UNIVERSITY OF TWENTE

BACHELOR ASSIGNMENT

ELECTRICAL ENGINEERING

Design and Calibration of a 4 Channel Programmable Phase Shifter ($2 - 6GHz$)

Author
R.C.M. BATTISTIG

Supervisor
S. GOLABIGHEZELAHMAD
Dr.Ing. E. KLUMPERINK
Dr.Ing. P.T. DE BOER

July 5, 2018

Abstract

Modern day high frequency phase shifters are often limited in bandwidth, as the phase delays implemented in classical phase shifter architectures are inherently frequency dependent. Furthermore, these phase shifters often lack in linearity and high phase resolution. In this paper a design is presented of a 4-channel phase shifter with high linearity and a simulated phase resolution of 2° , with an operational frequency of $2GHz$ to $6GHz$. The design overcomes the classical phase shifter issues of bandwidth limitations by opting for an In-phase-Quadrature (I-Q) addition architecture, of which components offer higher bandwidth capabilities. Implementation of 7-bit attenuators with a minimal attenuation step of 0.25dB allows the system to overcome systematic phase inaccuracies, after correct calibration. Notably, the model proposed in this work does not take into account any other inaccuracies besides the systematic phase and amplitude unbalance of the components. In case a better estimation of the path phase and amplitude discrepancies can be made, the more accurate estimation can be used in this work's proposed calibration method. The theory behind this calibration is explained and can be used for different design situation of an I-Q based phase shifter. The dependence of maximum phase error and the minimal attenuation step of the system is mathematically estimated in this work, and using numerical analysis this relationship was proven to hold for all attenuation steps tested. In the case of this system's attenuation step (0.25dB) the maximum phase error is $\pm 0.87^\circ$, regardless of the phase resolution.

. . . .

Acknowledgements

I am highly indebted to both my day to day supervisor *S.Golabighezelahmad* and *Dr.ing.E.Klumperink* for their guidance and supervision as well as for providing necessary information regarding the project.

Contents

1 Introduction

Phase shifters are a critical component in phased array systems. It is through the use of phase shifters that different phases are applied to phased array system in order to perform electronic beam-steering of the array. It is paramount for the proper operation of a phased array that the multitude of steering signals accurate both accurate in phase and amplitude. Unfortunately, phase array system often fall short because of narrow bandwidths and high cost [1]. A wide-band, cheap solution is thus needed.

In this paper, the design of a 4 channel programmable phase shifter will be discussed. The designed phase shifter will not be used to directly work within a phased array system, instead it will serve as a test bench to emulate the behavior of a phased transmitter and be used to test beam forming of a phased array receiver. The design is therefore tailored to be best suited for this application. To verify whether this is the case, a set of requirements are needed to fulfill the accuracy requirements of the system. These system requirements include: phase resolution, phase range, programmability, high linearity and no spurs. A detailed explanation about the design choices can be found in section 2.1.

Furthermore, in section 2 a quick overview is made about previously used Design Architectures, from which one design architecture is chosen. The chapter continues with an explanation of the relevant theory (I-Q modulation) and finishes with a summary of the component selections made. Based on these component selections a certain phase and amplitude unbalance between the different RF-paths can be estimated. Section 3 explains how the designed system can be calibrated regardless of the amplitude of these discrepancies. Section 4 contains a numerical example of the calculations proposed in the calibration section, these simulations are later discussed in section 5. Finally, a conclusion will be drawn in section 6 about to what extend calibration has been achieved and if the system requirements have been fulfilled accurately.

2 Design

2.1 System Requirements

The design choices made for this project are all motivated to function as an optimal test bench for a beam forming receiver. a closer look has to be taken of the system requirements. The system will serve to emulate different incoming wave patterns to an antenna array for testing purposes of the receiver. As such, the minimum system requirements for the phase shifter are:

- A total output of 4 independent RF signals.
- The 4 output signals should have a maximum amplitude difference of 0.5dB between the different channels.
- The output signals have an independent programmable phase shift.
- Phase resolution of better than 4°.
- Programmable Phase range of 360°.
- Output frequency of operation of $2GHz$ to $6GHz$.
- Spur level of less than -30dB.
- High linearity
- Conserve fully the modulation of the input signal in all output signals.

2.2 Design Architectures

Generally, phase shifters are often used to apply only one specific phase shift, possibly over a range of frequencies. A solution would be to have an array of possible phase shifting paths, and have programmable logic to connect or bypass certain paths to form different phase shifts. An example of this classical architecture using lumped networks can be found in [2]. The disadvantage of this system is that the lumped networks produce a highly frequency dependent phase shift, which causes the system to be very band-limited; in fact, the phase shifter in [2] was only tested at 212MHz. There are also more complex circuits which can replace the lumped networks, and be used to overcome the frequency dependence normally associated with lumped network phase shift. Unfortunately, this increase in complexity also greatly raises the price of this solution. In conjunction with the scarcity of phase shifting networks working at higher frequencies than $(1-2)GHz$, making a direct implementation of this architecture is not viable. An argument can be made to still use an implementation of this architecture at an intermediate frequency, where this operation would be viable, and then mix the signal to the required operational frequency. This approach would however still not be ideal, as the mixers would possibly introduce inter-modulation products, as well as a general increase in complexity which is otherwise not needed.

A second architecture for a programmable phase shift uses I-Q addition. An example of a digital modulator using I-Q addition is the work of Hua Wang [3]. In this work, Wang uses a set of linear digital variable gain amplifiers (VGA) to weight the corresponding I-Q channels so that the sum forms a known phase shifted signal. Wang's phase shifter, in comparison with the aforementioned lumped network phase shifter ([2]), has a considerable greater bandwidth of $7.6GHz$. Wang's work specifically is not applicable to our case scenario as it is designed as a 11.25 deg phase step. Additionally, although the use of VGA is one of the reasons the large bandwidth is achieved, these components have linearity problems, threshold voltage variations, and overall they increase the design complexity. Although Wang's architecture using VGA's is not ideal, it is an example of the wide bandwidth solutions possible with an I-Q based phase shifting architecture.

A third architecture which also uses I-Q addition is namely the phase shifter PCB used in the experimentation of the work of Linxiao Zhang [4]. The design is fundamentally different from Wang's work by employing passive 7-bit attenuators instead of the VGA approach. The use of attenuators from a transmission perspective is a disadvantage compared to the positive gain achievable by using VGA. However, the current design will be used for test purposes and is not focused on achieving high gain. The digital attenuator approach therefore simplifies Wang's approach to I-Q addition, and eliminates issues otherwise related to the use of VGA. One advantage is namely the high linearity achievable when using resistive attenuators over VGA. Zhang's work is however designed for lower frequencies ($500MHz$). Even though a direct implementation of Zhang's work is not possible, the architecture employed is capable of fully achieving the set system requirements; with the use of appropriate components with a higher RF operating frequency.

The architecture proposed by Zhang can be seen in figure 1. The design principle is to split the input RF signal into 4 channels, using a cascade of power splitters, and subsequently form the I and Q (basis vectors) which will be weighed appropriately using the attenuators, and finally combine the adjusted basis vectors. A final attenuator located at the end of each channel is placed to bring down the amplitude of the different channels to the same level. A more detailed description of the component properties, which are needed to fulfill the system requirements, will follow in the next section 2.4.

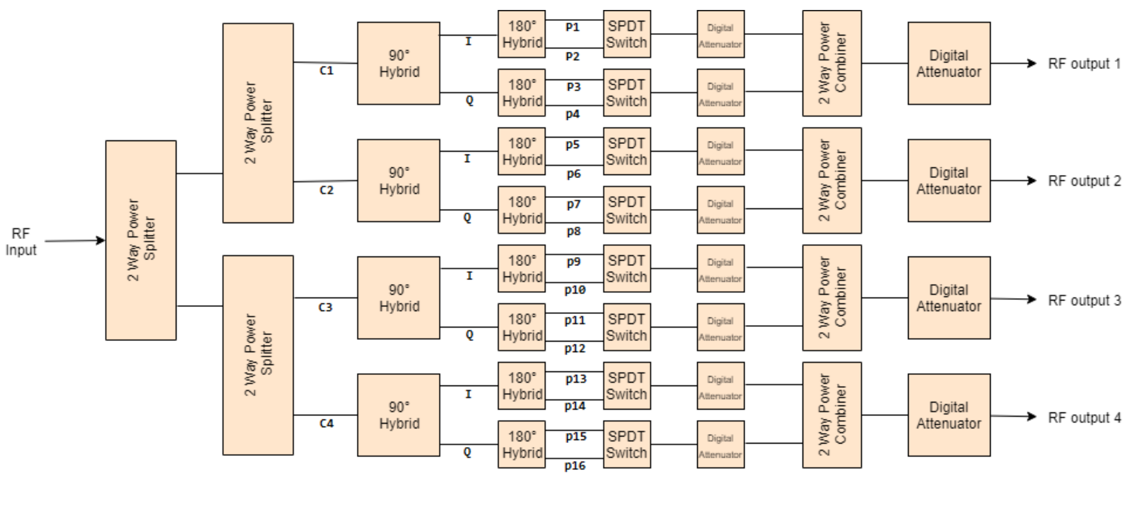


Figure 1: Block Diagram of 4-channel phase shifter, based on [4]

2.3 Vector Modulation

To better understand how vector modulation is utilized in the design to produce a specific phase shifted output signal, the theory of I-Q addition will be revisited. Fundamentally, I-Q addition is a normal vector addition, given the two added vectors have a 90° phase shift. The term vector refers to the signals amplitude and phase which characterizes the signal, and can therefore also be used to define a phasor/vector. When two signals are combined they superpose to form a new signal, as long as these two signals are both sinusoidal with the same frequency, the outcome will also be a sinusoid with the same frequency. Mathematically this can be expressed as seen in 1.

$$A\cos(\omega t+\alpha)+B\sin(\omega t+\beta)=\sqrt[2]{[A\cos(\alpha)+B\sin(\beta)]^2+[Asin(\alpha)-Bcos(\beta)]^2} \bullet \cos\left(\omega t+\tan^{-1}\left[\frac{A\sin\alpha-B\cos(\beta)}{A\cos\alpha+B\sin(\beta)}\right]\right) \quad (1)$$

Where A and B are the corresponding amplitude factors, α and β are the corresponding phases and ω is the angular frequency. As can be seen, the output phase as well as amplitude is well defined given that the initial phases and amplitudes are known. A special application of 1 is when the two vectors are exactly 90° out of phase. This scenario is what is referred to as I-Q addition and simplifies equation (1) as follows.

$$A\cos(\omega t)+B\sin(\omega t)=\sqrt[2]{A^2+B^2}\bullet\cos\left(\omega t+\tan^{-1}\left[\frac{B}{A}\right]\right) \quad (2)$$

By using vector addition the combined vector's phase can be controlled using the amplitudes of the input vectors according to equation (1). Ideally equation (2) could be used to perform this calculation, however, as the I-Q signals will have some deviation from the ideal 90° between each other, so equation 1 is used to account for this phase difference.

In the design proposed in (1) both I and Q signals can be attenuated using the digital attenuator. This choice is justified as it allows for two degrees of freedom (A and B) in equation 1. Unfortunately, as only passive attenuators are used, the amplitudes A and B can only be decreased. To gain further insight on how to control the output phase (θ) using the variables A and B, the behavior of the argument of the arctangent in 1 and 2 has to be understood. The general behavior of 1, for small α and β, can be simplified to the behavior of 3.

$$\theta=\tan^{-1}\left[\frac{B}{A}\right] \quad (3)$$

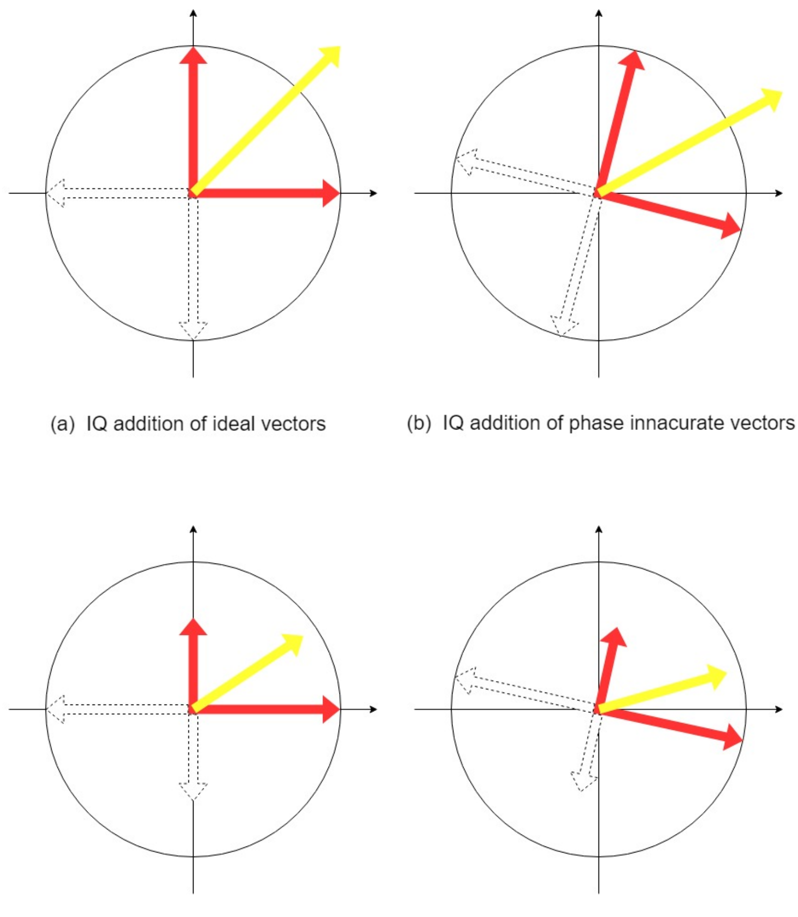
The arctangent function returns the value of $\frac{\pi}{4}$ for an input of 1, approaches 0 for smaller inputs, and grows to $\frac{\pi}{2}$ for higher inputs. From this behavior it can be deduced that attenuation of A will cause θ to approach $\frac{\pi}{2}$, likewise, attenuation of B will cause θ to approach 0. Furthermore, the two input signals should not be simultaneously be attenuated, as this will only cause the difference of $\frac{B}{A}$ to grow closer. In fact, attenuation of both signals at the same time would cause a needless loss in amplitude of the output signal, as can be deduced from either 1 and 2.

2.4 Design Overview

As the principle of operation of I-Q addition has been discussed, now the choices for the components will be reviewed. The design begins with one 2-way power splitter cascaded into other two 2-way power splitters to effectively create 4 equal power RF signal paths. This functionality could also be fulfilled in a single stage using a 4-way power splitter. Furthermore, there are commercially available 4-way power splitters which already apply phase shifts of 90° at each channel. Unfortunately, 4-way power splitters are very rare, as their functionality can be easily implemented using a cascade of 2-way power splitters, which are more readily available. Needless to say, for at the required frequency of operation, no commercially available 4-way power splitter was found. Instead the following 2-way power splitter was chosen, namely EP2W+ [5]. Since a 2-way option was ultimately chosen, it can be directly implemented also as the combiner to sum I-Q signals. The power splitter/combiner works at a frequency range of $0.7GHz$ to $6.0GHz$, with high isolation levels ($> 20dB$ at higher frequencies). The component is not perfect as there is a typical phase unbalance at $3-6GHz$ of 1.7°, furthermore, in the same frequency range the amplitude unbalance is $0.1dB$. These unbalances will cause an error in the phase shift. A visual representation of how phase and amplitude unbalances can affect I-Q addition can be found in figure 2.

After the four RF channels are generated, each signal has to be split up into base vectors for reconstruction of a specific phase shifted signal. Using a single 90° power splitter only two base vectors are created, commonly called I and Q. With only these two vectors a phase shift between 0° and 90° is possible. To increase the phase range of the system to 360° the design incorporates an additional 180° power splitter to generate the reversed polarity for I and Q signals, resulting in a total of four base vectors. It becomes apparent when searching for these components that wide-band options are scarce. Finally, QCH-63 was chosen as the 90° hybrid operating at $2-6GHz$ [6] . For the 180° hybrid a RF transformer was used to generate the inverse polarity signals of the I-Q signals [7], with a bandwidth of $1.7-6.7GHz$. Similar to the power splitter/combiner these components also have phase unbalance of a few degrees, as well as an amplitude unbalance of a few decibel. A complete description of the phase and amplitude unbalance will be given in section 3.

Once all I-Q basis vectors have been produced (four for each channel), the wanted I-Q basis vectors for addition have to be chosen. For each channel there are a total of 4 paths which can be added together, each (ideally at least) 90° phase shifted from one another. Not all path addition combinations form useful outputs, an example is if the I or Q signal are chosen to be added together with their inversed image, as the sinusoidal signals are ideally 180° out of phase this addition would lead to total destructive interference. The choice is therefore restricted to choosing between the original or inversed signal for both I and Q paths, namely a choice between the two terminals of the RF transformer (TCM2-672X+). Since this choice has to still be programmable to allow for a full 360° phase range, a digital single pull double throw (SPDT) switch was used. The



(a) IQ addition of ideal vectors

(b) IQ addition of phase innacurate vectors

(c) IQ addition of ampliude innacurate vectors

(d) IQ addition of phase and ampliude innacurate vectors

Figure 2: Example of Effects Phase and Amplitude Unbalances can Cause

Component Name	Phase Unbalance (°)	Amplitude Unbalance (dB)
RF transformer (TCM2-672X+)	1.00	-0.80
90 Hybrid (QCH-63)	3.50	-0.25
Power splitter/combiner (EP2W+)	0.04	-0.10

Table 1: Typical Phase and Amplitude Unbalance of Components at 5.8GHz

component chosen for this task is the HSWA2-63DR+ with a bandwidth of $0.1 - 6GHz$ with an isolation over that bandwidth of greater than $50dB$.

Finally, the last component is the digital attenuator. The important parameters of this component are the attenuation range and the attenuation step. The attenuation range for this application should be equal to attenuation needed when one of the I-Q signals has to be fully rejected. This figure is halted to the lowest isolation of the two signal paths, which in this case is the 90° power splitter with a typical isolation of 26dB. Any figure lower than that will deteriorate the signal summation further. The attenuation step on the other hand should be smallest possible. The reason being that the attenuation step represent the resolution at which the weighting factors of the IQ signals (A and B in equation 1) can be changed. A smaller attenuation step would allow for more precision in the choice of weighting factors. Taking these considerations into account the digital attenuator chosen is QPC6713 [8]. QPC6713 is a 7-bit attenuator with a attenuation range of 31.75dB and an attenuation step of 0.25dB, operating at a frequencies of $0.05 - 6GHz$.

In order to calibrate the system to overcome the phase and amplitude discrepancies created through the splitting processes in this system, the discrepancies have to first be defined. The other components (attenuator and SPDT switch) do not have phase nor amplitude unbalance as they are simply a 2 port system. In practice this is a simplification of the uncertainties present in the system, as there are additional discrepancies in phase and amplitude due to the manufacturing inaccuracies. These inaccuracies lead to each component have potentially different properties like phase and amplitude unbalance, as well as insertion loss. These inaccuracies are assumed to be negligible as there is insufficient data (besides maximum deviations) to model these inaccuracies. Instead, the typical values of the specified parameters were taken of the components from the their respective data sheets. It is important to note that these parameters are frequency dependent, an example of the calibration at 5.8GHz is shown, however if another frequency is chosen, the parameters have to be changed accordingly to what is specified for that particular frequency in the data sheets. The typical phase and amplitude imbalances of the splitting components are listed in table 1.

3 Calibration

For this system, the unideal properties are the phase and amplitude imbalances listed in table 1. These parameters represent the amplitude and phase differences of the second port compared to first port, where the second port has lower amplitude and a greater phase shift than it's ideal path phase shift; for example, according to these parameters the 90° hybrid produces an in phase signal and another with a 93.5° phase shift. By summing the phase and amplitude imbalances of the cascaded components the total path amplitude and phase unbalance can be calculated. This summation directly leads to a total path difference relative to the first path (where the paths are seen from top to down in figure 1). The total path amplitude error up to the first attenuators will be denoted as A_0 or B_0 and represents the attenuation of the path when the attenuator is set to 0dB. The total path phase error up to the first attenuators is on the other hand represented by α or β in equation (1), where α or β represent the phase offsets of the I-Q addition vectors. It is important to note that, for each in phase path an A_0 and α pair is be computed, and for each quadrature path a B_0 and β pair is computed. In total this will lead to a total of 4 weighting factors per channel, of which two A_0 and α pairs and two B_0 and β pairs.

In order to use the system, the required attenuation levels for the attenuators have to be set. The function of the final attenuators on each channel is solely to balance the amplitudes of all the channel outputs. To accomplish this the higher amplitude channels would have be attenuated to the level to match the lowest amplitude. In order to make this procedure automatic, the origin of the amplitude mismatch has to be taken into account. This origin is namely the different attenuation levels present in 4 channels, as they may be programmed to generate signals with different phase, meaning their weighting factors (A and B) are different. As can be seen in equation (1) this means that the final amplitudes (C) would differ as well. Luckily, the same equation (1) can be used to quantify this amplitude mismatch as seen in equation (4).

$$C = \sqrt[2]{[A\cos(\alpha) + B\sin(\beta)]^2 + [Asin(\alpha) - Bcos(\beta)]^2} \tag{4}$$

Using this formula it is possible to calculate all output amplitudes, given the attenuation levels of the initial set of attenuators (A and B) and the corresponding initial phase errors (α and β). This list can be used to refer the programmed output phase to the corresponding amplitude of that output signal. Knowing the output amplitudes the difference of each channel to the lowest amplitude of the four can be calculated and converted to dB to define the attenuation levels needed for the last attenuators of each channel.

The definition of the attenuation levels needed for the attenuators used to weigh the I and Q signals can be calculated in a similar manner from (1). The exact formula used in this case is the definition of the phase of the output signal, as seen in (5, which is an extended version of (3).

$$\theta = \tan^{-1}\left(\frac{A\sin\alpha - B\cos(\beta)}{A\cos\alpha + B\sin(\beta)}\right) \tag{5}$$

This equation can be used to calculate all possible output phases given the possible weighting factors (A and B). All the possible A and B values can be calculated from the initial amplitude differences A_0 or B_0 for each attenuation step of $-0.25dB$, with a maximum attenuation of $A_0 - 31.75dB$ or $B_0 - 31.75dB$.

Once all possible output phases have been calculated, the closest match to an interested output phase value can be found. Once these closest matches are found, the attenuation level required to produce that output phase can be reverse engineered. Finally, the outcome is a list of achievable output phases given an attenuation step of 0.25dB and their corresponding attenuation levels which produce them.

In the case that the attenuation step would be infinitely small, the system would be able to calibrate itself to all values of θ. As the attenuation step would always be a finite number, this is not possible, instead a discrete amount of θ's can be achieved. Therefore, there is a certain calibration error based on the difference of these discrete steps to the wanted θ, which is dependent on the attenuation step chosen. Intuitively, a higher attenuation step will lead to more closely packed discrete steps, thus a smaller maximum calibration error. Furthermore, it is known that the discrete segmentation of possible θ's is not linear, as the weighting factors (A and B) are reduced logarithmically. Even though the calibration error is easier to be found using numerical tools, an abstraction can be made to find the maximum calibration error possible depending on the attenuation step.

The maximum calibration error would occur at the largest discrete step between achievable θ's, the maximum error would be half this distance. This would occur between the first (0dB) attenuation level and the second one (-0,25dB for this specific system). Attenuation steps greater than the first step would cause an amplitude difference between the outcome and the last value that is smaller, as it is logarithmic scaling of attenuation. Using equation (5) to compute the first two achievable phases (θ_1 and θ_2), the distance of these phases can then be computed by finding their difference as shown in (6). In this equation A and B from (5) are set to the initial amplitude and phase uncertainties A_0 and B_0. Furthermore, attenuation of B is used as example, where the second attenuation level is defined as $B_0 * 10^{\frac{S}{10}}$, where S is the negative attenuation step in dB.

$$\theta_1 - \theta_2 = \tan^{-1}\left(\frac{A_0\sin\alpha - B_0\cos(\beta)}{A_0\cos\alpha + B_0\sin(\beta)}\right) - \tan^{-1}\left(\frac{A_0\sin\alpha - B_0 * 10^{\frac{S}{10}}\cos(\beta)}{A_0\cos\alpha + B_0 * 10^{\frac{S}{10}}\sin(\beta)}\right) \tag{6}$$

Using the inverse trigonometric identity of $\tan^{-1}$, equation (6) can be simplified to (5).

$$\theta_1 - \theta_2 \equiv \int_0^{\frac{A_0\sin\alpha - B_0\cos(\beta)}{A_0\cos\alpha + B_0\sin(\beta)}} \frac{1}{z^2+1}dz - \int_0^{\frac{A_0\sin\alpha - B_0 * 10^{\frac{S}{10}}\cos(\beta)}{A_0\cos\alpha + B_0 * 10^{\frac{S}{10}}\sin(\beta)}} \frac{1}{z^2+1}dz \tag{7}$$

$$\theta_1 - \theta_2 \equiv \int_{\frac{A_0 \sin\alpha - B_0 * 10^{\frac{S}{10}} \cos(\beta)}{A_0 \cos\alpha + B_0 * 10^{\frac{S}{10}} \sin(\beta)}}^{\frac{A_0 \sin\alpha - B_0 \cos(\beta)}{A_0 \cos\alpha + B_0 \sin(\beta)}} \frac{1}{z^2+1} dz \tag{8}$$

From equation (5) it can be seen that the initial hypothesis that a lower attenuation step leads to a lower maximum calibration error is correct. As S approaches 0, the lower bound of the integral in equation (5) approaches the upper boundary of the integral, meaning the integral area decreases. As the integral function is positive and real, the conclusion can be made that the distance $\theta_1 - \theta_2$ will decrease as well. Given the variables of the system designed, with $S = -0.25dB$, the maximum calibration error (over all channels) computed from equation (5) is 0.87°.

4 Simulations

The calculations described in section 3 were performed using Matlab R2016a. The script used to perform these calculations can be found in Appendix A. The script also includes possibility to vary the component phase and amplitude unbalances (for different frequencies), as well as the possibility to vary the attenuation step of the attenuator and the required phase resolution. Calculations are performed based on the values in table 1 as well as the attenuation step of -0.25dB.

First, all the possible attenuation steps are used to compute the achievable phases and amplitudes. These phases and amplitudes are shown in the form of an I-Q constellation shown in figure 3 of all channels combined. The channels are color coded, the red channel, channel 2, is identical to channel 3 and can therefore not be seen in the figure under the yellow channel 3.

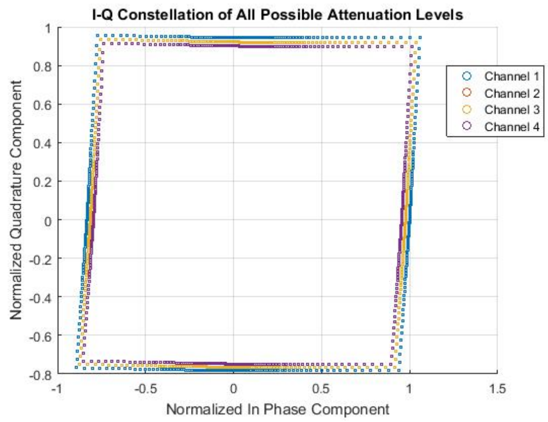


Figure 3: I-Q Constellation of All Possible Attenuation Levels

Using the possible output constellations, the nearest constellation points to a certain targeted phase are chosen. This calculation however requires a matrix of wanted phases, this means a required phase resolution has to be specified. As the estimated maximum error calculated in section 3 is 0.84°, the nearest integer is chosen which double of the error: a phase resolution of 2°. Based on this phase resolution the calibrated I-Q constellation of channel four is shown in figure 5. The calibrated I-Q constellations of the other three channels are fairly similar, and shown in appendix B.

From the calibrated constellation it is possible to find the error of this constellation to the ideally targeted phase. This error is shown for channel 4 in figure 5. The calibrated phase error of the other three channels are similar, and shown in appendix C.

5 Discussion

At first glance the I-Q constellation achieved in figure 3 seems odd, as generally I-Q constellations are altered to have a systematic shape (square or round). As is evident, this is not the case, instead slanted squares of different amplitudes are produced as a constellation. The slanted nature of the

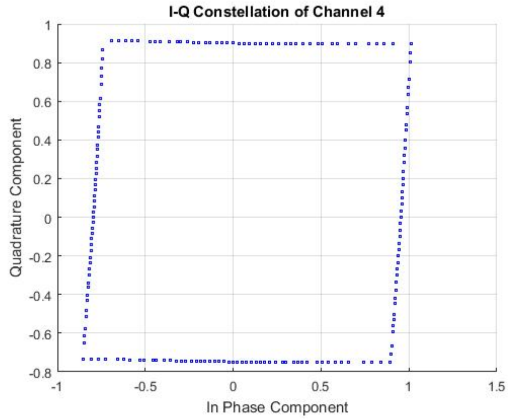


Figure 4: Calibrated I-Q Constellation of Channel 1 for a 2°phase resolution

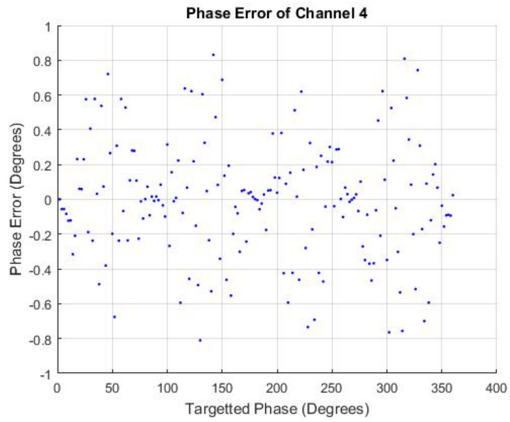


Figure 5: Phase error of channel 4 for a 2°phase resolution

constellation is due to the phase imbalance present prior to the attenuators. Similarly, the different square sizes represent the differences in the channel's attenuation imbalance. An example of how the possible I-Q constellation points would look like for an ideal system with no phase or amplitude imbalances is shown in appendix D, this example although not accurately portraying any uncertainties of the system, is a square constellation. Furthermore, even though this representation of the I-Q constellation is square, it only represents the I-Q addition constellation and not the complete output constellation. As the output constellation would be four points of this shown constellation (one for each channel) but then scaled in amplitude to be equal length vectors of the size of the shortest vector in the initial constellation. This means that the actual output constellation of this would be 4 points circularly placed around the origin.

It is evident from the phase error simulations shown in figure 5 that the previously hypothesized maximum calibration error from equation () is indeed accurate, as the maximum calibration error is under 0.87°. Another trend which is visible from this figure is that the phase errors seem to increase as they reach the corners of the I-Q constellation [45°, 135°, 225°, 315°] and decrease as the vertices of the I-Q constellations are reached [0°, 90°, 180°, 270°]. This phenomenon can be traced back to the constellation density shown in figure . The regions where there are more densely packed points means the possible resolution at that phase is higher, this increase of local resolution causes the calibration error at that phase to be lower.

Besides a general notion that the maximum calibration error 'decreases' or 'increases' depending on the targeted phase, a well defined trend can estimate this maximum error. Upon further empirical experimentation of calibration errors for systems with different attenuation steps, it was noticed that the maximum calibration errors seemingly follow sinusoidal pattern with a frequency of 2Hz and an amplitude as defined from equation (). To illustrate this pattern a boundary is sketched around figure 5, this example is shown in figure 6.

A further indirect effect of the logarithmically scaled attenuation levels is that the calibrated

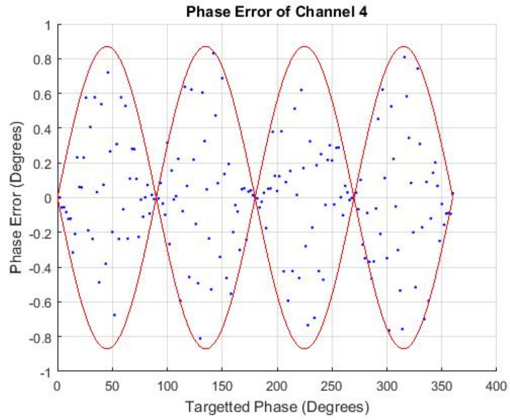


Figure 6: Example of sinusoidally bounded maximum calibration

I-Q constellation points are not linearly spaced. This means that some points are distanced closer together than others in the constellation. The closer the points the more susceptible the points are to become indistinguishable given unforeseen amplitude and phase noise. As no other uncertainties in phase and amplitude are modeled besides the phase and amplitude unbalance, it is possible that the non modeled sources of noise could cause the I-Q constellation to scramble and increase the phase error. Even though this system is not robust when faced with these additional phase and amplitude uncertainties, the uncertainties prior to the first attenuator can be measured as long as they are a systematic error and added to the initial phase and amplitude unbalances modeled. The new model will then calibrate the I-Q addition and formulate new attenuations levels to overcome these systematic errors. Unfortunately, using this method only systematic errors prior to the I-Q addition are calibrated. Therefore the errors generated through the last components (power combiners and attenuator) are not compensated for.

6 Conclusion

This paper aims to provide insight into important design choices of a programmable 4 channel phase shifter, as well as introduce a calibration technique applicable to all I-Q based phase shifters. The calibration aims to overcome the systematic errors in both phase and amplitude present in I-Q addition caused by the non-ideal splitting components. Through an analysis of different architectures, the I-Q addition architecture was chosen as this allowed for high linearity and could overcome band-limitations present in other architectures. Based on these components their amplitude and phase imbalance was used to directly estimate the systematic errors which are present in the system. As no experimentation of the system could be conducted yet, it is unsure how accurate this estimation of the systematic errors really is. In case the errors in this work are not accurate the calibration technique could still be used given the inputs of a better error estimation of each path.

A Matlab script was created to perform all the calculations needed for the calibration of the system. Based on these calculations an example of the I-Q constellation of the calibrated system with a phase resolution of $2°$ can be seen in figure 5. The resulting constellation also has a maximum phase error of $\pm 0.87°$. Based on these simulation results and the use of mathematically backed design choices, the system requirements listed in 2.1 have been fulfilled.

In addition to performing calculations made specifically for this design, the theory behind the used calibration method is extensive threated. This allows future designers of an I-Q addition based phase shifter to use this approach to advance their work. Designers can use equation (5) to quickly estimate the maximum phase error from their calibrated system based on their attenuation step. As well as directly implement the Matlab script for any change in amplitude and phase discrepancies, phase resolution and attenuation step.

References

[1] C. Wolff, "Radar basics - phased array antenna." http://www.radartutorial.eu/06.antennas/Phased%20Array%20Antenna.en.html. Accessed: 04-07-2018.

[2] D. Jagyasi, K. Ray, S. Chaudhary, and S. Krishan, "Six Bit Digital Phase Shifter Using Lumped Network for ST Radar," *International Journal of Computer Applications (0975 - 8887)*, 2013.

[3] H. Wang and A. Hajimiri, "A wideband cmos linear digital phase rotator," in *2007 IEEE Custom Integrated Circuits Conference*, pp. 671–674, Sept 2007.

[4] L. Zhang and H. Krishnaswamy, "Arbitrary analog/rf spatial filtering for digital mimo receiver arrays," *IEEE Journal of Solid-State Circuits*, vol. 52, pp. 3392–3404, Dec 2017.

[5] Mini-Circuits, "Datasheet of EP2W+." Available at https://ww2.minicircuits.com/pdfs/EP2W+.pdf.

[6] Mini-Circuits, "Datasheet of QCH-63." Available at https://ww2.minicircuits.com/pdfs/QCH-63.pdf.

[7] Mini-Circuits, "Datasheet of TCM2-672X+." Available at https://ww2.minicircuits.com/pdfs/TCM2-672X+.pdf.

[8] Mouser, "Datasheet of QPC6713." Available at https://nl.mouser.com/datasheet/2/412/QPC6713-1019059.pdf.

Appendices

A Matlab Script

```
clear all
close all

%for loop allowing for quick testing of different biases, Set by default to 1 for
normal application @5.8GHz.
for loop = 1:1

%Define Component Discrepancies
Pwr_phasebias = (loop-1)+0.04;
ninety_phasebias = 3.5;
hundredeighty_phasebias = 1;

Pwr_Attenuationbias = 0.1;
ninety_Attenuationbias = 0.25;
hundredeighty_Attenuationbias = 0.8;

%define attenuation step
n = 1;
step = 0.25/n;


amount = (31.75/step) + 1;

stepampl = 10^((-step)/10)

%define phase resolution in degrees
phase_res = 2;

%Compute maximum error

syms z;


%Calculate path Phase error (Alpha and Beta)
Phases = zeros(16,1);
Phase_matrix = zeros(16,4);
Phase_matrix = [0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1;0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1;
0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1;0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1]';

Phases(:,1)= Phase_matrix(:,1)*Pwr_phasebias + Phase_matrix(:,2)*Pwr_phasebias
+Phase_matrix(:,3)*ninety_phasebias + Phase_matrix(:,4)*hundredeighty_phasebias;
for xr = 1:8
    Phases((xr*2),1) = Phases((xr*2),1) + 180;
end

%Convert phase values from degrees to radians
Rads=(Phases(:)*pi)/180;


%Calculate path Attenuation error (A_0 and B_0)
Attenuation = zeros(16,1);
Attenuation_matrix = zeros(16,4);
Attenuation_matrix = - [0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1;
0,0,0,0,1,1,1,1,0,0,0,0,1,1,1,1;0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1;
0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1]';
```

```
Attenuation(:,1)= Attenuation_matrix(:,1)*Pwr_Attenuationbias + Attenuation_matrix(:,
2)*Pwr_Attenuationbias +Attenuation_matrix(:,3)*ninety_Attenuationbias
Attenuation_matrix(:,4)*hundredeighty_Attenuationbias;

%Convert Attenuation levels to Amplitude from dB to decimals
Amplitude= zeros(16,amount);
Amplitude(:,1)= 10.^((Attenuation(:))/10);

%Create all possible attenuation levels (A/B levels)
for n = 1:amount
    diff = 10^(((-step)*(n-1))/10);

    Amplitude(:,n)= Amplitude(:,1)* diff;
end

%Calculate output phases
for x = 1:4
    for y = 1:amount
        Out_phase(1+((x-1)*4),y) = (180/pi)*atan(((Amplitude(1+((x-1)*4),y)*sin(Rads
(1+((x-1)*4))))-(Amplitude(3+((x-1)*4),1)*cos(Rads(3+((x-1)*4)))))/((Amplitude(1+((x-
1)*4),y)*cos(Rads(1+((x-1)*4))))+(Amplitude(3+((x-1)*4),1)*sin(Rads(3+((x-1)*4))))));
        Out_phase(2+((x-1)*4),y) = (180/pi)*atan(((Amplitude(1+((x-1)*4),y)*sin(Rads
(1+((x-1)*4))))-(Amplitude(4+((x-1)*4),1)*cos(Rads(4+((x-1)*4)))))/((Amplitude(1+((x-
1)*4),y)*cos(Rads(1+((x-1)*4))))+(Amplitude(4+((x-1)*4),1)*sin(Rads(4+((x-1)*4))))));
        Out_phase(3+((x-1)*4),y) = (180/pi)*atan(((Amplitude(2+((x-1)*4),y)*sin(Rads
(2+((x-1)*4))))-(Amplitude(3+((x-1)*4),1)*cos(Rads(3+((x-1)*4)))))/((Amplitude(2+((x-
1)*4),y)*cos(Rads(2+((x-1)*4))))+(Amplitude(3+((x-1)*4),1)*sin(Rads(3+((x-1)*4))))));
        Out_phase(4+((x-1)*4),y) = (180/pi)*atan(((Amplitude(2+((x-1)*4),y)*sin(Rads
(2+((x-1)*4))))-(Amplitude(4+((x-1)*4),1)*cos(Rads(4+((x-1)*4)))))/((Amplitude(2+((x-
1)*4),y)*cos(Rads(2+((x-1)*4))))+(Amplitude(4+((x-1)*4),1)*sin(Rads(4+((x-1)*4))))));
        Out_phase(17+((x-1)*4),y) = (180/pi)*atan(((Amplitude(1+((x-1)*4),1)*sin(Rads
(1+((x-1)*4))))-(Amplitude(3+((x-1)*4),y)*cos(Rads(3+((x-1)*4)))))/((Amplitude(1+((x-
1)*4),1)*cos(Rads(1+((x-1)*4))))+(Amplitude(3+((x-1)*4),y)*sin(Rads(3+((x-1)*4))))));
        Out_phase(18+((x-1)*4),y) = (180/pi)*atan(((Amplitude(1+((x-1)*4),1)*sin(Rads
(1+((x-1)*4))))-(Amplitude(4+((x-1)*4),y)*cos(Rads(4+((x-1)*4)))))/((Amplitude(1+((x-
1)*4),1)*cos(Rads(1+((x-1)*4))))+(Amplitude(4+((x-1)*4),y)*sin(Rads(4+((x-1)*4))))));
        Out_phase(19+((x-1)*4),y) = (180/pi)*atan(((Amplitude(2+((x-1)*4),1)*sin(Rads
(2+((x-1)*4))))-(Amplitude(3+((x-1)*4),y)*cos(Rads(3+((x-1)*4)))))/((Amplitude(2+((x-
1)*4),1)*cos(Rads(2+((x-1)*4))))+(Amplitude(3+((x-1)*4),y)*sin(Rads(3+((x-1)*4))))));
        Out_phase(20+((x-1)*4),y) = (180/pi)*atan(((Amplitude(2+((x-1)*4),1)*sin(Rads
(2+((x-1)*4))))-(Amplitude(4+((x-1)*4),y)*cos(Rads(4+((x-1)*4)))))/((Amplitude(2+((x-
1)*4),1)*cos(Rads(2+((x-1)*4))))+(Amplitude(4+((x-1)*4),y)*sin(Rads(4+((x-1)*4))))));

        %Convert the values of -90 to 90 returned by atan matlab function
        %and check that this value is within the range 0-360

        if Out_phase(1+((x-1)*4),y)>0
           Out_phase(1+((x-1)*4),y) = 90+(90-Out_phase(1+((x-1)*4),y));
        else
           Out_phase(1+((x-1)*4),y)= -Out_phase(1+((x-1)*4),y);
        end

        if Out_phase(2+((x-1)*4),y)<0
           Out_phase(2+((x-1)*4),y) = 270-(90+Out_phase(2+((x-1)*4),y));
```

```
        else
           Out_phase(2+((x-1)*4),y) = 360- Out_phase(2+((x-1)*4),y);
        end

        if Out_phase(3+((x-1)*4),y)<0
           Out_phase(3+((x-1)*4),y) = 90-(90+Out_phase(3+((x-1)*4),y));
        else
           Out_phase(3+((x-1)*4),y) = 180 - Out_phase(3+((x-1)*4),y);
        end

        if Out_phase(4+((x-1)*4),y)>0
           Out_phase(4+((x-1)*4),y) = 270 + (90-Out_phase(4+((x-1)*4),y));
        else
           Out_phase(4+((x-1)*4),y) = 180 - Out_phase(4+((x-1)*4),y);
        end

        if Out_phase(17+((x-1)*4),y)>0
           Out_phase(17+((x-1)*4),y) = 360-(Out_phase(17+((x-1)*4),y));
        else
            Out_phase(17+((x-1)*4),y) = 0 - Out_phase(17+((x-1)*4),y);
        end

        if Out_phase(18+((x-1)*4),y)<0
           Out_phase(18+((x-1)*4),y) = 0 - (Out_phase(18+((x-1)*4),y));
        else
            Out_phase(18+((x-1)*4),y) = 360 - Out_phase(18+((x-1)*4),y);
        end

        if Out_phase(19+((x-1)*4),y)<0
           Out_phase(19+((x-1)*4),y) = 180 + (Out_phase(19+((x-1)*4),y));
        else
            Out_phase(19+((x-1)*4),y) = 180 - Out_phase(19+((x-1)*4),y);
        end

        if Out_phase(20+((x-1)*4),y)>0
           Out_phase(20+((x-1)*4),y) = 180 - (Out_phase(20+((x-1)*4),y));
        else
            Out_phase(20+((x-1)*4),y) = 180 - Out_phase(20+((x-1)*4),y);
        end

        %Compute output amplitudes
        Out_ampl(1+((x-1)*4),y) = sqrt((((Amplitude(1+((x-1)*4),y)*sin(Rads(1+((x-1
*4))))-(Amplitude(3+((x-1)*4),1)*cos(Rads(3+((x-1)*4)))))^2)+(((Amplitude(1+((x-1)*4)
y)*cos(Rads(1+((x-1)*4))))+(Amplitude(3+((x-1)*4),1)*sin(Rads(3+((x-1)*4)))))^2));
        Out_ampl(2+((x-1)*4),y) = sqrt((((Amplitude(1+((x-1)*4),y)*sin(Rads(1+((x-1
*4))))-(Amplitude(4+((x-1)*4),1)*cos(Rads(4+((x-1)*4)))))^2)+(((Amplitude(1+((x-1)*4)
y)*cos(Rads(1+((x-1)*4))))+(Amplitude(4+((x-1)*4),1)*sin(Rads(4+((x-1)*4)))))^2));
        Out_ampl(3+((x-1)*4),y) = sqrt((((Amplitude(2+((x-1)*4),y)*sin(Rads(2+((x-1
*4))))-(Amplitude(3+((x-1)*4),1)*cos(Rads(3+((x-1)*4)))))^2)+(((Amplitude(2+((x-1)*4)
y)*cos(Rads(2+((x-1)*4))))+(Amplitude(3+((x-1)*4),1)*sin(Rads(3+((x-1)*4)))))^2));
        Out_ampl(4+((x-1)*4),y) = sqrt((((Amplitude(2+((x-1)*4),y)*sin(Rads(2+((x-1
*4))))-(Amplitude(4+((x-1)*4),1)*cos(Rads(4+((x-1)*4)))))^2)+(((Amplitude(2+((x-1)*4)
y)*cos(Rads(2+((x-1)*4))))+(Amplitude(4+((x-1)*4),1)*sin(Rads(4+((x-1)*4)))))^2));
        Out_ampl(17+((x-1)*4),y) = sqrt((((Amplitude(1+((x-1)*4),1)*sin(Rads(1+((x-1
*4))))-(Amplitude(3+((x-1)*4),y)*cos(Rads(3+((x-1)*4)))))^2)+(((Amplitude(1+((x-1)*4)
```

```
1)*cos(Rads(1+((x-1)*4))))+(Amplitude(3+((x-1)*4),y)*sin(Rads(3+((x-1)*4)))))^2));
        Out_ampl(18+((x-1)*4),y) = sqrt((((Amplitude(1+((x-1)*4),1)*sin(Rads(1+((x-1)
*4))))-(Amplitude(4+((x-1)*4),y)*cos(Rads(4+((x-1)*4)))))^2)+(((Amplitude(1+((x-1)*4),
1)*cos(Rads(1+((x-1)*4))))+(Amplitude(4+((x-1)*4),y)*sin(Rads(4+((x-1)*4)))))^2));
        Out_ampl(19+((x-1)*4),y) = sqrt((((Amplitude(2+((x-1)*4),1)*sin(Rads(2+((x-1)
*4))))-(Amplitude(3+((x-1)*4),y)*cos(Rads(3+((x-1)*4)))))^2)+(((Amplitude(2+((x-1)*4),
1)*cos(Rads(2+((x-1)*4))))+(Amplitude(3+((x-1)*4),y)*sin(Rads(3+((x-1)*4)))))^2));
        Out_ampl(20+((x-1)*4),y) = sqrt((((Amplitude(2+((x-1)*4),1)*sin(Rads(2+((x-1)
*4))))-(Amplitude(4+((x-1)*4),y)*cos(Rads(4+((x-1)*4)))))^2)+(((Amplitude(2+((x-1)*4),
1)*cos(Rads(2+((x-1)*4))))+(Amplitude(4+((x-1)*4),y)*sin(Rads(4+((x-1)*4)))))^2));


        end
    end



%Convert output phases from degrees to radians
Out_rads = zeros(32,amount);
Out_rads = Out_phase* (pi/180);

phase_out=zeros(32,amount);

%Convert phasor representation to cartesian to plot
for s = 1:32
    for t = 1:amount
        Cartesian_all(t+((s-1)*amount),1)= Out_ampl(s,t) * cos(Out_rads(s,t));
        Cartesian_all(t+((s-1)*amount),2)= Out_ampl(s,t) * sin(Out_rads(s,t));
    end
end

Cartesian = zeros((amount*8),8);

figure(20+((loop-1)*20));
%         title('I-Q Constellation of All Possible Attenuation Levels');
        xlabel('Normalized In Phase Component');
        ylabel('Normalized Quadrature Component'); hold on
%Rearrange matrix
for l = 1:4
    Cartesian(1:(amount*4),1+((l-1)*2))= Cartesian_all(((((l-1)*amount*4)+1):
amount*l*4),1);
    Cartesian(1:(amount*4),2+((l-1)*2))= Cartesian_all(((((l-1)*amount*4)+1):
amount*l*4),2);
    Cartesian((amount*4)+1:(amount*8),1+((l-1)*2))= Cartesian_all(((((l-1)*amount*4)+
((amount*16)+1)):(amount*l*4+(amount*16))),1);
    Cartesian((amount*4)+1:(amount*8),2+((l-1)*2))= Cartesian_all(((((l-1)*amount*4)+
((amount*16)+1)):(amount*l*4+(amount*16))),2);

%      plot all possible attenuation levels

     scatter(Cartesian(:,1+((l-1)*2)),Cartesian(:,2+((l-1)*2)),5);

end

        legend('Channel 1','Channel 2','Channel 3','Channel 4')
```

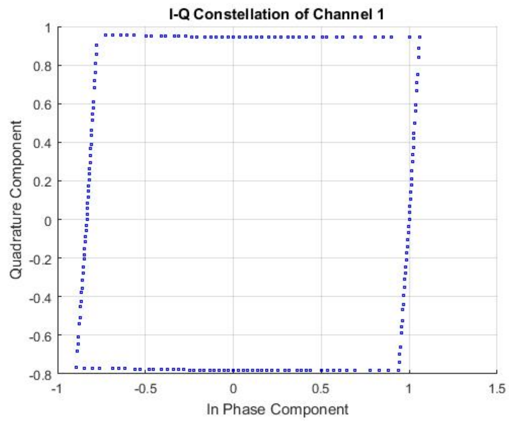
```
 hold off

% Plot all channels constallations together
% figure(5);
% scatter(Cartesian_all(:,1),Cartesian_all(:,2),5)

%Define resolution in degrees wanted
mul = phase_res;
size = 360/mul;
bestmatch= zeros(size,4);
Cartesian_best = zeros(size,8);

%Compute phase error attenuation levels of each channel
for w = 1:4
    for m = 1:size
        comp = (m-1)*mul;
        [d, ix] = min(abs((Out_phase((((w-1)*4)+1):(w*4),:)-comp)),[],2);
        [e, jx] = min(abs((Out_phase((((w-1)*4)+17):(w*4)+16,:)-comp)),[],2);

        [f,ir]= min(d);
        [g,jr]= min(e);

        if f > g
            bestmatch(m,w)= Out_phase((17+(jr-1)+((w-1)*4)),jx(jr));
            bestampl(m,w)= Out_ampl((17+(jr-1)+((w-1)*4)),jx(jr));
            bestatt(m,w) = (jx(jr)-1)*(-step);
        else
            bestmatch(m,w)= Out_phase(((ir-1)+1 +((w-1)*4)),ix(ir));
            bestampl(m,w)= Out_ampl((1+(ir-1) +((w-1)*4)),ix(ir));
            bestatt(m,w) = (ix(ir)-1)*(-step);
        end

    %Compute cartesian coordinates
    Cartesian_best(m,1+((w-1)*2))= bestampl(m,w) * cos(bestmatch(m,w)*(pi/180));
    Cartesian_best(m,2+((w-1)*2))= bestampl(m,w) * sin(bestmatch(m,w)*(pi/180));
    end

    tmp_x = Cartesian_best(: , 1+((w-1)*2));
    tmp_y = Cartesian_best(: , 2+((w-1)*2));

    %Plot seperate constallations for the calibrated phase
    if w == 1
        figure(6+((loop-1)*20));
        s1 = scatter(tmp_x,tmp_y,5);
        title('I-Q Constellation of Channel 1');
        xlabel('In Phase Component');
        ylabel('Quadrature Component');
        colour = s1.Marker;
        s1.MarkerEdgeColor = 'b';
        Channel_1 = Cartesian_best(: ,1:2);
    end


    if w == 2
        figure(7+((loop-1)*10));
```

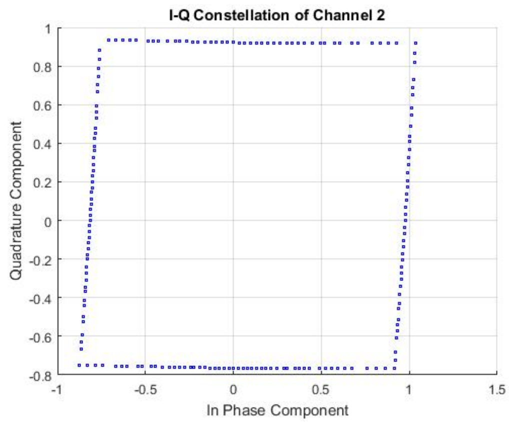
```
        s2 = scatter(tmp_x,tmp_y,5);
        title('I-Q Constellation of Channel 2');
        xlabel('In Phase Component');
        ylabel('Quadrature Component');
        colour = s2.Marker;
        s2.MarkerEdgeColor = 'b';
        Channel_2 = Cartesian_best(: ,3:4);
    end


    if w == 3
        figure(8+((loop-1)*10));
        s3 = scatter(tmp_x,tmp_y,5);
        title('I-Q Constellation of Channel 3');
        xlabel('In Phase Component');
        ylabel('Quadrature Component');
        colour = s3.Marker;
        s3.MarkerEdgeColor = 'b';
        Channel_3 = Cartesian_best(: ,5:7);
    end


    if w == 4
        figure(9+((loop-1)*10));
        s4 = scatter(tmp_x,tmp_y,5);
        title('I-Q Constellation of Channel 4');
        xlabel('In Phase Component');
        ylabel('Quadrature Component');
        colour = s4.Marker;
        s4.MarkerEdgeColor = 'b';
        Channel_4 = Cartesian_best(: ,7:8);
    end

end

%Find maximum error
for inversion = 1:2
    for channel = 1 : 4
    for path = 1: 4
        upbound(path, channel+((inversion-1)*4)) = tan(Out_rads((((channel↙
((inversion-1)*4))-1)*4 + path),1));
        downbound(path, channel+((inversion-1)*4)) = tan(Out_rads((((channel↙
((inversion-1)*4))-1)*4 + path),2));
        maximum_error(path, channel+((inversion-1)*4)) = (180/pi) * double((int((1↙
((z^2)+1)), z, upbound(path,(channel+((inversion-1)*4))), downbound(path,(channel↙
((inversion-1)*4))))))/2);
    end
        %Find maximum error for each channel
        [mx , lc] = max(abs(maximum_error(:,channel)));
        max_error(inversion,channel) = mx;
    end
end

%Create sinusoid which will be used in the plot to show it bounds all errors
t = [0:0.01:360];
```

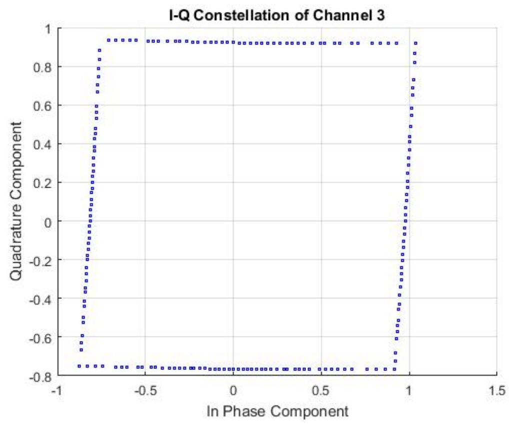
```
a = sin((2*pi*t)/180);

%Split up matrix
xChannel(1,1:size) = Cartesian_best(: ,1);
xChannel(2,1:size) = Cartesian_best(: ,3);
xChannel(3,1:size) = Cartesian_best(: ,5);
xChannel(4,1:size) = Cartesian_best(: ,7);

yChannel(1,1:size) = Cartesian_best(: ,2);
yChannel(2,1:size) = Cartesian_best(: ,4);
yChannel(3,1:size) = Cartesian_best(: ,6);
yChannel(4,1:size) = Cartesian_best(: ,8);


%Compute phase error of output phase after callibration
for lp = 1:size
compp = (lp-1)*mul;
output_error(lp,:) = bestmatch(lp,:) - compp;
end



%Plot results
figure(10+((loop-1)*10));
grid on
hold on
s5 = plot((mul*(1:size)'), output_error(:,1),'b.');
title('Phase Error of Channel 1')
ylabel('Phase Error (Degrees)')
xlabel('Targetted Phase (Degrees)')
% plot(t,(a*max_error(1,1)), 'r')
% plot(t,((-a)*max_error(1,1)), 'r')
hold off

figure(11+((loop-1)*10));
grid on
hold on
s5 = plot((mul*(1:size)'), output_error(:,2),'b.');
title('Phase Error of Channel 2')
ylabel('Phase Error (Degrees)')
xlabel('Targetted Phase (Degrees)')
% plot(t,(a*max_error(1,2)), 'r')
% plot(t,((-a)*max_error(1,2)), 'r')
hold off

figure(12+((loop-1)*10));
grid on
hold on
s5 = plot((mul*(1:size)'), output_error(:,3),'b.');
title('Phase Error of Channel 3')
ylabel('Phase Error (Degrees)')
xlabel('Targetted Phase (Degrees)')
% plot(t,(a*max_error(1,3)), 'r')
% plot(t,((-a)*max_error(1,3)), 'r')
hold off
```

```
figure(13+((loop-1)*10));
grid on
hold on
s5 = plot((mul*(1:size)'), output_error(:,4),'b.');
title('Phase Error of Channel 4')
ylabel('Phase Error (Degrees)')
xlabel('Targetted Phase (Degrees)')
% plot(t,(a*max_error(1,4)), 'r')
% plot(t,((-a)*max_error(1,4)), 'r')
hold off

end
```

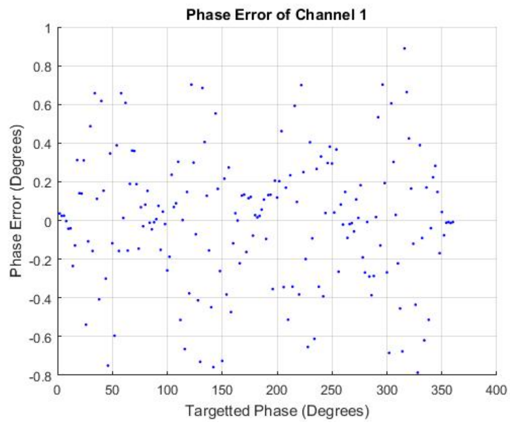
B Calibrated I-Q constellations of Channels 1-3 for a resolution of 2°

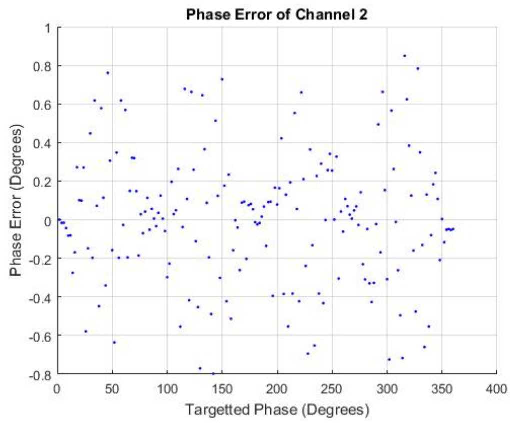


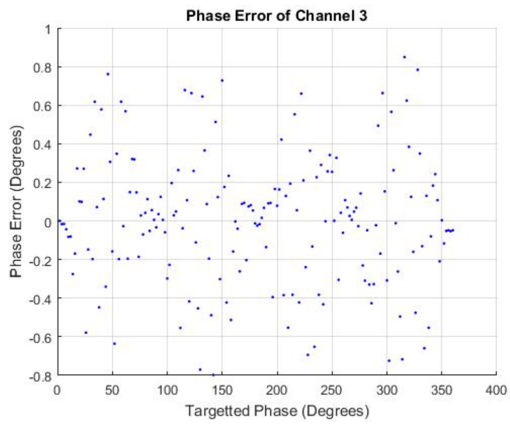




C Calibrated phase errors of Channels 1-3 for a resolution of 2°







D I-Q constellations given for no phase or amplitude uncertainty

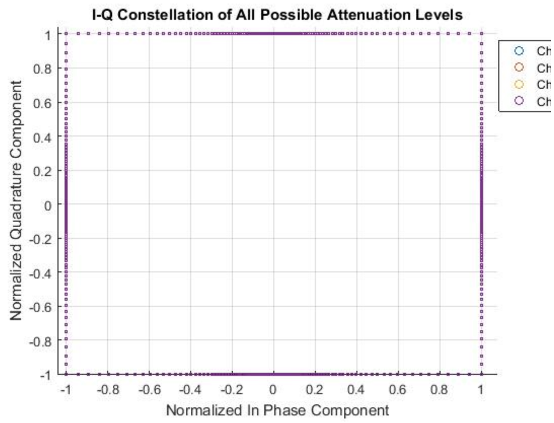


Figure 7: All possible I-Q constellations given for no phase or amplitude uncertainty

E The Scientific and Sociological Implications of 'Design and Calibration of a 4 Channel Programmable Phase Shifter ($2 - 6GHz$)'

The work of this project started simply as the task to design a programmable 4-channel phase shifter that could be used to test another system of interest. Through the conducted analysis on the system, made in order to motivate the design choices made in this 4-channel phase shifter, better insight into the workings of a programmable 4-channel phase shifter have been achieved. This insight is not limited to the specific application of this 4-channel phase shifter, as the analysis is flexible to allow for modeling of different systems with different parameters. For example two of such parameters are the minimal attenuation step of the system, as well as the desired phase resolution of the system. By conducting this analysis the application of this work has therefore gained value to other researchers who desire to design an In-phase-Quadrature (I-Q) phase shifters. In addition, this works provides motivation to why the architecture used in this design could be more helpful than the traditional lumped-network based phase shifter architectures. As this paper contributes to an expanding literature basis for phase shifter designs, it indirectly also contributes to designs who incorporate phase shifters. The most evident example of such system is the one mentioned in the paper, namely phased arrays. This is an expanding research field in the last decade, as these antenna systems have many advantages over classical stand-alone antenna architectures, some experts even speculating that in the future they will completely replace stand-alone antennas. Currently it is the belief that insufficient low cost solutions to design these systems have been found, which is the reason why these systems havenâĂŹt been commercially available. Research in this field, or (in this case) the field of phase shifters can contribute to the search of these low cost solutions and cause the implementation of phased arrays in a growing number of commercially owned devices, possibly leading to improvements like better radio frequency reception (such as cellular reception).