO relays. In: *2011 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS)*, 2011, S. 1–4

[24] Thales Alenia Space: *TAS multiMIND Payload Processing System*. https://www.thalesgroup.com//sites/default/files/database/document/2021-09/multiMIND_Flyer.pdf, . – [Online; accessed 18-April-2023]

[25] The MathWorks, Inc.: *RLS filter*. https://de.mathworks.com/help/dsp/ref/rlsfilter.html. – [Online; accessed 07-April-2023]

[26] Wikipedia: *Error vector magnitude — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/w/index.php?title=Error%20vector%20magnitude&oldid=1100404627, 2023. – [Online; accessed 11-April-2023]

[27] Wikipedia: *Least mean squares filter — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/w/index.php?title=Least%20mean%20squares%20filter&oldid=1134169312, 2023. – [Online; accessed 07-April-2023]

[28] Wikipedia: *Recursive least squares filter — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/w/index.php?title=Recursive%20least%20squares%20filter&oldid=1113585275, 2023. – [Online; accessed 07-April-2023]

[29] Wikipedia: *Weighted least squares — Wikipedia, The Free Encyclopedia*. http://en.wikipedia.org/w/index.php?title=Weighted%20least%20squares&oldid=1140153358, 2023. – [Online; accessed 07-April-2023]

[30] Zhang, Zhongshan ; Long, Keping ; Vasilakos, Athanasios V. ; Hanzo, Lajos: Full-Duplex Wireless Communications: Challenges, Solutions, and Future Research Directions. In: *Proceedings of the IEEE* 104 (2016), Nr. 7, S. 1369–1409. http://dx.doi.org/10.1109/JPROC.2015.2497203. – DOI 10.1109/JPROC.2015.2497203

[31] Zhang, Zhongshan ; Wang, Xiyuan ; Long, Keping ; Vasilakos, Athanasios V. ; Hanzo, Lajos: Large-scale MIMO-based wireless backhaul in 5G networks. In: *IEEE Wireless Communications* 22 (2015), Nr. 5, S. 58–66. http://dx.doi.org/10.1109/MWC.2015.7306538. – DOI 10.1109/MWC.2015.7306538

List of Abbreviations

AC	antenna cancellation
ADC	analog-to-digital converter
AGC	automatic gain control
AS	antenna separation
AWG	arbitrary waveform generator
AWGN	additive white Gaussian noise
AXI	advanced eXtensible Interface
BRAM	block random-access memory
BUFG	global buffer
BW	bandwidth
DAC	digital-to-analog converter
DMA	direct memory access
DSIC	digital self-interference cancellation
DSP	digital signal processor
EVM	error vector magnitude
FD	full-duplex
FF	flip-flop
FIR	finite impulse response
FMC	FPGA Mezzanine Card
FPGA	field-programmable gate array
GSPS	gigasamples per second
HD	half-duplex
HDL	hardware description language
I	in-phase
IBFD	in-band full-duplex
IP	intellectual property
IO	input/output
LMS	least mean squares
LTE	long-term evolution
LUT	lookup table
LUTRAM	lookup table random-access memory
MIMO	multiple-input and multiple-output
MMCM	mixed-mode clock manager

MPSoC	multiprocessor system-on-chip
MSB	most significant bit
MSIG	maximum self-interference gain
NLMS	normalised least mean squares
OFDM	orthogonal frequency-division multiplexing
PA	power amplifier
PCB	printed circuit board
PL	programmable logic
PS	processing system
PSK	phase-shift keying
Q	quadrature
QPSK	quadrature phase-shift keying
RA	receiver antenna
RF	radio frequency
RLS	recursive least squares
RMS	root mean square
RX	receiver
SI	self-interference
SINR	signal-to-interference-plus-noise ratio
SMA	SubMiniature version A
SNR	signal-to-noise ratio
TA	transmitter antenna
TAS	Thales Alenia Space
TX	transmitter
UART	universal asynchronous receiver-transmitter