Design of a revised DDS-PLL phase shifter architecture for phased arrays

Department of Electrical and Information Engineering
ELECTRICAL AND INFORMATION ENGINEERING
Ph.D. Program
SSD: ING-INF/01–ELECTRONICS

Final Dissertation

DESIGN OF A REVISED DDS-PLL PHASE SHIFTER ARCHITECTURE FOR PHASED ARRAYS

---

by
D'Amato Giulio:

Supervisor:

Prof. G. Avitabile

Coordinator of Ph.D. Program:
Prof. L. A. Grieco

Course n°30, 01/11/2014-31/10/2017

Politecnico di Bari

Department of Electrical and Information Engineering
ELECTRICAL AND INFORMATION ENGINEERING
Ph.D. Program
SSD: ING-INF/01–ELECTRONICS

Final Dissertation

DESIGN OF A REVISED DDS-PLL PHASE SHIFTER ARCHITECTURE FOR PHASED ARRAYS

---

by

D'Amato Giulio:

_____

Referees:

Prof. Zoran Stamenkovic

Prof. Petar Šolić

Supervisor:

Prof. G. Avitabile

_____

Coordinator of Ph.D Program:

Prof. L. A. Grieco

_____

Course n°30, 01/11/2014-31/10/2017

Summary

Preface

The growing interest towards electronically scanning arrays, both in warfare applications (e.g. synthetic aperture radars) and in consumer electronics (e.g. wireless communications), has resulted in an extensive research towards Beam Steering Units (BSUs) capable of implementing fine-grained phase shifts at each of their antennas. The scope of this phase reconfiguration is to conveniently focus and steer the radiation pattern of the phased array, and thus obtain a behavior that is not achievable through single element antennas. Among the many state of the art solutions that can be used to implement BSUs for phased arrays, those ones that employ LO path phase shifters, such as DDS-PLLs, seem the most promising. DDS-PLLs are circuit solutions capable to synthesize, with an exceptional phase resolution, phase shifted LOs in the GHz range from conveniently delayed low-frequency signals.

The purpose of this thesis is to present the design and the implementation of a revised DDS-PLL phase shifter, as well as its subsequent integration into a beam steering transmitter prototype. The proposed architecture has been designed for an eventual System-on-Chip (SoC) integration, in such a way that it could become a choice in space and weight constrained scenarios.

This thesis is organized as follows.

Chapter 1 is an introduction on the topic of phased arrays, presenting their theory of operation and some relevant state of the art architectures. An overview of traditional and emerging applications of electronically scanning arrays is also reported.

Chapter 2 recalls the working principles of common DDSs and PLLs in order to introduce the standard DDS-PLL architecture. Known variants to the standard architecture are also reviewed.

Chapter 3 deals with the design and the performance evaluation of a revised DDS-PLL architecture based on Synchronous Delay Lines (SDLs) with programmable lengths, along with a discussion on its FPGA-based prototype. Actual measurements performed in time- and in frequency-domain are reviewed.

Chapter 4 presents the design and the performance evaluation of a beam steering transmitter prototype based on DDS-PLLs. Actual measurements performed in time- and in frequency-domain are reviewed.

Appendix is a discussion on the topic of FPGA-based prototyping and IP core based designs, with a focus on SoC designs. Motivations, procedures and tools commonly employed in the academia and in the industry are presented.

1 Phased array antennas

Phased arrays are antennas made up of at least two stationary elements whose radiation pattern can be shaped and steered by assigning a convenient phase and amplitude relation to the currents fed to each of its elements. An electronic control of these parameters allows to implement directional radiation patterns that can be reconfigured without the need for moving parts, even when starting from non-moving individual antennas. Though phased arrays were first conceived in the early 1900s and have been used in warfare applications since World War II, their technology has evolved steadily, and nowadays the interest towards their application in consumer electronics is growing quickly. In radar applications, electronic steering provides dramatic improvements over mechanical scanning, since it is not affected by inertia. In wireless communications, enhanced directivity allows to implement energy efficiency due to an improved antenna gain. These benefits often overcome the cost and complexity of phased arrays, if compared to single element antennas, especially when these ones are unable to meet gain or radiation pattern requirements in critical applications.

1.1 Theory of operation

The main principle behind phased arrays is the constructive and destructive interference of the electromagnetic waves radiated by each of its antennas. It is now shortly recalled the mathematical model that describes how this physical principle affects the radiation pattern of a phased array, in order to understand how phased arrays work and in what manner reconfiguring the phase of the excitation currents fed to each of their elements impacts the

overall radiation pattern. The electric field induced at a given point in space from a group of antennas is:

$$E_{TOT} = \sum_{i=1}^{N} I_i \overline{f}_i(\theta_i, \varphi_i) \frac{e^{-jK_o R_i}}{4\pi R_i}$$

where I_i is the excitation current fed to the i[th] element of the array, $\overline{f}_i$ is its radiation pattern and R_i is its distance from that point. For an array of identical elements working under a far field condition:

$$E_{TOT} \cong f_i(\theta_i, \varphi_i) \frac{e^{-jK_o r}}{4\pi r} \sum_{i=1}^{N} I_i \, e^{-jK_o \bar{a}_r \cdot \bar{r}_i} = f_i(\theta_i, \varphi_i) \frac{e^{-jK_o r}}{4\pi r} F$$

where F is the array factor, r is the scalar distance of the observed point from origin and $\bar{a}_r \cdot \bar{r}_i$ is the scalar product between of the radius vector of the observed point from origin and the radius vector of the i[th] element of the array from the origin. The above equation allows to assert that the radiation pattern of an array made up of identical elements is the product between the radiation pattern of the antennas and the array factor, a concept that is also known as pattern multiplication.

Let's now focus the discussion on the most common phased array design, which is the 1-D linear phased array, namely a configuration of N identical antennas that are equally spaced along an axis by a distance d. It is now derived for this type of phased array the equation that relates the steering angle to the phase shifts assigned to the excitation currents of two adjacent elements. If the excitation current I_i can be written as:

$$I_i = I_0 e^{-ji\Delta\theta}$$

where $\Delta\theta$ is a constant phase shift between two adjacent antennas, the array factor of the phased array is:

$$F = I_0 \sum_{i=1}^{N} e^{-ji(\Delta\theta + K_o d\cos(\Phi_0))}$$

where $id\cos(\Phi_0) = \bar{a}_r \cdot \bar{r}_i$ from trivial geometrical considerations. Thus, the main lobe direction Φ_0, obtained when F reaches its maximum value, can be calculated through the following relation:

$$\Delta\theta = -K_o d\cos(\Phi_0)$$

In order to achieve a main lobe direction $\Phi_0 = 90°$, namely a main lobe directed towards broadside, $\Delta\theta = 0°$. For a main lobe direction $\Phi_0 = 0°$, namely a main lobe directed towards end-fire, $\Delta\theta = -K_o d$. Figure 1.1 provides a graphical representation, in polar coordinates, of the array factor of a 1-D linear phased array made up of 8 identical antennas equally spaced of λ/2 for three $\Delta\theta$ phase shifts, steering the main lobe direction Φ_0 to $90°$, $45°$ and $0°$.

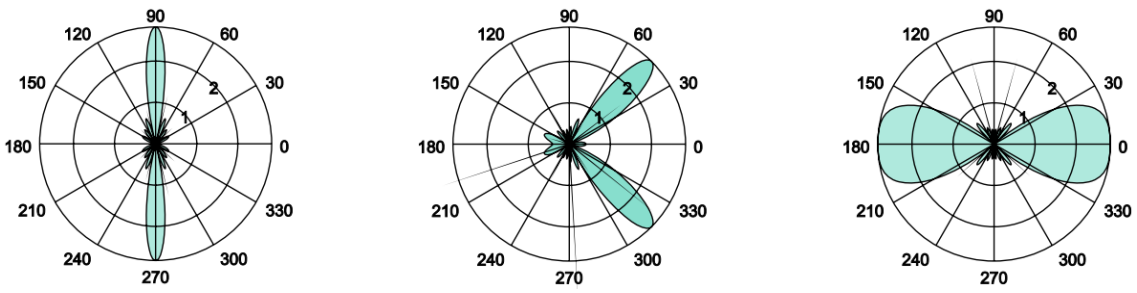


Figure 1.1 – Array factor of a 1-D linear phased array (N=8, d=λ/2). Φ_0=90°, 45° and 0°

The above study allows to understand how, in a 1-D linear phased array with a given N and d, by affecting the phase of the excitation currents, a different shape for the overall radiating pattern can be implemented. Many electronic subsystems that can provide these conveniently phase shifted I_i have been proposed in literature, implementing phased arrays that are referred to as electronically scanning arrays.

1.2 Benefits of phased arrays

The above theoretical background allows to elaborate more on the benefits of incorporating a phased array into a wireless system, such as spatial selectivity and gain boosting.

Spatial selectivity in phased arrays is the capability to implement, under a convenient working condition, radiation patterns that are characterized by a steerable main lobe, even from non-moving individual elements, thanks to the previously defined array factor. This means that, when working at the receiver end, undesired signals (namely interferers) that fall outside the width of the main lobe are significantly attenuated, whereas when working at the transmitter end, a reduced amount of power is transmitted towards undesired directions. For a 1-D linear phased array where $N = 4$ and $d = \lambda/2$, Figure 1.2 shows the array factor for three different values of Φ_0. The full width at half maximum (FWHM) of the main lobe (namely at -3dB in the figure) is a measurement that allows to quantify the beam width synthesized by the array. In this example, for $\Phi_0 = 60°$ and $\Phi_0 = 120°$ the resultant full width at half maximum is $FWHM \cong 30°$, whereas for $\Phi_0 = 90°$ the resultant full width at half maximum is $FWHM \cong 26°$. It seems important to remark that outside the beam width,

the received or the transmitted signal strength is less than a half of that in the steering direction.

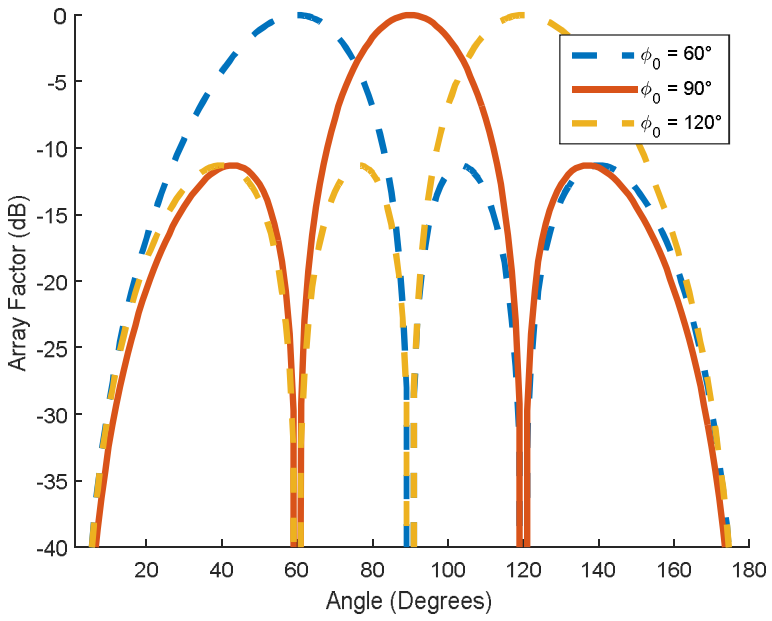


Figure 1.2 - Array factor of a 1-D linear phased array (N=4, d=λ/2). Φ_0=60°, 90° and 120°

The geometry of the array is another key parameter that determines radiation pattern of a phased array. The array factor of a 1-D linear phased array where $N = 4$ and $\Phi_0 = 90°$ is studied in Figure 1.3 when $d = \lambda/2, \lambda/3$ and $\lambda/1.5$. It can be noted that an increased spacing between the elements (with respect to $\lambda/2$) causes the emergence of grating lobes, whereas a decreased spacing causes a wider beam width for the main lobe. This is the reason why the excitation currents fed to each element of the array are often weighted in their amplitude, as well as mutually phase shifted, an operation often referred to as beam forming. Beam forming

allows to further shape the radiation pattern of a phased array, by tuning the position of its zeroes and attenuating its side lobes.

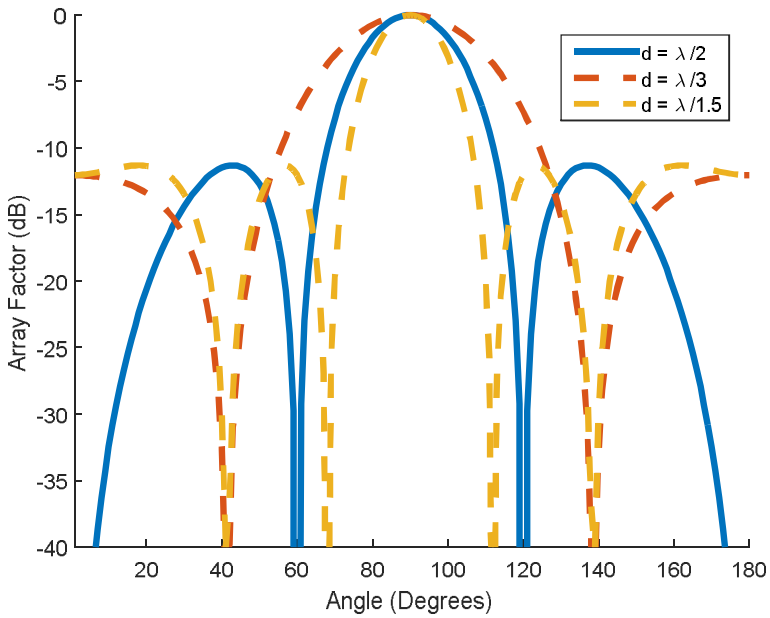


Figure 1.3 – Array factor of a 1-D linear phased array (N=4, Φ_0=90°). d=λ/2, λ/3 and λ/1.5

If suppressing strong interferers in electronic warfare attacks seems one among the most interesting applications of spatial selectivity in mission critical scenarios, even consumer grade wireless communication systems can benefit from its implementation. In fact, spatial selectivity can be used to mitigate the effects of signal propagation in real environments, such as the multi-path interference, in which the transmitted signal reaches the receiver from multiple directions with different phases due to reflections caused by obstacles. Focusing the

receiver towards a preferential direction, namely operating a spatial filtering, reduces this undesired effect and improves the overall quality-of-service.

A direct consequence of spatial selectivity is the so called gain boosting. In fact, since the law of conservation of energy always applies, if a reduced amount of power is transmitted towards undesired directions, an enhanced amount of power is transmitted towards desired directions, in such a way that the total amount of transmitted power is conserved (Figure 1.4).

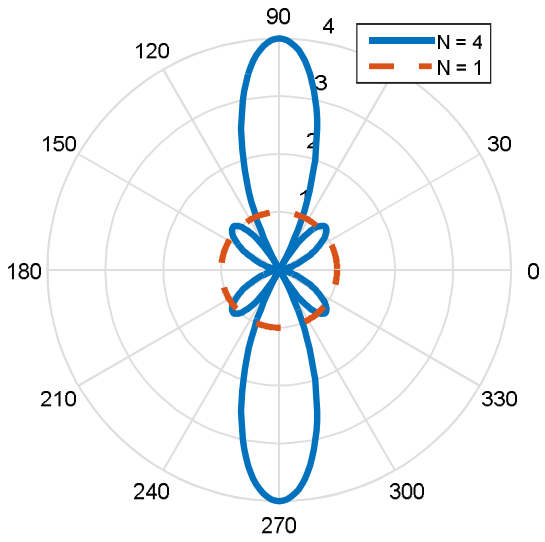


Figure 1.4 - Array factor of a 1-D linear phased array (d=λ/2, Φ_0=90°). N=4 and 1

This is the reason why phased array transmitters can reach receivers that are further away in space, for a given transmitted power and receiver sensitivity. Moreover, considering

unintended receivers placed outside of the main lobe, a dual effect is obtained, since a reduced amount of power originating from the transmission, in this case acting as an interferer, will reach them. The above considerations on gain boosting, discussed when a phased array is employed at the transmitted end, hold true when a phased array is employed at the receiver end. Thus, when the array is focusing towards a given direction, the received signal strength is improved by an enhanced antenna gain through the array factor and vice versa.

1.3 Electronically scanning arrays

An electronically scanning array is a phased array antenna where phase shifts assigned to the excitation currents are implemented through an electronic circuit. In literature, many solutions have been proposed to implement a Beam Steering Unit (BSU) for electronically scanning arrays, though being circuit topologies characterized by severely different complexities and costs. This depends on the physical principle used to implement phase shifts, which in turn depends on the region of the overall system architecture where phase shifts are assigned, a choice that also depends on the bandwidth requirement for the phased array. Phased array architectures can be partitioned into three distinct categories, depending on the circuit path (RF, LO or IF) where phase shifters are located.

RF phase shifting architectures are the ones that have traditionally been used to implement phased arrays. In these architectures, phase shifting is operated in the RF path, namely after up-conversion or before down-conversion (Figure 1.5).

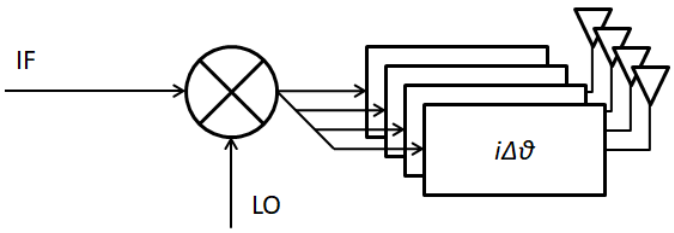


Figure 1.5 – RF phase shifting architecture block diagram

This means that RF phase shifting architectures use only one mixer, and are thus characterized by just one LO distribution point that has to be managed. Moreover, these architecture are the best at strong interferer filtering, since they perform it before the received signal goes into the mixer stage, which is where it may cause a harmful saturation of the input dynamic range, and thus corrupt the behavior of the phased array. However, working in the RF path means that the phase shifting devices operate at high frequencies, where parasitic effects are significant. For this reason, they typically require large die areas and expensive technology platforms to be fabricated. A common technique to implement such RF phase shifters is that of the switched transmission lines. In these topologies, the RF signal is routed through a chain of transmission lines characterized by different lengths, in such a way that phase shifts are achieved through actual propagation delays. In fact, for a quasi-TEM mode propagating in these structures the phase shift introduced is:

$$\Delta\varphi = \frac{2\pi}{\lambda}\Delta L$$

where ΔL is the mismatch between the two transmission lines lenghts and λ is the wavelength of the propagating signal. Figure 1.6 illustrates the conventional switched-line phase shift

element based on PIN diodes reviewed in [1], by Maloratsky (2010) among many other common PIN diodes-based phase shifting solutions.

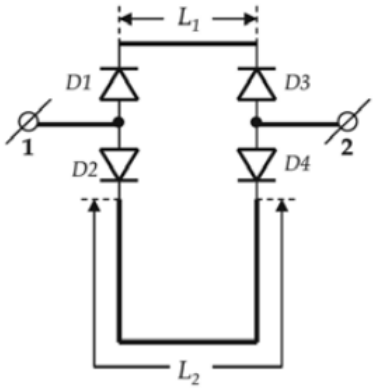


Figure 1.6 – Conventional switched-line phase shift block based on PIN diodes

Switching is achieved by conveniently changing the bias point of the PIN diodes from forward to reverse direction and vice versa. The author asserts that since phase shifts in these devices only depends on the transmission lines length, they are very stable over time and temperature. This is because parameters drift in PIN diodes causes an effect that is more noticeable from an insertion loss point of view, rather than from a change in the phase shift characteristics. Conventional switched-line phase shifters based on PIN diodes are a popular choice for the implementation of 90° and 180° phase shifts at a given frequency, since they only require imposing the length of the transmission lines equal to $\lambda/4$ and $\lambda/2$. However, a phase distortion is introduced at frequencies that differs from the nominal one. Among the drawbacks associated to the above structure, the authors report the undesired resonance

effects that may occur when the length of a transmission line is a multiple of 0.5·λ, a condition which causes a reflection of the signal power to the input port. The electrical length of a transmission line can be changed by varying the reactive load assigned to its ports (e.g. through diodes whose bias voltage can be tuned), thus achieving different propagation delays with a lower number of circuit blocks. In [2], Sharma et al. (2016) proposed a 6-bit phase shifter targeting high-power airborne IFF applications working at 1030- to 1090-MHz (Figure 1.7), whose lower bits implementation is based on the loaded line topology. The proposed 5.625° phase bit topology is that of a tandem stub configuration, where two diodes are switched between the ON and the OFF state to implement (what the authors expect to be) respectively, a resistive load equal to 0.5Ω and a capacitive load equal to 0.65pF.

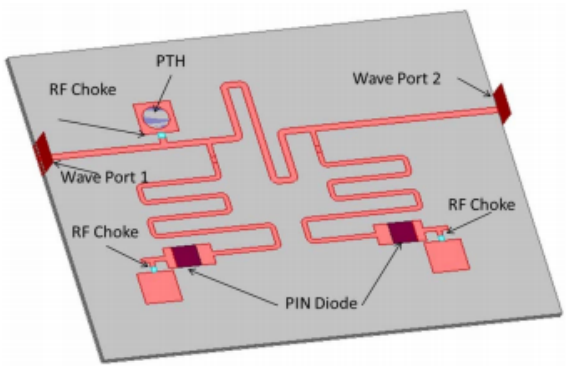


Figure 1.7 – Layout of the loaded line topology for 5.625° phase bit

In [3], Karabey et al. (2011) proposed a continuously tunable loaded line phase shifter for microwave applications based on liquid crystals as a tunable dielectric. The approach

followed by the authors has been to implement a periodic structure composed by chunks of non-tunable transmission lines and shunt loads based on liquid crystals (Figure 1.8).

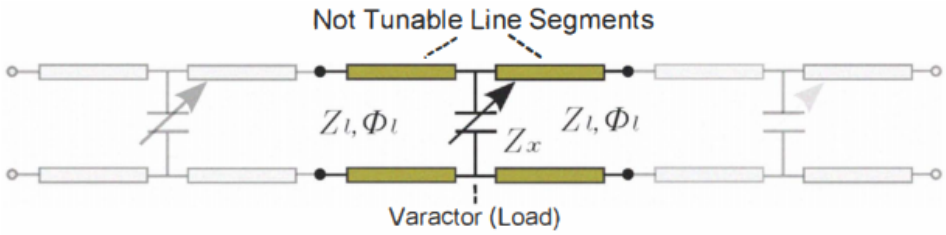


Figure 1.8 – Periodically loaded line with shunt LC based varactors

The structure consists of two stacked substrates encapsulating the liquid crystals in between, and has been reported to work at 12-GHz (with TMM® 10 laminates as substrate) implementing phase shifts up to 145°. It must be noted that this exotic solution arises many non-trivial technological challenges to be implemented, such as the need for a cavity between the stacked substrates. Propagation delays in transmission lines can be synthesized through LC blocks that, in some cases, exploit the parasitic elements of the switching devices themselves. In [4], Miyaguchi et al. (2002) proposed a 5-Bit phase shifter MMIC using series/parallel LC circuits working from 6- to 18-GHz. The reported solution, designed for high performance and small size, allows to select among 32 phase states through SPDT switches (Figure 1.9), with a maximum RMS phase error of 7°. At 10-GHz, the insertion loss has been reported to be 9-dB, and the RMS phase error 2.60°. The total chip size is 4.27-mm x 3.17-mm, including the 5-bit phase shifter section whose footprint is 3.47-mm x 3.17-mm.

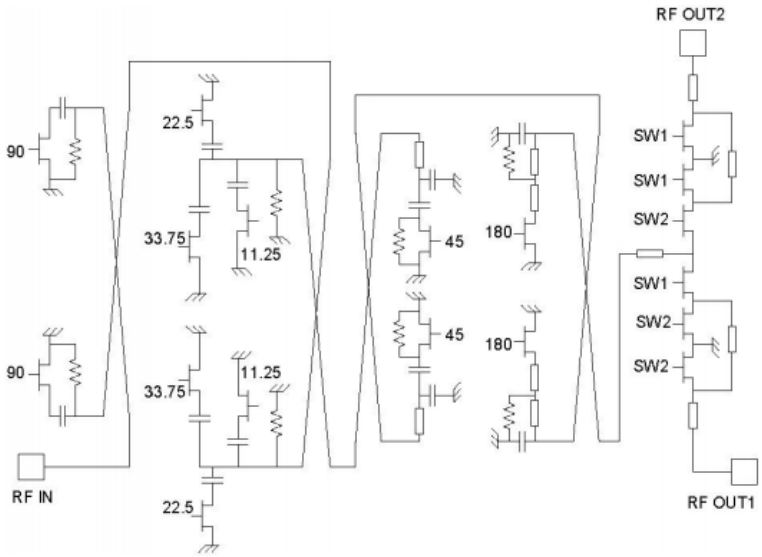


Figure 1.9 – Schematic diagram of the 5-bit phase shifter MMIC with SPDT switch

Modern topologies employing MEMS devices as tunable capacitors or switches have also been reported. In [5], Kim et al. (2001) proposed a monolithic TTD network, based on direct metal-to-metal contact RF MEMS switches, capable to operate from DC to 40-GHz (Figure 1.10). Actual phase shifts have been implemented through switched transmission lines with different lengths. The reported phase shifts cover the [0°; 360°) range in tuning steps of 22.5° at 10.8-GHz (namely, a 4-bit resolution). Die area is 6-mm x 5-mm, substrate is GaAs. It must be noted that this solution requires a control voltage for the MEMS switches as high as 70-V, which may require some effort in common use cases (such as in battery powered devices).

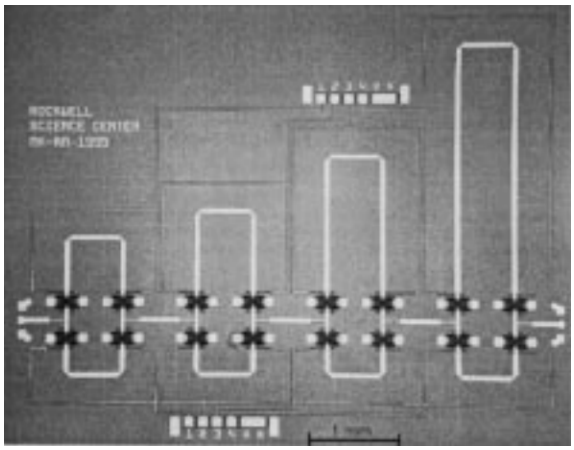


Figure 1.10 - Photograph of the four-bit RF MEMS TTD network

Broadband topologies based on photonic devices represent another state of the art solution for RF phase shifters. In [6], Jiang et al. (2013) proposed a microwave photonic phase shifter, based on an optical phase modulator and a fiber Bragg grating, capable to operate continuous phase shifts from 20- to 30-GHz (Figure 1.11) through an electro-optical effect. The proposed architecture is based on a discrete-fiber grating, thus it is not targeting space and weight constrained scenarios. However, it must be noted that it supports working with bandwidths up to 10-GHz.

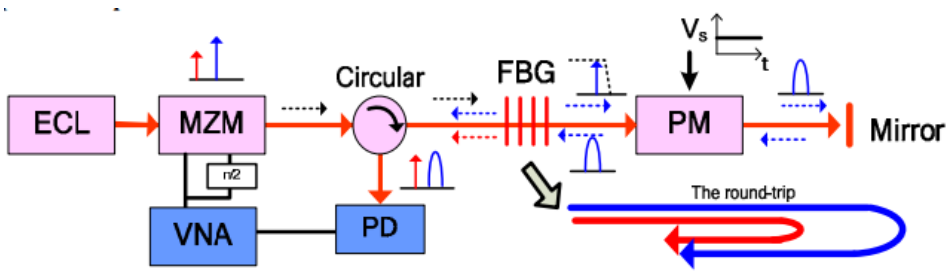


Figure 1.11 - Schematic diagram of the photonic microwave phase shifter

In [7], Burla et al. (2012) proposed a CMOS-compatible optical delay line for Ku-band satellite communications, based on four optical ring resonators, capable of continuous time delays from 0- to 277-ps in the DVB-S communication band (from 10.7- to 12.75-GHz) (Figure 1.12). The proposed solution has been implemented as a monolithic optical circuit through a Si-photonics technology platform, and can thus be a choice in future portable applications. However, the absence of silicon-based light sources requires further technological efforts to implement a single-chip solution, such as the integration of III-V semiconductors through wafer bonding.

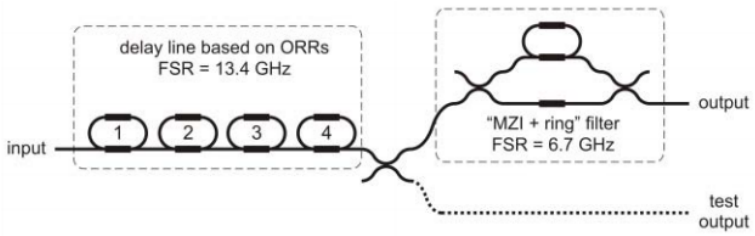


Figure 1.12 – Schematic diagram of the delay unit

LO and IF phase shifting architectures are based on the evidence that the phase of an RF signal can be tuned by changing the phase of one among its own components. In IF phase shifting architectures, phase shifting is operated before up-conversion or after down-conversion (Figure 1.13). This means that phase shifters operating in the IF path operate at a much lower frequency than the ones operating in the RF or in the LO path, and thus their requirements are much more relaxed.

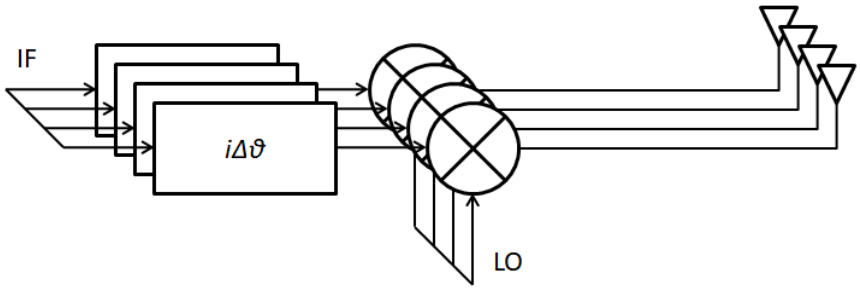


Figure 1.13 – IF phase shifting architecture block diagram

However, IF phase shifting architectures require that each antenna is equipped with a phase shifter and a mixer. Moreover, these architectures are the worst at strong interferer filtering, since they perform it after the received signal goes into the mixer stage. This means that in those applications where strong interferers from undesired directions must be suppressed, each mixer must also be characterized by a conveniently high dynamic range. IF phase shifters can be implemented in the form of switched transmission lines synthesized through LC blocks, such as the ones previously cited. However, in systems oriented towards the digital synthesis of waveforms, IF phase shifters can also be implemented through digital signal processing. Figure 1.14 depicts the generic block diagram of an AESA featuring both analog and digital beamforming functions.

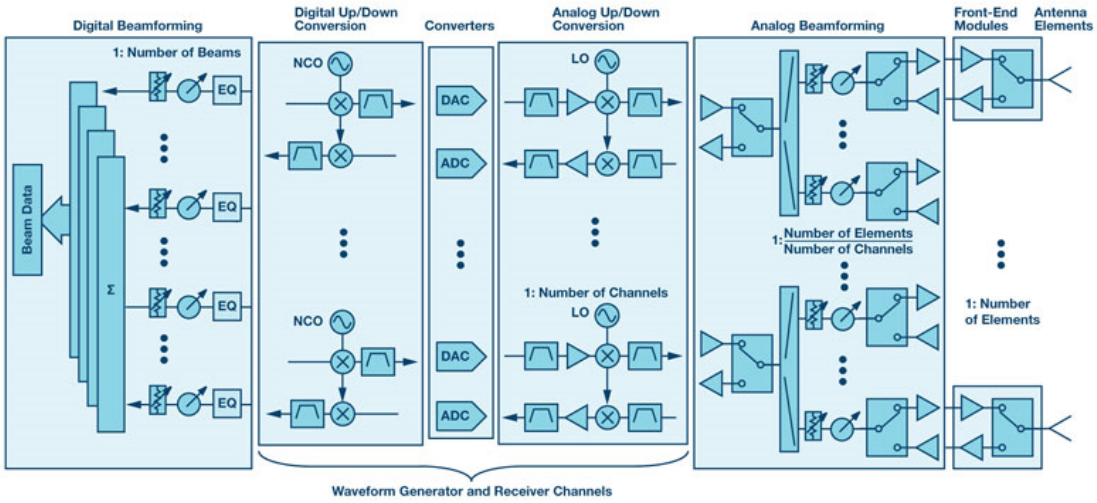


Figure 1.14 – AESA block diagram featuring analog and digital beamforming functions

Nowadays, many systems working in the L- and S-band are developed without an analog beamforming network, pursuing an all-digital beamforming architecture. In such systems, beamforming in the transmit path is accomplished at first by equalizing the channels, and then by imposing phase shifts and amplitude weights to digital streams of samples, before they are converted to analog signals through a DAC. However, this means that each antenna must be equipped with an ADC and a DAC, and that the bandwidth of the involved signals must be small enough so that calculations can be performed in real time by a DSP. A digital phased array can synthesize multiple beams, perform adaptive enhancements to the radiation pattern, and estimate directions of arrival. Unfortunately, these advantages are achieved at an expensive cost, in terms of power consumption and actual price of the components. In [8], Digdarsini et al. (2016) reported the realization of a FPGA-based digital beam forming (DBF) system capable to drive a phased array receiver made up of 16 elements. Figure 1.15

depicts the block diagram of the proposed DBF system. The system prototype implements a linear phased array where each antenna is followed by a down-conversion stage towards IF, where signals are digitized and fed to an FPGA for beam forming. The complete solution has been implemented in subsystems capable to drive 4 antennas, integrating one FPGA and 4 ADCs, for a total number of 4 FPGAs and 16 ADCs.

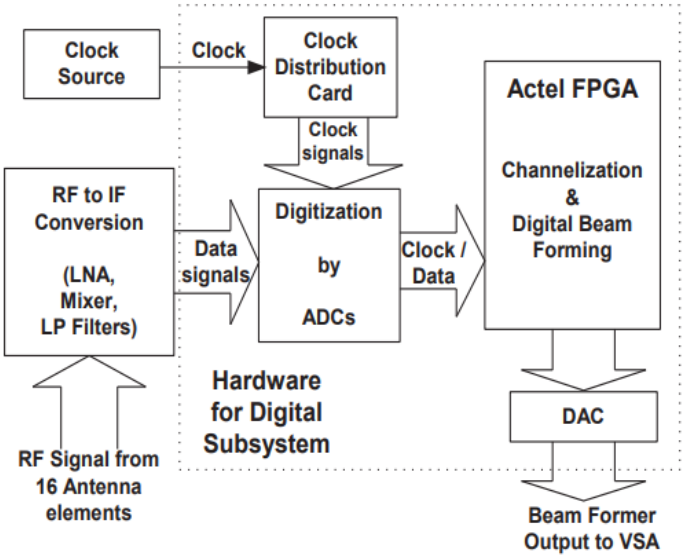


Figure 1.15 – Block diagram of the DBF system

The authors remark how a proper clock distribution network at the ADC sampling stage is critical to avoid introducing phase mismatches to the signals received at each antenna. Moreover, the solution exploits a dynamic calibration of the digitized signals to correct phase and amplitude imbalances, related to unavoidable mismatches in the analog channels. Among the benefits of the proposed solution, the authors underline the opportunity to reprogram

the digital hardware to serve in different scenarios. However, the cost of the proposed solution, in terms of power consumption, form factor and actual price of the components cannot be neglected.

In LO phase shifting architectures, the LO is the only component that is phase shifted in order to perform beam steering (Figure 1.16).

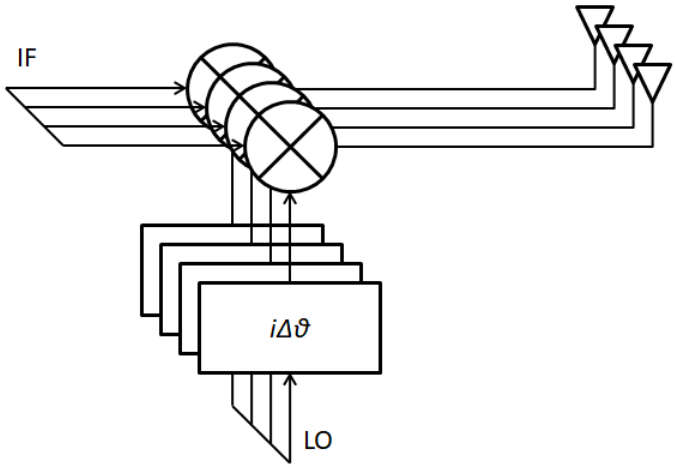


Figure 1.16 – LO phase shifting architecture block diagram

One of the main advantages of these architectures working in the LO path is that they do not interfere with the circuit topology of the signal path. This is the reason why typical performance degradations (e.g. losses, non-linearity and noise) due to the insertion of phase shifters in the signal path can be neglected in this case. Moreover, certain requirements for the phase shifting devices are more relaxed (e.g. in terms of bandwidth) if compared to the ones that operate in the RF path, since the latter must behave consistently across the whole

signal bandwidth. However, LO phase shifting architectures suffer from the same drawback of the IF phase shifting ones, namely they require that each antenna is equipped with a phase shifter and a mixer. It must be noted that LO phase shifting architectures implement an approximation of the time delays that are actually required to drive the phased array. In fact, rather than actual time delays they introduce constant phase offsets. The difference between the two approaches is immediately understood in the frequency domain, where the time delay transfer function is:

$$H_{delay}(f) = e^{-j2\pi fT}$$

whereas the phase offset transfer function is:

$$H_{offset}(f) = e^{-j\Delta\theta}$$

Time delays are linear (frequency dependent) phase shifts, whereas phase offsets (such as the ones introduced in RF signals through LO phase shifting) are flat (frequency independent) phase shifts. The above statements can be further contextualized to LO phase shifting architectures by considering two cosines, such as an LO and a single tone. When those two cosines are multiplied and then time delayed:

$$2\cos(\omega_1 t)\cos(\omega_2 t) = \cos((\omega_1 + \omega_2)t) + \cos((\omega_1 - \omega_2)t)$$

$$\rightarrow \cos\big((\omega_1 \pm \omega_2)(t + \Delta t)\big) = \cos\big((\omega_1 \pm \omega_2)t + \varphi\big)$$

where $\varphi = (\omega_1 \pm \omega_2)\Delta t$, and thus a frequency dependent parameter. Instead, when the LO is phase shifted and then multiplied by the single tone:

$$2\cos(\omega_1 t + \varphi)\cos(\omega_2 t) = \cos\big((\omega_1 + \omega_2)t + \varphi\big) + \cos\big((\omega_1 - \omega_2)t + \varphi\big)$$

where φ is the phase shift assigned to the LO, and thus a frequency independent parameter. The effect of this approximation is a linear phase distortion (namely a dispersion effect) that degrades the performance of the RF channel in terms of SNR. The above discussion allows to assert that, due to the phase distortion they introduce, LO phase shifting architectures are most suited in narrowband systems, namely in those scenarios where the bandwidth of the payload is less than the 10% of the LO frequency. Even LO phase shifters can be implemented in the form of transmission lines. In [9], Lu et al. (2013) proposed an LO-phase shifting receiver front-end, where a tunable transmission line loaded with switched capacitors is used to implement fine-grained phase shifts in the first down-conversion stage (Figure 1.17). The reported phase shifts cover the [0°; 360°) range when combined to a coarse phase shifter operating in the second down-conversion stage, achieving a phase shift resolution of 5.4° at 44- to 54-GHz (namely a 6-bit resolution).

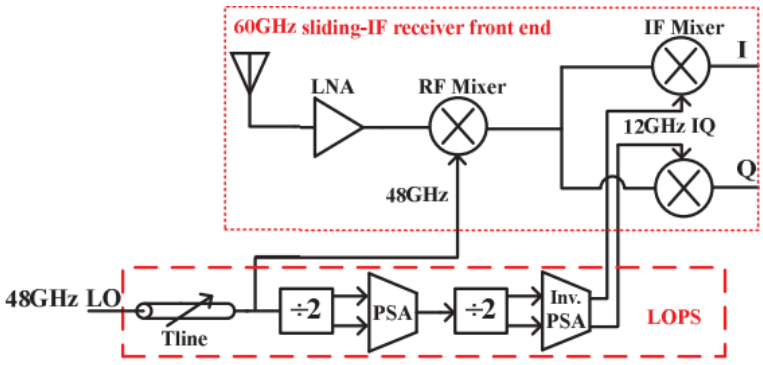


Figure 1.17 – Block diagram of the 60 GHz LO-phase shifting receiver front-end

The proposed solution has been fabricated in a 40-nm CMOS process, and occupies a die area of 550-μm x 260-μm. The authors reported a power consumption of 16.9-mW from a power supply of 1.2-V. It seems interesting to denote that, at each frequency, the achieved distribution of phase shifts is uneven, and that across frequencies, phase resolution is not preserved, having a worst case and average phase shift step respectively of 5.4° and 3.5°. Integrated topologies that exploits ring VCOs to generate more than one output phase at once have also been reported. In [10], Hashemi et al. (2005) proposed a fully integrated 24-GHz LO-phase shifting receiver for phased arrays, based on a 19.2-GHz CMOS ring VCO (Figure 1.18). The reported phase shifts cover the [0°; 360°) range with a 4- to 5-bit resolution, obtained through a 16 phases VCO and an optional first order interpolator.

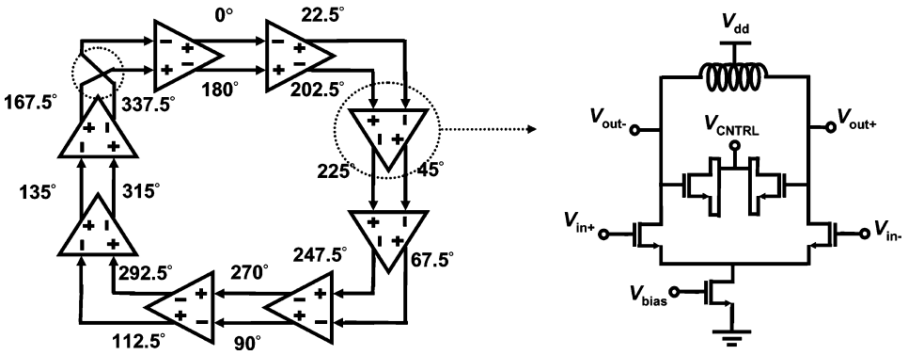


Figure 1.18 – Schematic diagram of the 16-phases 19.2-GHz CMOS ring VCO

Figure 1.19 depicts the block diagram of the fully integrated 24-GHz phased-array receiver. The receiver drives a phased array of 8 antennas, and has been implemented with a two-step down-conversion architecture. The IC integrates a third order PLL, used to synthesize the

LOs for both down-conversion stages, namely 19.2-GHz (for the RF to IF stage) and 4.8-GHz (for the IF to baseband stage) that is obtained through a divide-by-four block from the first one. In this solution, phases can be routed to each antenna independently through an analog multiplexing circuit, and a convenient network of dummy elements has been implemented to maintain a constant load at each output of the ring VCO. The proposed solution has been fabricated through a 0.18-μm SiGe BiCMOS process (with a f_T for HBTs of 120-GHz), and occupies a die area of 3.3-mm x 3.5-mm.

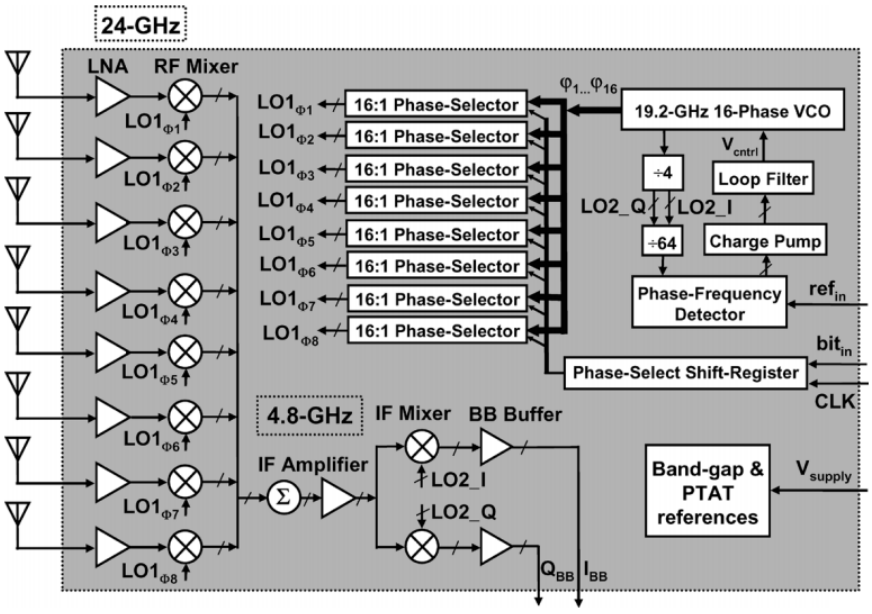


Figure 1.19 – Block diagram of the fully integrated 24-GHz phased-array receiver

In their work, the authors also validate the viability of the LO phase shift approach in narrowband applications. Figure 1.20 is a calculation of the EVM (expressed in %) as a function of the incidence angle for two different signal bandwidths (750-MHz,

corresponding to 1-Gb/s and 7.5-GHz, corresponding to 10-Gb/s) for an 8 antennas receiver working at 24-GHz. The authors reports that for a signal bandwidth of 750-MHz (that is 3.125% of the carrier frequency), the worst case EVM that is obtained for an incidence angle of 90° is lower than 2%, whereas for a signal bandwidth of 7.5-GHz (that is 31.25% of the carrier frequency), the worst case EVM is approximately 10%. This demonstrates that the LO phase shift approach can be used in wireless communications, though the narrowband condition must be satisfied to preserve signal integrity. As already discussed, the LO phase shift approach introduces a constant phase shift across the signal bandwidth instead of a linear one. This means that at the carrier frequency the phase of the signal received at the different paths is coherent, whereas a linear phase error is introduced at the other frequencies, which results in noise. In digital communications, this noise has the effect to spread the constellation diagram (corresponding to a degraded EVM).

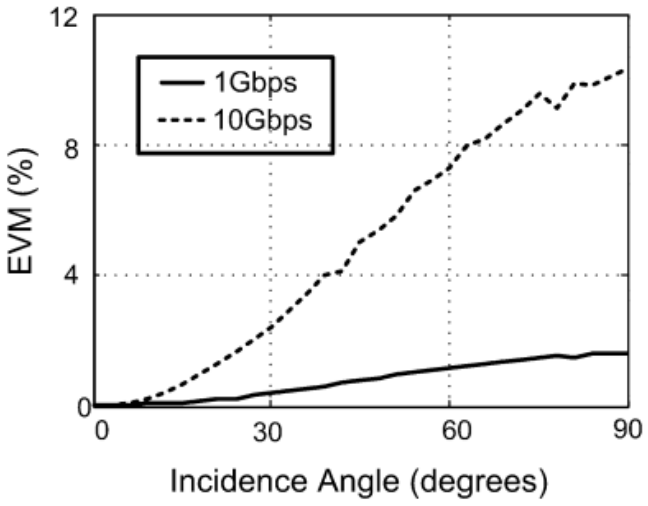


Figure 1.20 – EVM vs. incidence angle for two different signal bandwidths (750-MHz and 7.5-GHz)

The authors also elaborated around the error that is introduced by the quantization of LO phase shifts. In fact, when a discrete number of phase shifts is available, only at certain incidence angle a coherent reconstruction of the carrier can be achieved. Figure 1.21 is a calculation of the EVM (expressed in %) as a function of the incidence angle when 8 (3-bit), 16 (4-bit), 32 (5-bit) or continuous phase shifts are available, for a signal bandwidth of 7.5-GHz.

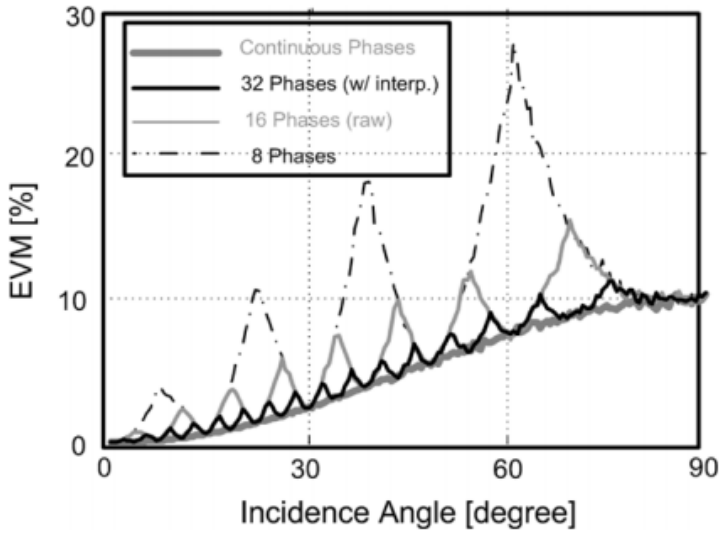


Figure 1.21 – EVM vs. incidence angle for 8 (3-bit), 16 (4-bit), 32 (5-bit) or continuous phase shifts available

When working with continuous phase shifts, the worst case EVM is recorded for an incidence angle of 90°, which is the condition where the largest time delay between antennas is expected. Instead, when working with quantized phase shifts, peaks occurs at specific incidence angles, and are characterized by a shape that also depends on the bandwidth of the

received signal. For example, using a 4-bit phase shift resolution leads to a peak EVM that is 1.5 times higher than the one obtained using continuous phase shifts.

Table 1.1 presents a summary of the above cited phase shifting architectures.

Reference	Technique	Technology	Resolution	Frequency
[1]	RF path	Discrete PIN diodes	-	-
[2]	RF path	Discrete PIN diodes	6-bit	1030- to 1090-MHz
[3]	RF path	Liquid crystals + PCB Substrate	Continuous	12-GHz
[4]	RF path	Monolithic LC circuits	5-bit	6- to 18-GHz
[5]	RF path	Monolithic MEMS	4-bit	DC to 40-GHz
[6]	RF path	Fiber-based photonic circuit	Continuous	20- to 30-GHz
[7]	RF path	Integrated optics	Continuous	10.7- to 12.75-GHz
[8]	IF path	FPGA-based hybrid circuit	-	-
[9]	LO path	Monolithic 40-nm CMOS	6-bit	44- to 54-GHz
[10]	LO path	Monolithic 0.18-µm SiGe BiCMOS	4- to 5-bit	24-GHz

Table 1.1 – Summary of the above cited phase shifting architectures

The architectures compared in Table 1.1 allow to derive some further considerations about state of the art phase shifters. In fact, among the various solutions that have been reviewed, the most interesting ones are those that can fit into an IC fabricated through a cheap technology process. Even though most suited to narrowband systems, the LO path phase shifting approach seems the most promising, particularly when LO phase shifts are implemented during the very synthesis process of the LO. In this scenario, a renowned LO phase shifting solution has been proposed in literature, based on the DDS-PLL architecture. The DDS-PLL architecture will be extensively presented in the following chapter, as a solution to implement cheap, compact and lightweight BSUs for phased arrays characterized by an exceptional phase resolution, low complexity and a frequency independent theory of operation.

1.4 Applications of phased arrays

Traditional applications of phased arrays are in warfare, where they have been extensively used to implement radar systems for guidance, tracking and imaging solutions (among many others). Radar systems analyze reflected electromagnetic waves in order to reconstruct information about the state of the region of space they are inspecting, by transmitting continuous- or pulsed-wave signals towards a certain direction. In radar applications, electronically scanning arrays are used to steer this direction without the need of mechanical movements, as well as to narrow the beam width. Thus, phased arrays allow to implement radar systems that are less prone to failure and, at the same time, are characterized by unmatched performance. Moreover, a further advantage of phased arrays is that the physical configuration of its individual elements can be changed to follow the shape of its housing.

These topologies are also called conformal arrays, and constitute an important opportunity for designers that cannot be implemented with single element antennas. The *IEEE Standard Definition of Terms for Antennas* gives the following explanation for conformal arrays: "An array that conforms to a surface whose shape is determined by considerations other than electromagnetic; for example, aerodynamic of hydrodynamic" (Figure 1.22).

Figure 1.22 – Phased array radar inside the nose of an aircraft

Another traditional application of phased arrays is in astronomy, where they have been used to implement radio telescopes with special technical requirements. It is the case of arrays replacing extremely large apertures, such as the ones implemented through single parabolic antennas. Apart from the practical aspects that such deployments ease, the previously discussed enhanced sensitivity of phased arrays must be kept in mind, as well as spatial selectivity, that can be used to mitigate the effects of interferers coming from undesired

directions. Many research projects are trying to develop radio telescopes whose phased array elements are sparse over ever growing areas, such as the so called square kilometer array (SKA). Among the various options proposed to implement the SKA, ASTRON presented a phased array solution featuring over one million of receiving elements with a mixed RF/digital adaptive beam former. Apart from their traditional applications, phased arrays are now becoming an attractive technology in many consumer applications. For example, automotive radars are expected to be a key enabling technology for the development of autonomous driving cars, as well as collision avoidance, parking aid and blind spot detection systems. The frequency bands allocated for these applications is 24- to 77-GHz, and this leads to phased arrays made up of small antenna elements that can be configured to hide through a conformal shape. Several manufacturers are now able to implement volume production processes with suited performance and low cost potential for this class of applications, such as the ones based on the SiGe BiCMOS technology. For this reason, driver assistance systems based on millimeter-wave radar sensors are expected to be introduced in the full range of newly introduced car models. In [11], Hasch et al. (2012) presented an overview of the key issues in semiconductor technology, packaging and antenna systems for millimeter-wave technology used in automotive radar sensors. Phased arrays are also expected to play a crucial role in upcoming wireless communication networks (e.g. 5G). In fact, they can be used to increase network capacity, extend coverage, mitigate multi-path effects and implement user localization services. For example, phased arrays can be used to synthesize multiple beams at the base station, in order to implement spectral efficiency through spatial multiplexing. Radio spectrum is a scarce resource in the cellular industry, and next generation networks are soon to exacerbate this situation, since they are expected to be

an enabling technology for a wider range of use cases and related applications (e.g. ultra-high definition video streaming, augmented reality, Internet-of-Things, …) (Figure 1.23).

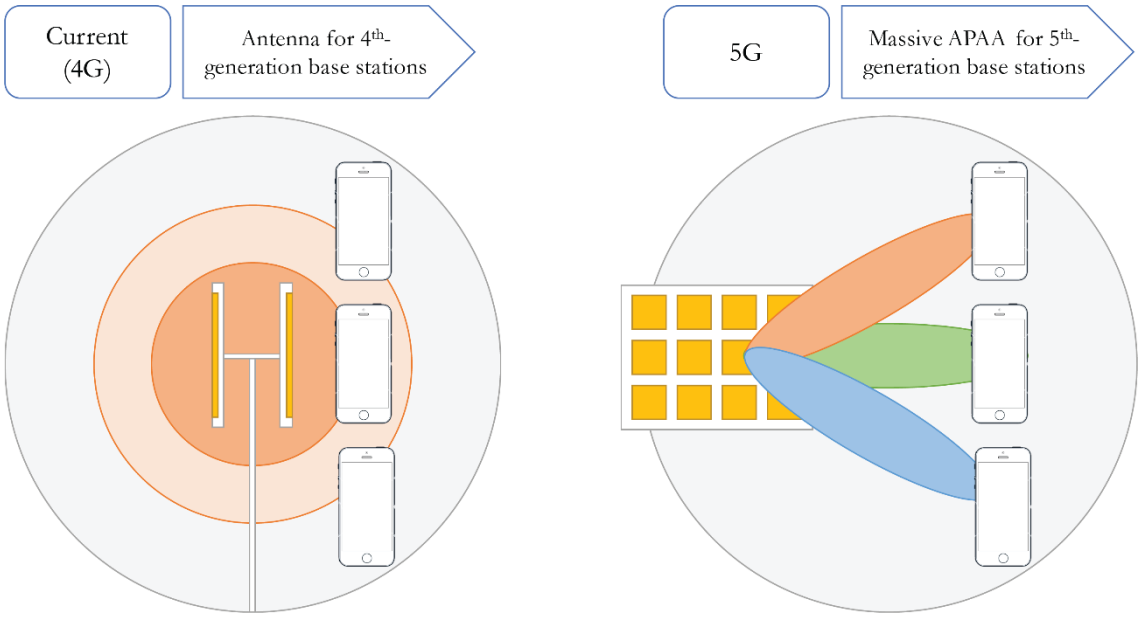


Figure 1.23 – Phased arrays for multi-beam wireless communications

Emerging applications of phased arrays can also be found in biomedical engineering, such as in microwave imaging methodologies for cancer detection through the identification of dielectric discontinuities. These methodologies are proven to deliver an improved contrast ratio with respect to X-ray and ultrasound, and thus are expected to yield higher detection rates. Moreover, they are less harmful for patient than X-ray, as well as cheaper than magnetic resonance imaging. It has also been demonstrated that phased arrays can be used to treat tumours by inducing hyperthermia in diseased tissues through an electromagnetic beam that is focused by following the principles of beam forming.

1.5 Towards a SoC architecture for phased arrays

The growing number of applications empowered by phased arrays is making researchers face new constraints in the implementation of BSUs. In fact, nowadays BSUs must be low-power and low-cost devices that come in a compact and lightweight form factor. To name one example, the above requirements are of primary importance in space applications, where the size and the weight of payloads are critical quantities for a given mission, as well as in consumer applications (e.g. handheld devices), where battery-life and ergonomics can have an impact on the commercial success of the final product. In this scenario, a System-on-Chip (SoC) solution seems the most desirable to implement BSUs, if considered the cost-effectiveness of some mixed-signal BiCMOS processes on the market and the robustness of some open-source IP cores that are synthesizable with standard cells, such as the ones implementing application processors.

In [12], Kane et al. (2005) elaborated around the above statements, and asserted that new applications are continually enabled from the advance in scaling and performance of the BiCMOS technology, prospecting (and persuading) the opportunity to implement phased array SoCs through that platform. The authors state that a SoC for phased arrays would certainly lead to enhancements in terms of cost, weight and size if compared to multichip systems. Moreover, the authors report that the primary limitations to a ubiquitous implementation of phased arrays consists in their very cost, weight, size and power consumption, and that these limitations might have constrained the usage of the phased array technology to aerospace. If traditional phased array systems are organized in modules (Figure 1.24), so that discrete functions implemented in disparate technologies could be integrated

to form a unit (such as GaAs or InP for LNAs, mixers and PAs, PIN diodes for switches and phase shifters, ceramics for passive filters and other silicon ICs for analog or digital baseband functions) the authors state that a monolithic implementation of these architectures will finally lead to a desirable revolution in terms of cost, size and weight.

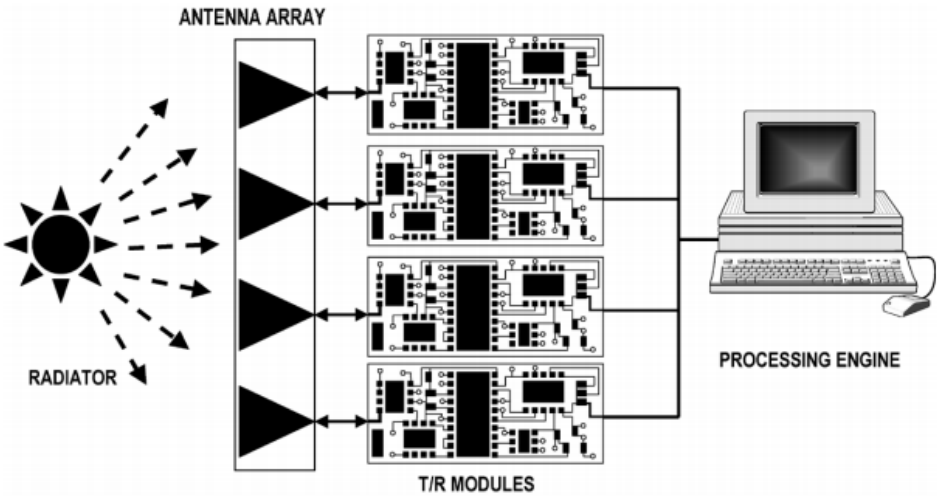


Figure 1.24 – Phased array system based on T/R modules

The authors underlined that the complexity of phased array solutions based on T/R modules is typically related to the connectivity effort, power management and packaging, and that modern IC technologies can be a choice to implement single IC solutions where a dense digital subsystem is coupled to a high-performance RF design, leading to chips that are ready for large-scale production. Figure 1.25 depicts their proposed high-level block diagram for a hypothetical SoC for phased arrays.

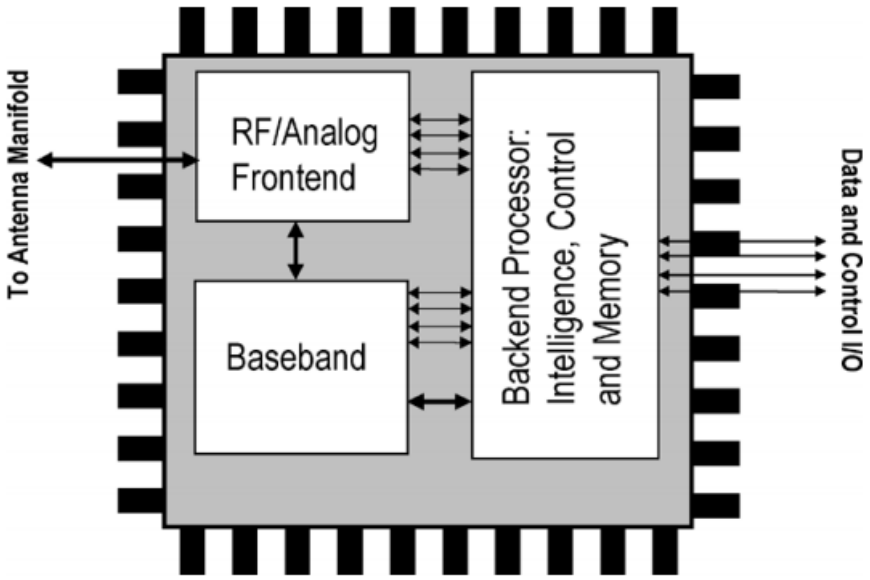


Figure 1.25 – High-level block diagram of a SoC architecture for phased arrays

The block diagram includes RF/analog frontend, baseband and backend processor, implementing all the digital logics that are necessary to control the apparatus. In this system architecture, the authors proposed to integrate the discrete T/R modules into the RF/analog frontend, and to develop the IC in a SiGe BiCMOS technology. The authors remark that the SiGe BiCMOS technology is an optimal choice for SoCs in wireless applications, since it is a platform rich of well-characterized passives (e.g. resistors, capacitors and inductors), specialized devices (e.g. varactors, voltage-controlled oscillators and fuses), and high-performance active devices suited for the implementation of monolithic wireless frontends. For this reason, the authors believe that a SiGe BiCMOS architecture may avoid the need for a solution based on multiple technologies, being a platform that is capable to deliver the complete set of devices that are necessary to implement a desired system, thus overcoming

the limitations of off-chip interconnections for an improved signal integrity and packaging. Moreover, the opportunity to implement, on the same chip, bipolar as well as CMOS transistors allows to broaden the design solution space. If the RF-CMOS platform has been a driver for a broad number of consumer electronics solutions (such as Bluetooth and 802.11 devices), being cheaper and quicker processes than the BiCMOS ones, the authors identify the SiGe HBT BiCMOS technology as the key enabling platform to implement SoCs for phased arrays, due to its improved noise figure (NF), phase noise and transconductance. In fact, SiGe HBTs are the most suited to support the shift towards higher frequencies in phased array applications, that are moving to 20-GHz and above. In the first instance, operating at millimeter-wave frequencies allows to work with smaller antenna apertures, and thus manufacture miniaturized systems that are better supposed to match the requirements of portable electronics. In the second instance, it allows to implement higher data rates in wireless metropolitan area networks (e.g. links above 1-Gbps), automotive radars that are less sensitive to long-range interferences (due to high propagation losses) as well as special monitoring activities. The implementation of a modern SoC for phased arrays allows to take advantage of reprogrammable digital circuits, that allows dynamic software and digital signal processing engine updates even through the integration of FPGA blocks. This leads to solutions that can adapt to new use cases and trade-offs even after they are manufactured. Figure 1.26 depicts a detailed block diagram of the phased array SoC architecture proposed by the authors.

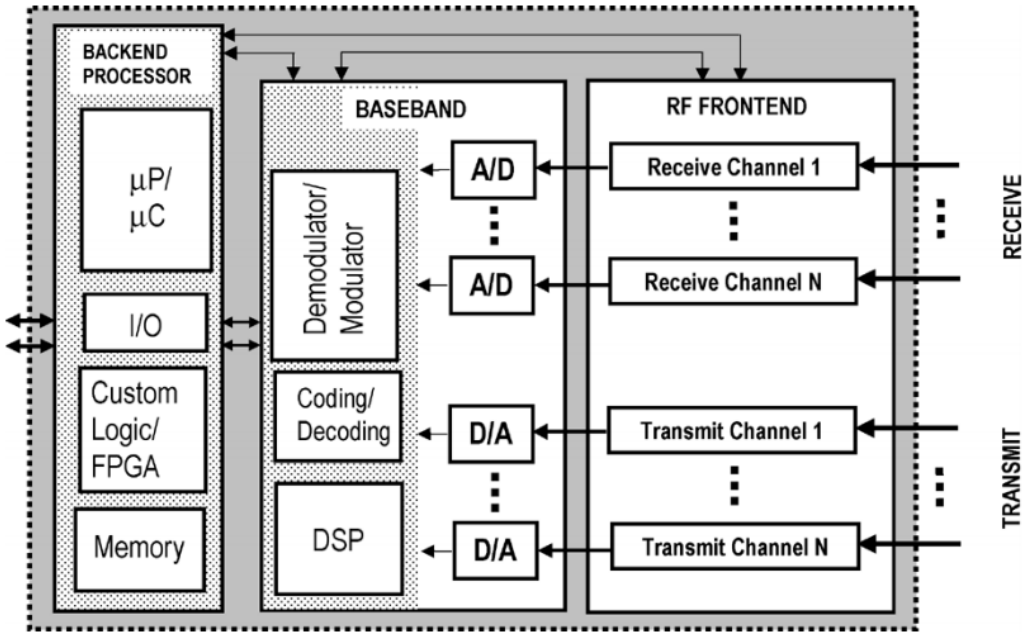


Figure 1.26 – Reference block diagram of a SoC architecture for phased arrays

The proposed block diagram can be customized to implement BSU architectures that follow an RF, IF or LO phase shifting approach, and can thus be thought as a reference architecture for designers. The authors themselves do not pursue a specific phase shifting technique, yet they report about a successful prototype working at 24-GHz that follows the LO phase shifting approach with a 4-bit resolution that has already been reviewed in this chapter ([10]). However, the authors also prospect the opportunities of all-digital beam steering architectures, such as customization through software modification, which come at the cost of an increased power consumption and complexity due to an increased number of A/D and D/A blocks and the need for a high-performance DSP core to perform operations on multiple streams of samples in real-time at challenging high data rates.

Having the limits of all-digital beam steering architectures in mind, the LO phase shifting approach seems the most promising to implement fully integrated solutions for phased arrays. Among the others, a carrier frequency independent LO phase shifting technique, such as the DDS-PLL, seems even more interesting to develop a multifunction and customizable SoC device. This is the reason why the main purpose of this work is the design and implementation of a revised DDS-PLL phase shifter that could eventually be integrated into a SoC.

2 DDS-PLL phase shifter architectures

The purpose of this chapter is to provide a detailed overview of DDS-PLL phase shifters, namely circuit topologies that can synthesize phase shifted LOs to implement beam steering in phased arrays. In its standard architecture, the output of a DDS-PLL is a PLL synthesized sinusoid (with a frequency in the GHz range) whose phase is fine-grained adjusted according to a DDS generated waveform (with a frequency in the MHz range). Being an LO path phase shifting approach based on how LOs are actually synthesized, a DDS-PLL must be assigned to each antenna of the phased array. For this reason, many attempts to decrease the complexity of DDS-PLL phase shifters have been proposed, replacing full-featured DDSs with simpler all-digital hardware solutions.

In the following discussion, the main principles behind common DDSs and PLLs are reported, as well as the standard DDS-PLL architecture. Moreover, known variants to the standard architecture are reviewed.

2.1 Direct Digital Synthesizers

The process of generating an analog signal (e.g. a sine wave) through a digital-to-analog conversion of its samples takes the name of direct digital synthesis, and the electronic circuits used to implement this function are known as direct digital synthesizers. The main blocks (Figure 2.1) at the basis of a DDS are the phase accumulator, the phase-to-amplitude converter and the digital-to-analog converter (DAC). When these three blocks are integrated into a single IC, it is usually referred to as a complete DDS solution.

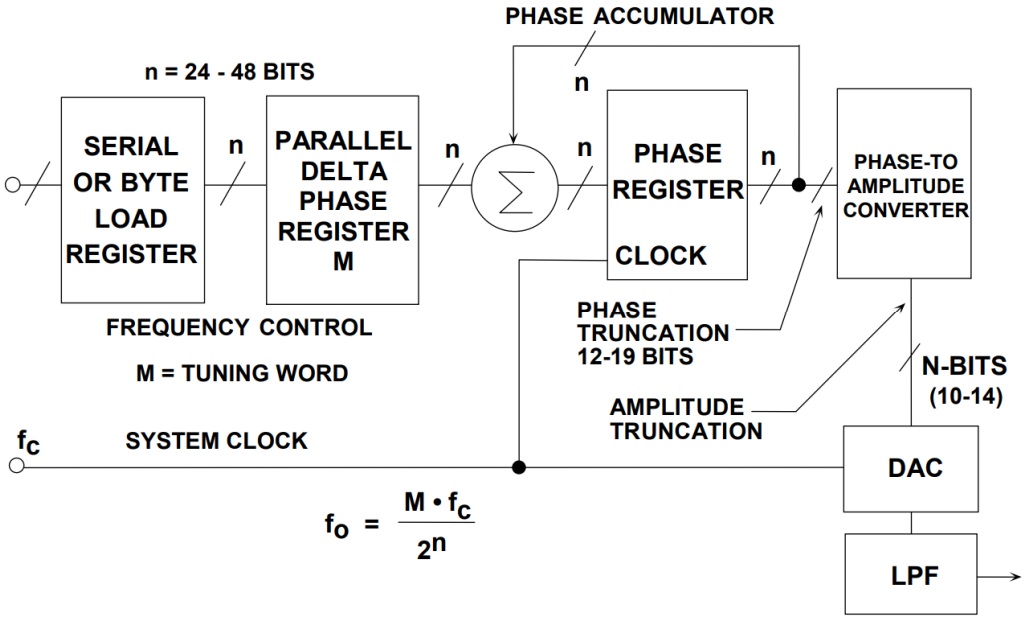
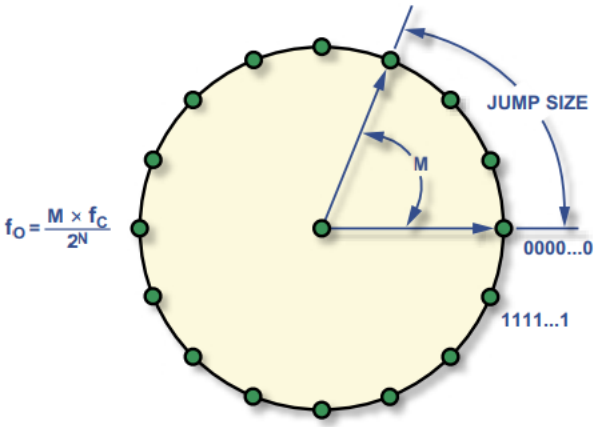


Figure 2.1 – Block diagram of a DDS

As soon as the phase-to-amplitude converter is a sine look-up table (LUT), the DDS can produce sine waves at various frequencies, depending on its reference clock frequency and the tuning word programmed into its frequency control register. This tuning word is a binary number that is fed to the phase accumulator in order to select, at each clock cycle, the correct amplitude sample from the LUT. Then, this amplitude value is converted into a voltage output through the DAC. In order to generate a fixed-frequency sine wave the frequency tuning word must be kept constant in such a way that, at each clock cycle, the same phase increment is applied to the output signal. The larger the phase increment, the faster the synthesized output frequency is; the shorter the phase increment, the slower the synthesized

output frequency is. This is because a different number of clock cycles will be necessary to complete a revolution around the phase wheel (Figure 2.2).



n	NUMBER OF POINTS
8	256
12	4096
16	65535
20	1048576
24	16777216
28	268435456
32	4294967296
48	281474976710656

Figure 2.2 - Digital phase wheel graphical representation

Considering the phasor representation of a sine wave, one revolution around the phase wheel, performed at a constant speed, results in one complete cycle of the output sine wave. This is because the phase accumulator provides angular values that correspond to certain rotations of the above phasor around the phase wheel. The phase accumulator is a modulus-M counter, incrementing its current value at each clock cycle by the content of the above cited frequency tuning word (namely the jump size around the phase wheel). This overall behavior translates into the following equation, which determines the output frequency of

the DDS based on the frequency tuning word (M), the reference clock frequency and the length of the accumulator register (n):

$$f_{OUT} = \frac{M \cdot f_C}{2^n}$$

Since the Nyquist–Shannon sampling theorem dictates that at least two samples per period are needed to reconstruct a sine wave, the maximum value for f_{OUT} is $f_C/2$. In actual implementations, f_{OUT} is at most $1/3$ of this limit value in order to mitigate non-ideal effects in the reconstructed signal (e.g. approximation of zero-width impulses by rectangles at DAC output). When generating a fixed-frequency signal, the output of the phase accumulator is a sampled ramp, since its content is a growing binary number that overflows. The length of the phase accumulator register is typically higher than the number of samples in the sine look-up table. This is because the size of the LUT must be small enough to be practical, whereas the accumulator register must support a high phase and frequency tuning resolution; a convenient truncation of the less significant bits is thus performed at the interface between the two subsystems. Moreover, typical sine LUTs only describe one quarter of a sine period, in order to save die area. This can be done due to the symmetry of sine waves. DDSs are not constrained to synthesize sine waves, in fact they can generate any waveform that complies the Nyquist–Shannon sampling theorem (namely whose highest frequency domain component is below ½ the clock frequency).

It seems important to emphasize the key advantages of high-performance and functionally integrated DDS ICs, in order to understand their importance in communication systems. In fact, they provide: i) digitally controlled micro-hertz frequency-tuning and sub-

degree phase-tuning; ii) fast hopping speed in tuning output phase and frequency (without the analog-related drawbacks due to the circuit dynamics); iii) no component aging or manual tuning; iv) remote configuration and optimization through digital control interface. This is the reason why many modern applications rely on DDSs as programmable waveform generators. However, a number of trade-offs are associated to the DDS technique, such as: i) bandwidth; ii) spectral purity; iii) cost, complexity and power consumption. Trying to elaborate more around spectral purity in DDSs, it must be said that DDSs are rarely affected by phase noise problems. In fact, they produce outputs whose changes are related to digital phase numbers that are produced through a fixed clock signal. Conversely, DDS systems are affected by spurs, which are discrete (narrowband) spectral impurities originated by quantization errors and DAC errors. Since the actual output from a DAC is a quantized representation of an ideal amplitude value, an error signal is summed to the desired output. Moreover, the DAC can introduce sources of errors such as distortions into its trans-characteristic. Another main source of spurs in DDSs is the ratio of the output frequency to the clock frequency. The spurious-free dynamic range (SFDR) is the ratio (expressed in dB) between the power of the fundamental signal and the power of the highest spurious signal in the spectrum. The SFDR is an important figure of merit for DDSs, and is particularly important in communication systems where a channelized frequency spectrum is shared between devices (and thus signals can be corrupted from spurious power generated in neighboring transmissions). Trying to summarize the above considerations, DDSs are particularly interesting synthesizers when the following design objectives must be pursued [13]: i) sub-microsecond settling time; ii) multi-octave operation; iii) very fine frequency resolution; iv) phase continuous frequency changes; v) exceptionally linear analog or data

modulation; vi) exceptionally linear sweep and chirp modulation; vii) quadrature generation over multiple octaves. It seems interesting to report the applications that the authors of [13] reported as particularly sensitive to those requirements, that are: i) frequency hopping and spread spectrum radios; ii) electronic warfare and hammer systems; iii) doppler and chirp radars; iv) high speed and/or high resolution PSK or FSK; v) radio and television broadcast equipment; vi) test equipment.

2.2 Phase Locked Loops

Phase Locked Loops (PLLs) are feedback systems used to synthesize periodic waveforms that verify a constant phase and frequency relation towards a given reference signal. Among the main applications of PLLs, one is that of frequency multiplication (a.k.a. frequency synthesis), where a steady and high frequency sine wave is generated from a low frequency periodic oscillation. Basic PLLs are composed by 5 building blocks, namely the phase-frequency detector (PFD), the charge pump (CP), the loop filter, the VCO and the feedback divider (Figure 2.3) [14].

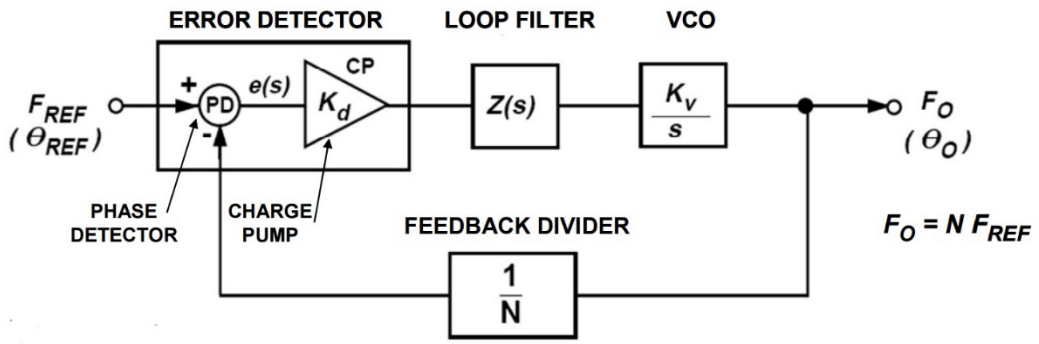
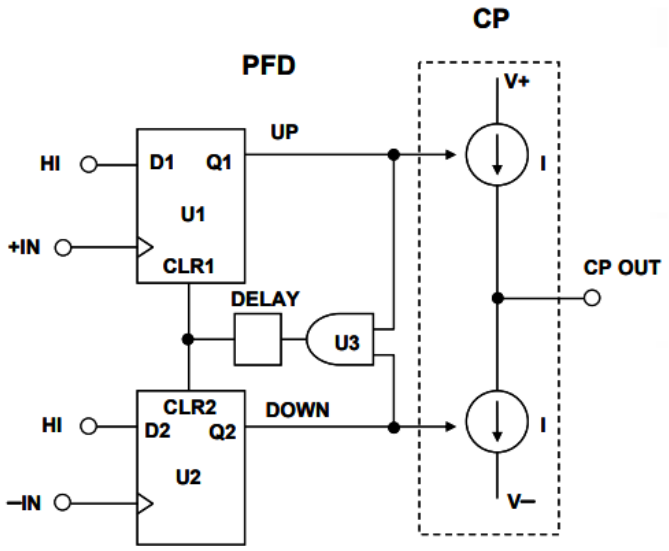


Figure 2.3 – Basic PLL block diagram

The PFD and the CP implement the error detector, which is the subsystem where the negative feedback of the PLL actually takes place. The main purpose of the negative feedback is to bind the phase- and frequency-lock condition to an error signal $e(s)$, that goes to zero when $F_O = N \cdot F_{REF}$, which (conceptually) is the main goal of the PLL. It seems important to recall that: i) the signal at the output of the VCO oscillates at a frequency that is F_O; ii) N is the number of times F_O is divided before being compared with the reference signal; iii) F_{REF} is the reference signal frequency. Thus, when the two inputs of the error detector are equal in phase and frequency, the error signal will be constantly equal to 0 and (conceptually) the phase- and frequency-lock condition is reached. A typical implementation of the PFD block in "Type II" PLLs is depicted in Figure 2.4, and consists of two D-type flip flops. The outputs of these flip flops respectively drive a positive and a negative current source, in such a way that the charge pump (that is a pair of current sources) outputs either a positive or a negative current pulse, if not in three state. The delay element placed after the AND gate of the PFD ensures that the CP output never enters a high-impedance state for more than one PFD cycle. This condition is reached when a perfect phase match is achieved among the inputs of the PFD, and leads to an undesired phenomenon known as the dead zone effect. In fact, when the CP output enters a long high-impedance state, the VCO output phase can drift in such a way that a significant error may be developed. This error is the cause of a cyclical presence of pulses at the CP output, which in turn leads to a modulation of the CP output at a sub-harmonic of the PFD frequency. Since these spectral components fall inside the bandwidth of the loop filter (as will be further explained later), they cannot be attenuated and thus would result in spurs ruining the output spectrum characteristics of the PLL. For

this reason, the above delay element is employed to guarantee that a short current pulse is generated at each PFD cycle, even when a close phase alignment is achieved.



UP	DOWN	CP OUT
1	0	+I
0	1	–I
0	0	0

Figure 2.4 – Schematic diagram of the PFD driving the CP

The main principles behind PLLs have been explained, on the other hand PLLs must be analyzed as dynamic systems in order to develop a complete understanding of their behavior, especially during transients. It is now shortly recalled the linearized model of Type II PLLs (such as the ones reported above). The open-loop transfer function of such PLLs is:

$$G_0(s) = K_{PFD} I_{CP} G_{LPF}(s) \frac{K_{VCO}}{s}$$

where K_{PFD} is the gain of the PFD, $G_{LPF}(s)$ is the transfer function of the loop filter and K_{VCO} is the gain of the VCO. The above transfer function is derived assuming that the output of the VCO is a sine wave whose frequency is $\omega_{out} = \omega_{FR} + K_{VCO} v_{cont}$. A recurrent configuration for the loop filter is the lead/lag filter (Figure 2.5).

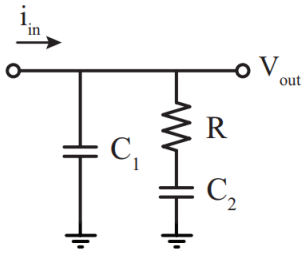


Figure 2.5 – Lead/lag filter schematic

Its transfer function is:

$$G_{LPF}(s) = \frac{1 + \frac{s}{\omega_z}}{s(C_1 + C_2)\left(1 + \frac{s}{\omega_p}\right)}$$

where:

$$\omega_z = \frac{1}{R_1 C_2} \qquad \omega_p = \frac{C_1 + C_2}{R_1 C_1 C_2}$$

The transfer function is often approximated as follows, since C_1 is many times smaller than C_2:

$$G_{LPF}(s) \approx \frac{1}{sC_2}\left(1+\frac{s}{\omega_z}\right)$$

The lead/lag filter is a circuit topology that is commonly used because of the improved loop stability it introduces with respect to a single load capacitance, which is the simplest form of integrator one can implement at the output of the PFD. Since the PLL is a feedback system, its transfer function can be written in the following form:

$$\frac{Y(s)}{X(s)} = \frac{G_0(s)}{1+G_0(s)H(s)}$$

and thus:

$$\frac{\Phi_{out}(s)}{\Phi_{in}(s)} = \frac{K_{PFD}\frac{1}{sC_2}\left(1+\frac{s}{\omega_z}\right)\frac{K_{VCO}}{s}}{1+K_{PFD}\frac{1}{sC_2}\left(1+\frac{s}{\omega_z}\right)\frac{K_{VCO}}{s}} = \frac{\frac{K_{PFD}}{C_2}\left(1+\frac{s}{\omega_z}\right)K_{VCO}}{s^2+\frac{K_{PFD}}{C_2}K_{VCO}\left(1+\frac{s}{\omega_z}\right)}$$

It is possible to put the denominator of the above transfer function into the standard second order form:

$$D(s) = s^2 + \frac{K_{PFD}}{C_2}K_{VCO}\left(\frac{s}{\omega_z}\right) + \frac{K_{PFD}}{C_2}K_{VCO} = s^2 + \frac{s\omega_0}{Q} + {\omega_0}^2$$

and thus derive:

$$\omega_0 = \sqrt{\frac{K_{PFD}}{C_2}K_{VCO}} \qquad Q = \frac{\omega_z}{\omega_0}$$

as well as the position of the poles:

$$s_{1,2} = \frac{-\frac{\omega_0}{Q} \pm \sqrt{\left(\frac{\omega_0}{Q}\right)^2 - 4{\omega_0}^2}}{2}$$

The derived loop gain has two poles in the origin, which is the main characteristic of Type II PLLs. Through the above model, typical root locus design techniques can be employed to shape the closed loop system response. However, a more detailed simulation framework must be implemented for verification (e.g. SPICE level simulation), including phenomena such as non-linear effects.

It can be said that the PLL acts as a low-pass filter to reference input phase changes. This means that whenever the reference input phase changes through slow variations, the output will be able to follow those variations. Contrarily, when the reference input phase changes through fast variations, the output will not follow those variations, due to the low-pass behavior of the loop. Thus, the lower the cut-off frequency of the loop filter, the better the reference jitter rejection. On the other hand, the PLL acts as a high-pass filter to the VCO output. Thus, the higher the cut-off frequency of the loop filter, the better the VCO output jitter rejection. This means that a trade-off decision must be carried out, since the PLL bandwidth must be set to minimize both jitter sources, and choosing a low PLL bandwidth can actually degrade its output jitter performance. For example, this can be done by improving the spectral purity of the VCO (e.g. through a voltage-controlled crystal oscillator, VCXO) or by cleaning the reference signal before it is fed to the PLL (e.g. through a jitter-cleaning IC). Two critical aspects of any PLL design are long- and short-term steadiness of the synthesized oscillation. Long-term steadiness relates to fluctuation that may take place in

a relatively long period of time, such as an hour, a day or a month. It is typically measured as a Δf over f ratio and expressed in dB. Instead, short-term steadiness relates to fluctuation that may take place in a relatively short period of time, which is within a second. This evaluation requires a spectrum analyzer, and allows to discriminate between spurious spectral components located at discrete frequencies and random noise fluctuations that leads to the broadening of the output spectrum (Figure 2.6).

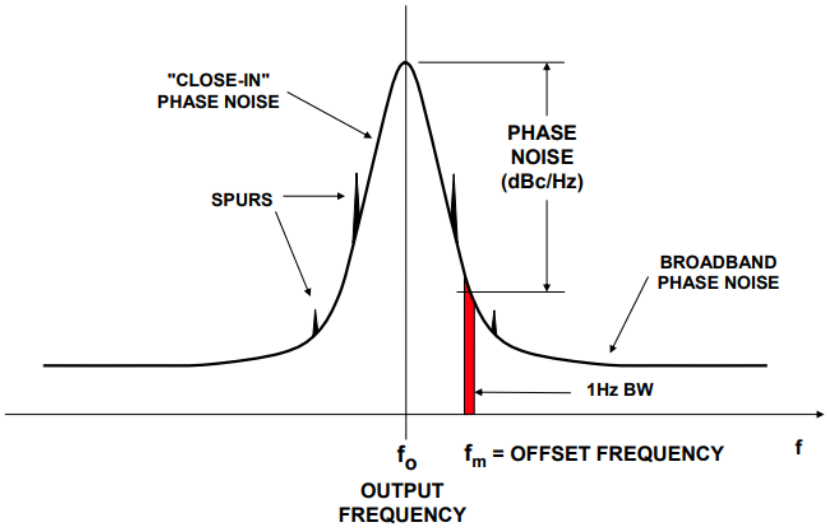


Figure 2.6 – Conceptual representation of noise and spurs in PLLs

It seems interesting to report that among the most detrimental spurs to the PLL output spectrum are the ones that are originated by the reference signal falling inside the loop bandwidth. In fact, these spurs are reproduced at the VCO output and gained-up. Phase noise is measured as the ratio between the noise power in a 1-Hz bandwidth at a specified frequency offset from f_o and the oscillator signal amplitude at f_o, and is expressed in

dBc/Hz. Figure 2.7 depicts phase noise as a function of the frequency offset. It allows to determine a number of regions where the curve has a different slope, depending from the noise source that is eventually giving its contribution.

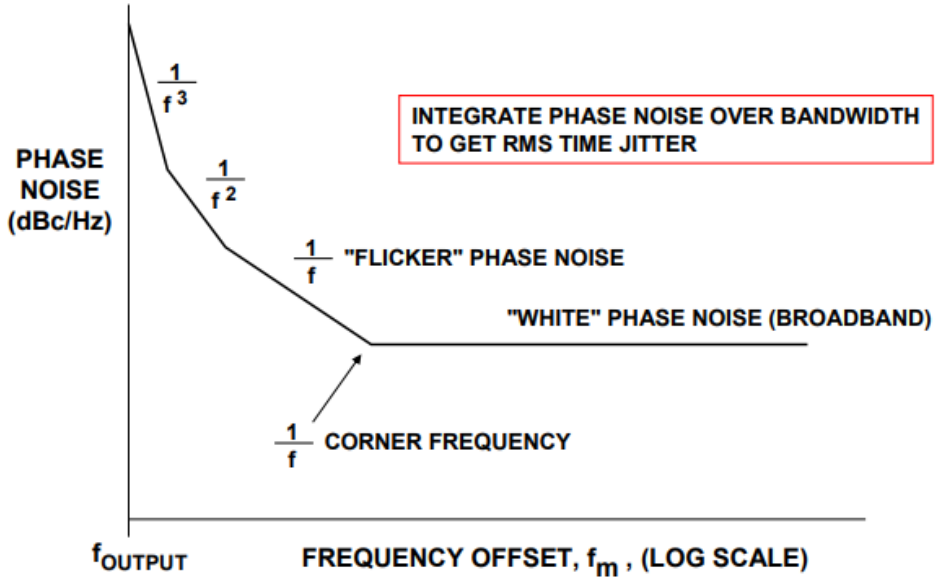


Figure 2.7 – Conceptual diagram of phase noise (dBc/Hz) vs. offset from output frequency

The region where the slope is $1/f$ is related to flicker phase noise, whereas the region where the slope is 1 is related to white phase noise. The $1/f$ corner frequency is a figure of merit that allows to rank the noise performance of a PLL. The lower the $1/f$ corner frequency, the better the loop performance. It seems interesting to cite that integer-N PLLs are not the only PLL synthesizers that can be implemented: offset-PLLs and fractional-N PLLs are renowned architectures whose further discussion is, however, outside the scope of this work.

2.3 DDS-PLL standard architecture

The DDS-PLL architecture allows to combine the high frequency performance of PLL synthesizers with some of the unmatched characteristics of DDSs, to generate LO signals whose phase (and frequency) can be tuned in exceptionally fine-grained steps. This is the reason why DDS-PLLs seem an interesting solution to implement LO phase shifters in phased arrays. DDSs can achieve extraordinary frequency and phase resolutions (e.g. up to 10^{-6}-Hz), have an output frequency that can span over a range that can even exceed 40 octaves (e.g. from 1-μHz to 150-MHz), can make extremely fast output frequency changes (even thousands of times faster than PLLs), can be synchronized to implement multiple DDS architectures (e.g. for the generation of quadrature signals) and can be used to implement high-speed digital modulations (such as PSK and FSK). However, certain applications can arise their limitations, such as the impossibility to implement, under certain circumstances, exact frequencies and phases due to quantization errors caused by the digital nature of their functioning. For example, a DDS clocked at 100-MHz with a frequency resolution of 32-bits cannot synthesize an output frequency that is exactly 20-MHz, since those frequencies are not in a power of two relationship. PLLs are able to perform phase and frequency lock to the reference signals at their input and perform extremely precise synthesis of equally spaced frequencies that are even thousands of times faster than it. However, they are not designed to change their frequency instantaneously and their resolution is far from the sub-Hz steps achievable through DDSs. In DDS-PLL phase shifters, the PLL is used to scale-up (in the GHz range) the frequency of a DDS generated waveform (in the MHz range), such as a sine wave, whereas the phase of the two waveforms is related through a deterministic relationship.

When certain circumstances are met, the phase resolution of the DDS is also maintained at the PLL output (as will be demonstrated in the following discussion).

In literature, three main techniques have been proposed to implement a DDS-driven PLL, whose differences reside in the role that the DDS plays inside the loop. A technique can be to employ the DDS in the feedback path of the PLL, acting as a fractional divide-by-N stage (Figure 2.8). This allows to improve the output frequency resolution of the PLL, but at the expenses of a degraded spectral purity, since the spurs originating from the DDS that fall inside the loop bandwidth are gained-up by an $\frac{f_{OUT}}{f_{REF}}$ factor and then transferred to the output. Moreover, the above configuration can only be implemented with PLL ICs that support the usage of external dividers.

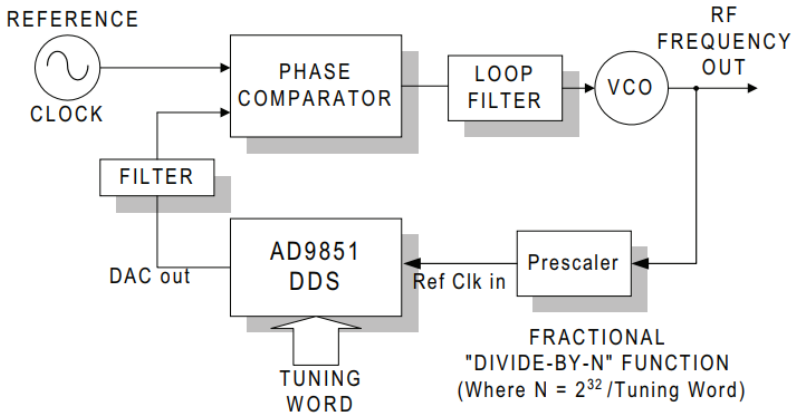


Figure 2.8 – DDS in the feedback path of the PLL acting as a fractional divide-by-N stage

Another technique can be to employ the DDS as an offset frequency generator in an offset-PLL (Figure 2.9), that is a PLL where an analog mixer is inserted in the feedback path. The

main advantage of this configuration is that DDS spurs within the loop bandwidth will pass unchanged to the output, rather than amplified as in the previous case. However, the already introduced $\frac{f_{OUT}}{f_{REF}}$ gain is applied to spurs that fall inside the loop bandwidth originating from the reference signal. This is a practical limitation to the PLL multiplication factor, as well as the need to implement an adequate filter at the mixer output.

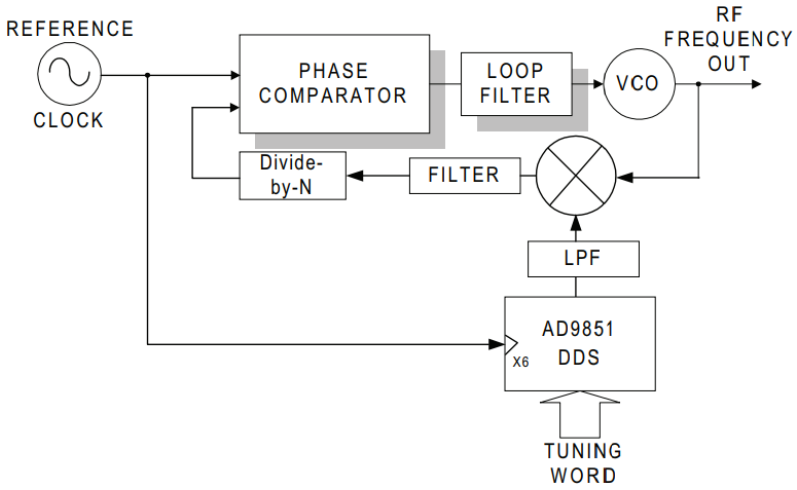


Figure 2.9 – DDS as an offset frequency generator in an offset-PLL

In [15], Bonifanti et al. (2003) implemented a DDS-based PLL for 2.4-GHz frequency synthesis following this scheme. The authors achieved an 80-MHz tuning range for their solution, with a settling time of 3-µs and 240-Hz accuracy. Measured phase noise integrated over a bandwidth of 1-kHz to 1-MHz is 0.9°. When the PLL is locked, the output frequency is defined by the following equation:

$$f_{OUT} = N \cdot M \cdot f_{REF} + M \cdot f_{DDS}$$

The authors state that, since in this configuration f_{REF} and f_{DDS} are independent terms, f_{REF} can be high so that a large PLL bandwidth can be implemented, and thus a faster frequency switching can be achieved. Moreover, the authors report that this allows to lower down the N and M multiplication factors, in such a way that PFD and divider phase noises are reduced. In their prototype, the DDS clock was set to 280-MHz, thus its highest output frequency results 70-MHz. Since an 80-MHz tuning range was the target specification, a divide-by-M prescaler was inserted before the mixer, thus allowing to reduce the required frequency range (Figure 2.10). The authors underline that this prescaler stage gains-up the DDS noise and spurs within the loop bandwidth (that in their work is 2-MHz) by M before passing to the output, thus $M = 2$ was chosen as the best tradeoff.

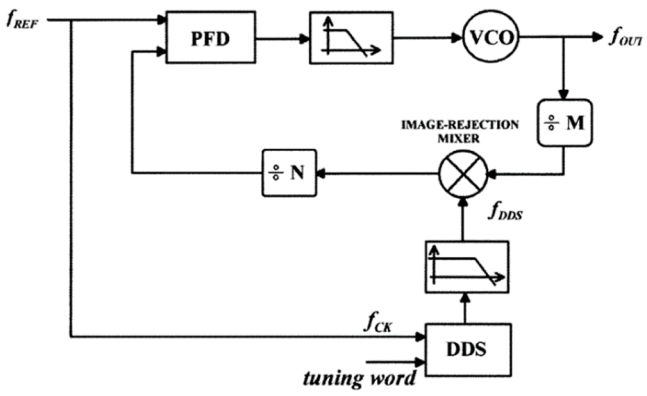


Figure 2.10 – DDS-based PLL for 2.4-GHz frequency synthesis

The authors also reported the power consumption of the overall DDS-PLL solution, as a budgetary analysis for a future IC integration (Figure 2.11). The DDS-DAC blocks represent

the most important contribution to power consumption, and the authors assert that this may be the main weakness for a fully-integrated implementation of the system. This can be considered a key motivation to the research of a revised DDS-PLL phase shifter architecture for phased arrays, especially if considered the previously discussed revolution that an SoC for phased arrays is expected to introduce.

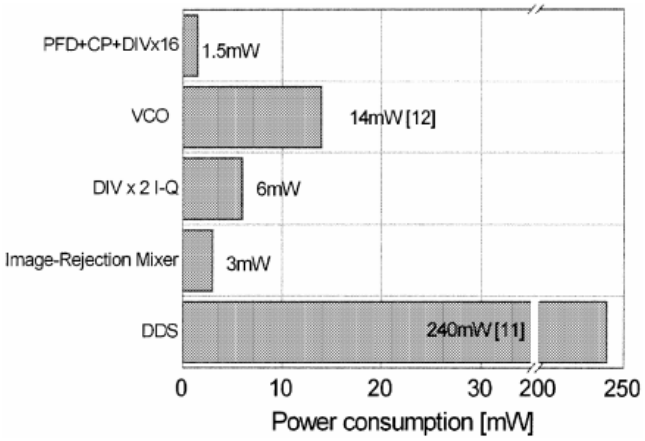


Figure 2.11 – Budgetary power consumption of the proposed DDS-PLL solution

Finally, a DDS can be employed as the reference signal generator for the PLL. This is the simplest DDS-PLL architecture that can be implemented, but in this scheme the DDS noise and spurs within the loop bandwidth are once again gained-up by an $\frac{f_{OUT}}{f_{REF}}$ factor before passing to the output. Since DDSs are far from being ideal sources, this effect can become an impairing limitation to the implementation of this technique. In [16], Li et al. (2014) proposed an active array antenna with remote controllable radiation patterns for mobile communications based on this architecture. The proposed array consists of 8 antenna

elements equipped with DDS-PLL phase shifters, implemented through two AD9959 DDS ICs and 8 ADF4350 PLL ICs. The authors report a 0.1° beam steering resolution, with an LO frequency tuning range from 1730- to 1790-MHz. Figure 2.12 depicts the system architecture of the phased array, whereas Figure 2.13 depicts the block diagram of the transceiver architecture assigned to each antenna.

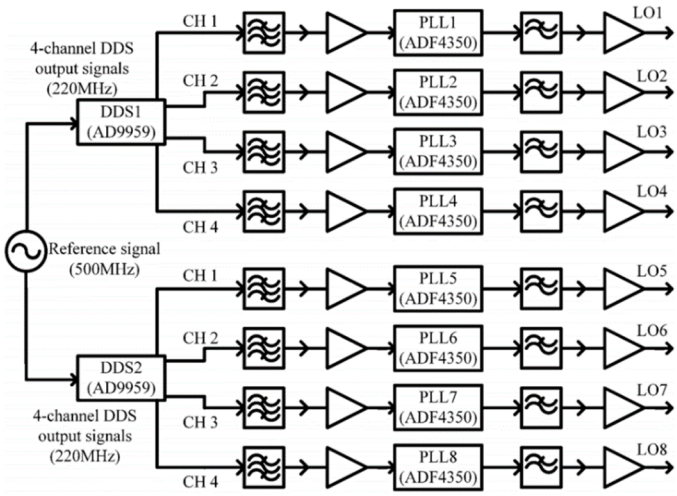


Figure 2.12 – System architecture of the proposed phased array

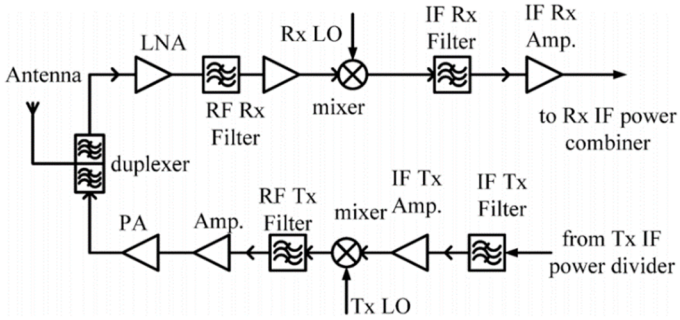


Figure 2.13 – Block diagram of the transceiver architecture assigned to each antenna

Each antenna operates from 1920- to 1980-MHz in reception, and from 2110- to 2170-MHz in transmission, since the target application is a WCDMA base station system. In this implementation, a bandpass filter is used at the output of the DDS to eliminate spurs before its synthesized signal is fed into the PLL. The DDS output frequency was set to 220-MHz, whereas the DDS clock frequency was 500-MHz. The PLL implementation is conventional, thus when the loop is locked, the output phase and frequency are locked to the ones of the reference signals from the DDSs. The following equation governs the relationship between input and output frequencies, namely f_{DDS} and f_{LO}:

$$f_{LO} = \frac{N}{R} f_{DDS}$$

whereas the following equation governs the natural relationship between input and output phases, namely θ_{DDS} and θ_{LO}:

$$\theta_{LO} = \frac{N}{R} \theta_{DDS}$$

where N and R are respectively the multiplication and the division factors. The output frequency and phase of the DDS are determined by the frequency tuning word (FTW) and the phase tuning word (PTW):

$$f_{DDS} = \frac{FTW}{2^A} f_{CLK} \qquad \theta_{LO} = \frac{PTW}{2^B} 2\pi$$

where A is the length of the FTW and B is the length of the PTW expressed in bits. Thus, the following equation returns the output frequency and phase of the LO signal based on the DDS configuration:

$$f_{LO} = \frac{N}{R} \cdot \frac{FTW}{2^A} f_{CLK} \qquad \theta_{LO} = \frac{N}{R} \cdot \frac{PTW}{2^B} 2\pi$$

whereas the following equations return the DDS-PLL frequency and phase resolution:

$$\Delta f_{LO} = \frac{N}{R} \cdot \frac{1}{2^A} f_{CLK} \qquad \Delta\theta_{LO} = \frac{N}{R} \cdot \frac{1}{2^B} 2\pi$$

The chosen DDS ICs are characterized by a 32-bit frequency tuning word and a 14-bit phase tuning word. Considering the above equations, this leads to a frequency tuning step equal to 1.86-Hz and a phase tuning step of 0.35° (since $N = 16$ and $R = 1$). The equations are presented in [16] without introducing further considerations. This is because the authors did not use any technique to preserve the phase resolution of the DDS at the PLL output. This was possible since the phase resolution and the output frequency of the chosen DDS were high enough to achieve a 10-bit resolution anyway. Figure 2.14 shows the simulated radiation patterns for 11 equally spaced angle of radiations.

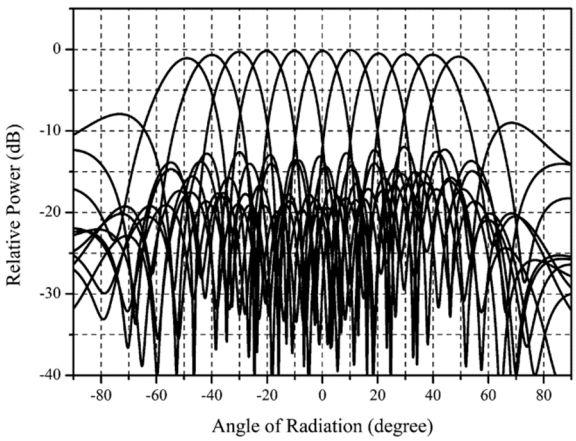


Figure 2.14 – Simulated radiation patterns for 11 equally spaced angle of radiations

The size of the antennas is 65-mm x 75-mm, fabricated on a 1.5-mm thick substrate with a dielectric constant of 3. The authors state that the structure of the antenna is optimized to give a larger beam width in the H-plane to support a wide beam scanning range, and simulation radiation patterns are depicted in Figure 2.15.

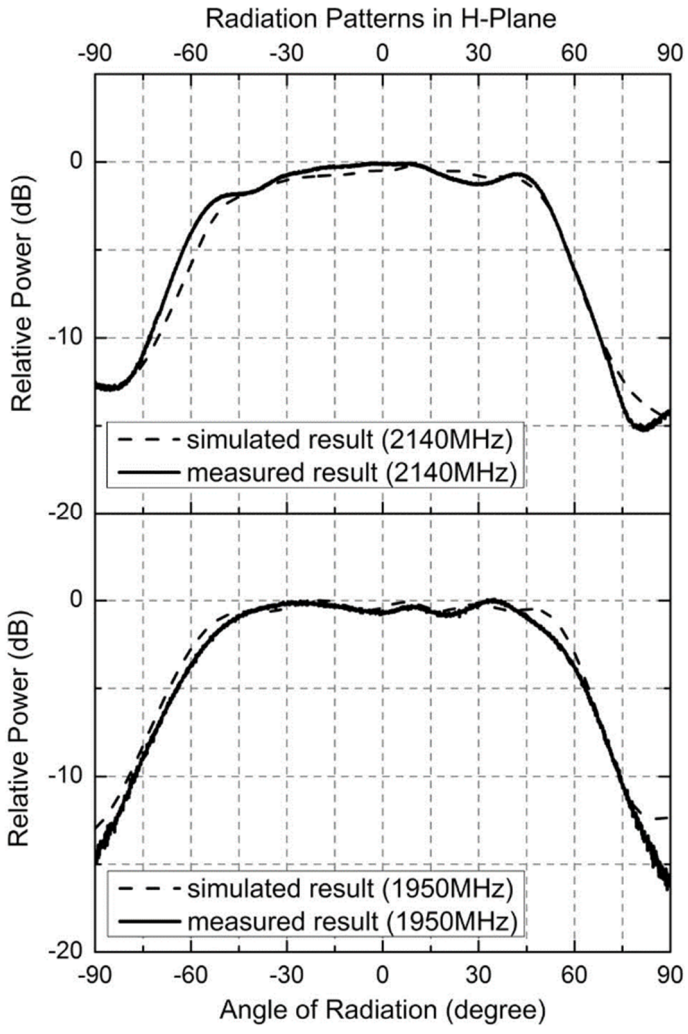


Figure 2.15 – Simulated radiation patterns of the antenna elements employed in the phased array

The authors implemented a calibration procedure (for both TX and RX mode) to compensate for phase mismatches across the channels, that typically originate from ordinary differences among components, antennas and routing of the signals in the PCB board. The calibration procedure for the TX mode has been performed in an anechoic chamber where a spectrum analyser was connected to a horn antenna, placed in such a way that the far-field condition was satisfied (if considered the position of the phased array under test). A steering step of 10° was used to demonstrate the beam steering accuracy. A vector signal generator was used to synthesize an IF signal, and mutual phase shifts were set among a reference channel and one of the other 7 DDS-PLLs, leaving the remaining 6 output channels turned off. This test was used to determine the PTW that returned the lowest received signal strength at the spectrum analyzer, a condition that corresponds to the π phase offset condition. The authors assert that looking for the minimum in the received signal strength vs. PTW characteristics is more significant that looking for its maximum, which is the condition that corresponds the 0 phase offset condition, since in this latter case the recombined signal "is distorted by noise and power imbalance". The overall system has been measured to derive the RF channel performance and the radiation pattern. Among the many parameters that the authors derived, it seems important to report the achieved worst case EVM that is 1.5% for a 5-Mbps data rate both in TX and RX mode (for a specified output power of 21-dBm, a specified input power of -55-dBm and a QPSK modulation scheme), the beam position error with respect to simulated results that is better than 2° and the beam width error with respect to simulated results that is better than 2.5°.

2.4 Known variants to the standard architecture

Some works in literature report the attempt to reduce the complexity of the standard DDS-PLL architecture, in order to lower down its cost and power consumption. In [17], Avitabile et al. (2006) proposed a DDS-PLL phase shifter where the DDS subsystem is replaced by an all-digital circuit that feeds square waves rather than sine waves to the PFD inputs of the PLLs, without degradation of the phase noise performance. In fact, in modern PFDs the phase and frequency mismatch detection are eventually operated by converting the reference input signals into square waves, and then working on their rising edges. For this reason, the proposed variant gets rid of the unnecessary digital-to-analog transformation at the DDS outputs as a mean to lower down the complexity of the circuit topology. Figure 2.16 depicts a conceptual block diagram of the proposed solution.

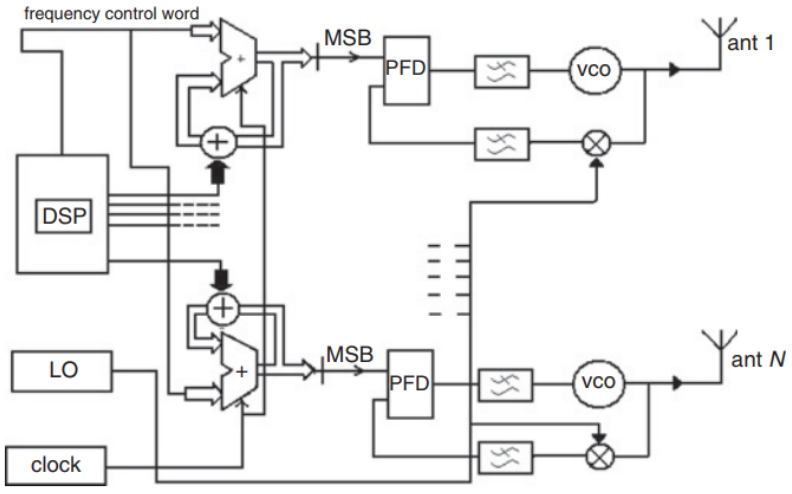


Figure 2.16 – Block diagram of a revised architecture based on an accumulator register and offset-PLLs

The implemented PLLs output frequency is 1.2-GHz and work in an offset configuration. The offset configuration has been used to preserve the output phase in the feedback path. In fact, when using an N divider in the feedback path, the signal that goes into the PFD is:

$$s(t) = A \cdot \cos\left(\frac{\omega_{RF}}{N}t + \frac{\varphi}{N}\right)$$

Whereas when using an offset PLL, the signal that goes into the PFD is:

$$s(t) = A \cdot cos((\omega_{RF} - \omega_{LO})t + \varphi)$$

This means that the output phase φ is directly compared to the one of the reference signal, and thus the phase resolution implemented at the reference signal generator is certainly preserved at the PLLs output. The proposed solution has been tested by the authors in the [0°; 77°] phase shift range by sampling 12 phase configurations (Figure 2.17), and returned a maximum error of 2.12°.

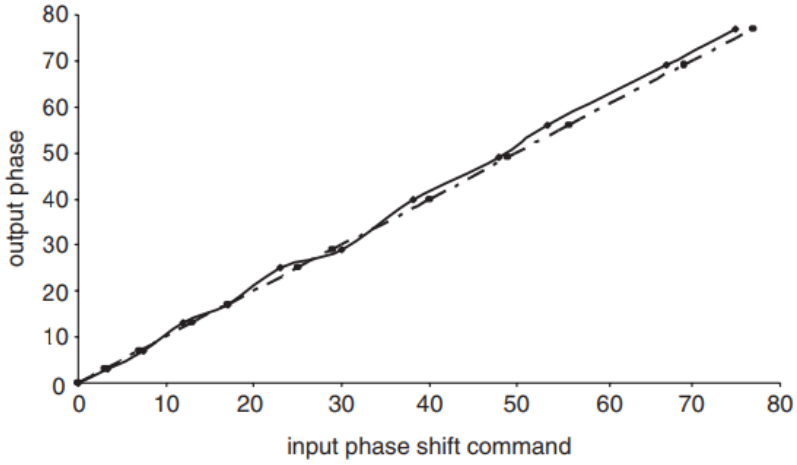


Figure 2.17 – Measured vs. ideal output phase characteristics

The reference frequency generator is based on an accumulator register, and the actual reference signal is the value of the most-significant bit (MSB) at its output. It seems interesting to report how the authors measured the implemented phase shifts. In fact, they perform an indirect measurement of phase shifts between two PLLs through the use of a frequency mixer. Since the two PLLs were locked to the same frequency, the DC output of the mixer is a voltage signal whose amplitude depends on their relative phase (Figure 2.18).

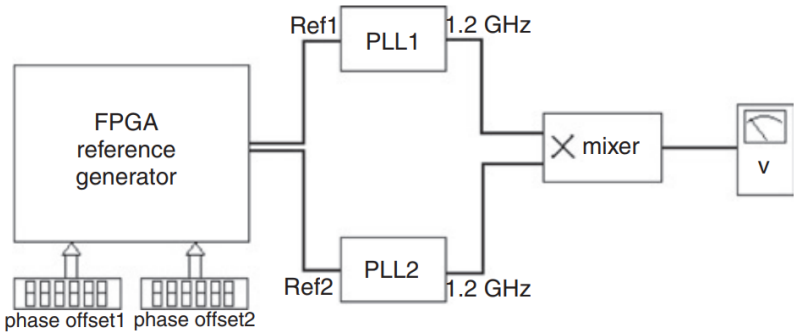


Figure 2.18 – Measurement setup for indirect quantification of phase shifts

In [18], Avitabile et al. (2008) proposed an improved DDS-PLL phase shifter based on an accumulator register, comparators and integer-N PLLs. The proposed solution overtakes the need for the offset-PLL configuration in [17] by taking advantage of the periodicity of sine waves. Figure 2.19 depicts the block diagram of this improved DDS-PLL phase shifter. The phase control unit (PCU) is the all-digital hardware architecture that replaces the full-featured DDSs. The accumulator register is used to implement the phase wheel with a k-bit resolution and the frequency of its clock signal is f_{CLK}.

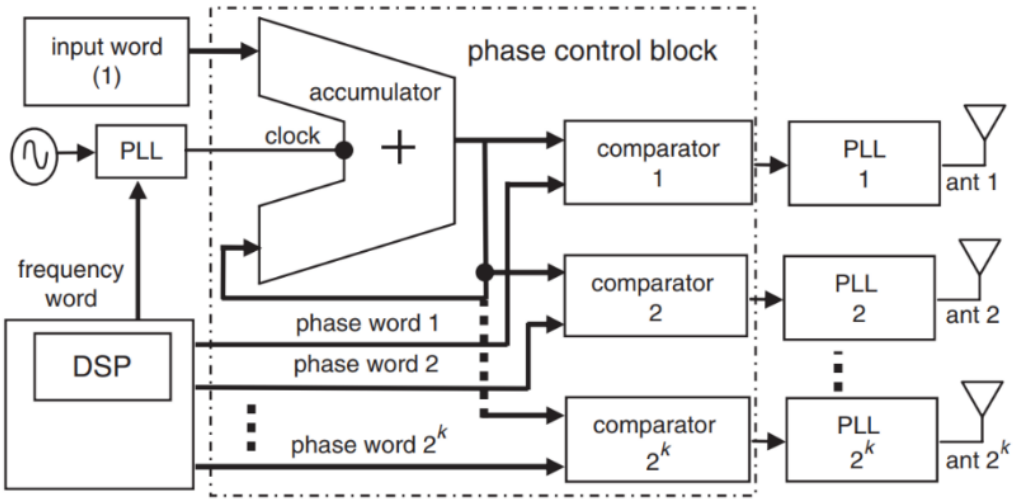


Figure 2.19 – Block diagram of the revised DDS-PLL architecture based on the accumulator register

The k-bit wide output of the accumulator register can be interpreted as a ramp signal whose frequency is $\frac{f_{CLK}}{2^k}$, and its 2^k different outputs are mapped to an equal number of phases. One comparator per antenna converts the above ramp signal into a square wave. The comparators are designed in such a way that a pulse is issued when the accumulator register matches a given phase word, without considering the most significant bits (Figure 2.20). This happens two times during each ramp period, and the above pulse is used to clock a type-D flip-flop. This means that the frequency of the square wave (namely the PCU output) is $\frac{f_{CLK}}{2^k}$. The comparison is performed through $k-1$ XNOR gates and an AND gate with $k-1$ inputs. The most significant bits of the two words are fed to the D input of the above flip-flop trough an XOR gate, in such a way that the polarity of the Q output of the flip-flop could be chosen accordingly to the above phase word.

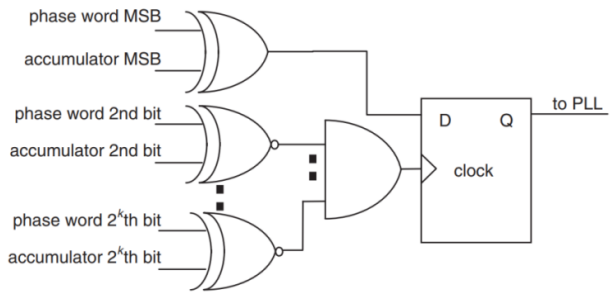


Figure 2.20 – Circuit diagram of the implemented comparators

Figure 2.21 depicts the above considerations when k is equal to 3 and the phase word is 011. In this example, the pulse is triggered when the output of the accumulator register is 011 or 111, with a respective output equal to 0 and 1 at the D-type flip-flop. For a phase word equal to 111, the pulse would have been triggered for the same outputs of the accumulator register, but the respective output at the flip-flop would have been inverted.

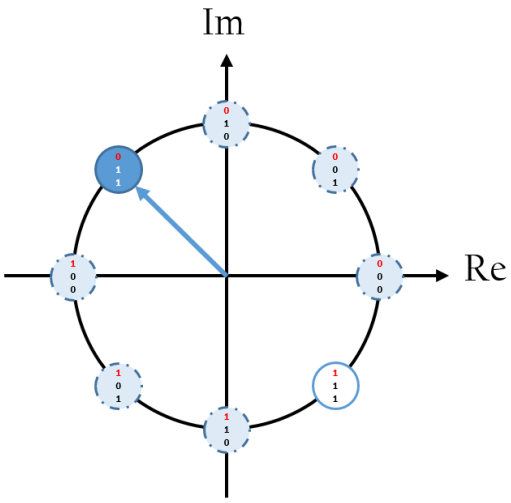


Figure 2.21 – Graphical representation of the comparators working principle

Figure 2.22 depicts the sampled output of a 4-bit wide accumulator register and the output of a comparator when the phase word is equal to 0000.

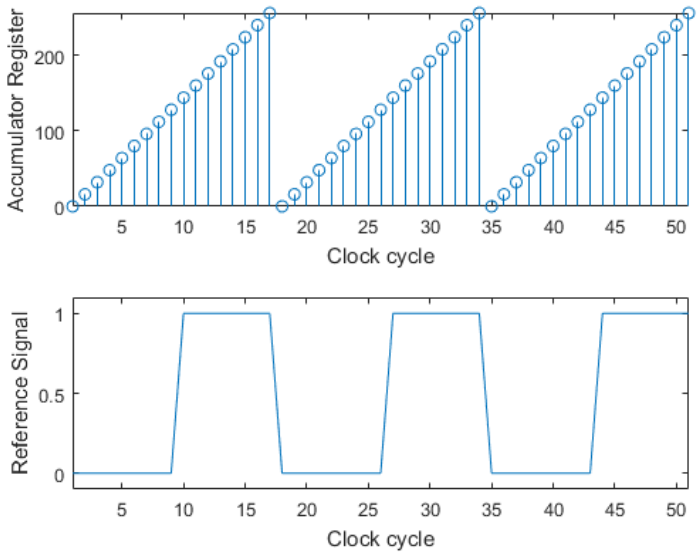


Figure 2.22 – Output of a 4-bit accumulator register and output of a comparator when phase word is 0000

Considering the duration of one period of the clock signal (T_{CLK}) assigned to the accumulator register and the duration of one period of the reference signal (T_{REF}) assigned to the PLLs, the minimum phase shift that can be assigned to the reference signals ($\Delta\varphi_{REF}$) is:

$$\Delta\varphi_{REF} = \frac{T_{CLK}}{T_{REF}} 2\pi$$

The corresponding phase shift $\Delta\varphi_{OUT}$ at the PLL outputs is:

$$\Delta\varphi_{OUT} = \frac{T_{CLK}}{T_{OUT}} 2\pi = N \frac{T_{CLK}}{T_{REF}} 2\pi = N \cdot \Delta\varphi_{REF}$$

where T_{OUT} is the duration of one period of the PLLs output sine waves. The above equations explicate that $\Delta\varphi_{OUT}$ is greater than $\Delta\varphi_{REF}$, and that its magnitude depends on N, that is the ratio between the input and output frequency of signals at the PLLs. This has already been discussed during the review of [16]. However, further considerations are now taken into account, in such a way that a technique to preserve the output phase resolution of the PCU at the PLL output is derived.

In DDS-PLLs (and their variants) the output of the PLL is locked to the one of the PCU, but its frequency can be N times higher when acting as a frequency multiplier. This means that the relationship between the PCU and the PLL output phase is not trivial, even if it depends only on the phase tuning word (PTW) assigned to the PCU. When N is greater than 1, the sequence of output phases becomes scrambled, following a law that is now illustrated. To this end, one can quantify the output phase difference due to the assignment of two different PTWs ($\Delta PTW = PTW_1 - PTW_0$):

$$\frac{2\pi}{2^k} \cdot N \cdot \Delta PTW = \frac{2\pi}{2^k} \cdot \beta + 2\pi x \qquad \text{Eq. 2.1}$$

where $2\pi/2^k$ is the phase resolution of the PCU, and $\beta \in [0 \dots 2^k]$ and x are integer numbers. The right side of Eq. 2.1 simply decomposes the obtained phase difference into two convenient terms, namely an effective phase difference and an integer number of turns around the phase wheel. Since this latter contribution can be neglected in the case of LO synthesis, one can rewrite Eq. 2.1 as follows:

$$\left(\frac{2\pi}{2^k} \cdot N \cdot \Delta PTW\right) \bmod 2\pi = \frac{2\pi}{2^k} \cdot \beta$$

For a generic design, the output phase difference of two consecutive PTWs can be greater than $2\pi/2^k$. However, this does not necessarily imply a degradation of the output phase resolution. In fact, when a given output phase ($\Delta\varphi_{out} = [2\pi/2^k] \cdot \beta$) has to be synthesized, the above equation merely needs to be inverted. For example, this can be done through a lookup table.

It is now shown how, in a DDS-PLL phase shifter, as long as N is an odd number, the output phase resolution is equal to the phase resolution of the PCU. Let's assume that $N = 2m + 1$ and that the effective output phase difference is 0 (namely that two PTWs would return the same effective output phase, and thus the phase resolution would not be maintained). Eq. 2.1 would become:

$$\Delta PTW = \frac{x}{(2m+1)} \cdot 2^k$$

Keeping in mind that ΔPTW must be an integer number (constraint A), that ΔPTW must be less than 2^k (constraint B), and that all the dividers of 2^k are powers of 2, the only possible solution for which the equation derived would respect the constraint A is x equal or multiple of $(2m + 1)$. However, this solution would violate constraint B, because ΔPTW would have to be equal to 2^k. In other words, two different PTWs cannot produce an effective phase difference equal to 0, therefore if N is an odd number the output phase resolution of a DDS-PLL is preserved. Let's now assume that $N = 2m$ and that the effective output phase difference is again 0. Eq. 2.1 would become:

$$\Delta PTW = \frac{x}{m} \cdot 2^{k-1}$$

Keeping in mind the previously defined constraints A and B, and that all the dividers of 2^{k-1} are powers of 2, there are two possibilities for which the above equation respects constraint A. The two possibilities are that x is equal to m with m being an odd number, or that m is an even number. If m is an odd number, two PTWs would return the same effective output phase, and as byproduct the output phase resolution is halved. If m is an even number, more than two PTWs would return the same effective output phase, depending on the greatest common divisor (GCD) of m and 2^{k-1}. The above considerations demonstrate that the output phase resolution (OPR) of a DDS-PLL is preserved if, and only if, N is an odd number, and that:

$$OPR = \frac{2^k}{GCD(N, 2^k)}$$

3 A revised DDS-PLL architecture for phased arrays

The purpose of this chapter is to present my attempt to decrease the complexity of DDS-PLL phase shifters through the implementation of a revised PCU where synchronous delay lines (SDLs) with programmable lengths (namely reconfigurable chains of memory elements) are used to replace full-featured DDSs (as well as the already discussed PCU variants). The proposed circuit topology is capable to introduce a given set of phase shifts among the PLLs reference signals employing a simple working principle that involves the management of just two signal paths: the actual signal that has to be phase shifted, and a clock signal fed to its memory elements.

3.1 SDLs with programmable lengths

The SDL architecture at the basis of the proposed PCU (Figure 3.1) is a sequential logic circuit that consists of 2^n D-Type flip-flops and n 2x1 multiplexers, organized into n delay blocks (where n is the desired phase shift resolution expressed in bits). The 2^n flip-flops are arranged into n shift registers, each one assigned to a delay block. The shift registers are made up of 2^k flip-flops, where $k \in [0;\ n-1]$ is the position that their corresponding delay block covers into the SDL. Each multiplexer is assigned to a delay block in order to route, at its output, the logic level at the input or at the output of its shift register, thus allowing to program the overall length of the SDL based on a PTW stored into a memory register. The output of each SDL is fed to a pipeline flip-flop, in order to mitigate phase errors related to the physical routing of signal paths assigned to each PTW configuration.

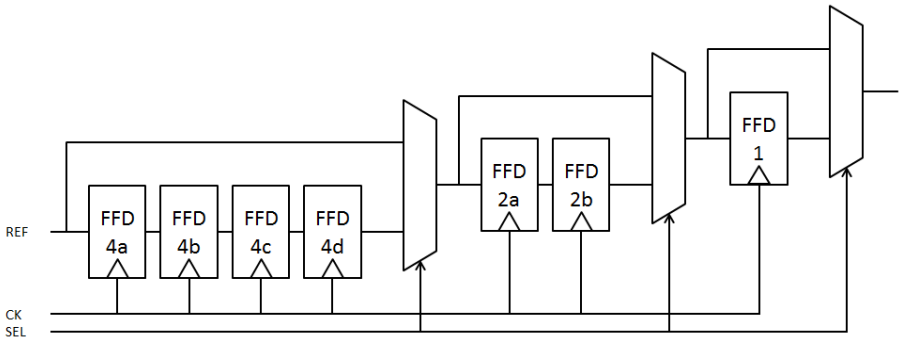


Figure 3.1 – Three shift registers chained into a SDL with programmable length (*k* equal to 2, 1 and 0)

From a z-domain perspective, the flip-flops in the SDL implement z^{-1} blocks, since each of them delays the square wave reference signal at the SDL input (REF) by one period of the clock signal assigned to it (T_{CLK}). In order to synthesize phase shifts in the [0°; 360°] range, T_{CLK} must be related to the period of the input square wave signal (T_{REF}) as follows:

$$T_{CLK} = \frac{T_{REF}}{2^n}$$

Given that a non-trivial implementation requires n to be greater than one, f_{CLK} is always equal or more than twice f_{REF}, and thus the Nyquist-Shannon sampling theorem requirements are always met by the system. It seems interesting to observe that phase shifts and time delays are equivalent concepts when considering the ideal representation of square wave signals.

A PCU based on this approach needs one SDL for each PLL of the BSU. Figure 3.2 depicts the block diagram of a 4 channel PCU based on SDLs with programmable lengths. In the block diagram, the SDLs integrate a parallel-input parallel-output (PIPO) register, in

such a way that the PTWs that must be assigned to the selection inputs of each multiplexer can be latched. This means that the I/O inputs of the PCU are multiplexed among the SDLs.

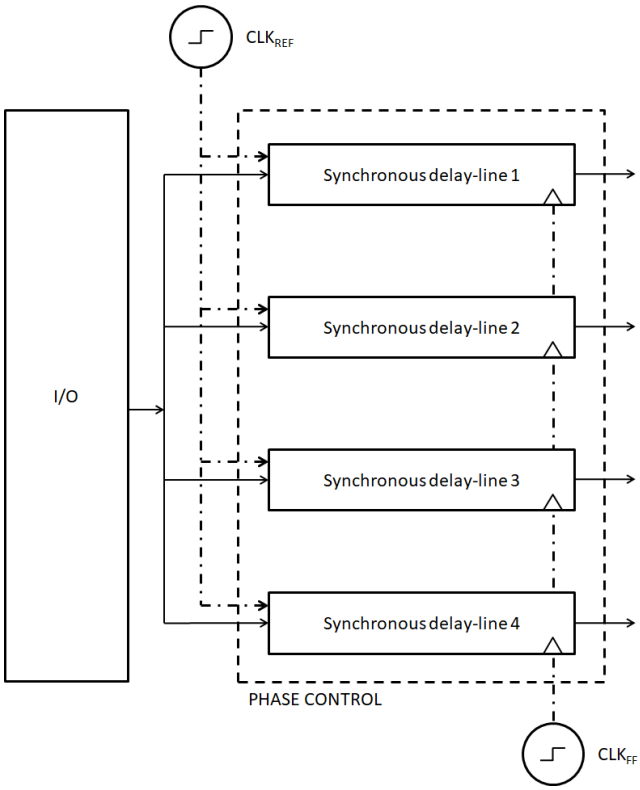


Figure 3.2 – Block diagram of a 4 channel PCU based on SDL with programmable lengths

This PCU architecture has been described in VHDL, then synthesized and compiled for FPGA. Considering the fact that the block in position $n-1$ merely implements a 180° phase shift, an XOR gate (acting as a controlled inverter) has been used in its place, saving 2^{n-1} flip-flops and one multiplexer. The PCU has been designed to provide a phase shift resolution of 8-bits (corresponding to a phase tuning step as low as 1.40625°) on four

independent output channels. This means that the overall PCU is made up of 4 SDLs with n equal to 8. The target FPGA device was the Altera EP4CE225F29C7 (114,480 logic elements). The synthesis process has been performed in Quartus II 14.1. The FPGA usage (Table 3.1), in terms of Logic Elements (LEs), reported in the compilation report is less than 1%.

	IP	**LEs**	**Registers**	**Memory**
This work	PCU	117 (< 1%)	79	448 (< 1%)

Table 3.1 – Compilation report

Figure 3.3 depicts the RTL netlist view of the PCU. The four 8-bit PIPO registers (named "bypass_c") are made up of 8 flip-flops. These registers are loaded through 8 inputs (named "bypass_ex") of the VHDL component and are selectively enabled through 4 other inputs (named "bypass_s"). The outputs of each PIPO register are routed to the 8 multiplexers of its assigned SDL (named "i_delay_block_c"), in such a way that the desired delay is independently implemented at each channel. The 4 outputs (named "output") of the VHDL component are available for further assignment operations.

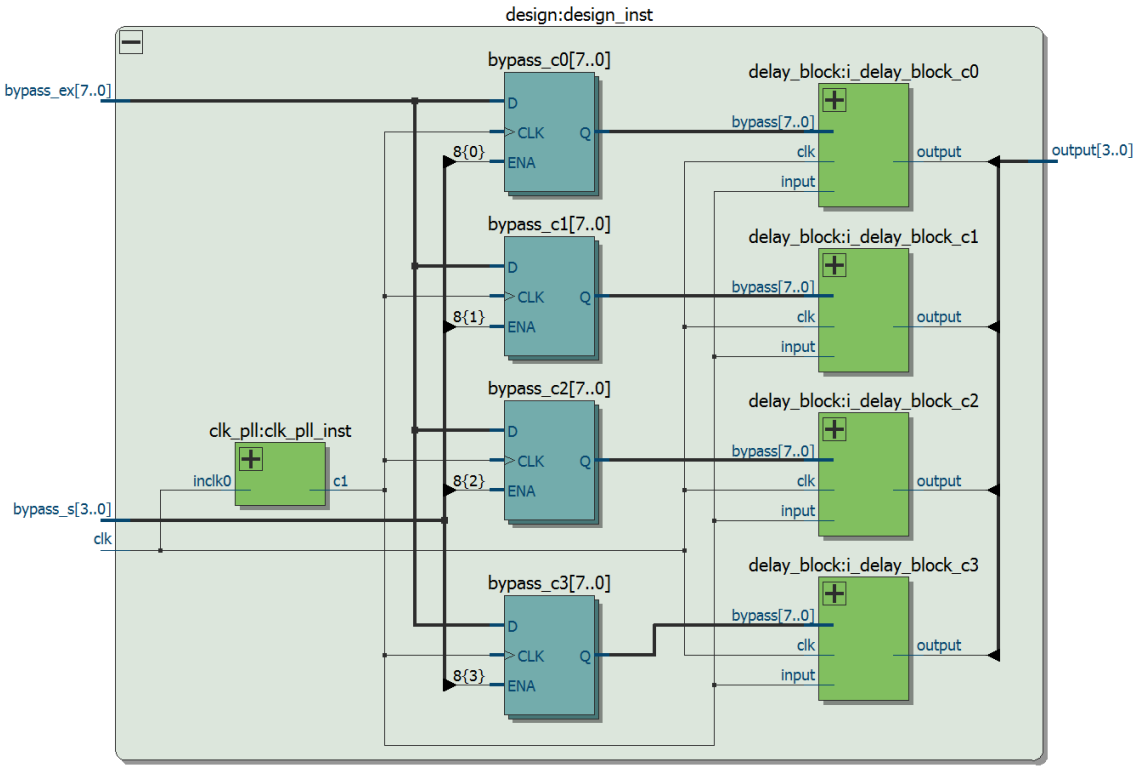


Figure 3.3 – RTL netlist view of the PCU entity

Figure 3.4 depicts the RTL netlist view of one of the SDL entities. As already mentioned, the rightmost flip-flop is used to pipeline the output of the SDL in order to force its alignment to the edges of the clock signal. Best performance is achieved when this clock signal is ultra-clean (e.g. is generated from an external clock cleaner circuit), whereas the reference clock signal can be derived from it (e.g. through one of the all-digital PLLs integrated into the FPGA [19], dividing its frequency by 2^n). This is because the clock signal assigned to the flip-flops is the one that ultimately determines the output characteristics in terms of jitter performance.

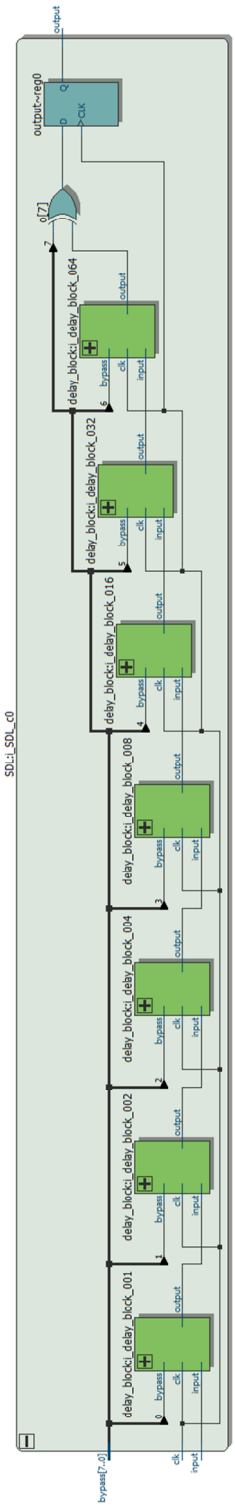


Figure 3.4 –RTL netlist view of one of the SDL entities

Figure 3.5 depicts the RTL netlist view of a delay block ($k = 7$). The number of flip-flops in the shift register constitutes the only difference among the various instances of this entity chained into the SDL.

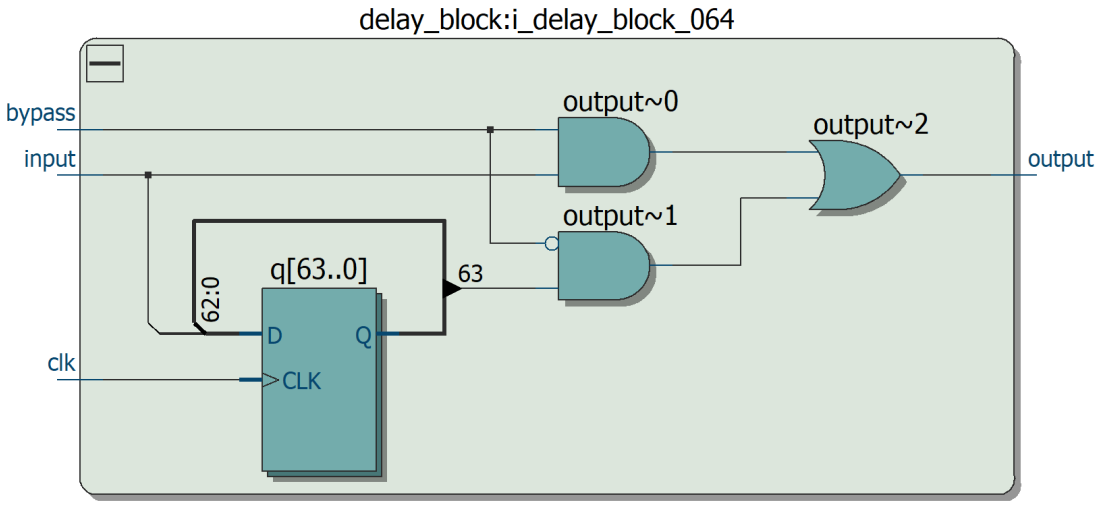


Figure 3.5 – RTL netlist view of a delay block ($k = 7$)

The chosen FPGA device supports working with f_{CLK} and f_{REF} frequencies respectively equal to 256-MHz and 1-MHz (among the other possible configurations). These have been the ones chosen during the development of the prototype, and that will always be used in the following discussion. Once again, the frequency of the reference signals fed to the PLLs is 1-MHz, and that the frequency of the clock signal assigned to the memory elements is 256 times faster than it. This is done in such a way that the reference signals can be delayed with an 8-bit resolution, namely in 256 steps of their period.

The above discussed PCU, implemented as a VHDL component, can be integrated to other VHDL entities to prototype a full-featured digital subsystem into the FPGA. For example, the integration of a microcontroller unit (MCU) may allow to control the PCU through a convenient firmware routine and a well-known serial interface, as well as to execute other general-purpose code. This form of integration, when extended to the analog and RF subsystems of a BSU, can lead to a single IC solution that follows the auspicated characteristics of SoCs for phased arrays presented in Chapter 1.

3.2 A BSU architecture for phased arrays

After designing the PCU, my research focused towards the implementation of a BSU prototype. This has been done through the acquisition of an 8051 IP core, and the design of a frequency scaling unit (FSU), namely the PLL subsystem that synthesizes phase shifted LOs starting from the PCU-generated reference signals. The integration of an MCU as a BSU component seems desirable for a wide class of applications that require firmware code execution along with the enhanced connectivity given by phased arrays of antennas. This is even more evident if considering the complexity of modern communication protocols, and how the management of their entire stack into a SoC solution could be beneficial in terms of space occupancy and net weight.

Figure 3.6 presents the block diagram of the BSU, designed for an array of four antennas.

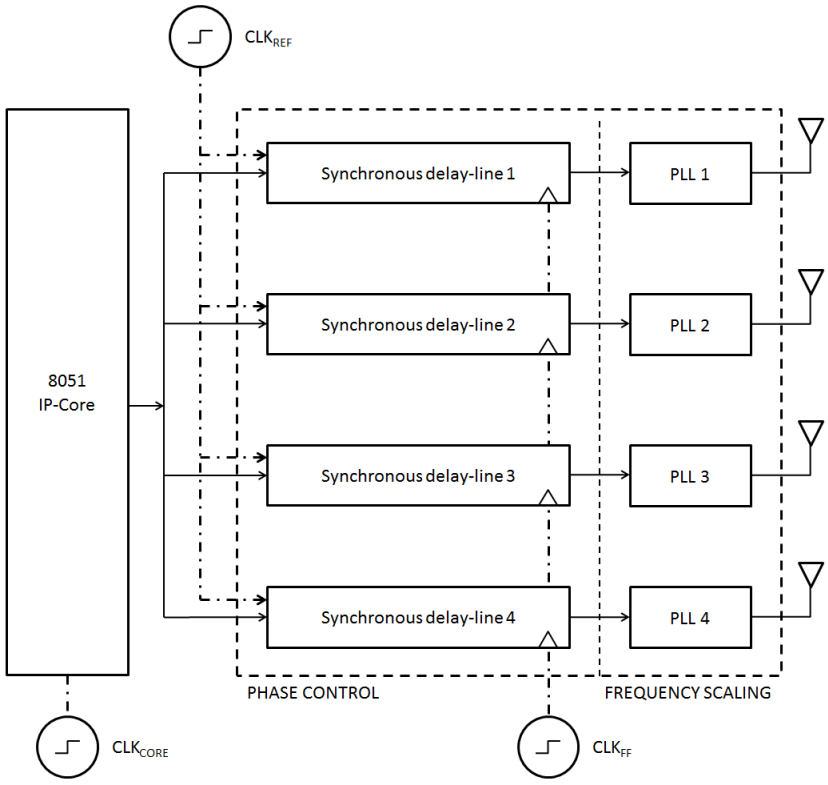


Figure 3.6 – Block diagram of the BSU

3.2.1 8051 IP core

Focusing the discussion on the full-digital portion of the above architecture, it has been designed to be compatible with FPGA and ASIC design flows, and it has been synthesized and compiled for the previously mentioned FPGA IC. Instead of using an external MCU or one of the powerful microprocessor IP cores from FPGA vendors (that cannot be compiled to standard cell-based ASIC layouts, since they are not intended to be used outside their proprietary development environment), the choice has been an open-source 8051 IP core

from Oregano Systems, distributed under the "GNU Lesser General Public License" (LGPL).

The IP core is provided in plain VHDL, thus it can be synthesized for both FPGAs and standard cell based ASICs from major foundries. This means that the FPGA-based design that is being discussed can be ported to ASIC following the standard digital design flow. The IP core is binary compatible to the well-known 8051 processor from Intel and is supported by many toolchains and integrated development environments (IDEs) for firmware compilation. For example, the IDE that has been used to compile the firmware codes for this work is Keil μVision 5. Many virtual prototyping environments are also available for its architecture. The 8051 is extremely popular due to its historical importance and its continuous reimplementation in modern ICs. It is based on the Harvard architecture, thus it has two separate storage and signal paths for instructions and data. Being an 8-bit processor, it can read, write and process words of data whose length is 8-bit. For a deeper discussion on the 8051 please refer to [20].

The IP core has been configured to feature two timer/counter units, one serial communication interface (SCI) and two external interrupt sources. It has been linked to a 128 bytes internal RAM, an 8192 bytes external RAM and an 8192 bytes ROM. In this work, the clock frequency assigned to its core (f_{CORE}) is 25 MHz, that is the maximum speed the IP core is rated for. Ten of its GPIOs are used to configure the PCU, whereas other 12 GPIOs are used to configure the FSU. The SCI is used to interface a host PC through the RS232 level shifter present on the development board.

A VLSI implementation of an application processor based on this chosen IP core has been reported in [21] by Chu et al. (2015) for a TSMC 0.18μm technology, occupying a die area of 1.96-mm^2. In their work, the authors integrated the IP core with an in-system programming (ISP) mechanism, in order to make it independent from specific tooling for its burn-in phase. This feature is not present in the original 8051 microprocessor and constitutes an example of a research effort that is focused on improving a state of the art IP and silicon-proving its final implementation.

It seems interesting to recall that typical digital ASIC design flows are heavily based on CAD tools capable of: i) compiling hardware description language code (e.g. VHDL files) into a tree of interconnected standard cells from a chosen process design kit (PDK); ii) allowing the user to prepare a floorplan for the die area and partition his design; iii) placing cells under an optimized strategy; iv) routing an optimized clock tree as well as the rest of the interconnections. Of course, the goal of the expert designer is to conduct the CAD tool towards the best implementation in terms of functional and non-functional requirements (e.g. yield). This is done by generating a proper set of constraints (e.g. related to timings) that the system can tolerate, leading to a rich set of inputs (from which the so-called design space is derived) for the subsequent optimizations that are conducted by the tool.

The original 8051 has been reimplemented by Oregano Systems obtaining the following block diagram (Figure 3.7).

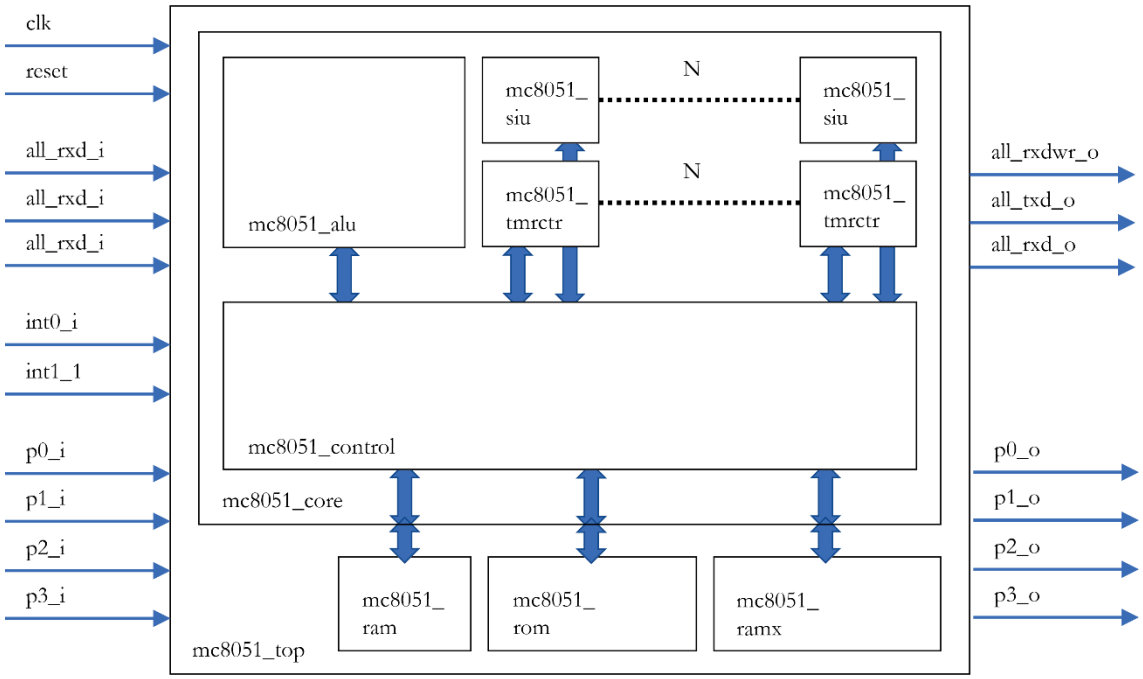


Figure 3.7 – 8051 architecture implemented by Oregano Systems

The main difference between the two implementations is that the one from Oregano Systems is optimized to reduce the number of cycles needed to perform certain instructions. Thus, even if the instruction set is the same, code execution can be different in the revised implementation. For this reason, special care must be taken when implementing firmware code, especially when generating delays through the execution of routines made up by dummy code.

The IP core is partitioned into the following hierarchy diagram (Figure 3.8), which depicts the relationship among the design units.

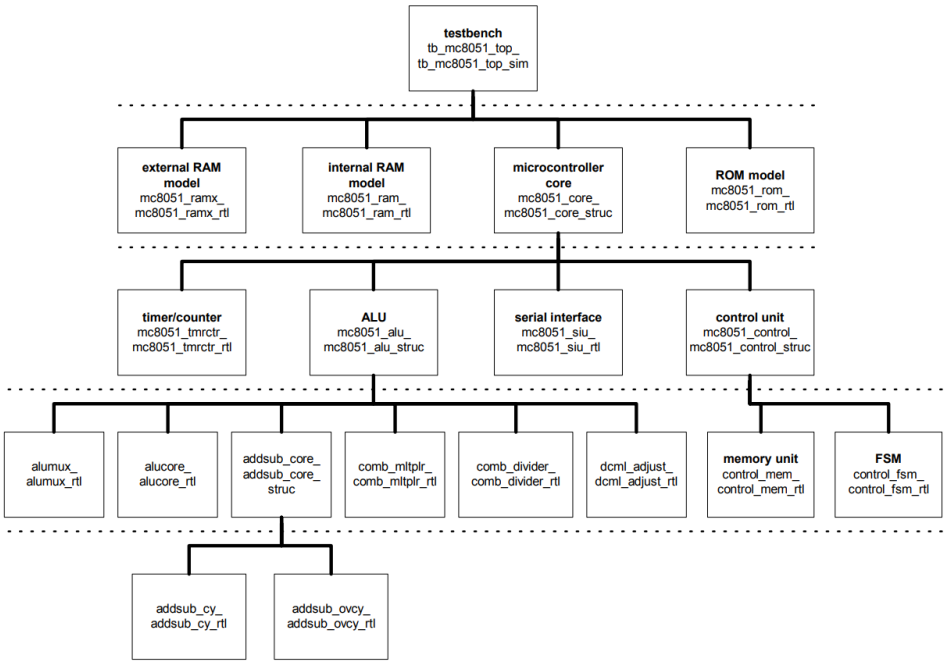


Figure 3.8 – Hierarchy diagram of the IP core architecture

It seems important to remind that this hierarchy must be carefully followed when preparing the project for compilation. In fact, the diagram explicates dependencies among files, and thus those units that must be declared before others make reference to them. The mc8051_core unit is the actual block distributed by Oregano Systems whereas the mc8051_ram, mc8051_rom and mc8051_ramx are instances of third-party memory IP cores (e.g. from the FPGA vendor) that are to be bound to the rest of the design.

Table 3.2 summarizes the main parameters extracted from the compilation report (performed in Quartus II 14.1) of the the full-digital portion of my proposed BSU architecture, that is the top-level entity of the FPGA design.

	IP	LEs	Registers	Memory
This work	MCU + PCU	4,261 (4%)	602	198,080 (5%)

Table 3.2 – Compilation report

The RTL netlist view of is depicted in Figure 3.9. Three GPIO ports of the 8051 IP core. Twelve GPIOs have been used as outputs to program the PLLs whereas other 12 GPIOs have been bound to the PCU component for its control.

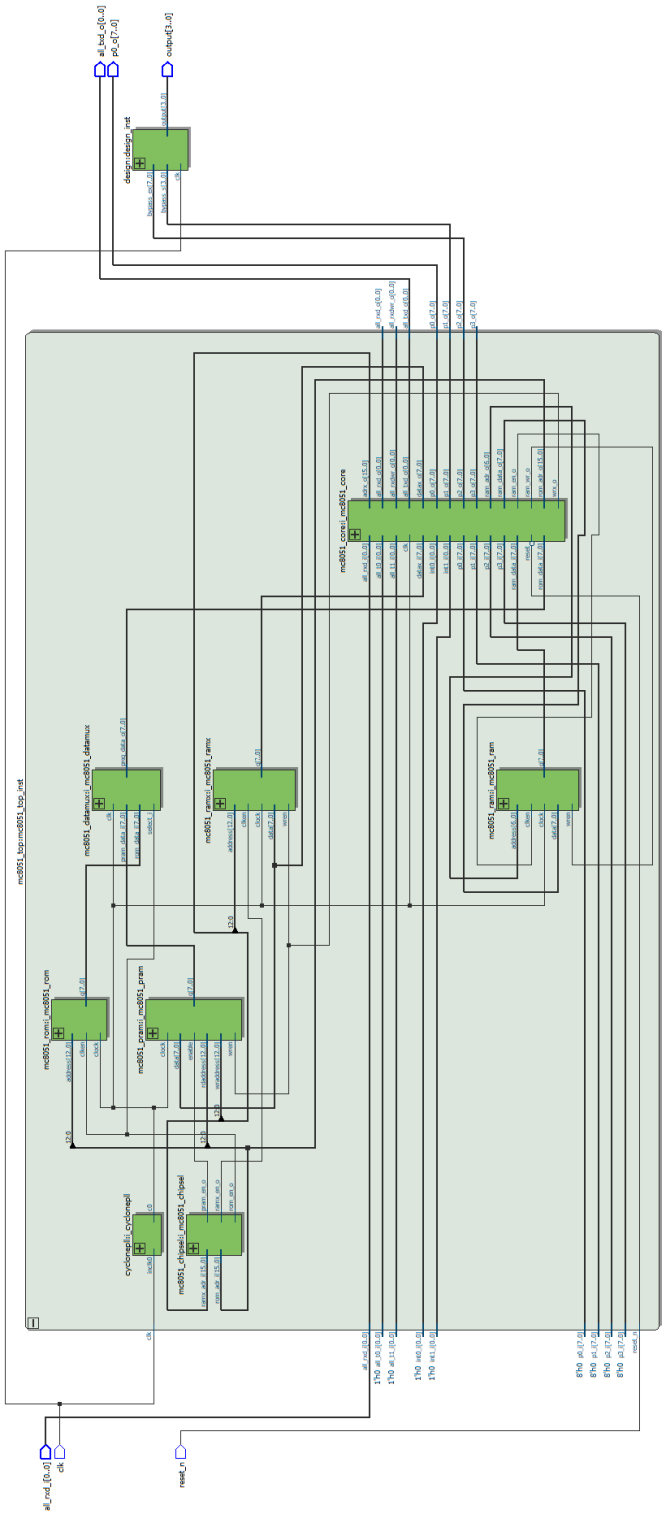


Figure 3.9 – RTL netlist view of the full-digital portion of my proposed BSU architecture

The above hardware runs firmware codes compiled into binary files, used to setup the content of the system ROM IP during its instantiation. For example, a custom firmware used to support the following qualification process of the BSU in an automated test environment has been implemented. The high-level flow chart of this firmware is depicted in Figure 3.10, and comprises a setup procedure to initialize peripherals (such as the serial communication interface, SCI) and two distinct procedures, one to configure the PLLs [22] in the FSU and one to store new PTWs into the PCU. These procedures are executed when convenient ASCII commands are sent from a host computer to the MCU through its SCI.

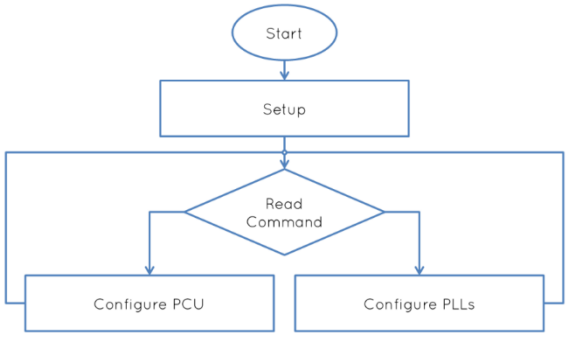


Figure 3.10 – High-level flow chart of the firmware code

The inputs and the outputs of the top-level entity have been bound to actual pins of the FPGA device, to interface the hardware to the real-world.

3.2.2 FSU design

Four PLLs centered at 2.453-GHz (optimized for reference signals of 1-MHz) make up the FSU. The PLLs are based on the ADF4118 from Analog Devices and the VCO190-

2453TY from Sirenza. The digital supply voltage has been set to 3-V whereas the charge pump and VCO supply voltage has been set to 5-V. The PFD frequency has been set to 1-MHz (thus the reference signal does not undergo any frequency division). Being the ADF4118 an Integer-N PLL chip, this prototype can synthesize sine waves equally spaced in the frequency domain that are 1-MHz apart from one other (due to the above PFD frequency configuration). The loop bandwidth (LBW) has been set to 10-kHz through a passive second-order loop filter. This bandwidth value has been chosen as an adequate trade-off between phase jitter and frequency and phase locking time, respectively 0.43° RMS and 333-µs (simulated in ADIsimPLL). Table 3.3 presents the simulated time-to-lock (TTL) at 10-Hz, TTL at 1°, and phase jitter for various LBWs (1-kHz, 10-kHz, 20-kHz) implemented through passive second-order loop filters. Figure 3.11 depicts the transient analysis of the output frequency and of the phase error for the same above listed LBWs.

LBW	**TTL at 10-Hz**	**TTL at 1°**	**Phase Jitter**
1-kHz	9.74ms	9.63ms	0.11° RMS
10-kHz	377µs	333µs	0.43° RMS
20-kHz	188µs	145µs	0.75° RMS

Table 3.3 – Frequency and phase TTL

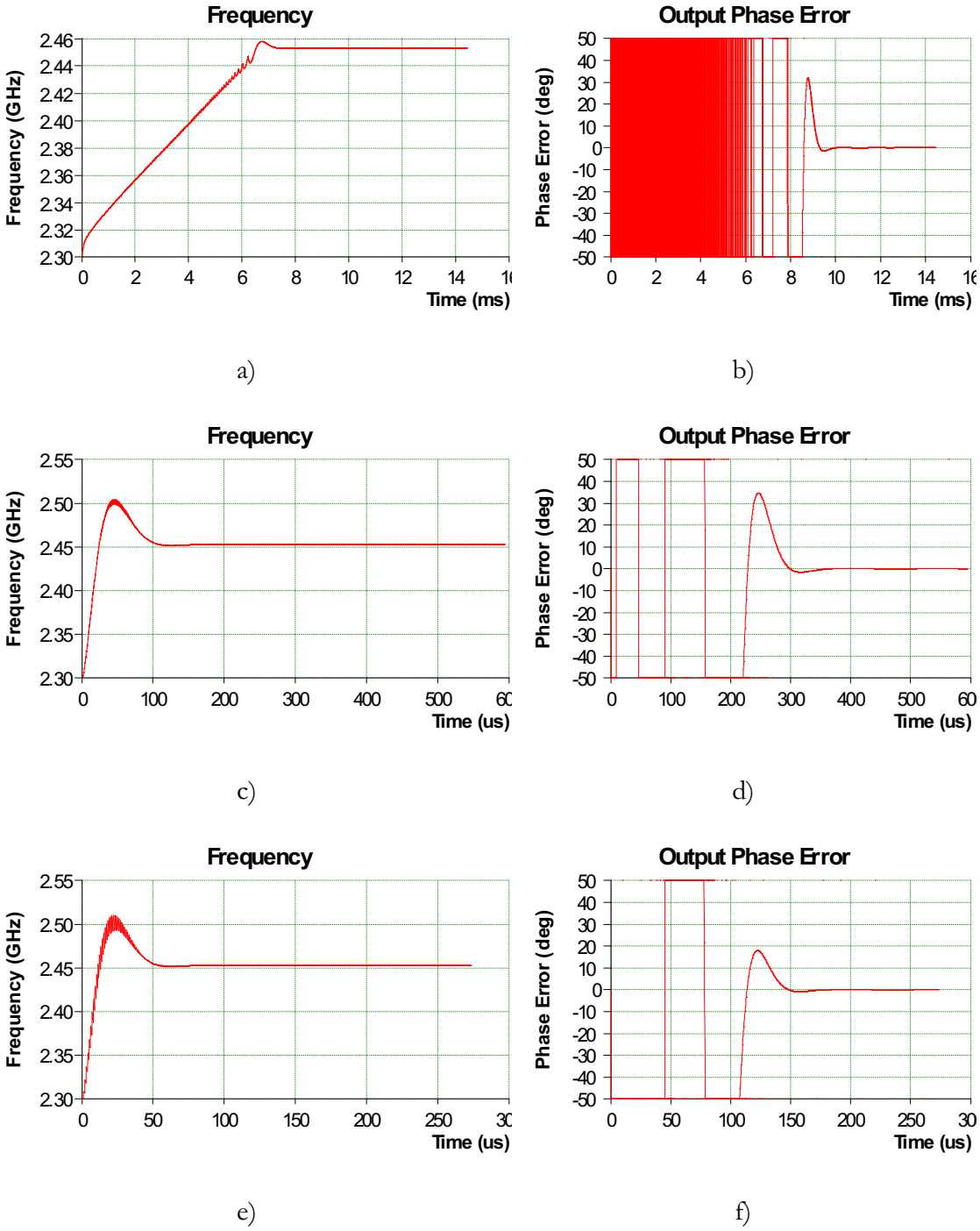


Figure 3.11 – Transient analysis of output frequency and phase error for various LBWs (1-kHz, 10-kHz, 20-kHz)

The FSU has been designed to be powered through an external power supply (5.5-V to 12-V), and each channel is equipped for independent power management through two linear voltage regulators. It accepts the four reference signals for its PLLs from a header or from SMA connectors. Figure 3.12 depicts the top view of the FSU. The size of the board is 240-mm x 100-mm and its thickness is 0.8-mm. The board has been designed on 2 layers. The bottom layer is used as a ground plane and for the routing of supply voltages. The distance between the outputs is equal to 6.115-cm, namely $\frac{\lambda}{2}$ (where λ is the wavelength of the LO frequency, that is 2.453-MHz, in free space). The layout has been carried out through the implementation of a sub-circuit that has been replicated four times. The traces handling high-frequency signals have been designed as microstrips in Advanced Design System (ADS) from Keysight Technologies, with an impedance matched to 50-Ω. It must be noted that the most critical transmission line is the feedback path from the VCO to the ADF4118 IC, if considering how short the length of other transmission lines is.

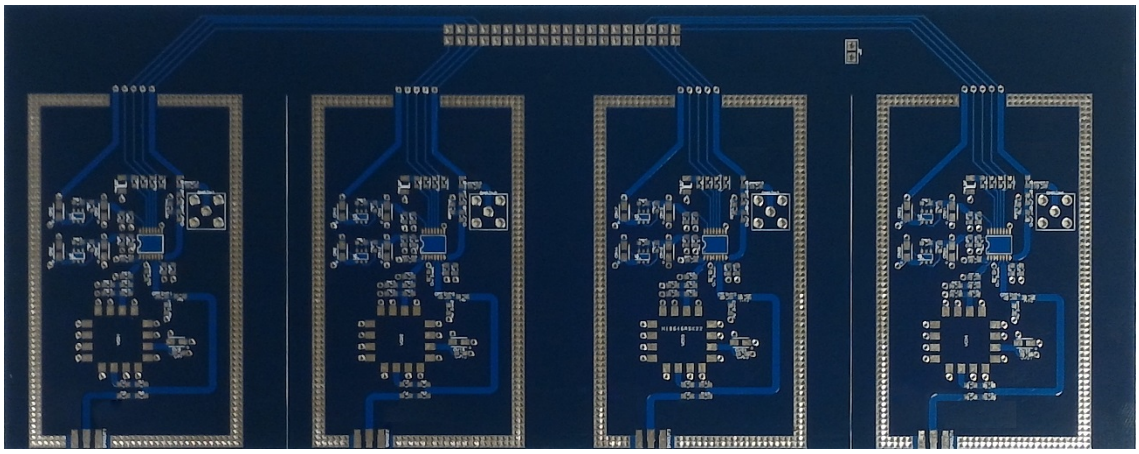


Figure 3.12 – Top view of the FSU

3.3 Performance evaluation

It is now presented the performance evaluation of the BSU architecture developed during this research. This has been done through experimental measurements (performed in time- and frequency-domain) of phase shifts among the signals at the output channels of the PCU and of the FSU. Preliminary beam steering performance are also evaluated and will further be discussed in Chapter 4, where a beam steering transmitter prototype and a more in-depth examination of the measurement process will be presented.

3.3.1 Measurement setup

Figure 3.13 and Figure 3.14 depict the block diagrams of the measurement setups employed during the time domain qualification of the PCU and FSU outputs. The PCU output signals are square waves, characterized by a frequency of 1-MHz, a peak-to-peak amplitude of 3-V and a mean amplitude of 1.5-V. The FSU output signals are high-frequency tones, characterized by a frequency of 2.453-GHz and a typical output power level of 3-dBm (output stages matched to 50-Ω loads). The above setups are based on a 4-GHz oscilloscope (LeCroy HDO9404) configured for a 20-GSPS sampling rate in real-time. Phase shifts have been quantified on digitized signals.

Figure 3.13 – Measurement setup diagram (time domain, PCU outputs)

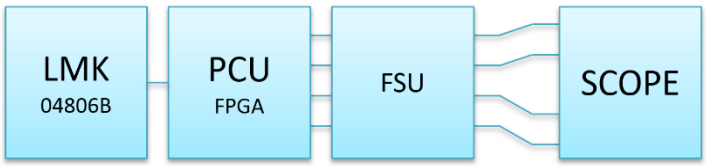


Figure 3.14 – Measurement setup diagram (time domain, FSU outputs)

In order to improve the consistency of the phase shift measurements, the coaxial cables used to implement the interconnections towards the oscilloscope are semi-rigid (6-inch coaxial cables with 50Ω characteristic impedance and SMA connectors). This is because semi-rigid cables are meant to keep their shape, and thus reduce random phase changes related to accidental movements. However, improper bends must be avoided when using semi-rigid interconnections, since they are sources of stress for the material that may induce phase drifts over time. This is an undesirable event that can ruin the evaluation of a high-resolution system such as the one under test. The above setups allow to perform a complete test of the PCU and of the FSU output signals by acquiring 256 phase difference measurements among the 6 possible output couples (since the prototype is characterized by an 8-bit resolution and 4 outputs). A total number of 1536 samples have to be collected. This explains the need for an automated test bench assisted by the 8051 IP core. The automated test bench uses a MATLAB script running on a host computer to orchestrate the measurement process. The script uses GPIB commands over TCP/IP to communicate with the oscilloscope, as well as ASCII commands over RS232 to communicate with the 8051 IP core. GPIB commands are used to acquire measurement results after a fixed amount of time has passed. ASCII

commands are used to invoke the PLLs configuration routine or to change the PTWs stored in the SDLs PIPO registers.

Figure 3.15 depicts the block diagram of the measurement setup employed during the preliminary frequency domain qualification of the beam steering performance. The setup is based on an EXA vector signal analyzer (Keysight Technologies N9010A) configured for 1001 points acquisition over a span of 100-KHz with a resolution bandwidth of 47-KHz and a video bandwidth of 47-KHz. The received signal strength (RSS) is evaluated, in a clean RF environment, on the received waves propagating in free space and originated by the BSU.

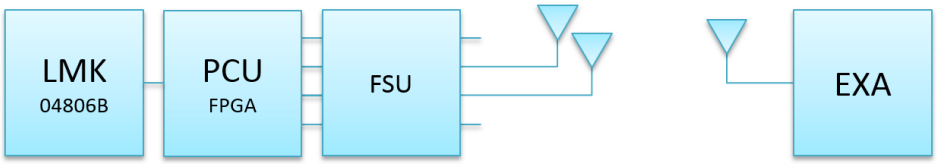


Figure 3.15 – Measurement setup diagram (frequency domain, FSU outputs – 2 channels)

The EXA signal analyzer input was connected to a patch antenna working at 2.4-GHz with a characteristic impedance of 50-Ω. A linear array of patch antennas was instead connected to the outputs of the FSU (Figure 3.16). The antenna design has been conducted in CST Studio Suite (in Chapter 4 the design and simulation of a variant centered at 3.35-GHz will be discussed). This has been done designing the patch, and then replicating its structure four times on the same substrate with λ/2 spacing.

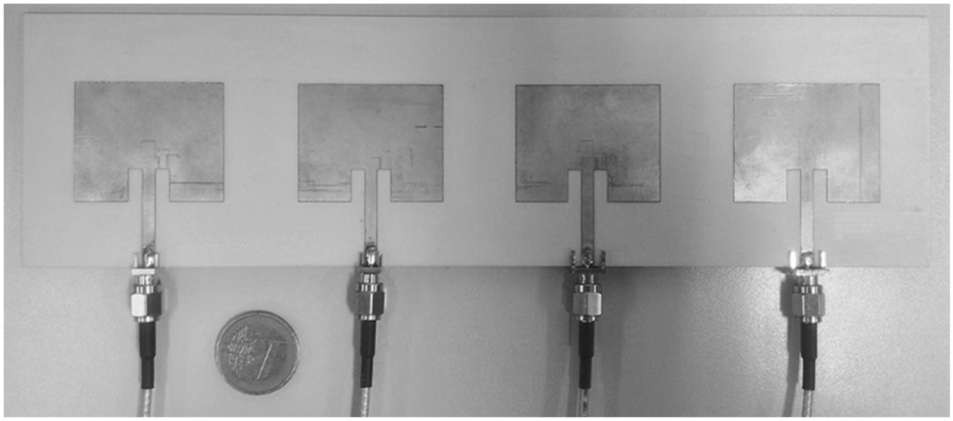


Figure 3.16 - Linear array of patch antennas connected to the outputs of the FSU

The RSS has been computed from the peak amplitude of the tone received when adjacent channels were powered on in couples (and placed at an equal distance from the receiver). The distance between transmitters and receivers during measurements respected the far-field condition, governed by the following equation:

$$R_{far} = \frac{2D^2}{\lambda}$$

where D is the aperture of the array. The above setup allows to perform a complete test of the FSU by acquiring 256 RSS measurements among the 3 adjacent output couples (since the prototype is characterized by an 8-bit resolution and 4 outputs). This means that up to a total number of 768 samples can be collected. The automated test bench follows the same scheme as the previously introduced one, with the only difference that GPIB commands over TCP/IP are used to communicate with the EXA signal analyzer to acquire peak power at the LO frequency.

3.3.2 Measured performance of the PCU

Figure 3.17 to Figure 3.19 depict the results of the conducted performance measurements at the PCU outputs, in terms of PTW to output phase trans-characteristic, phase error and standard deviation on the acquired samples. Measurements have been averaged over 100k periods.

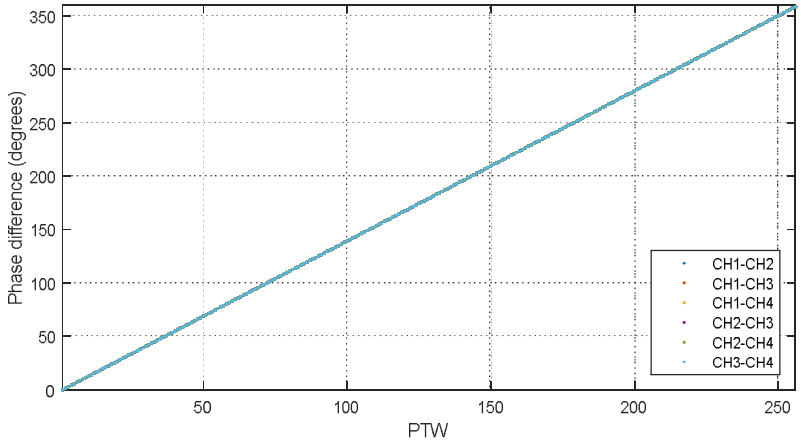


Figure 3.17 – PTW vs. phase difference trans-characteristic (PTW=0 to PTW=255)

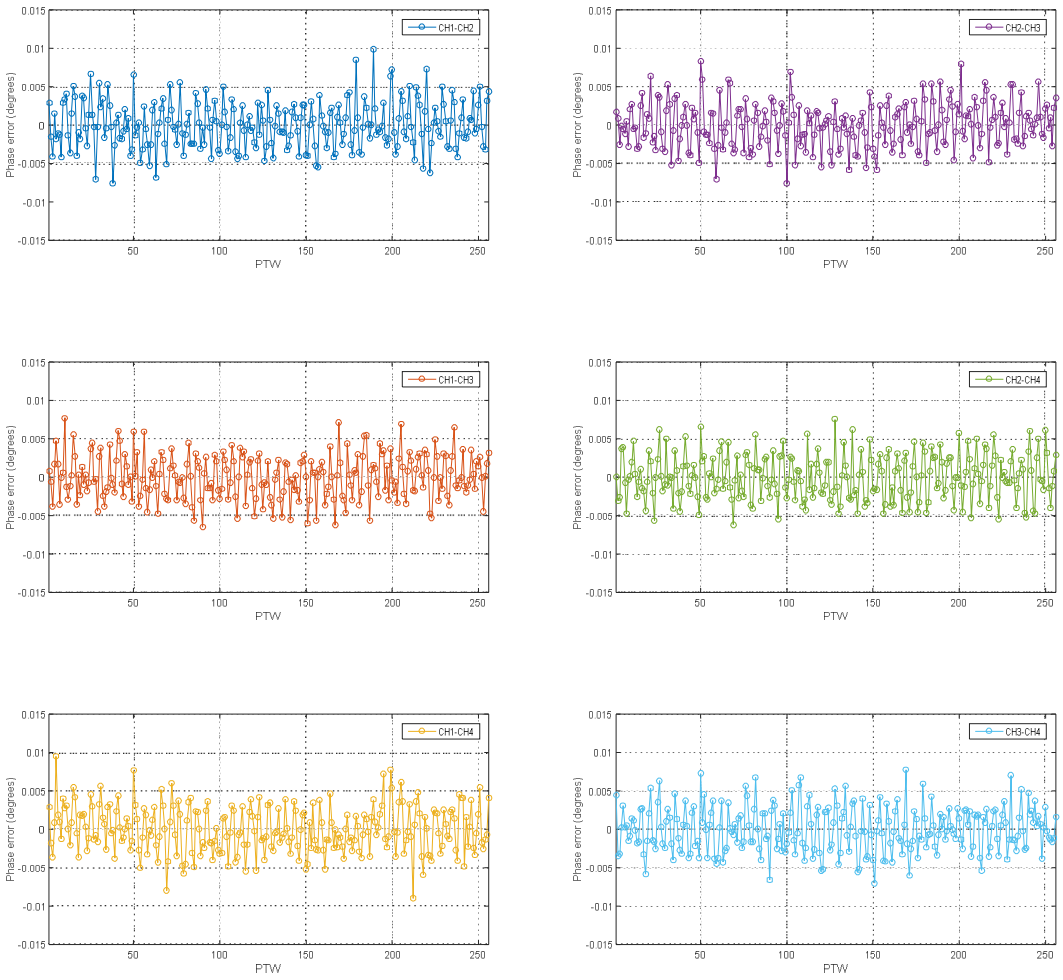


Figure 3.18 – PTW vs. phase error for CH3-CH4 (PTW=0 to PTW=255)

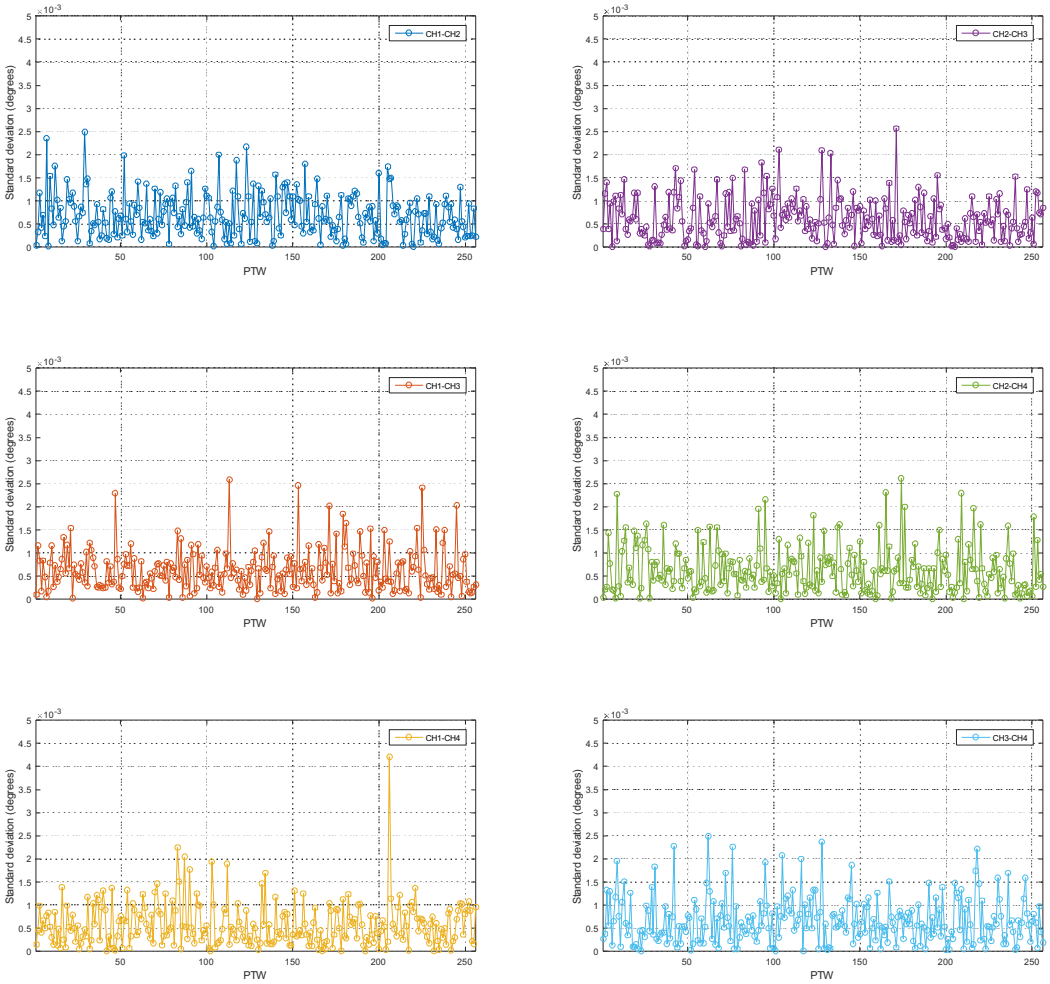


Figure 3.19 – PTW vs. standard deviation for CH1-CH2, CH1-CH3, CH1-CH4 (PTW=0 to PTW=255)

The mean phase error across PTWs is proximal to 0, and the mean standard deviation in the acquired phase difference samples is less than 0.0025°. The phase error maintains well within the range [-0.01°; +0.01°].

3.3.3 Measured performance of the FSU (coaxial cable)

Figure 3.20 to Figure 3.22 depict the results of the conducted performance measurements at the FSU outputs, in terms of LUT address to output phase trans-characteristic, phase error and standard deviation on the acquired samples. Measurements have been averaged over 100k periods.

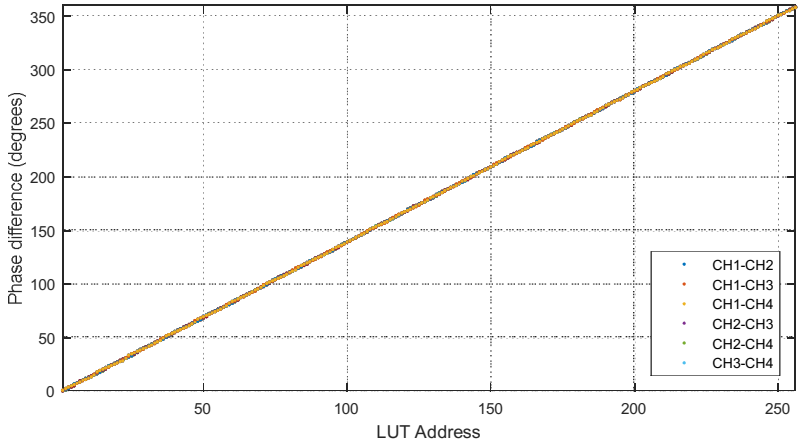


Figure 3.20 – LUT address vs. phase difference trans-characteristic (LUTa =0 to LUTa =255)

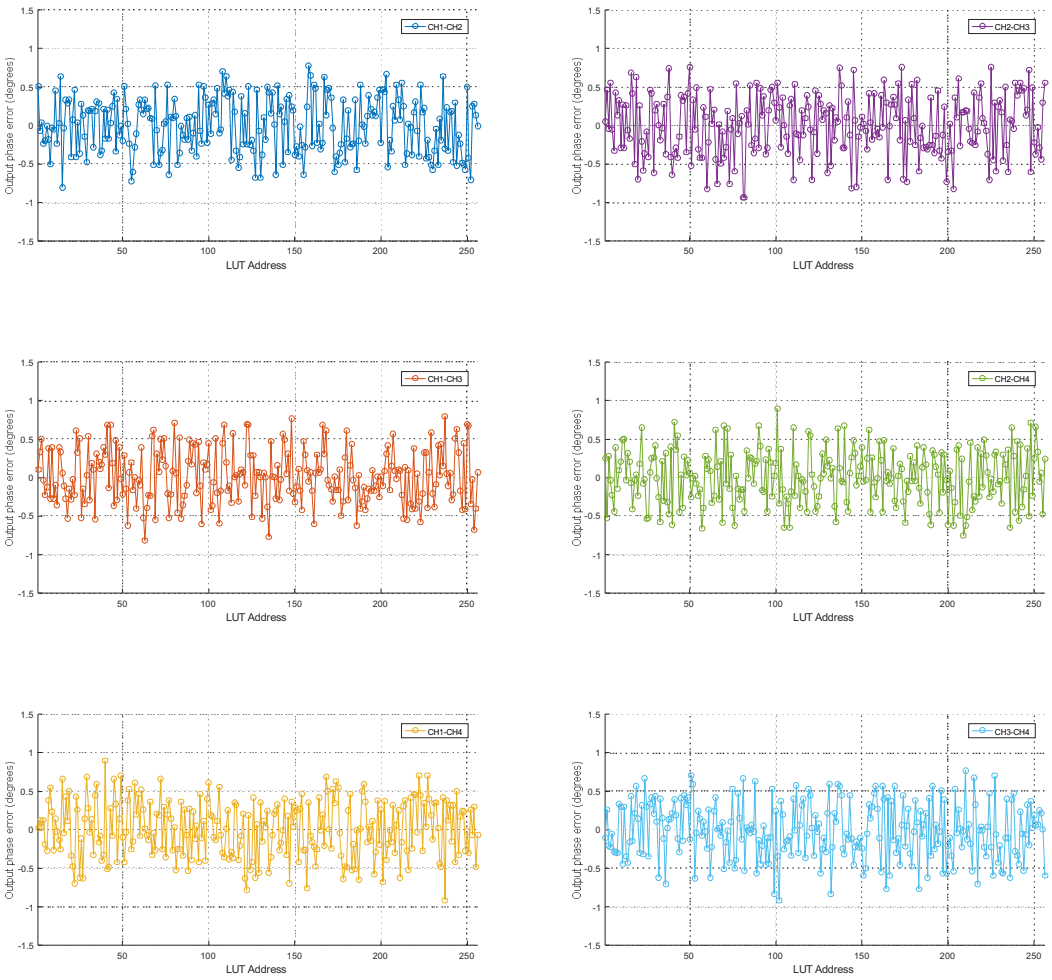


Figure 3.21 – LUT address vs. phase error for CH1-CH2 (LUTa =0 to LUTa =255)

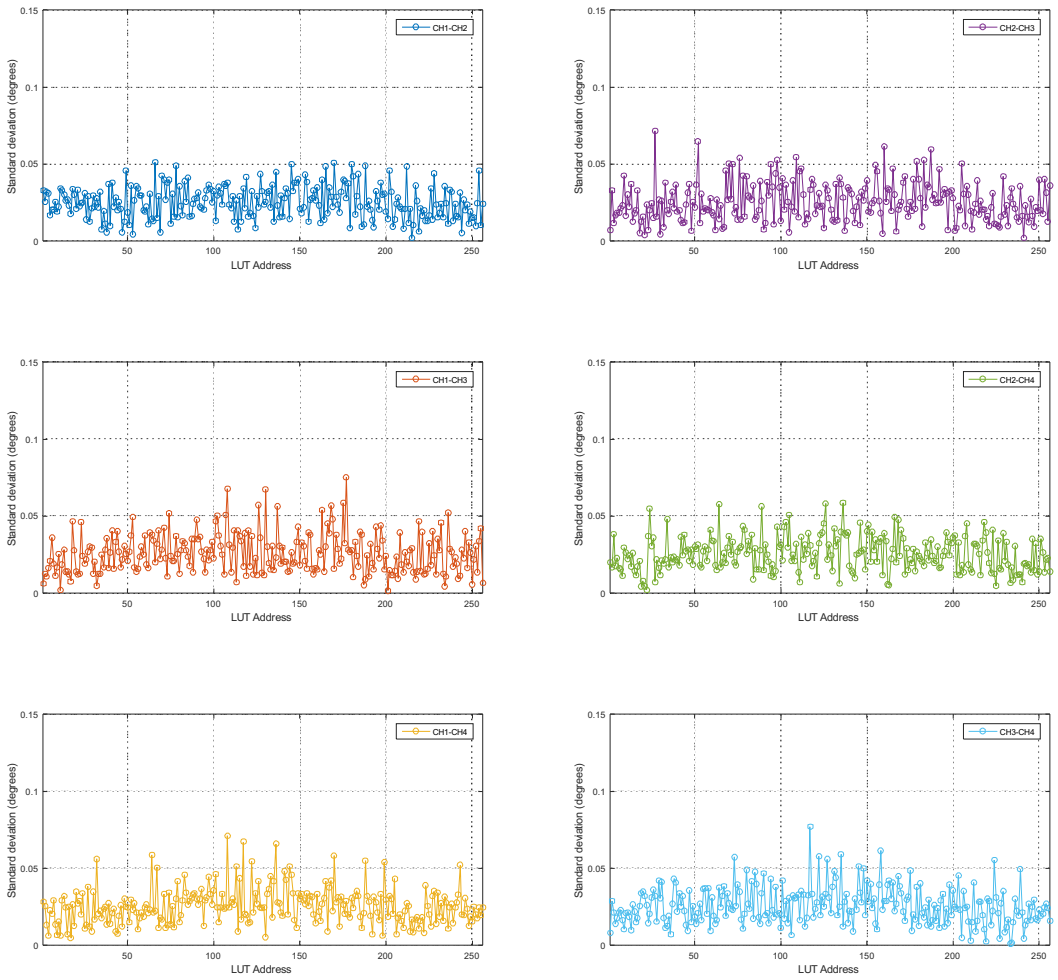


Figure 3.22 – LUT address vs. std. deviation for CH1-CH2, CH1-CH3, CH1-CH4 (LUTa =0 to LUTa =255)

The mean phase error across LUT addresses is proximal to 0, and the mean standard deviation in the acquired phase difference samples is less than 0.01°. The phase error maintains well within the range [-0.9°; +0.9°]. The FSU phase error depends on the PCU phase error and on other sources of random phase noise, such as the VCO phase noise. It seems important to remark that the PTWs that have been assigned to LUT addresses have been descrambled following the theory of operation discussed in Chapter 2.

3.3.4 Measured performance of the FSU (free space)

Figure 3.23 depicts the results of the conducted performance measurements at the FSU outputs, in terms of LUT address to the received signal strength (normalized power, dBm). During the test, the phase of one of the two adjacent outputs was fixed, whereas the phase of the signal connected to the second output was changed for a monotonically growing series of output phases.

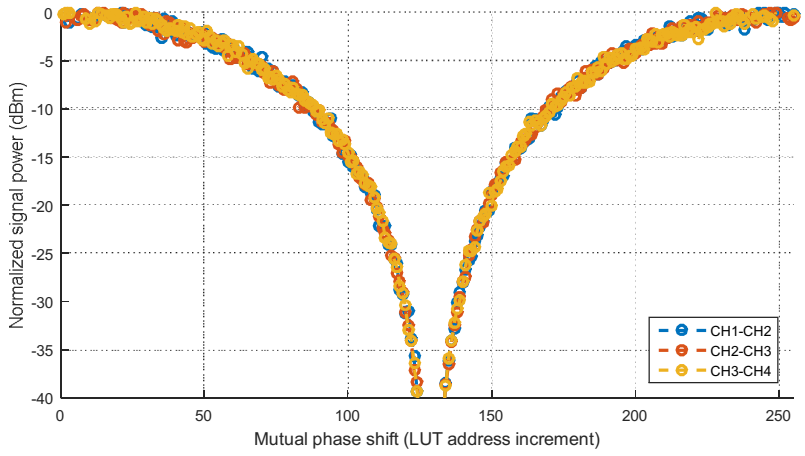


Figure 3.23 – Mutual phase shift vs. normalized power (LUTa=0 to LUTa =255)

When two antennas are used, the RSS curve exhibits a zero in correspondence to the antiphase condition for the electromagnetic waves, namely when a 180° phase shift is assigned to the outputs of the FSU.

3.3.5 Discussion

Given the above measurement results, it can be said that the proposed architecture is suited to implement BSUs for phased arrays following the LO phase shifting approach. In fact, it can independently drive the phase of four, or even more, independent outputs with a high resolution. If compared to the state of the art solutions presented in Chapter 1, the achieved phase resolution is higher, and the architecture has the advantage of being frequency independent. This means that the same scheme can be ported to higher LO frequencies without revisions. This can be extremely beneficial to implement "agile" transceivers, which are solutions useful in software defined radio (SDR) apparatuses, capable to tune over a broad range of frequencies. The modular structure of the proposed phase shifter (at SDL and at system level) can be exploited to develop different phase shift resolutions with a small redesign effort. In fact, the phase shift resolution can be improved by implementing SDLs with a larger number of delay blocks (e.g. ten, for a 0.35° phase resolution). However, since for a given f_{REF} the value of f_{CLK} grows exponentially with n, an impractical system clock frequency is soon to be approached. This is the intrinsic limit of any all-digital topology that implements discrete time delays of 1-bit quantized signals. In fact, even though a clock signal can be used to drive a PLL instead of a sine wave, the synchronous techniques that can synthesize fine-grained delays in their replicas are deeply different. This is because the analog reconstruction of a sine wave from its samples, given that the Nyquist-Shannon sampling theorem requirements are met, allows to implement delays that are not constrained by the clock frequency of the DDS circuitry that is synthesizing the signal, which instead limits the maximum output frequency. It is the samples

lookup table resolution the main parameter constraining the minimum phase tuning step, when a $\frac{\sin(x)}{x}$ reconstruction filter is used. Conversely, when working with all-digital topologies that implement discrete time delays of 1-bit quantized signals, the minimum time delay step depends on the system clock frequency, since outputs are bound to two amplitude values, and can change state only when a transition of the system clock occurs. Thus, the minimum time delay among two outputs is exactly equal to one period of the system clock.

4 A beam steering transmitter prototype for narrowband communications

The purpose of this chapter is to present the design, implementation and evaluation of a beam steering transmitter prototype for narrowband communications. The proposed architecture is centered at 3.350-GHz, and its beam steering operation is based on the BSU presented in Chapter 3. The transmitter is based on a custom RF frontend module and a linear array of patch antennas centered at 3.350-GHz. The RF frontend module performs the up-conversion and the conditioning of the IF signal at its input, in order to transmit it through the array. Actual measurements are reported at block- and system- level, and the performance of a test WCDMA transmission is also presented. Furthermore, a modulator-less approach for low data-rate communications is discussed.

4.1 Beam steering transmitter architecture

The architecture of the proposed phased array transmitter is made up of three subsystems: i) a four-channel LO synthesizer; ii) a custom RF frontend module; iii) a linear array of patch antennas. As will be further analyzed in the following discussion, the main purpose of this architecture is to transmit an up to 10-MHz wide IF signal (centered at 897-MHz) at 3.350-GHz, and to operate its beam steering through the BSU based on revised DDS-PLLs presented in Chapter 3.

Figure 4.1 depicts the block diagram of the above described architecture, and the three subsystems that have been cited are presented afterwards.

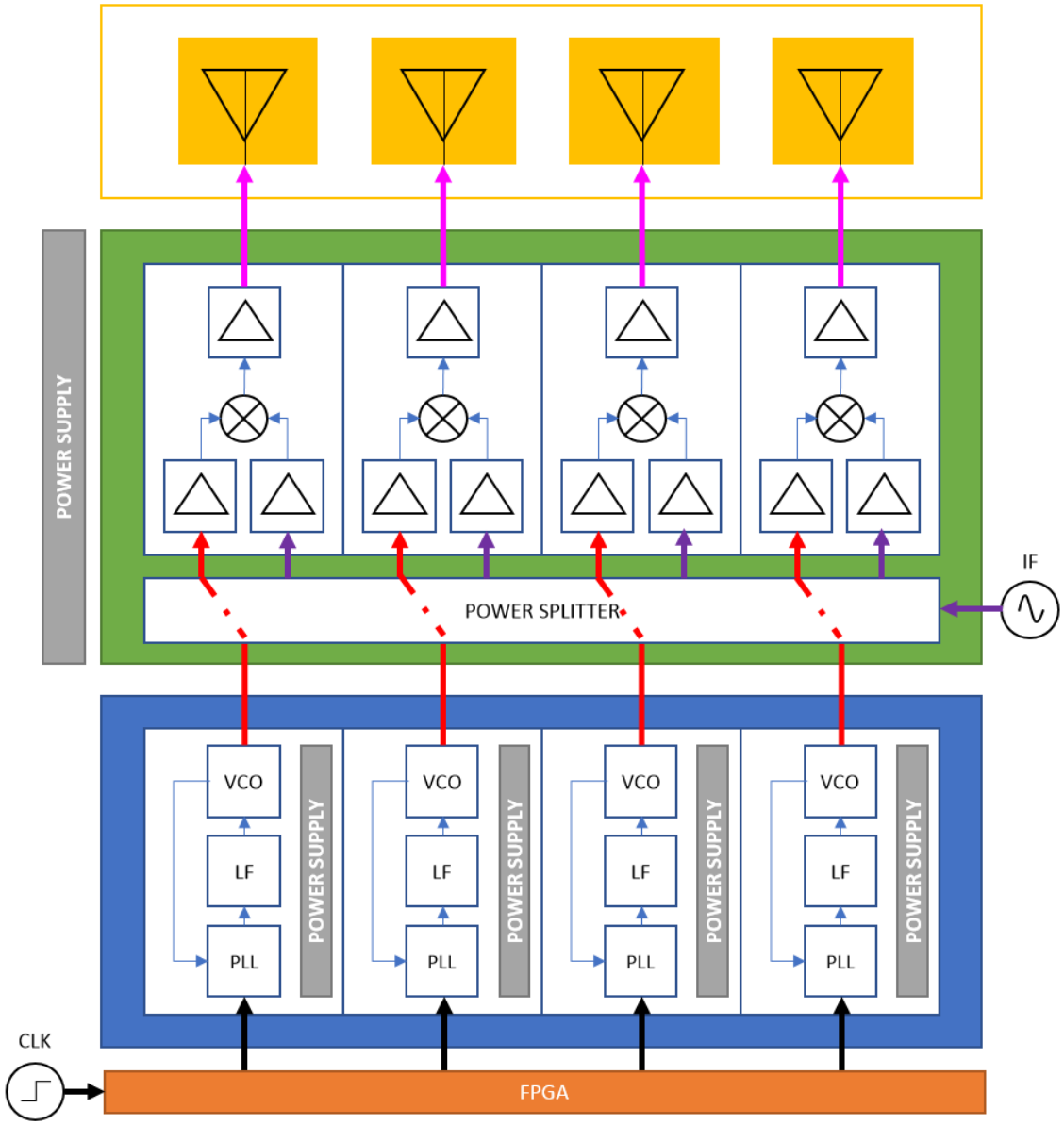


Figure 4.1 – System architecture of the proposed phased array transmitter

The LO synthesizer is the BSU presented in Chapter 3. It is centered at 2.453-GHz and implements mutual phase shifts among its output channels in the [0°; 360°] range with an 8-bit resolution. Figure 4.2 depicts a closeup of one among the four identical LO synthesizers

present on its PCB when the RF shielding is removed. The overall size of the PCB is 240x100-mm^2, and its reference inputs and RF outputs are spaced apart by 61.15-mm.

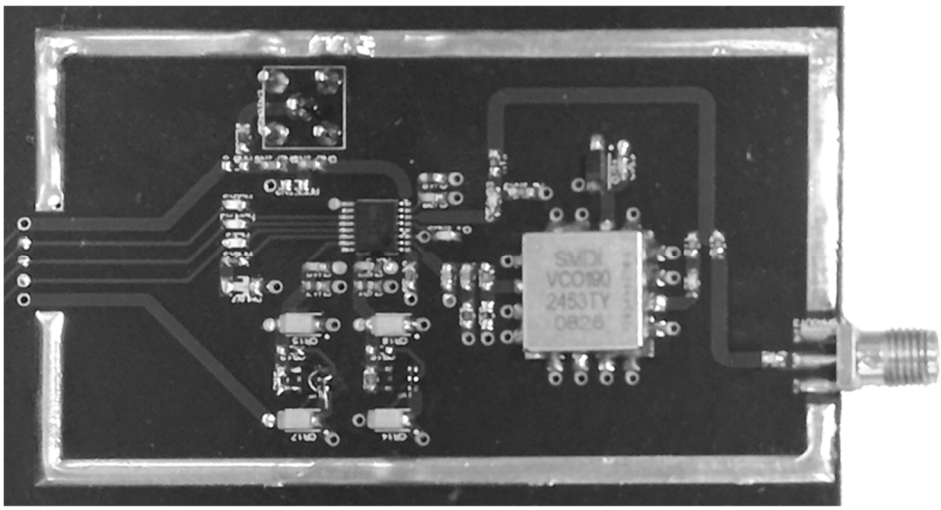


Figure 4.2 – LO synthesizers PCB (one channel)

The RF frontend module performs the IF signal up-conversion at 3.350-GHz, and the other signal conditioning required to drive the antennas. The up-conversion is performed, channel by channel, through passive frequency mixers, and every signal path has been designed to accommodate a fixed attenuator, a gain stage and a filter at its input and at its output. This has been done to feed the IF and the LO ports of the mixers an input power level to optimize their operation, and to achieve an output power at the antennas equal to 0-dBm. This means that, by design, the target input power at the IF ports was -10-dBm whereas the target input power at the LO ports was 7-dBm. Given the above constraints, the choice of the attenuators and of the gain stages were derived from: i) the LO synthesizer output

power, ii) the insertion loss of the filters; iii) the conversion loss of the mixers and, iii) the fixed gains present in the chosen IC series. The IF signal generation can be considered outside the scope of this work and has been delegated to a vector signal generator (VSG), whose output power is divided on-board through a 4-way passive splitter. Figure 4.3 depicts a closeup of one among the four up-conversion stages present on its PCB when the RF shielding is removed. The overall size of the PCB is 80x215-mm^2. The LO inputs are spaced apart by 61.15-mm whereas the RF outputs are spaced apart by 44.75-mm. It must be noted that the spacing of the input and output connectors is matched to their respective $\lambda/2$ value. This means that the four stages are identical, but the routing towards the RF outputs is matched in length but could not be identical in shape.

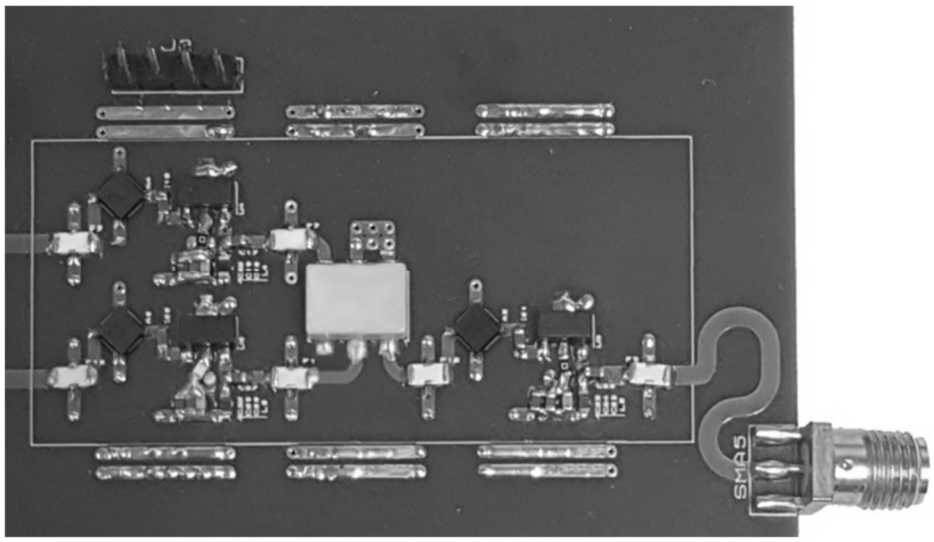


Figure 4.3 – Active up-conversion PCB (one channel)

The linear array of patch antennas is made up of four radiating elements with a $\lambda/2$ spacing, centered at 3.350-GHz, and designed to guarantee a 10-MHz bandwidth. The overall size of the PCB is 185x50-mm^2 and the SMA connectors on it are spaced apart by 44.75-mm. The chosen substrate was the RO4003C laminate from Rogers Corporation, with a thickness of 0.060-in. The design has been carried out using Advanced Design System (ADS) from Keysight and based on conventional patch antennas design theory. Figure 4.4 depicts the PCB placed near to a coin to get a reference for its size.

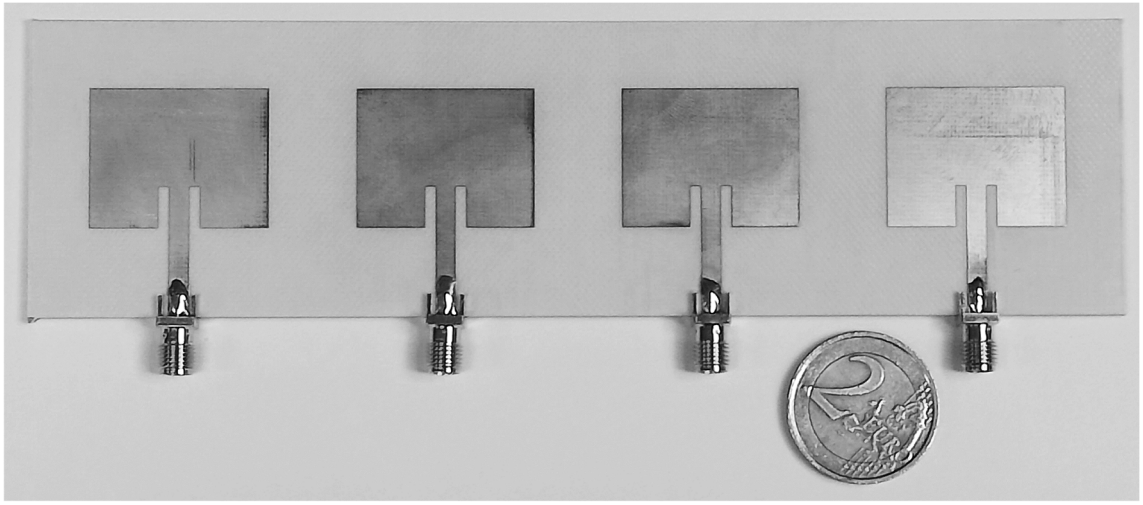


Figure 4.4 – Phased array of patch antennas PCB

The three PCBs have been designed in such a way that they could be assembled together as demonstrated in Figure 4.5. It must be noted that the FPGA and the ultra-clean clock synthesizer boards are not present in the figure but are required parts for the synthesis of the LOs.

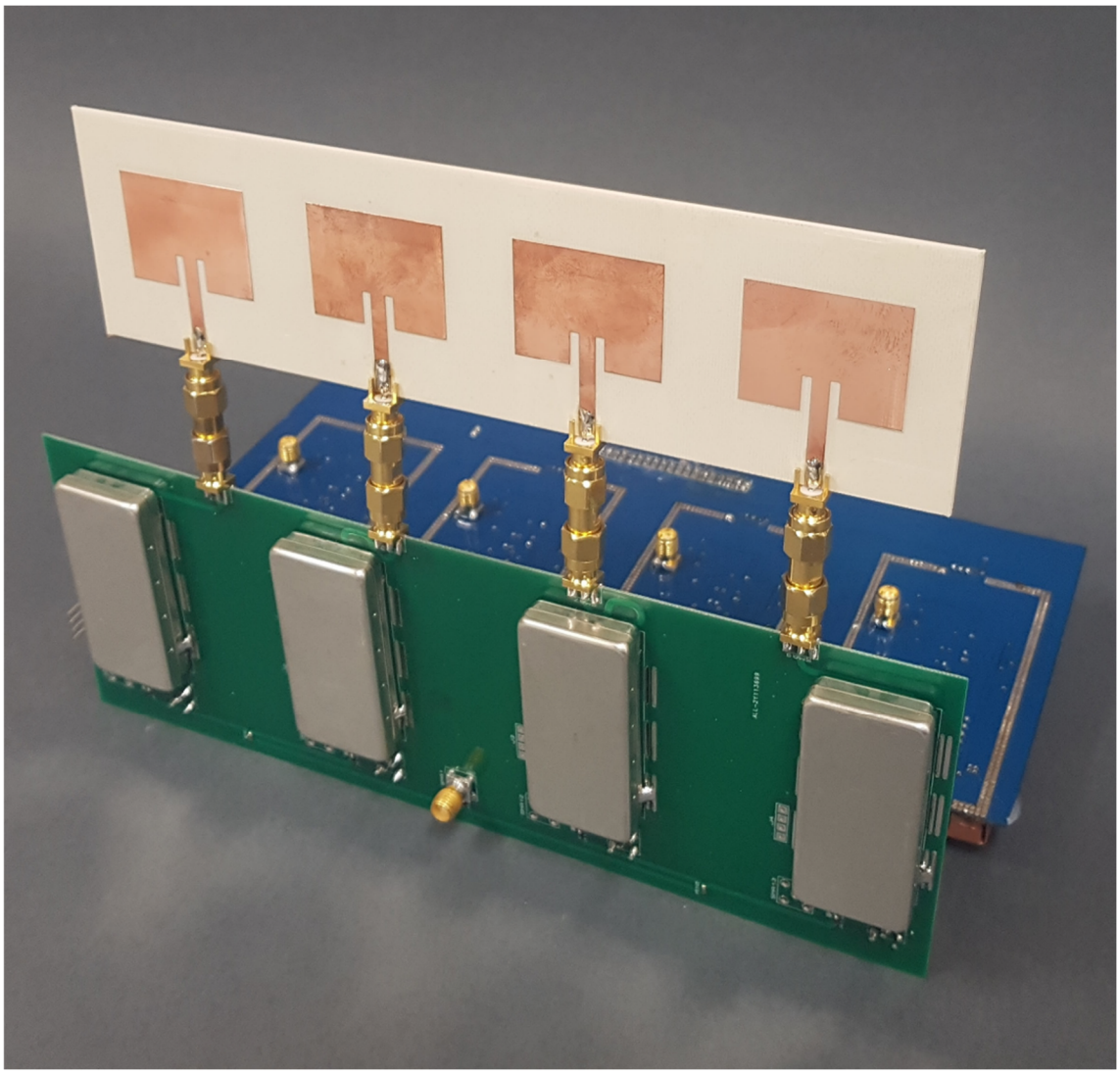


Figure 4.5 – Phased array transmitter prototype

4.2 Antenna design and simulation

Some relevant aspects on the design and simulation of the previously introduced array of patch antennas are now presented, with a focus on the version centered at 3.350-GHz. It must be noted that a previous version centered at 2.453-GHz has also been designed,

simulated and used during the preliminary beam steering tests introduced in Chapter 3. However, due to the similar nature of the results and the actual scope of this work, only the 3.350-GHz variant will be analyzed.

First, the design of one patch antenna has been conducted in ADS through the Momentum 3D Planar EM Simulator. Then, the design has been tuned and replicated four times with a $\lambda/2$ spacing to implement the phased array. Finally, the 2D layers have been imported in CST Microwave Studio to derive the following simulation results. Figure 4.6 depicts a pictorial representation of the PCB and the 3D reconstruction of the radiation pattern when the mutual phase shift among the input ports ($\Delta\varphi$) is 0° (namely, when transmitting towards broadside).

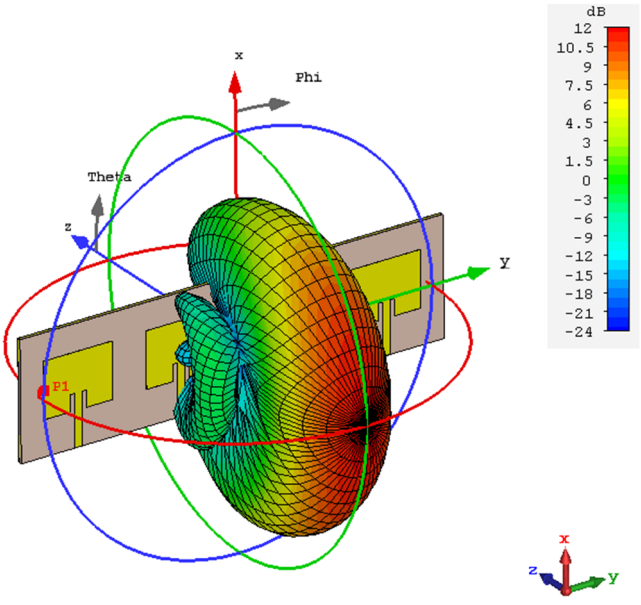


Figure 4.6 – 3D reconstruction of the radiation pattern when $\Delta\varphi$ = 0°

Figure 4.7 depicts the 3D reconstruction of the radiation pattern for a set of four different values of $\Delta\varphi$ (0°, 30°, 60°, 90°), looking at the YZ-plane.

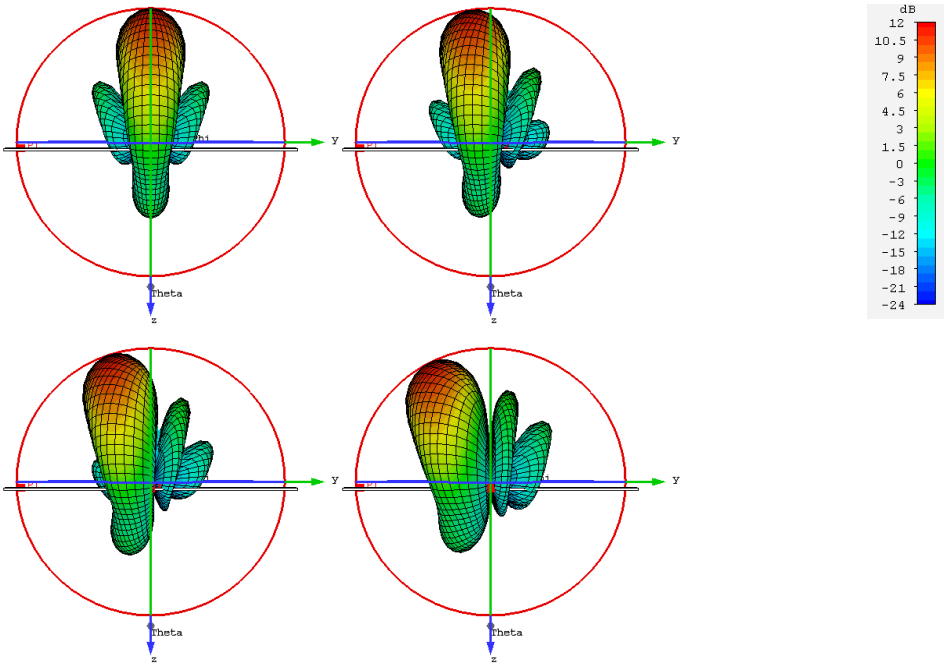


Figure 4.7 – 3D reconstruction of the radiation pattern for a set of four different values of $\Delta\varphi$ (0°, 30°, 60°, 90°)

It seems important to remark that the above 3D reconstructions concern the radiation pattern of the antenna, and thus are derived by the far fields observed during the simulated experiment. Figure 4.8 depicts a 2D reconstruction of the electric field (absolute value) for a set of four different values of $\Delta\varphi$ (0°, 30°, 60°, 90°). This is a near-field study of the antenna, which allows to visualize the effect of phase shifts in the free space propagation of the electromagnetic signal.

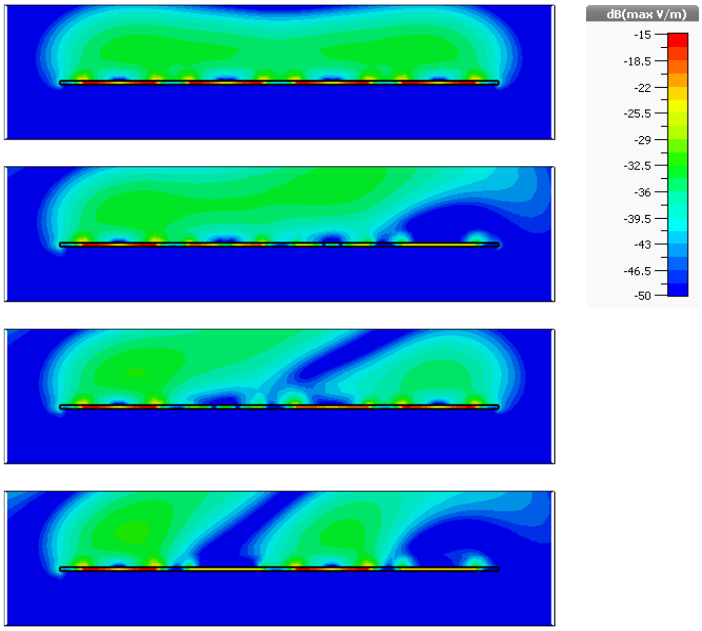


Figure 4.8 – Reconstruction of the electric field (absolute value) for a set of four different values of $\Delta\varphi$ (0°, 30°, 60°, 90°).

Figure 4.9 depicts the reflection coefficient (in dB) at the four input ports, namely the S-parameters whose indices are equal.

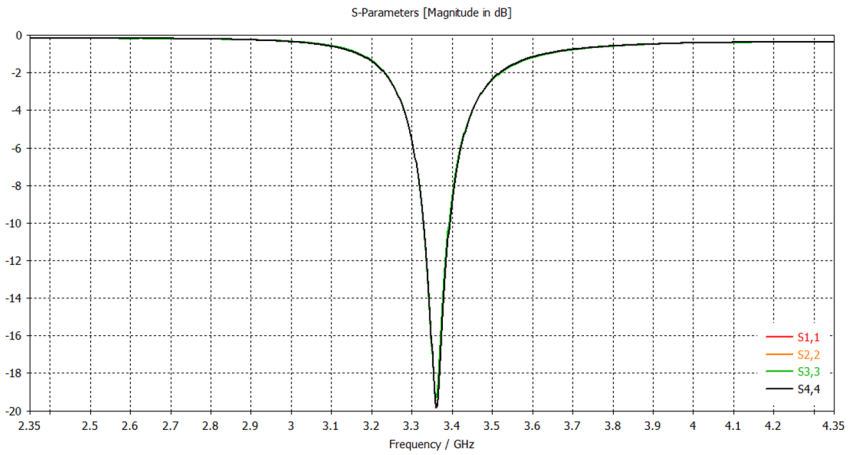


Figure 4.9 – Reflection coefficient at input ports (dB)

It seems important to recall that the reflection coefficient of an antenna quantifies how much of its input power is reflected to the feed line when its excitation signal is swept in the frequency domain. This allows to measure the bandwidth of the antenna, that is the frequency range where a certain amount of input power is delivered to it. For example, when at least 90% of the input power must be delivered to the antenna, a reflection coefficient equal or less than -10-dB must be verified in the bandwidth. Figure 4.10 sums up the power accounting for the antenna array. The radiated power is a share of the overall stimulated power (e.g. due to losses in dielectrics and metal).

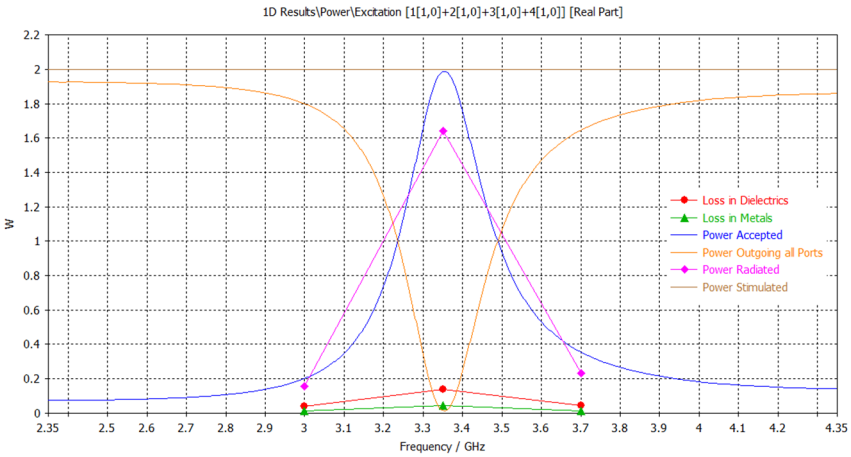


Figure 4.10 – Power accounting

Figure 4.11 represents the coupling (in dB) among the four input ports, namely the S-parameters whose indices are not equal. Three groups of curves can be identified and ranked by descending values of their peak. The first one describes the coupling between adjacent antennas (1-2, 2-3, 3-4), the second one between antennas spaced by one (1-3, 2-4), the third

one between antennas spaced by two (1-4). Due to the reciprocity of the problem, some of the displayed solutions coincide.

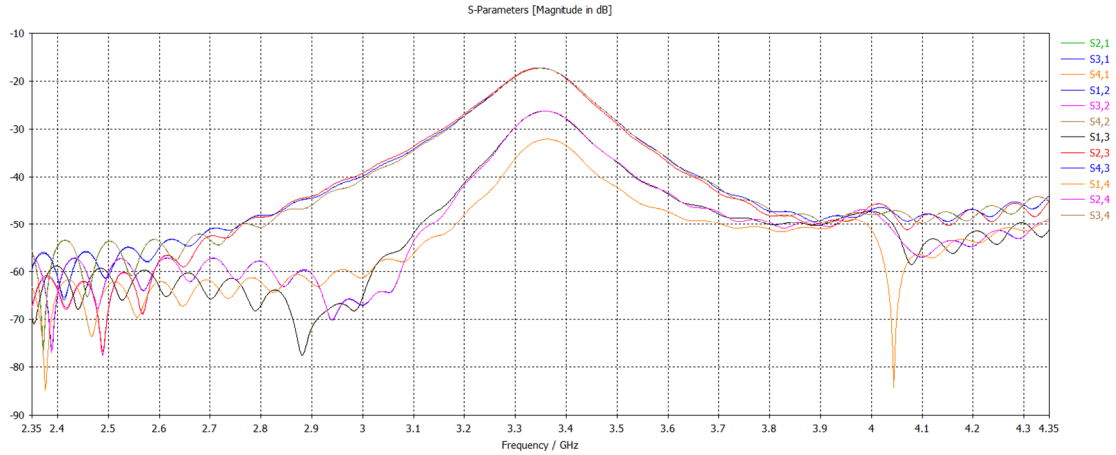


Figure 4.11 – Coupling among the four input ports (dB)

Figure 4.12 depicts the radiation efficiency of the single antennas and of the overall array.

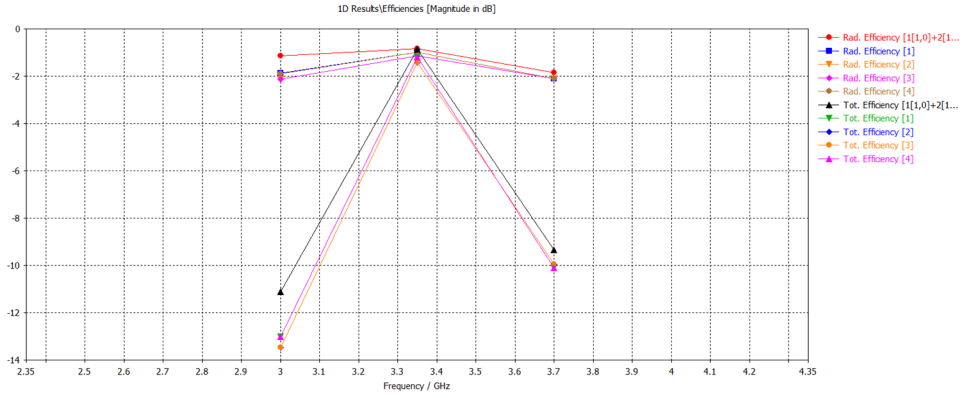


Figure 4.12 – Radiation efficiency of the single antennas and of the overall array

Figure 4.13 depicts the radiation pattern of the array in the YZ-cut for a set of four different values of $\Delta\varphi$ (0°, 30°, 60°, 90°) whereas Figure 4.14 depicts the radiation pattern of the single antenna.

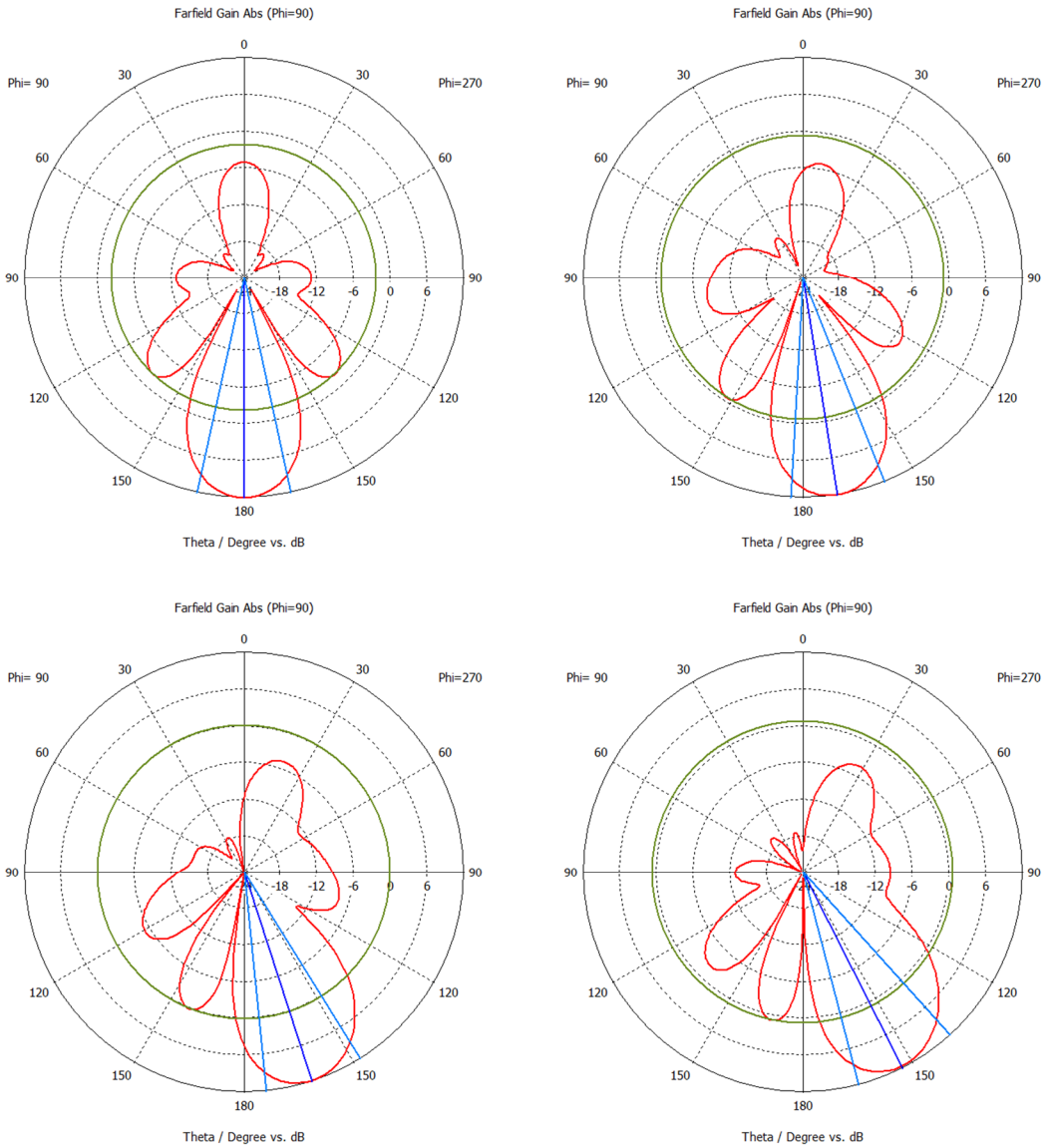


Figure 4.13 – Radiation pattern in the YZ-cut for a set of four different values of $\Delta\varphi$ (0°, 30°, 60°, 90°)

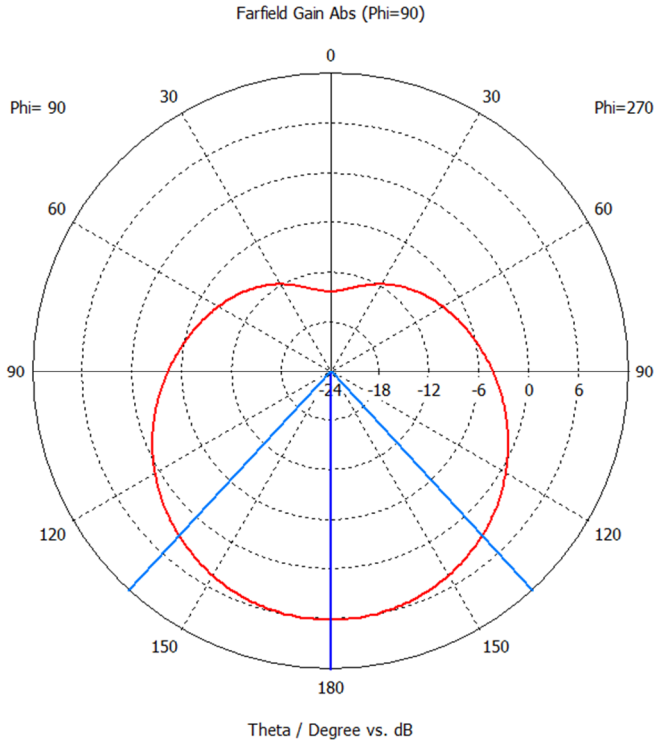


Figure 4.14 – Radiation pattern of the single antenna in the YZ-cut

Considering the above results, the array exhibits a significantly narrower 3-dB angular width of the main lobe if compared to the single antenna. Moreover, its beam steering is clearly demonstrated. However, as the steering angle grows due to an increased $\Delta\varphi$, the peak magnitude of the side lobes, the so-called side lobe level (SLL), is increased. This is an undesirable effect that must be avoided. For example, this can be done limiting the scan angle to a value that ensures certain pass criteria. Figure 4.15 depicts 3-dB angular width, main lobe magnitude (compared to SLL) and main lobe direction for 10 values of $\Delta\varphi$ equally spaced in the range [0°; 180°] (namely any relevant value of $\Delta\varphi$).

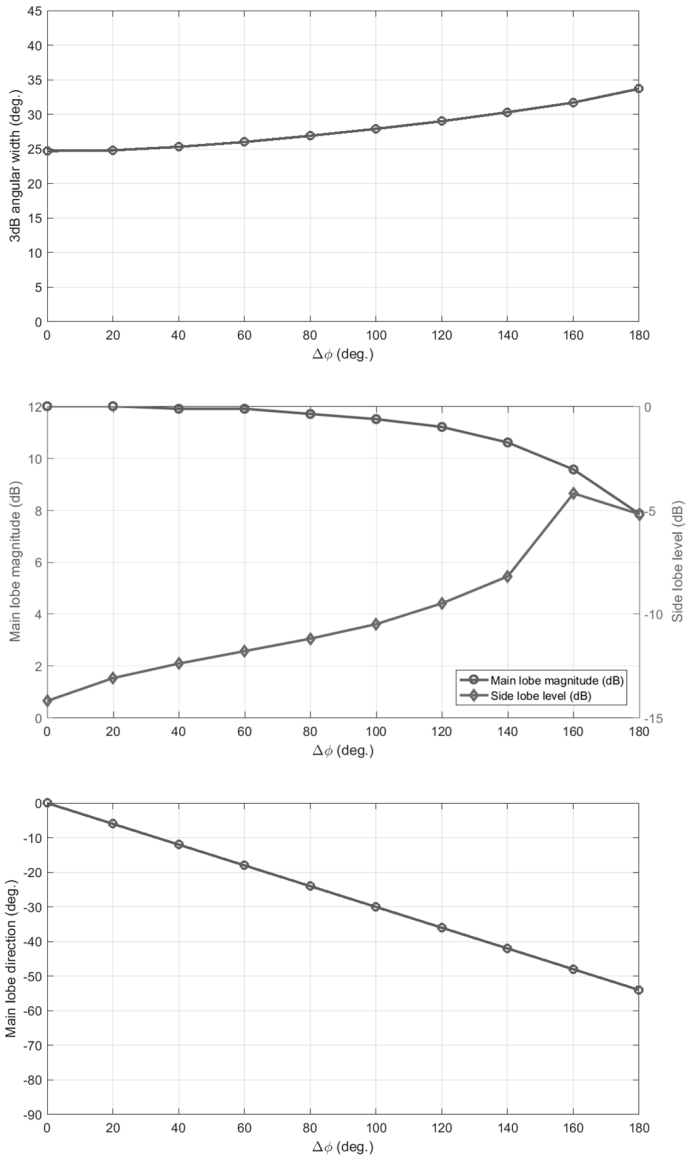


Figure 4.15 – 3-dB angular width, main lobe magnitude (compared to SLL) and main lobe direction vs. $\Delta\varphi$

The main lobe direction dependence from $\Delta\varphi$ maintains linear in the whole range. The main lobe magnitude decreases for the upper values of the steering angle, accompanied by a worsened SLL. A broader (worsened) 3-dB angular width is obtained following the same trend.

4.3 Performance evaluation

To implement and characterize a beam steering transmitter based on DDS-PLLs, it is necessary to develop a system calibration procedure. This must be done to put the array into a known state, from which predictable mutual phase shifts can be assigned following the previously reviewed formulations. In Chapter 3, the system calibration has been performed in time domain, probing the FSU outputs through an oscilloscope and coaxial cables. In this case, the calibration has been performed in the frequency domain, sampling the RSS of the actual propagating signal at the receiver end. This is because the overall system, including the antennas, had to be calibrated at once. The following technique has thus been developed.

Mismatches due to fabrication tolerances must be compensated for through a phase alignment. This is because tolerances lead to unpredictable phase offsets among the outputs that must necessarily be eliminated. This has been achieved exciting the IF port of the transmitter with a single frequency and then aligning each adjacent pair of outputs through the following steps:

1) Every PTW is set to 0 to put the array into its reset state.

2) One channel of the array is arbitrarily assumed as calibrated (a.k.a. the reference channel).

3) All channels are muted, except the reference and its adjacent one that is going to be calibrated.

4) The two unmuted outputs must have an equal electrical distance from the receiver.

5) The calibration PTW for the non-reference channel is found searching the minimum RSS condition (a.k.a. the antiphase condition). Phase alignment is obtained adding 2^{k-1} to this PTW and then performing a modulus 2^k operation.

6) The last calibrated channel is elected as the new reference and is used to calibrate its adjacent one.

7) Steps 3 to 6 are repeated for all pairs of outputs that has to be calibrated.

The above steps return a vector of PTWs that ensures phase alignment, namely the calibration vector for the transmitter. After this calibration, beam steering can be accomplished using the above PTWs as offset values to be added to the theoretical PTWs obtained inverting the equation introduced in Chapter 3.

It is useful to note that different strategies can be used to find the PTW corresponding to the minimum RSS of each channel. One option is to inspect all the PTWs, and then to choose the one corresponding to the minimum RSS that has been obtained. Unfortunately, this solution can become too onerous if the resolution of the phase shifter is high or if the number of radiating elements is large. An effective workaround is to exploit the characteristics of the RSS curve. The key lies in the fact that the expected curve has just one minimum point. Thus, interval-halving methods can be used to lower the number of measurements one needs to perform. A final consideration concerns the repeatability of the above technique and the applicability of its results. Any perturbation of the signal path affects the output phase of its corresponding channel. This means that the above calibration has to be performed any time a perturbation of the signal path characteristics is suspected. However, it is the PCU to PLLs interface the one that is most sensitive to fabrication or

assembly related mismatches. The reason for this is evident if considering the previously reviewed theory of operation.

It should be noted that in this case the RSS is only used to quantify the electromagnetic interference effect related to phase mismatches among coherent waves and that the RSS is not per se an index of the quality of a wireless link. This is especially true for phased arrays based on the LO phase shifting approach, remembering that a linear phase distortion across the signal bandwidth is always present. A measurement of the error vector magnitude (EVM) of the communication under test must always accompany the RSS value to get a clear picture of the actual capability of the apparatus to transmit intelligible information contents.

4.3.1 Measurement setup

The performance of the prototype has been evaluated in an RF quiet environment. The test bench included a vector network analyzer (VNA) and two vector signal generators (VSGs) from Keysight in addition to the instruments that have already cited in Chapter 3. The same precautions to ensure that the measurements were repeatable have also been used (e.g. semi-rigid coaxial cables secured through a torque meter wrench).

The first test has been conducted on the up-conversion PCB. This test was performed, channel by channel, in the frequency domain by inputting known LO and IF signals generated through the VSGs, and measuring the output power and the EVM of a QPSK transmission following the WCDMA protocol. The LO frequency was 2.453-GHz and the IF center frequency was 897-MHz. These values produce, after mixing, an upper-sideband

signal centered at 3.350-GHz. The measured EVM (RMS) was within 1% of the theoretical value.

The second test was conducted on the antenna elements used in the design. The measured reflection coefficient matched the CST simulated results. For a bandwidth of 10-MHz centered around 3.350-GHz, the signal's power transferred to the antenna is above 99% (Figure 4.16).

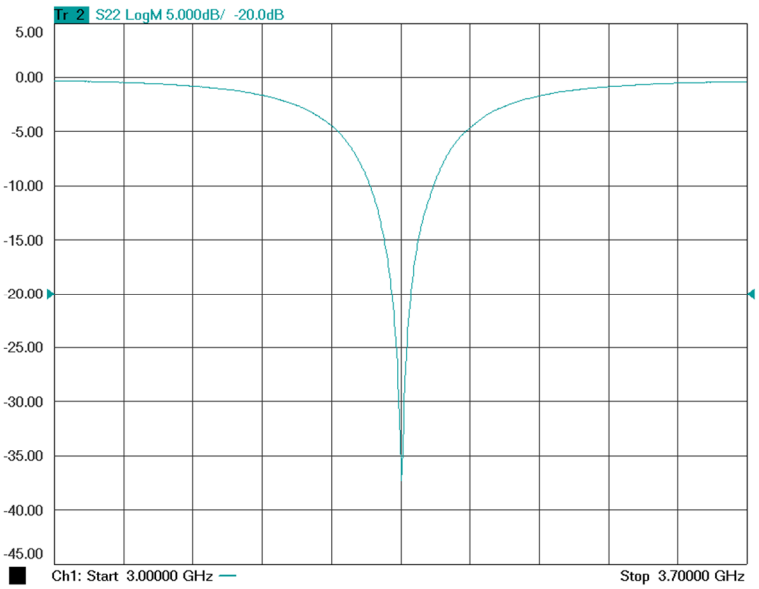


Figure 4.16 – Reflection coefficient at input port (dB)

At the system-level, the overall architecture has been tested attaching a patch antenna to the VSA and probing the RSS for different directions and for various scan angles. The signal and frequency configuration of this test was the same as the one from the up-conversion PCB. However, the proposed transmitter architecture is not bound to this particular

configuration. It seems useful to remark that this hardware architecture can be employed to transmit any modulation scheme with any LO and IF frequency configuration.

Figure 4.17 shows the QPSK constellation of the received signal and the EVM (200 symbols) at the VSA when the transmitter is focusing towards it, placed at broadside.

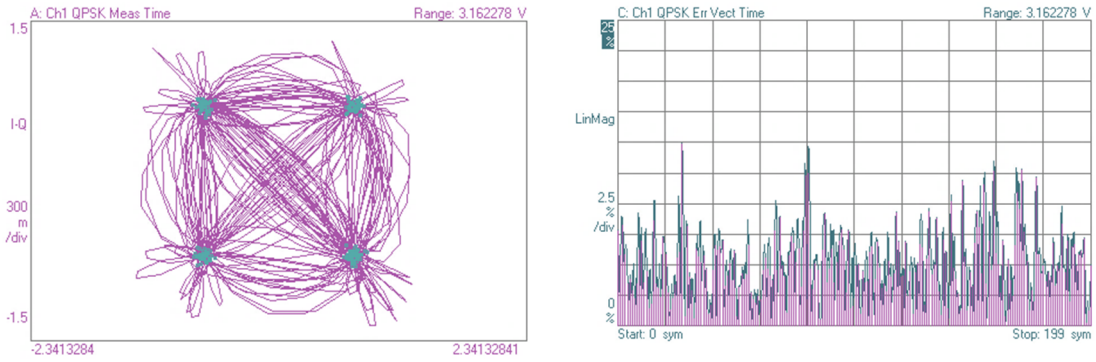


Figure 4.17 – QPSK constellation and EVM (200 symbols) at the receiver when transmitting broadside ($\Delta\varphi = 0°$)

Table 4.1 summarizes the values of the main figures of merit measured by the instrument.

Table 4.1 – Main figures of merit measured by the instrument

Measurement	Value	
EVM (RMS)	5.7193	%
EVM (peak)	14.955	%
Mag. Err. (RMS)	4.0760	%
Mag. Err. (peak)	-12.674	%
Phase Err.	2.3033	deg
Frequency Err.	1.1533	kHz
SNR (MER)	24.853	dB

Figure 4.18 shows a normalized map of the RSS sampled for various values of $\Delta\varphi$ and different angular positions of the receiver. The achieved beam steering is clearly recognized, as well as the worsened 3-dB angular width and SLL near the limits of the scan angle. The experimental results illustrate how the BSU prototype presented in Chapter 3 can be successfully used to implement a beam steering transmitter for phased arrays. It is important to note that improving the resolution of phase shifts is more beneficial when the 3-dB angular width of the main lobe is small, that is when the number of elements in the phased array is large. In this case, the angular resolution of the array becomes comparable to the beam width, and thus an exceptionally fine-grained spatial selectivity can be achieved.

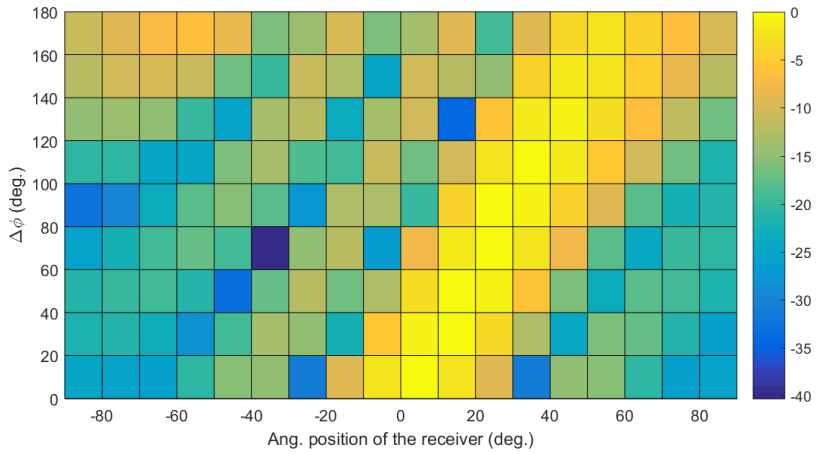


Figure 4.18 – Normalized RSS for various values of $\Delta\varphi$ and different angular positions of the receiver

4.4 Modelling the effects of the time delay approximation

Known that the LO phase shifting approach introduces a linear phase distortion across the signal bandwidth, the above beam steering transmitter has been constrained to narrowband signals, that in this context are signals whose bandwidth is less than 1% of the LO frequency. However, a deeper understanding of the effects that the above distortion introduces had to be carried out. For this reason, a model has been developed to simulate the effects of the time delay approximation over a bandwidth that is wider than 10% of the carrier frequency. The model considers an array of four isotropic antennas at the transmitter end, one isotropic antenna at the receiver end (at a far distance from the transmitter) and an ideal wireless channel.

Figure 4.19 shows the normalized RSS (linear scale) calculated for seven mutual phase shifts (equally spaced between 0° and 180°) at the antennas of the transmitter (across a 400-MHz bandwidth centered at 3.35-GHz) when the receiver is placed in various angular positions (equally spaced between 0° and 90°).

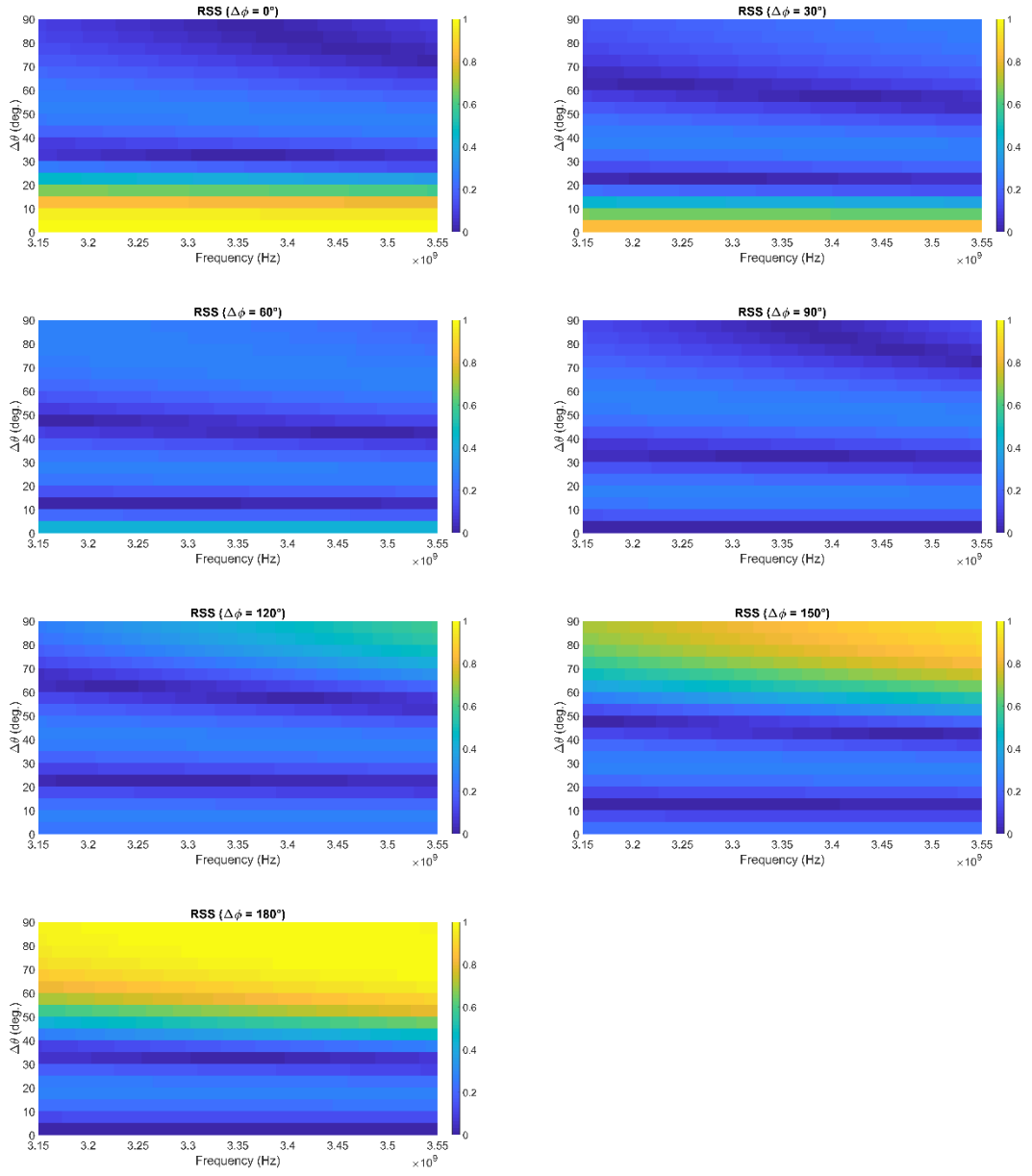


Figure 4.19 – Normalized RSS (linear scale) calculated for seven mutual phase shifts when the receiver is placed in various angular positions

Figure 4.20 shows the normalized RSS (linear scale) calculated for four angular positions (equally spaced between 0° and 90°) of the receiver when the mutual phase shifts at the

antennas of the transmitter (across a 400-MHz bandwidth centered at 3.35-GHz) is varied among various values (equally spaced between 0° and 180°).

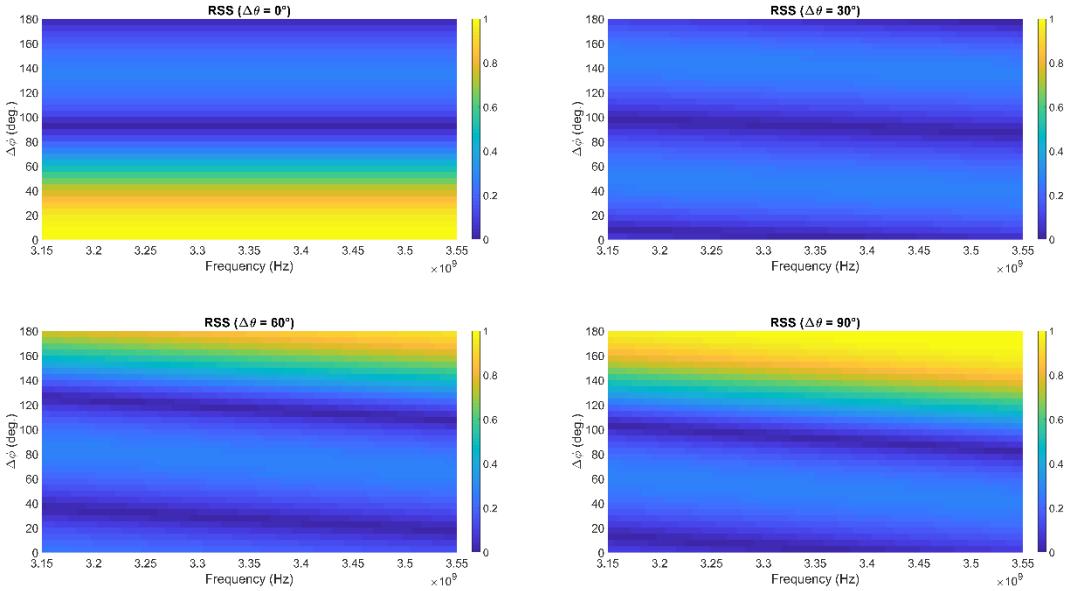


Figure 4.20 – Normalized RSS (linear scale) calculated for four angular positions of the receiver when the mutual phase shifts at the antennas is varied among various values

The above data allows to derive the following considerations. The RSS varies across the inspected bandwidth, when keeping constant $\Delta\theta$ and $\Delta\varphi$. This is because at different frequencies the same phase shift corresponds to a different time delay, giving an uneven distribution of the lobes in the radiation pattern and a beam squint across the signal bandwidth. These two effects must be governed to implement a successful broadband communication following the LO phase shifting approach.

4.5 A modulator-less beam steering transmitter

The purpose of this section is to present an alternative beam steering transmitter architecture for low data-rate communications. The main feature of this solution is that it does not rely on modulators (or mixers) to transmit data. In fact, it implements phase shift keyed (PSK) signals through the very LO phase shifting capability of the BSU presented in Chapter 3, and data transmission is bound to the dynamics of the PLLs phase reconfiguration.

If many revised DDS-PLL phase shifter architectures have been presented in literature, the opportunity to exploit their hardware to implement a PSK communication has never been focused on. This is mainly because the target of these works was to investigate the LO phase shifting under a quasi-static steering condition, where transients related to phase reconfiguration can be neglected. Instead, in this approach, the BSU is used to synthesize both quasi-static LO phase shifts for beam steering, and fast phase shifts for the transmission of PSK symbols. Under this working condition transients cannot be neglected, since their duration is comparable to the one of the symbols that are being transmitted.

The theory of operation of the above transmitter is now discussed. If one of the PLL output phases is considered as a reference phase, φ_0, it can be said that some of its other output phases (related to an equal number of reference signal delays) can be interpreted as the symbols of a PSK constellation. In fact, each output phase shift from φ_0 can be interpreted as a rotation of the vector that represents the synthesized PLL output in the IQ plane. If a LUT exists that can map this transformation (and it exists, since it is the same one constructed for the mutual phase shifts among PLLs), a modulator can be implemented

exploiting the very BSU hardware. All it is needed is an operator that sums the phase rotation assigned to the PSK symbol *a*, to the phase shift β assigned to the beam steering (Figure 4.21). The *a* angle changes with a frequency that is defined by the symbol rate of the communication, thus the transient response of the PLLs at each transition must extinguish in a fraction of the symbol time duration, a specification that ultimately depends on the loop bandwidth. The angle β can be interpreted as a phase offset that changes with a much slower frequency (its variation is only needed to reshape the radiation pattern). The phase rotations *a*, due to the transmission of the PSK symbols, do not affect beam steering. In fact, beam steering is related to mutual phase shifts at the BSU outputs, that only depend on β (*a* is equal for all outputs).

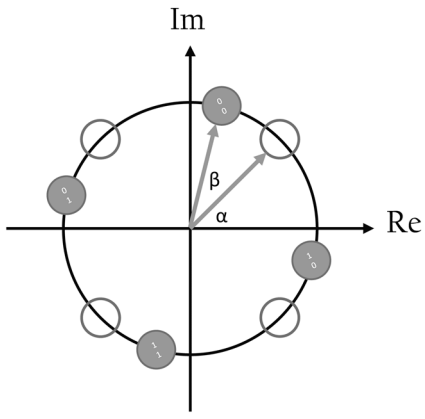


Figure 4.21 - IQ plane representation of α (phase rotation assigned to the PSK symbol) and β (phase shift assigned for beam steering)

Being $B = [\beta_0, \beta_1, \beta_2, \beta_3]$ the vector of phase shifts to synthesize the desired radiation pattern, and *a* the phase rotation assigned to the PSK symbol being transmitted, the resultant phase state vector for the array is $B' = [a + \beta_0, a + \beta_1, a + \beta_2, a + \beta_3]$. Thus, the configuration

vector for the delay lines is $C = [\chi(a + \beta_0), \chi(a + \beta_1), \chi(a + \beta_2), \chi(a + \beta_3)]$ where χ is the transformation, operated through the LUT, that maps output phases to binary Phase Tuning Words (PTWs) in the PCU. The transformation is operated by finding the LUT pointer (*PTR*) that returns the PTW needed to obtain the $a + \beta_i$ output phase, and then reading from that address the matching PTW. For a LUT where the PTWs are stored in memory for increasing output phases, the desired *PTR* is:

$$PTR = \text{round}\left(\frac{(\alpha + \beta_i) \bmod 360}{360} \cdot 2^8\right)$$

where $PTR \in [0, 1, \dots 255]$, namely *PTR* is an 8-bit unsigned integer. It must be noted that *PTR* can also be computed as follows:

$$PTR = a_1 + a_2 = \text{round}\left(\frac{\beta_i \cdot 2^8}{360}\right) + \frac{\alpha \cdot 2^8}{360}$$

where both addenda $a_{1,2} \in [0, 1, \dots 255]$, that is the addenda are two 8-bit unsigned integers. In fact, $\beta_i \in [0°; 360°)$, and for any 2^n-PSK constellation ($n \in [1, 2, \dots 8]$):

$$\alpha = m \cdot \frac{360}{2^n}$$

where $m \in [0, 1, \dots 2^n - 1]$. This allows to lower the number of instructions needed to find the *PTW*, since no modulus operation is needed to compute the *PTR*.

The most important constraint in this technique is that the transient response of the PLLs at each *a* angle transition must extinguish in a fraction of the symbol time duration. This specification fixes the Time To Lock (TTL) at 10-Hz and 1° of the PLLs. Table 4.2 reports these values for different LBWs, calculated for a Charge Pump (CP) current of 1.00mA and

a phase margin of 45°. It must be noted that the reported timings are calculated for VCOs in free-run as their initial condition, which is not the case during symbol transition.

Table 4.2 – Frequency and phase TTL

LBW	TTL at 10-Hz	TTL at 1°	Phase Jitter
1-kHz	9.74ms	9.63ms	0.11° RMS
10-kHz	377µs	333µs	0.43° RMS
20-kHz	188µs	145µs	0.75° RMS
50-kHz	78.2µs	55µs	1.21° RMS
100-kHz	36.7µs	24.5µs	1.40° RMS

4.5.1 Performance evaluation

The purpose of this section is to present the performance evaluation conducted on a proof-of-concept prototype. The prototype has been configured to transmit data according to a 16-PSK modulation scheme (4-bits per symbol). The symbol rate of the communication has been fixed to 8-kbaud (that is the symbol duration time for data transmission, a.k.a. UI, is 125 µs). Thus, the data transmission rate is:

$$R_b = 8\frac{kbaud}{s} \cdot 4\frac{bits}{baud} = 32kbps$$

Given a UI of 125 μs, the designed LBW of the PLLs has been set to 100kHz. This means that the worst-case TTL during a symbol transition is 19.6% of UI. However, a phase margin of 46.2° has been implemented when working with actual components, thus the worst-case TTL during a symbol transition is 24.4% of UI (Table 4.3).

Table 4.3 – Frequency and phase TTL for the actual LBW

Loop BW	TTL at 10-Hz	TTL at 1°	Phase Jitter	% of UI
100-kHz	45.1μs	30.5μs	1.37° RMS	24.4

A receiver based on a TRF371125 IQ demodulator has been implemented to perform the test communication. During the measurements, the receiver was connected to the RF outputs of the BSU through a 4-way passive combiner and four semi-rigid coaxial cables with matched lengths (Figure 4.22). This allowed to simulate the position of the receiver with respect to the transmitter as if it was broadside and at a far distance. The BSU was configured to transmit towards the receiver, with β_1, β_2, β_3 and β_4 set to the same value. A random sequence of bits was used to validate the transmission.

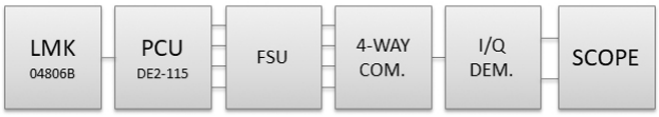


Figure 4.22 - Measurement setup employed during the qualification of the transmitter

The passive 4x1 power combiner has been designed so that the four signal paths had matched lengths. The spacing between the input ports is 6.115 cm, the same spacing as at the FSU outputs. Traces have been designed as microstrips with an impedance of 50 Ohm at 2.453 GHz (Rogers RO4003C substrate, ε = 3.55, H = 0.812mm, W = 1.83mm).

Given the above test bench, Figure 4.23 shows actual IQ signals (before filtering) for the signal transitions a = 135° and a = 315°. The transient response of the PLLs is shorter than the designed 19.6% of UI, and thus validates the viability of the proposed solution.

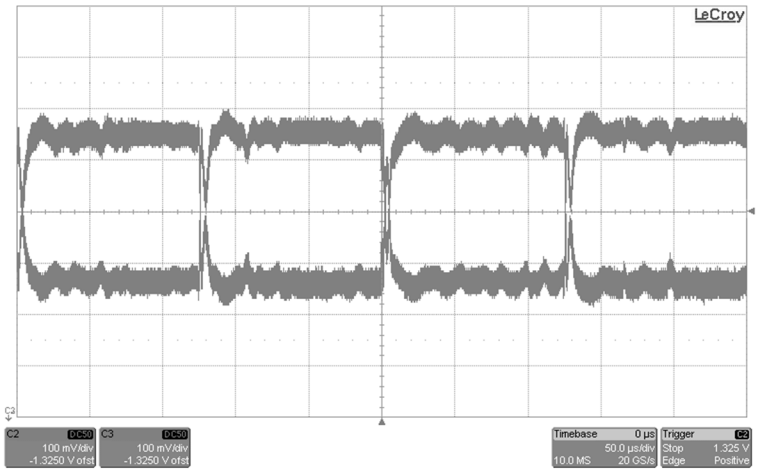


Figure 4.23 - Actual IQ signals for the signal transitions α = 135° and α = 315°

The narrowband beam steering polar transmitter presented in this section demonstrates how BSUs based on DDS-PLL phase shifters can be used to introduce spatial selectivity in low data-rate communications without the need for external parts. However, it also reveals a number of drawbacks that must be taken into account. First, there is no guarantee that the

radiation pattern is preserved during transients related to phase reconfigurations. Second, a broader LBW leads to a worsened spectral purity of the synthesized LOs. These drawbacks can further be studied through theoretical and experimental approaches. For example, the benefits of using reference signals with a higher frequency could be investigated.

Conclusions and future work

This thesis presented the design and the implementation of a revised DDS-PLL phase shifter, as well as its subsequent integration into a beam steering transmitter prototype.

After an introduction on the topic of phased arrays and the review of some relevant state of the art architectures, the working principles of common DDSs and PLLs has been presented, in order to introduce the standard DDS-PLL architecture and its known variants. Then, the design and the performance evaluation of a revised DDS-PLL architecture, whose PCU is based on SDLs with programmable lengths, has been presented, along with a discussion on its FPGA-based prototype. The resultant BSU allows to independently set the phase of its four outputs with an 8-bit resolution at 2.453-GHz, though the modular structure of the proposed PCU can be exploited to develop different resolutions with a small redesign effort. Moreover, the proposed architecture is frequency independent.

The phase shifting performance of the prototype has been quantified in time and frequency domain. At the PCU outputs, the phase error maintains well within the range [-0.01°; +0.01°]. At the FSU outputs, the phase error maintains well within the range [-0.9°; +0.9°]. When two antennas are used, the RSS curve exhibits a zero in correspondence to the antiphase condition for the electromagnetic waves, namely when a 180° phase shift is assigned to the outputs of the FSU.

Finally, the implementation and evaluation of a beam steering transmitter prototype for narrowband communications has been presented. The proposed architecture is centered at 3.350-GHz, and its beam steering operation is based on the above BSU. The transmitter is

based on a custom RF frontend module and a linear array of patch antennas centered at 3.350-GHz. The RF frontend module performs the up-conversion and the conditioning of the IF signal at its input, in order to transmit it through the array.

A test WCDMA transmission has been used to evaluate the performance of the overall transmitter and derive an RSS map for various steering angles and angular positions of the receiver. The achieved beam steering is clearly recognized, as well as the worsened 3-dB angular width and SLL near the limits of the scan angle. A modulator-less approach for low data-rate communications has also been presented, implementing a PSK communication through the BSU without the need for external parts.

The experimental results illustrate how the BSU prototype can be successfully used to implement beam steering transmitters for phased arrays. This demonstrates the viability of the proposed approach, and thus the opportunity to plan more research activities based on these results.

Future research effort will focus on the implementation of a beam steering receiver based on the same working principle and the integration of a transceiver architecture into an actual ASIC design, developed in a BiCMOS technology. This perspective is supported by the cost effectiveness of many mixed-signal processes on the market, the measured results achieved through the prototype and how animated this research field is.

Appendix

The purpose of this appendix is to discuss the topic of FPGA-based prototyping and IP core based designs, with a focus on SoC designs. Motivations, procedures and tools commonly employed in the academia and in the industry are presented.

A.1 FPGA-based prototyping

FPGA-based prototyping is a consolidated methodology used to reduce development costs, shorten time-to-market (TTM), and reduce risks during the design of ASICs. In fact, as the complexity of digital circuits grows unceasingly, powerful methodologies are needed to carry on hardware verification before fabrication, in order to avoid costly re-spins. In [23], Darren Zacher (2010) reported that the development cost of a 90-nm ASIC design is around 20-M$, with an estimated cost for the mask set exceeding 1-M$. This makes clear that pursuing a fault-free design during the first fabrication pass, or at least trying to minimize the number of re-spins, is crucial to the commercial success of any ASIC. Moreover, apart from its cost, each wafer run requires months to be completed, and thus can dramatically increase the TTM of a product. The above considerations, true for any ASIC design are even more pertinent when approaching the development of SoCs.

In [24], D. Amos et al. (2011) presented a snapshot of best practices in the field of FPGA-based prototyping with particular reference to SoC designs. The authors propose a clear definition of SoC, which is "a superset of ASICSs" that "always includes at least one CPU and runs embedded software". The authors also assert that, among the many product designs that can benefit from FPGA-based prototyping, SoC designs are the ones that benefits the

most of it due to its "unique ability to provide a fast and accurate model of the SoC in order to allow validation of the software". In fact, among the challenging aspects of SoC designs is that software development has to be counted among the dominant efforts in the product design. To better explain how crucial a convenient prototyping strategy (such as the FPGA-based ones) can become for the success of a SoC design, the authors present a case study reporting the performance of a SoC development project. For example, the authors illustrate how development costs for an actual wireless headset design performed in a 65-nm technology (estimated as 31,650-k$) are spread during months. If hardware design related efforts dominate the development costs in the first 12 months, the first engineering samples are not ready before month 19, and thus post silicon validation cannot be carried on before that date. Moreover, software development consumes 40% of the total cost of the design (due to OS, low-level software and high-level application support, porting or development) and ramps up just before RTL is finalized. For an average volume of 1.5 million units per month and an average selling price of 5.50$, the authors state that the project is expected to reach break-even 3 years after the starting product development, that is an extremely challenging scenario. For this reason, they suggest that any SoC design benefits from a "start software sooner" approach. The above example has been introduced to demonstrate the benefits of prototyping, by employing techniques such as simulators, emulators, and FPGA-based prototypes.

Before reporting the peculiarities of these techniques (with a particular focus toward FPGA-based prototypes) it seems important to recall what the authors summarized about user priorities in prototyping, leading to different prototyping options as the best trade-off.

1. *Time of availability* is the opportunity to develop a model for a software validation environment as soon as the specifications for the design are frozen.
2. *Execution speed* is primarily related to the level of abstraction that is implemented, and thus to how accurate is the representation of actual hardware.
3. *Accuracy* is the effectiveness of the model, which must not introduce issues related to its own implementation.
4. *Capacity* is the opportunity to host an entire SoC design into the prototyping solution, as well as its future upgrades.
5. *Development cost* is the cost of the prototyping solution, intended as the actual cost of production plus the overhead cost of tooling design, and suggests if it can be replicated to furnish development teams.
6. *Bring-up cost* is the cost needed to startup the model implementation, without considering what is already needed to develop the silicon.
7. *Deployment cost* is the cost of distributing multiple copies of the prototype such as actual deployment, maintenance and support for end users.
8. *Debug insight* is the detail that inspection tools can provide when analyzing signals, registers and state of the hardware/software design.
9. *Execution control* is an important feature to debug systems, and is carried on through the use of assertions in the hardware or breakpoints in the software that synchronously stops all the involved components.

10. The opportunity to implement *system interfaces* into the prototype is crucial to prototype protocol stacks (e.g. USB, network, wireless …).
11. *Turnaround time* is the time required to update the prototype if a newer version of the model is developed.
12. *Value links to low power and verification* are related to the opportunity to interconnect portions of the SoC prototype to the other ones in order to verify their behavior, as well as to assess power consumption of the various components to evaluate low power strategies.

It seems also important to report what the authors state to be the chip design trends in future SoC projects: "i) further miniaturization towards smaller technology nodes; ii) a decrease in overall design starts; iii) programmability combined with a rapid increase of embedded software content; iv) IP reuse; v) application specificity; vi) adoption of multicore architectures; vii) low power; viii) an increase in the analog/mixed signal portion of chips". The above trends make clear that future designs are oriented where the need for prototyping is the greatest, namely where the risk per design increases dramatically. That is, for example, when moving to smaller technology nodes. Moreover, software development is critical to determine the TTM of SoC devices, since the effort required for its design is equal, if not surpassing, the one of the hardware development. For this reason, the authors state that multiple prototyping solutions are in active use among the industry, since different priorities are taken into account when considering the various aspects of the hardware/software development flow of a SoC product. Figure A.0.1 represents the results of a poll conducted

during an international conference on SoCs, when 116 participants where asked what methodologies they were using to develop hardware dependent software.

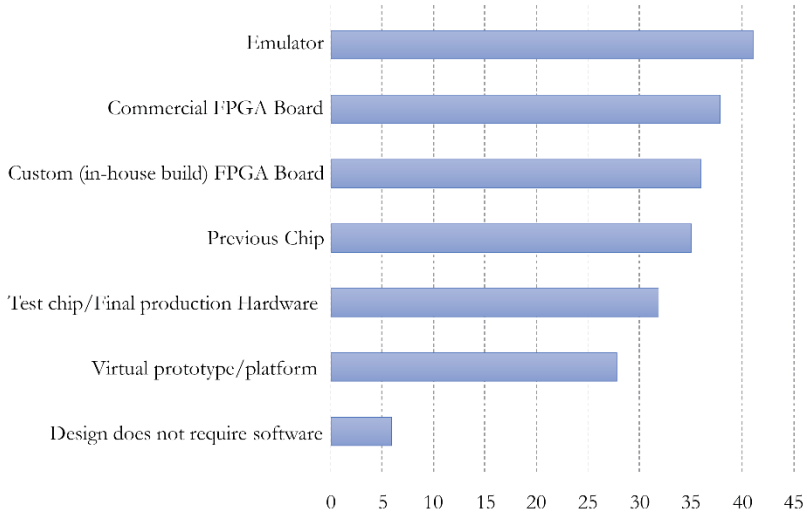


Figure A.0.1 – Poll on methodologies used to develop hardware dependent software

The above chart clearly indicates that among all the prototyping solutions used in the industry, there's no preferential choice for users. This may depend on the different pros and cons of each prototyping solution, which frequently leads to the usage of multiple tools during the different stages of the product development in order to take advantage of the peculiar benefits of each of them. Of course, the main impediment to the implementation of a hybrid approach is its high cost.

It seems now interesting to recall the particularities of the main prototyping solutions cited in the above chart, that are virtual prototypes, emulators and FPGA-based prototypes (being them based on commercial development kits or custom boards). These three solutions share their independence from the actual silicon implementation of the SoC being designed, such as test chips or previous generation chips.

1. Virtual prototypes are prototyping solutions that can be employed even in the earliest stages of the design flow. They can be used to model, in software, the entire design (e.g. SoCs, boards, IOs, user interfaces) but are not suited to perform cycle accurate investigations. Virtual prototypes use instruction set simulators to execute actual software for the target architecture, and can provide several tens of MIPS when timing accuracy is not required. It is the case of virtual prototypes shipped with software development kits (SDKs), where the main aim is to let third-party developers programming and interfacing to a high-level application programming interface (API). In this case, the prototype must provide a sufficient level of detail to provide what the authors summarize as "just enough accuracy" in order to fool the application into thinking that it is running on the final platform. Virtual prototypes are the best when it comes to development cost and deployment cost considerations, and this is because they are also needed to empower the ecosystem of third-party developers and thus must be replicable as much as possible.

2. Emulators constitute a class of hardware-based prototyping solutions where RTL designs are mapped into a dedicated hardware in order to improve compile time

and get powerful insights about the behavior of the design during code execution. It must be noted that the execution speed of emulators is extremely low (less than 10 MIPS) if compared to simulators, since they are based on a much more detailed model of the system. Emulators are preferred when an improved degree of accuracy is needed during execution. Emulators are not able to empower hardware/software co-development strategies since they need a finalized RTL to work (and thus a completely specified hardware design). They are especially useful to extend the verification environment and allow low-level software development, such as boot ROM code, due to the slow running speed of its execution that cannot support actual OS bootstrap. Emulators are the best when it comes to accuracy and debug insight, and this is the reason why they are used as a tool for early RTL verification.

3. FPGA-based prototypes are another class of hardware-based prototyping solutions that implement powerful tools to empower software development and validation, when a finalized RTL is ready for the design. FPGAs are able to run designs at almost real-time speed (tens of MIPS) while maintaining the same accuracy of emulators. This is because in an FPGA the RTL design is mapped to a certain number of cells in actual silicon. Moreover, FPGA-based prototypes can be coupled to an extensive number of interfaces and stimulus, in order to develop a prototyping environment that is as close as possible to the final product. However, FPGA-based prototypes are not able to reproduce the insights that an emulator can provide. This is the reason why they are used in latter stages of the

> RTL verification, such as when the RTL is relatively mature. Thus, they are not suited for hardware/software co-design. Another reason for this choice is that porting an RTL to FPGA-based prototypes can be a complex task, and when an FPGA-based prototype is already working for a given design, it is unlikely that the hardware design would be subject to further modifications. FPGA-based prototypes are cheaper than emulators, but their price is sufficiently high to inhibit extensive deployments as it can be the case of software-based prototyping such as simulators. FPGA-based prototypes are the best when it comes to accuracy, execution speed and system interfaces, and this is the reason why they are used as a tool for hardware-software introduction, where tricky bugs can arise that must be identified before fabrication. Due to their cycle-accurate yet fast model execution, FPGA-based prototypes are the most representative prototypes of the final design. For this reason they are extensively used in the industry to prepare entire hardware and software ecosystems around the novel SoC design, in such a way that when actual chips are available, everything is ready to put them running.

The authors state that FPGA-based prototypes are best suited when it comes to: i) high performance and accuracy; ii) interfacing with the real world; iii) in-lab usage; iv) out-of-lab demonstration; v) extended RTL test and debug. The authors also assert that these needs are typically related to: i) test real-time dataflow; ii) early hardware-software integration; iii) early software validation; iv) test real-wold data effects; v) test real-time human interface; vi) debug rare data dependencies; vii) proof-of-concept; viii) testing algorithms; ix) public exhibitions; x) encourage investors. The above list of use cases prefigures that FPGA-based prototypes

are useful under both an engineering and a business perspective. In fact, these prototypes are the closest one to the final product when considering the latter stage of its development, allowing accurate testing and verification into the lab (even through the interface to external components) as well as the implementation of full-featured demonstrators. However, FPGA-prototyping are also suited to implement an early stage proof-of-concept of a given subsystem, such as the case of this research effort.

It can be said that FPGA-prototyping is the most suited for exploratory activities that lack significant funding, if considering the cost of commercial development kits available on the market. These kits are perfect to test the behavior of circuits in a real-world scenario, starting from small subsystems and ending with complex logic architectures, overcoming the need for accurate, yet time-consuming, SPICE simulations. Moreover, they can be extended with daughter cards to connect to external devices, such as mixed-signal ICs (e.g. data converters). This leads to the prototypal implementation of complete solutions that are extremely close to the behavior of an ASIC. This is the reason why the prototypes implemented during the experimental work reported in this thesis are based on FPGAs.

Prototyping is an essential strategy in today's product development. Figure A.0.2 depicts a qualitative overview of the cost of fixing a problem (e.g. a bug), when identified during different stages of a typical IC development cycle.

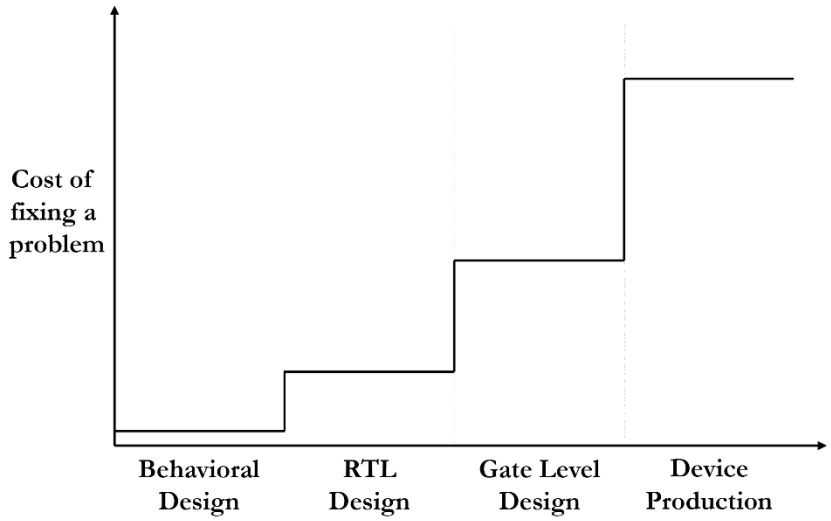


Figure A.0.2 – Qualitative overview of the cost of fixing a problem when identified during different stages of a typical IC development cycle

If a bug discovered during behavioural or RTL design stage (where prototyping is capable to provide most of its benefits) is relatively not harmful, this cannot be said when the IC is already in production (e.g. fabrication of test chips). This is because the production stage is costly from an economical and a temporal point of view. When a proper verification strategy, according to several forms of prototyping, is implemented within the RTL design stage, chances of a successful product implementation are maximized.

A.2 IP core based design

As the number of functions getting integrated into single chip solutions (such as SoCs) increases, design reuse is becoming essential to accelerate TTM, through the seamless integration of formerly designed and verified components and subsystems. These reusable

units take the name of Intellectual Property (IP) cores and implement logic, cells or chip layouts, which are among the most valuable assets of a design center. IP cores can be developed and reused internally, or licensed from other parties, such as companies that developed vertical knowledge into a particular domain. IP core based architectures can be pursued when making FPGA or ASIC designs, and IP cores are typically distributed as "soft", "hard" or "firm" IPs.

1. Soft IP cores are offered as synthesizable or synthesized RTL models, and shipped in technology independent hardware description language code (such as VHDL) or generic gate level netlists. Instead, hard IP cores are offered as layout designs, shipped in a layout format (such as GDS) bound to a specific ASIC foundry process (that cannot be changed without a redesign).

2. Hard IP cores are ready to be integrated in the final layout of chips, whereas soft IP cores need at least a synthesis, placement and route (SPR) flow to be ready for integration.

3. Firm IP cores constitutes an intermediate category that is not clearly defined in literature. Authors refer to firm IP cores as synthesizable or synthesized RTL models optimized for a target FPGA architecture (namely a soft IP core specialized for a certain FPGA vendor) [25], or as a soft IP core that is shipped with a preliminary form of layout, such as a floor-plan or a placement (but without routing).

Soft IP cores are the most suited for digital logic cores such as microprocessors, DSP blocks and cryptography whereas hard IP cores are the ones that are typically employed to ship

analog and mixed signal designs such as ADCs, DACs, PLLs and physical layer interfaces. Many soft IP cores are also "hardened" by vendors, in such a way that a hard IP core is pre-implemented for a given set of process technologies. This methodology extends the opportunities offered by the flexibility of soft IP cores through the advantages of hard IP cores, such as readiness for ASIC fabrication. For example, one can integrate a soft IP core into an FPGA-based prototype and then one of its hardened variants into the final chip layout.

Figure A.0.3 and Figure A.0.4 depict the design flow for soft and hard IP cores.

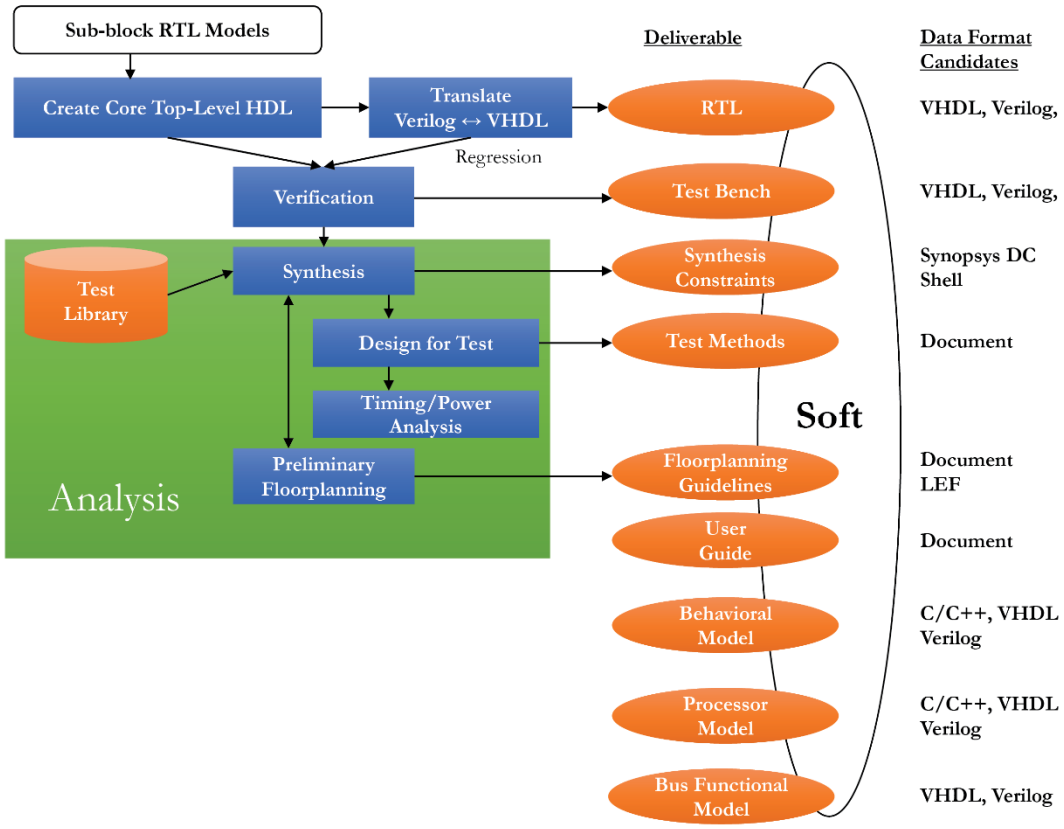


Figure A.0.3 – Design flow for soft IP cores

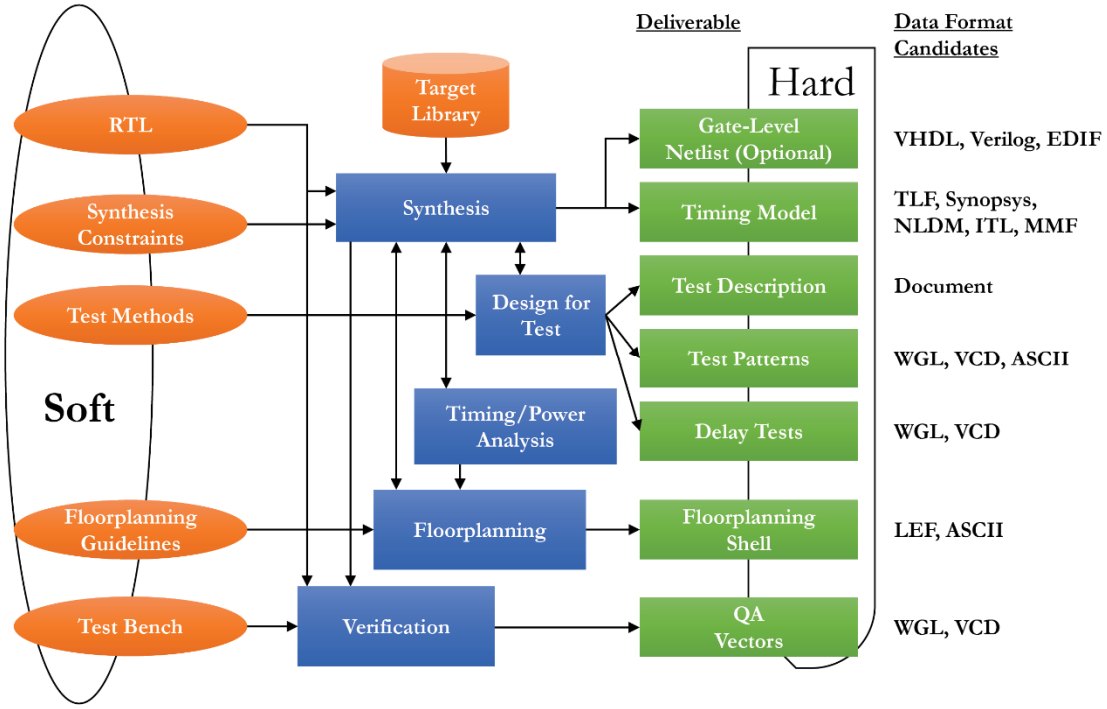


Figure A.0.4– Design flow for hard IP cores

It seems interesting to elaborate more on this industry trend, which sums up various other trends in the IC design market. First of all, nowadays the window of opportunity for a product is so tight that designs are developed with a first-to-market design-wins strategy in mind, which leads to design reuse as a mean to reach the market quickly in order to anticipate competitors. Then, the number of subsystems that are now part of a design (in particular a SoC design) is growing so much that design reuse can be interpreted as a way to move complex SoC designs toward a high-level integration effort. Moreover, the standardization of interfaces has also lead to this reuse opportunity, when considering the number of standardized components that are integrated into designs. Finally, the need for extremely

optimized products, the growing complexity of involved subsystems as well as the number of domains involved in modern SoC designs (requiring vertical knowledge in analog and mixed-signal design strategies) are driving the need for externalizing portions of the design to vendors that are specialized on a particular set of IPs.

The above considerations drove the experimental work conducted in this research. For example, the BSU architecture that has been prototyped had, as a functional requirement, the presence of a full-featured MCU capable to run general purpose firmware code. Implementing such logic subsystem would have been an extremely tough and time consuming task, especially if the final MCU had to be based on a custom architecture, not to consider the effort required to design a custom toolchain for firmware compilation. This is the reason why a third-party microcontroller IP core has been used to satisfy the above requirement. Such a strategy allowed to move from a design/implementation to an integration effort. This means that instead of developing the MCU from scratch, an IP core has been configured and interconnected to the other components of the design hierarchy. A reuse of existing code has thus been performed, following the relevant industry trend in SoC design that has been discussed above. It must be noted that open-source projects are becoming popular even in the area of IP core designs, inspired by the same ideals and licensing agreement contracts of the free software movement. These designs constitute an excellent opportunity for researchers, since they allow to gain an in-depth understanding of complex architectures that have been developed by formal or informal teams of volunteer engineers, and to develop customizations at any level of the IP core architecture. This is a tremendous advantage for researchers, which have the opportunity to focus on specific

improvements in a vast catalogue of subsystems, or to integrate these subsystems into novel solutions. Of course, the lack of open-source IPs would have been a difficult to overcome entrance barrier to many research groups, which could not be able to afford the cost of implementing every component from scratch. Many IP cores are also associated to the advantage of being silicon-proven. This means that a working layout for it has already been implemented, fabricated, tested and has been integrated into a final product that currently is (or that has been) in production. Since, as already mentioned, the fabrication stage costs money and time, silicon-proven IP cores are the most valuable for ASIC implementations, due to the fact that it is at least known that someone succeeded into taking them into an IC. Silicon-proven IP cores are often distributed as hardened IPs, however their cost can make customers take the choice of designing the layout on their own anyway. Open-source IP cores may also be related to silicon-proven designs, especially when their main contributors are formal teams. For such teams it is a common practice to distribute open-source soft IP cores for free (yet under a licensing agreement contract), and to provide their hardened implementation to customers that are willing to pay a royalty fee. This revenue is what sustains the open-source project, which acts as a form of knowledge sharing as well as a strategy to engage new customers.

Among the many IPs on the market, memory blocks (e.g. RAM, ROM …) are the most recurrent. In fact, being circuit designs ultimately organized as a dense replication of a base storage cell, their layout implementation is licensed by foundries and specialized companies. These IPs may be distributed in the form of fixed block sizes or can be synthesized through memory compilers, which are software tools capable to synthesize memory blocks for

various combination of foundries and technology nodes. The availability of these IPs is crucial to the development of a SoC, since common MCU implementations require at least internal program memory (ROM) and data memory (RAM).

References

[1] L. G. Maloratsky, "Electrically tunable switched-line diode phase shifters Part 1: Design procedure." in High frequency electronics, p.p. 16-21, 2010.

[2] K. Sharma, A. Basu and S. Koul, "High power 6-bit digital phase shifter for airborne IFF application," 2016 Asia-Pacific Microwave Conference (APMC), New Delhi, 2016, pp. 1-4.

[3] O. H. Karabey, F. Goelden, A. Gaebler, S. Strunck and R. Jakoby, "Tunable loaded line phase shifters for microwave applications," 2011 IEEE MTT-S International Microwave Symposium, Baltimore, MD, 2011, pp. 1-4.

[4] K. Miyaguchi et al., "A 6-18 GHz 5-Bit Phase Shifter MMIC Using Series/Parallel LC Circuit," 2002 32nd European Microwave Conference, Milan, Italy, 2002, pp. 1-4.

[5] M. Kim, J. B. Hacker, R. E. Mihailovich and J. F. DeNatale, "A DC-to-40 GHz four-bit RF MEMS true-time delay network," in IEEE Microwave and Wireless Components Letters, vol. 11, no. 2, pp. 56-58, Feb. 2001.

[6] H. Y. Jiang et al., "Microwave photonic phase shifter using a phase modulator and a fiber Bragg grating in a round-trip," CLEO: 2013, San Jose, CA, 2013, pp. 1-2.

[7] M. Burla et al., "CMOS-compatible integrated optical delay line for broadband Ku-band satellite communications," 2012 IEEE International Topical Meeting on Microwave Photonics, Noordwijk, 2012, pp. 120-123.

[8] D. Digdarsini, M. Kumar and T. V. S. Ram, "Design & hardware realization of FPGA based Digital Beam Forming system," 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN), Noida, 2016, pp. 275-278.

[9] C. Lu et al., "A 48 GHz 6-bit LO-path phase shifter in 40-nm CMOS for 60 GHz applications," 2013 Proceedings of the ESSCIRC (ESSCIRC), Bucharest, 2013, pp. 73-76.

[10] H. Hashemi, Xiang Guan, A. Komijani and A. Hajimiri, "A 24-GHz SiGe phased-array receiver-LO phase-shifting approach," in IEEE Transactions on Microwave Theory and Techniques, vol. 53, no. 2, pp. 614-626, Feb. 2005.

[11] J. Hasch, E. Topak, R. Schnabel, T. Zwick, R. Weigel and C. Waldschmidt, "Millimeter-Wave Technology for Automotive Radar Sensors in the 77 GHz Frequency Band," in IEEE Transactions on Microwave Theory and Techniques, vol. 60, no. 3, pp. 845-860, March 2012.

[12] B. C. Kane, L. A. Geis, M. A. Wyatt, D. G. Copeland and J. A. Mogensen, "Smart Phased Array SoCs: A Novel Application for Advanced SiGe HBT BiCMOS Technology," in Proceedings of the IEEE, vol. 93, no. 9, pp. 1656-1668, Sept. 2005.

[13] Intel, "Introduction to direct digital synthesis," in Application Note 101, Mar. 1990

[14] Analog Devices, "Fundamentals of Phase Locked Loops (PLLs)," in MT-086, 2009

[15] A. Bonfanti, F. Amorosa, C. Samori and A. L. Lacaita, "A DDS-based PLL for 2.4-GHz frequency synthesis," in IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 50, no. 12, pp. 1007-1010, Dec. 2003.

[16] L. Li, W. Hong, Y. Zhang, Z. Chen and P. Chen, "Design and Implementation of an Active Array Antenna With Remote Controllable Radiation Patterns for Mobile Communications," in IEEE Transactions on Antennas and Propagation, vol. 62, no. 2, pp. 913-921, Feb. 2014.

[17] G. Avitabile, F. Cannone and A. Vania, "Phase shifter based on DDS-driven offset-PLL," in Electronics Letters, vol. 42, no. 25, pp. 1438-1439, December 7 2006.

[18] G. Avitabile, F. Cannone and G. Coviello, "Low-power Tx module for high-resolution continuous programmable phase shift," in Electronics Letters, vol. 44, no. 18, pp. 1052-1053, August 28 2008.

[19] https://www.altera.com/en_US/pdfs/literature/ug/ug_altpll.pdf

[20] D. M. Calcutt, F. J. Cowan and G. H. Parchizadeh, "8051 Microcontrollers: hardware, software and applications," Elsevier, 1998

[21] Chu, W. C. C., "VLSI Implementation of 8051 MCU with In-System Programming." Intelligent Systems and Applications: Proceedings of the International Computer Symposium (ICS) Held at Taichung, Taiwan, December 12–14, 2014. Vol. 274. IOS Press, 2015

[22] http://www.analog.com/media/en/technical-documentation/data-sheets/ADF4116_4117_4118.pdf

[23] http://www.soccentral.com/results.asp?CatID=596&EntryID=30794

[24] D. Amos, A. Lesea and R. Richter, "FPGA-based prototyping methodology manual: Best practices in design-for-prototyping, " Synopsys Press, 2011

[25] http://www.embedded.com/design/configurable-systems/4218841/The-basics-of-core-based-FPGA-design--Part-1---core-types---trade-offs

Further references

[26] D. Parker and D. C. Zimmermann, "Phased arrays - part 1: theory and architectures," in IEEE Transactions on Microwave Theory and Techniques, vol. 50, no. 3, pp. 678-687, March 2002.

[27] D. Parker and D. C. Zimmermann, "Phased arrays-part II: implementations, applications, and future trends," in IEEE Transactions on Microwave Theory and Techniques, vol. 50, no. 3, pp. 688-698, March 2002.

[28] L. Josefsson, "A perspective on array antenna developments," 2006 First European Conference on Antennas and Propagation, Nice, 2006, pp. 1-7.

[29] P. Kasemir N. Sutton M. Radway B. Jeong T. Brown D. S. Filipovič "Wideband analog and digital beamforming" 2009 9th International Conference on Telecommunication in Modern Satellite Cable and Broadcasting Services pp. 372-375 2009.

[30] R. V. Garver, "Broad-Band Diode Phase Shifters," in IEEE Transactions on Microwave Theory and Techniques, vol. 20, no. 5, pp. 314-323, May 1972.

[31] M. Hangai, M. Hieda, N. Yunoue, Y. Sasaki and M. Miyazaki, "S- and C-Band Ultra-Compact Phase Shifters Based on All-Pass Networks," in IEEE Transactions on Microwave Theory and Techniques, vol. 58, no. 1, pp. 41-47, Jan. 2010.

[32] D. Adler and R. Popovich, "Broadband switched-bit phase shifter using all-pass networks," 1991 IEEE MTT-S International Microwave Symposium Digest, Boston, MA, USA, 1991, pp. 265-268 vol.1.

[33] C. F. Campbell and S. A. Brown, "A compact 5-bit phase-shifter MMIC for K-band satellite communication systems," in IEEE Transactions on Microwave Theory and Techniques, vol. 48, no. 12, pp. 2652-2656, Dec. 2000.

[34] D. Jagyasi, K.P. Ray, S. Chaudhary, S. K. Elissa, "Six Bit Digital Phase Shifter using Lumped Network for ST Radar", International Conference on Communication Technology, no. 3, 2013.

[35] M. Y. Frankel R. D. Esman M. G. Parent "Array transmitter/receiver controlled by a true time-delay fiber-optic beamformer" IEEE Photonics Technology Letters vol. 7 no. 10 pp. 1216-1218 Oct. 1995.

[36] X. Wang, E.H.W. Chan, R.A. Minasian, "Optical-to-RF phase shift conversion-based microwave photonic phase shifter using a fiber Bragg grating", Optics Letters, vol. 39, no. 1, pp. 142-145, 2014.

[37] J. Perruisseau-Carrier A.K. Skrivervik "Composite right/left handed transmission lines metamaterial phase shifters (MPS) in MMIC technology" IEEE Trans. on Microwave Theory and Techniques vol. 54 no. 4 pp. 1582-1589 Apr. 2006.

[38] W. Hu, O. H. Karabey, A. Gäbler, A. E. Prasetiadi, M. Jost and R. Jakoby, "Liquid crystal varactor loaded variable phase shifter for integrated, compact, and fast beamsteering antenna systems," 2014 44th European Microwave Conference, Rome, 2014, pp. 1604-1607.

[39] E. Pels W. Liang "A method of array steering by means of phase control through heterodyning" IRE Transactions on Antennas and Propagation vol. 10 no. 1 pp. 100-100 January 1962.

[40] P.-S. Wu H.-H. Chang M.-D. Tsai T.-W. Huand H. Wang "New miniature 15–20-GHz continuous-phase amplitude control MMICs using 0.18-μm CMOS technology" IEEE Trans. on Microwave Theory and Techniques vol. 54 no. 1 pp. 10-19 Jan. 2006.

[41] H. Kwon H. Lim B. Kang "Design of 16–18 GHz wideband phase shifters using radial stubs" IEEE Microwave and Wireless Components Lett. vol. 17 no. 3 pp. 205-107 Mar. 2007.

[42] D. Viveiros J.D. Consonni A.K. Jastrzebki "A tunable all-pass MMIC active phase shifter" IEEE Trans. on Microwave Theory and Techniques vol. 50 no. 8 pp. 1885-1889 Aug. 2002.

[43] B. Lakshminarayanan T.M. Weller "Design and modeling of a 4-bit slow-wave MEMS phase shifters" IEEE Trans. on Microwave Theory and Techniques vol. 54 no. 1 pp. 120-127 Jan. 2006.

[44] K. Topalli, M. Unlu, O. A. Civi, S. Demir, S. Koc and T. Akin, "A monolithic phased array using 3-bit DMTL RF MEMS phase shifters," 2006 IEEE Antennas and Propagation Society International Symposium, Albuquerque, NM, 2006, pp. 517-520.

[45] M. Longbrake "True time-delay beamsteering for radar" 2012 IEEE National Aerospace and Electronics Conference (NAECON) pp. 246-249 2012.

[46] C. Fulton, M. Yeary, D. Thompson, J. Lake and A. Mitchell, "Digital Phased Arrays: Challenges and Opportunities," in Proceedings of the IEEE, vol. 104, no. 3, pp. 487-503, March 2016.

[47] A. Garrod "Digital module for phased array radar" Proc. IEEE Int. Symp. Phased Array Syst. Technol. pp. 81-86 1996-Oct.

[48] W. Shang, Z. Dou, W. Xue and Y. Li, "Digital beamforming based on FPGA for phased array radar," 2017 Progress In Electromagnetics Research Symposium - Spring (PIERS), St. Petersburg, 2017, pp. 437-440.

[49] X. Guan H. Hashemi A. Hajimiri "A fully integrated 24-GHz eight-element phased-array receiver in silicon" IEEE J. Solid-State Circuits vol. 39 no. 12 pp. 2311-2320 Dec. 2004.

[50] A. Natarajan A. Komijani X. Guan A. Babakhani A. Hajimiri "A 77-GHz Phased-Array Transceiver With On-Chip Antennas in Silicon: Transmitter and Local LO-Path Phase Shifting" IEEE Journal of Solid-State Circuits vol. 41 no. 12 pp. 2807-2819 Dec. 2006.

[51] C. Serrano M. Diaz A. Gomez M. Arranz S. Tenorio "Active antenna—Efficient mobile broadband networks" 73rd IEEE Veh. Technol. Conf. pp. 1-4 2011.

[52] M. Caretti M. Crozzoli G. M. Dell'Aera A. Orlando "Cell splitting based on active antennas: Performance assessment for LTE system" IEEE 13th WAMICON 2012-Apr.

[53] Z. Q. Yu J. Y. Zhou J. N. Zhao T. Zhao W. Hong "Design of a broadband MIMO RF transmitter for next-generation wireless communication systems" Microwave J. vol. 53 pp. 22-26 Nov. 2012.

[54] R. Fantacci D. Marabissi "Cognitive Spectrum Sharing: An Enabling Wireless Communication Technology for a Wide Use of Smart Systems" Future Internet 2016 vol. 8 no. 2 pp. 23.

[55] G.L. Stuber J.R. Barry W Y. Li A. Ingram T.G. Pratt "Broadband MIMO-OFDM wireless communications" Proc. IEEE vol. 92 no. 2 pp. 271-294 Feb. 2004.

[56] D.G. Rahn M.S. Cavin F.F. Dai N.H.W. Fong R. Griffith J. Macedo A.D. Moore J.W.M. Rogers M. Toner "A fully integrated multi-band MIMO WLAN tranceiver RFIC" EEE J. Solid State Circuits vol. 40 no. 8 pp. 1629-1641 Aug. 2005.

[57] D. Yang, Z. Zhu and P. Liu, "Design of 14–14.5GHz semiconductor microwave digital phase shifters," 2016 17th International Conference on Electronic Packaging Technology (ICEPT), Wuhan, 2016, pp. 1244-1246.

[58] G.M. Rebeiz K-J. Koh "Silicon RFICs for Phased Arrays" IEEE Microwave pp. 96-103 May 2009.

[59] A. Hajimiri, "mm-Wave Silicon ICs: Challenges and Opportunities," 2007 IEEE Custom Integrated Circuits Conference, San Jose, CA, 2007, pp. 741-747.

[60] Y. Zheng C.E. Saavedra "An ultra-compact CMOS variable phase shifter for 2.4-GHz ISM Applications" IEEE Trans. on Microwave Theory and Techniques vol. 56 no. 6 pp. 1349-1354 June 2008.

[61] L. Wu, A. Li and H. Luong, "A 4-path 42.8-to-49.5GHz LO generation with automatic phase tuning for 60GHz phased-array receivers," 2012 IEEE International Solid-State Circuits Conference, San Francisco, CA, 2012, pp. 270-272.

[62] S. Patnaik and R. Harjani, "A 24-GHz phased-array receiver in 0.13-μm CMOS using an 8-GHz LO," 2010 IEEE Radio Frequency Integrated Circuits Symposium, Anaheim, CA, 2010, pp. 465-468.

[63] L. Xie, X. Yin, L. Yang, C. Lu and H. Zhao, "Multi-channel phased signal generator for scalable active antenna array," Proceedings of 2014 3rd Asia-Pacific Conference on Antennas and Propagation, Harbin, 2014, pp. 87-89.

[64] B. Floyd, A. Sarkar, K. Greene and Y. Yeh, "Toward efficient, reconfigurable, and compact beamforming for 5g millimeter-wave systems," 2017 IEEE Bipolar/BiCMOS Circuits and Technology Meeting (BCTM), Miami, FL, 2017, pp. 66-73.

[65] N. Ebrahimi, M. Bagheri, P. Wu and J. F. Buckwalter, "An E-band, scalable 2×2 phased-array transceiver using high isolation injection locked oscillators in 90nm SiGe BiCMOS," 2016 IEEE Radio Frequency Integrated Circuits Symposium (RFIC), San Francisco, CA, 2016, pp. 178-181.

[66] J. Zhang, E. H. W. Chan, X. Wang, X. Feng and B. Guan, "Broadband Microwave Photonic Sub-Harmonic Downconverter with Phase Shifting Ability," in IEEE Photonics Journal, vol. 9, no. 3, pp. 1-10, June 2017, Art no. 5501910.

[67] M. Elmer, B. D. Jeffs, K. F. Warnick, J. R. Fisher and R. D. Norrod, "Beamformer Design Methods for Radio Astronomical Phased Array Feeds," in IEEE Transactions on Antennas and Propagation, vol. 60, no. 2, pp. 903-914, Feb. 2012.

[68] W. Yang, J. Zhou, K. Zhou, Z. Yu, W. Qin and J. Chen, "An active antenna array based on frequency shiftable vector modulator module," 2016 IEEE MTT-S International Wireless Symposium (IWS), Shanghai, 2016, pp. 1-4.

[69] C. Shu, L. Tian, C. Guo and W. Hong, "Design and implementation of a multi-channel frequency synthesizer for 5G wireless communication systems," 2018 IEEE MTT-S International Wireless Symposium (IWS), Chengdu, 2018, pp. 1-4.

[70] G. Serafino et al., "Photonics-Assisted Beamforming for 5G Communications," in IEEE Photonics Technology Letters, vol. 30, no. 21, pp. 1826-1829, 1 Nov.1, 2018.

[71] G. Babur, G. O. Manokhin, A. A. Geltser and A. A. Shibelgut, "Low-Cost Digital Beamforming on Receive in Phased Array Radar," in IEEE Transactions on Aerospace and Electronic Systems, vol. 53, no. 3, pp. 1355-1364, June 2017.

[72] X. Zhang et al., "Beam-based vehicular position estimation in 5G radio access," 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona, 2018, pp. 1-6.

[73] C. Ding, Y. J. Guo, P. Qin and Y. Yang, "A compact phase-shifting unit for phased array antennas," 2014 44th European Microwave Conference, Rome, 2014, pp. 1746-1749.

[74] C. Rowen, "Reducing SoC simulation and development time," in Computer, vol. 35, no. 12, pp. 29-34, Dec. 2002.

[75] Heng-Chia Chang, "Analysis of coupled phase-locked loops with independent oscillators for beam control active phased arrays," in IEEE Transactions on Microwave Theory and Techniques, vol. 52, no. 3, pp. 1059-1066, March 2004.

[76] Analog Devices, "Fundamentals of Direct Digital Synthesis (DDS)," in MT-085, 2009

[77] W. Kester, "Converting Oscillator Phase Noise to Time Jitter," in MT-008, 2009

[78] R. Cushing, "Replacing or Integrating PLL's with DDS solutions," 1999

[79] I. Collins, "Selecting PLLs for timing and phase control," Feb. 2006

[80] J. Vankka K. Halonen Direct Digital Synthesizers: Theory Design and Applications USA MA Norwell:Kluwer Academic 2001.

[81] R. Suryavanshi, S. Sridevi and B. Amrutur, "A comparative study of direct digital frequency synthesizer architectures in 180nm CMOS," 2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS), Vellore, 2017, pp. 1-5.

[82] L. Li W. Hong P. Chen "Design and implementation of an active antenna array for TD-LTE system based on DDS phase shifter" 2013 Asia-Pacific Microwave Conference Proceedings (APMC) pp. 672-674 2013.

[83] J. Lu M. Wu X. Jin Z. Fang "Active phased array antenna based on DDS" Proc. IEEE Int. Phased Array Syst. Technol. Symp. pp. 511-516 2003-Oct.

[84] Yinghao Zhang Wen Wu Guo Yang Jianzhong Zhao "DDS-PLL phased source for beam control active phased array" 2008 Asia-Pacific Microwave Conference pp. 1-4 2008.

[85] G. Avitabile G. Coviello F. Cannone "An improved Architecture for High-phase Resolution Phased Array Transmitters" Proceedings of the 3rd International Conference on Circuits Systems Communications Computers and Applications (CSCCA'14) pp. 104-112 22-24 Nov. 2014.

[86] J. Zhang, R. Zhang and Y. Dai, "Design and FPGA implementation of DDS based on waveform compression and Taylor series," 2017 29th Chinese Control And Decision Conference (CCDC), Chongqing, 2017, pp. 1301-1306.

[87] Mu Xuehua and Wang Jinzhang, "Development of frequency synthesizer based on DDS+PLL," Proceedings of 2011 IEEE CIE International Conference on Radar, Chengdu, 2011, pp. 1251-1254.

[88] A. Bonfanti, D. De Caro, A. D. Grasso, S. Pennisi, C. Samori and A. G. M. Strollo, "A 2.5-GHz DDFS-PLL With 1.8-MHz Bandwidth in 0.35-um CMOS," in IEEE Journal of Solid-State Circuits, vol. 43, no. 6, pp. 1403-1413, June 2008.

[89] K. Wan and Q. Xue, "Indirect Controlled Phased Source," in IEEE Microwave and Wireless Components Letters, vol. 16, no. 12, pp. 702-704, Dec. 2006.

[90] A.J. Salim, M. Othman, and M.A. Mohd Ali, "8051-based Digital Signal Controller in Xilinx FPGA", in International Conference on Computer and Communication Engineering 2006 ICCCE'06, K.L. May 2006, pp. 1141-5.

[91] S. Saponara, L. Fanucci and P. Terreni, "Architectural-Level Power Optimization of Microcontroller Cores in Embedded Systems," in IEEE Transactions on Industrial Electronics, vol. 54, no. 1, pp. 680-683, Feb. 2007.

[92] A. Milenkovic and D. Fatzer, "Teaching IP core development: an example," Proceedings 2003 IEEE International Conference on Microelectronic Systems Education. MSE'03, Anaheim, CA, USA, 2003, pp. 16-17.

[93] I. Mezzah, H. Chemali, S. Mezzah, O. Kermia and O. Abdelmalek, "MCIP: High configurable 8-bit microcontroller IP-core," 2015 Science and Information Conference (SAI), London, 2015, pp. 1387-1390.

[94] Y. Soliman and R. Mason, "Phase calibration techniques for injection-locking LO-path based phase-shifting phased-array architectures," 2012 IEEE Radio and Wireless Symposium, Santa Clara, CA, 2012, pp. 415-418.

[95] R. Sorace, "Phased array calibration," in IEEE Transactions on Antennas and Propagation, vol. 49, no. 4, pp. 517-525, April 2001.

[96] R. Long, J. Ouyang, F. Yang, W. Han and L. Zhou, "Fast Amplitude-Only Measurement Method for Phased Array Calibration," in IEEE Transactions on Antennas and Propagation, vol. 65, no. 4, pp. 1815-1822, April 2017.

[97] J. Schuss et al., "Large scale phased array calibration," 2016 IEEE International Symposium on Phased Array Systems and Technology (PAST), Waltham, MA, 2016, pp. 1-2.

List of my publications in this research area

[98] Tatoli, T.; D'Amato, G.; Conteduca, D.; Innone, F.; De Palo, M.; Dell'Olio, F.; Ciminelli, C.; Avitabile, G.; Armenise, M. N., "Optoelectronic beamformer for spaceborne X-band synthetic aperture radar," in GE 2015 Poster Session, 24-26 Jun. 2015

[99] Talarico, C.; D'Amato, G.; Coviello, G.; Avitabile, G., "A high precision phase control unit for DDS-based PLLs for 2.4-GHz ISM band applications," in Circuits and Systems (MWSCAS), 2015 IEEE 58th International Midwest Symposium on , vol., no., pp.1-4, 2-5 Aug. 2015

[100] D'Amato, G.; Avitabile, G.; Coviello, G.; Talarico, C., "A beam steering unit for active phased-array antennas based on FPGA synthesized delay-lines and PLLs," in Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), 2015 International Conference on , vol., no., pp.1-4, 7-9 Sept. 2015

[101] Coviello, G.; Avitabile, G.; Piccinni, G.; D'Amato, G.; Talarico, C., "Effects on phased arrays radiation pattern due to phase error distribution in the phase shifter operation.," in 20th International Conference on Circuits, Systems, Communications and Computers, vol. 76, pp.1-6, 14-17 Jul. 2016

[102] D'Amato, G.; Avitabile, G.; Coviello, G., "Design and Implementation of a Novel ECG Teleconsultation Platform," in International Journal of Communications, 1, 173-177

[103] Talarico, C.; D'Amato, G.; Avitabile, G.; Piccinni, G.; Coviello, G., "A systematic design approach for nanoscale inductor-less regulated cascode stages," in Integrated Circuits and Systems Design, 29th Symposium on, vol., no., pp.1-5, 29-3 Aug.-Sep. 2016

[104] D'Amato, G.; Talarico, C.; Avitabile, G.; Coviello, G., "A fully integrated 2.4GHz phase shifter for an enhanced SoC solution," in Very Large Scale Integration (VLSI-SoC), 24th IFIP/IEEE International Conference on, vol., no., pp.1-2, 26-28 Sep. 2016

[105] D'Amato, G.; Talarico, C.; Avitabile, G.; Coviello, G., "Toward a Novel Architecture for Beam Steering of Active Phased-Array Antennas," in Circuits and Systems (MWSCAS), 2016 IEEE 59th International Midwest Symposium on , vol., no., pp.1-4, 16-19 Oct. 2016

[106] G. D'Amato, G. Avitabile, G. Coviello and C. Talarico, "A DDS-PLL beam steering polar transmitter for narrowband IoT communications," 2017 2nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), Split, 2017, pp. 1-4.

[107] G. D'Amato, G. Avitabile, G. Coviello and C. Talarico, " An FPGA-based beam steering transceiver prototype for narrowband IoT communications," ECBS '17 Proceedings of the Fifth European Conference on the Engineering of Computer-Based Systems, Article No. 22

[108] Coviello, G.; Avitabile, G.; Piccinni, G.; D'Amato, G.; Talarico, C., " Comprehensive study on the role of the phase error distribution on the performances of the phased arrays systems based on a behavior mathematical model.," in WSEAS Transactions on Circuits and Systems, Volume 16, 2017, pp. 55-63

[109] G. D'Amato, G. Avitabile, G. Coviello and C. Talarico, " A modulator-less Beam Steering Transmitter based on a revised DDS-PLL Phase Shifter Architecture," in Journal of Communications Software and Systems, vol. 14, no. 1, pp. 1-9, Mar. 2018.

[110] G. D'Amato, G. Piccinni, G. Avitabile, G. Coviello and C. Talarico, "An Integrated Phase Shifting Frequency Synthesizer for Active Electronically Scanned Arrays," 2018 IEEE 21st International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), Budapest, 2018, pp. 91-94.

[111] G. D'Amato, G. Avitabile, G. Coviello and C. Talarico, "A beam steering transmitter prototype for IoT communications," 2018 3nd International Multidisciplinary Conference on Computer and Energy Science (SpliTech), Split, 2018, pp. 1-4.