5Ghz Chirp Signal Generator for Broadband FMCW Radar Applications

Peter Hobden†and Saket Srivastava†

†School of Engineering, University of Lincoln, United Kingdom

16633091@students.lincoln.ac.uk, Ssrivastava@lincoln.ac.uk

Abstract – Direct digital synthesis (DDS) is a method of producing an RF analogue waveform which is usually a sine wave. However, there are a limited number of devices capable of producing a high frequency output (of more than 2 GHz). To generate complex waveforms, you ideally require a high-end expensive FPGA board with on-board high speed SerDes transceivers coupled with a DAC. The 'Analog devices' AD916X series is one of the few devices that can output frequencies over 5 GHz. In this paper, we present a low-cost implementation scheme for producing high frequency waveform patterns using Xilinx FPGAs and AD9164, with the minimum of latency. Our proposed solution makes use of the Xilinx 7 series and Ultrascale devices, using the high speed SerDes channels over the FMC connector together with the PCIe bus for fast loading of patterns. With our proposed solution it is easy to generate and play back complex waveforms, while maintaining a jitter free and low phase noise output. One of the most important application areas that would benefit from our proposed implementation is the generation of high frequency FMCW radar chirps and simulating target responses especially in the upcoming 77GHz frequency range where the baseband can sweep to 4 GHz.

Keywords—Direct digital synthesis (DDS), FFH frequency hopping, SerDes (Serializer/Deserializer), Inter Integrated Circuit (I2C), Serial Peripheral Interface (SPI), Peripheral Component Interconnect Express (PCI), Intellectual Property (IP), System-on-Chip (SoC), Verilog, VHDL, Xilinx Vivado, dynamic range (SFDR), noise spectral density (NSD).

I. INTRODUCTION

With recent changes in legislation, radar designers are moving away from 24 GHz to 77GHz [1], where they are able to sweep up to 4GHz, instead of 200 MHz at 24 GHz. One of the biggest challenges is to be able to create complex frequency patterns that make use of this wide bandwidth. Apart from the issue of generating the high frequency, it is also challenging to transmit data quick enough to successfully generate the complex pattern without any chance of phase jitter. Traditionally, the DDS chip generates the output from registry settings which are normally set using an SPI interface. This method is sometimes too slow for complex patterns. Alternatively, a comparable high frequency DDS can be generated using an FPGA and a high-performance DAC such as the FMC230 from 4DSP running on a Virtex 7, using the DDS IP core block from Xilinx [2]. This is a much more expensive option, with additional issues such as higher phase noise. The preferred route is always to use a dedicated chip based DDS.

To the best our knowledge, no papers exist on implementation of a microwave frequency DDS coupled with a high-end FPGA and utilising the PCIe bus & SPI for the loading of Patterns and solving the solution of 2.5 GHz SerDes link bar-

rier. Companies such as Keysight and Rohde and Schwarz produce Microwave Signal Generators capable of 5 GHz generation, but as the market stand, these are very expensive and can only be used as test equipment. One recent work made use of multiple AD9164s for single tone generation [1], but this does not expand in to complex waveform generation.

The AD9164 brings a new level of performance to the RF DDS class and enables RF signal generation up to 7.5 GHz to be performed without the need to up-convert. In addition, higher update rates create wider first Nyquist zones, which in turn enable the converter to directly synthesize higher output frequencies. In this paper we will make use of the AD9164 to provide the best of both worlds, namely generating high frequency complex signal patterns while keeping the jitter noise and development costs extremely low compared to the current state of the art. In our proposed solution, we have targeted the challenges of implementing a low-cost microwave frequency generator for complex patterns up to 7.5GHz that could be embedded in a commercial product such as FMCW Radar. We have addressed this by using the current state of art AD9164 DDS from Analog devices, combined with a Xilinx's Virtex / Kintex 7 FPGA, connected to an optional PC via a Gen2 PCIe bus.

The rest of the paper is as follows. Section II provides a brief technical background. Section III describes our proposed design. We discuss experimental setup in section IV and the results of our proposed design in Section V. The Conclusion and Summary are presented in Section V.

II. TECHNICAL DESCRIPTION

The AD9164 chip has a hardware DDS onboard for single tone generation, but like the DAC it also supports a high speed SerDes interface, allowing us to transfer and output complex patterns at near to real time. This capability allows the AD9164 to generate frequencies above 2.5 GHz. Table 1 highlights the specifications of AD9164 board that make it ideal for our proposed application. The following sub-sections outline the major hardware components of AD9164 chip.

Table 1 AD9164 Specifications

AD9164 Specifications
• DAC update rate up to 12 GSPS (AD9129 – 5.7)
• Direct RF synthesis at 6 GSPS (AD9129 – 2.85)
• DC to 2.5 GHz in baseband mode (AD9129 – DC to 1.425)
• DC to 6 GHz in 2× nonreturn-to-zero (NRZ) mode
• 1.5 GHz to 7.5 GHz in Mix-Mode (AD9129 – 1.424 GHz to 4.2 GHz)
• By passable interpolation 2×, 3×, 4×, 6×, 8×, 12×, 16×, 24×

A. DIRECT DIGITAL SYNTHESIZER

A DDS produces a sine wave at a given frequency. The frequency depends on two variables, the system-clock frequency and the binary number programmed into the frequency register (tuning word). Continuous-time sinusoidal signals have a repetitive angular phase range of 0 to 2π. The digital implementation (shown in Fig. 1) is no different. The counter's carry function allows the phase accumulator to act as a phase wheel in the DDS implementation [3].

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

Traditionally, DDS were implemented as in hardware on a dedicated chip. However, the recent developments in high speed FPGAs and DAC technology now make it possible to implement a DDS as a soft IP block [2]. In the case of AD9164, we have an option to either use the onboard hardware DDS or else make use of the high-speed DAC in conjunction with an FPGA to implement an IP block for a DDS, thus providing us a wide range of design flexibility. To generate a single tone frequency (Fig. 2. Single Tone (DDS generated), the dedicated hardware DDS can run independently. The IP Block solution can be utilised whenever we require continuous clocking by the FPGA as in case of a radar application. On the other hand, a dedicated hardware DDS has the advantage of added feature sets of NCOs and interpolators in DACs, which relieves the FPGA or ASIC from the resources and power consumption of implementing those features. This also enables the DACs to operate on lower data transfer rates than otherwise would be required. The lower data rate reduces overall power consumption in the system, and in some cases makes it possible for the digital chip, on which the fabric speed may range up to 300 MHz - 400 MHz, to keep pace with the converter.
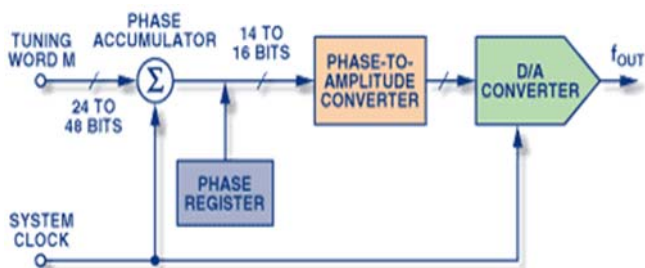

Fig 1. Components of a Direct Digital Synthesizer [3]

B. FREQUENCY HOPPING (FFH) NCO ENGINE

The AD9164 includes a fast FFH Numerically Controlled Oscillator (NCO) engine. The FFH NCO engine is implemented with thirty-two 32-bit NCOs, each with its own phase accumulator, and a selection block that enables the fast frequency hopping. This capability allows us to pre-program up to 32 different frequencies to generate a complex signal pattern which is good for an application such as FMCW radar. The NCO has two selectable modes of operation, either phase continuous or phase discontinuous frequency switching. In phase continuous switching, the frequency tuning word (FTW) is updated but the phase accumulator is not reset, re-

sulting in a continuous phase change in frequency. In phase discontinuous mode, the phase accumulator is reset when the FTW is updated. Phase continuous frequency hopping is important for test applications and radar applications that need to track the phase of an exciter signal for use later. Phase continuous frequency hopping enables changing from one frequency to another and back to the original one again, without losing track of the original frequency's phase accumulation. 32 different frequencies can be pre-programmed, allowing for frequency hopping of less than 300 ns.

The limitation of 32 frequencies for the FFH NCO, is crucial for applications that require complex waveforms needing
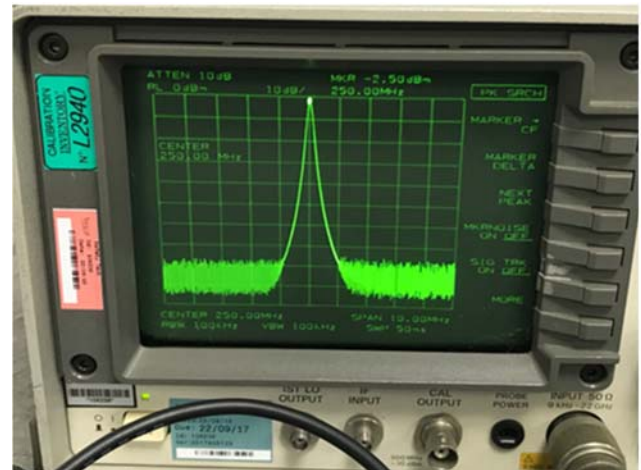

Fig. 2. Single Tone (DDS generated)

more than 32 frequencies such as QAM-256. However, we have managed to get over this limitation using a clever technique as discussed later in section 0 below.

FIR Filter

The high frequency signal generated using a DDS has an inherent problem that the generated wave does not conform to the actual shape of a real sine wave. This is primarily because it is heavily aliased. The FIR filter block present in AD9164 can convert this heavily aliased waveform in to a proper high frequency sine wave. We have seen that it is not possible to generate a proper signal for any waveform above 5GHz without making use of this FIR filter block. For our proposed scheme, where we generate a complex chirp signal above 5GHz, we need to set the FIR filter accordingly. According to the technical specifications of AD9164, the DAC is capable of a maximum sampling rate of up to 12 GSPS, which results in a highly aliased output signal, thus making it necessary to utilize the FIR filter block.

Serial Peripheral Interface (SPI)

The SPI signals [4] [5] [6] are mostly used for setting the register values on the AD9164 board. The signal integrity of SPI is guaranteed up to 100 MHz to enable fast updating of the FTW. With the 100 MHz SPI, this means a new FTW can be chosen in 240 ns, with the single byte write. While this is fast but may not be fast enough for creating high frequency
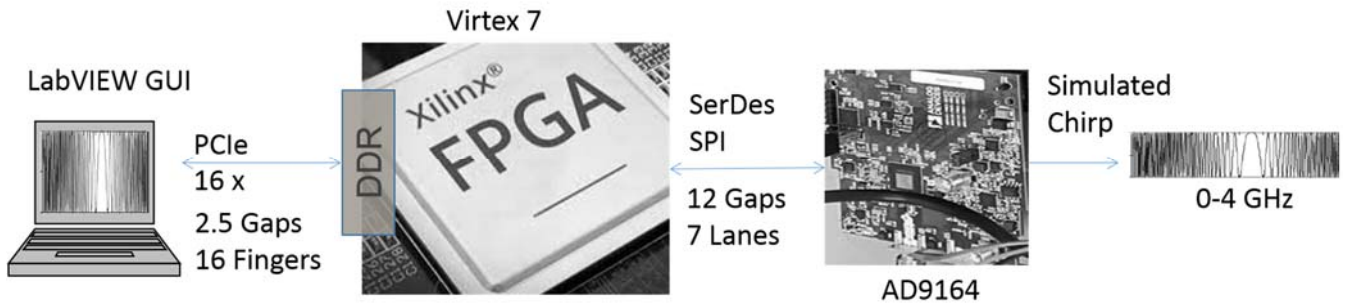
Fig. 1. Block diagram of the proposed setup

complex pattern such as those used in FMCW return signals, The AD9164 solves this by providing a by an 8-lane, 12.5 GBPS JESD204B SerDes serial interface [7] that enables practical management and transfer of output data. In JESD204B SerDes mode, you can think of the AD9164 acting as a high-speed DAC with FIR filter on the output.

Serializer/Deserializer (SerDes)

SerDes is a pair of functional blocks commonly used in high speed communications to compensate for limited input(s)/output(s). These blocks convert data between serial data and parallel interfaces in each direction. The term "SerDes" generically refers to interfaces used in various technologies and applications. SerDes is now preferred over LVDS to provide data transmission over a single/differential line as it minimizes the number of I/O pins and interconnects.

As mentioned above (in section B), due to limitation of NCO, AD9164 cannot be totally used for the generation of complex signal patterns that require more than 32 frequencies. However, in this paper we present a solution to overcome this limitation as follows:

- *NCO engine via SPI*
  This method is used when we generate the required complex wave form above 2.5 GHz frequencies that needs less than 32 frequency hops.

- *Using the SerDes link*
  On the other hand, if the complex waveform requires more than 32 frequencies, and is below 2.5 GHz, we shall use the SerDes link, to set the output DAC. This novel use of AD9164 of combining all the methods of setting the DDS together, allows us to break the 2.5 GHz limit for complex pattern generation. To achieve this, we first have to understand how we are going to use the SerDes for complex waveforms below 2.5 GHz. Xilinx and others provide a high-speed IP SerDes block making use of the on-chip resources, for the new JESD204 standard. JESD204 is a high-speed serial interface for connecting data converters (ADCs and DACs) to logic devices. Revision B of the standard supports serial data rates up to 12.5 Gbps and ensures repeatable, deterministic latency on the JESD204 link.

Peripheral Component Interconnect Express (PCIe)

Another part of our novel implementation is the inclusion of a PCI bus, which allows a host PC to capture our generated

complex frequency pattern directly from / to FPGA's RAM via DMA at 2.5 GSPS as shown in Fig 3. This capability allows us to dynamically change the pattern at close to real-time update rate. PCI Express is a high-speed serial computer expansion bus standard, designed to replace the older PCI, PCI-X, and AGP bus standards.

III. PROPOSED DESIGN

As mentioned in section II.0, our proposed solution for using the AD9164, uses (1) multiple JESD204B SerDes lines (8 in total) for sending the complex wave forms from the DDR memory for frequencies below 2.5 GHz and (2) the SPI link for frequencies above 2.5 GHz. For the hardware platform shown in Fig. 3, we have opted for the Xilinx Virtex VC707 board as this provides high speed JESD204B SerDes for operating up to 12 GSPS per lane. We have also tried the cheaper Kintex KC705, but the maximum SerDes data rate is only 6 GSPS. The AD9164 evaluation board connects via the FMC HPC port. So that complex patterns can be initially generated on a PC, we have also added a high speed 2.5 GSPS per finger (lane) PCIe DMA memory bus. The USB has also been used to provide SPI communication to the AD9164, so that register settings can be set.

Initially, to get a device up and running quickly for testing, Analog devices have provided a Window's GUI which runs on their own development board that connects to our AD9164 Eval board. This board also uses a high performance Virtex 7, but lacks the PCIe bus. The advantage with using this board is that it has given us a reference for the development and a ready-made windows GUI. Unfortunately, there's no low-level source code provided and the board itself is really only designed for one purpose, that is to test and demonstrate the AD9162/64 Eval boards. The patterns provided are limited and have to be pre-loaded and it can't generate patterns above 2.5 Ghz. This does give us a bench mark to verify us proposed designed against.

A. DATA VERIFICATION

To verify the data integrity of the payload data stream through the TX JESD204B IP core and transport layer, the DAC's JESD core is configured to check short transport layer test pattern that is transmitted from FPGA's test pattern generator. To verify that data from the FPGA digital domain is successfully sent to the DAC analogue domain, the FPGA is

configured to generate the reference clock. The AD9508 clock fan-out buffer provides periodic SYSREF pulses for both the AD9162 and JESD204B IP Core. We can use this to check the integrity of our link to the AD9164.

B. NYQUIST ZONES - CONSIDERATIONS

We must be mindful of the Nyquist zones when using a digital DAC in a DDS. This is where a generate frequency can appear in multiple zones know as Nyquist zones. A good example is that 100-MHz signal may appear when output from a microwave DAC using a sampling frequency, say of 3 Gbps [8]. The signal in the first Nyquist zone may also alias appear in the second Nyquist zone and images will also appear in the third and fourth zones. As we can predict the frequencies, appropriate filtering around the frequencies of interest is all that is needed to isolate the required frequency.

C. MODES SUCH AS NON-RETURN TO ZERO

For frequencies about 5 GHz it is recommended to use a different coding schemes which include NRZ, return to zero (RTZ) and an RF mode—where consecutive samples are inverted. In addition to these, another mode, Narrow Return to Zero (NRTZ) has been shown to provide extremely good performance in the first and second Nyquist zones. The bandwidth of the device should be greater than 5 GHz to provide performance in the third and fourth Nyquist zones.

D. SPURS CONSIDERATIONS

DDS's also have an issue with generating spurs and other unwanted frequency components. To help move the spurs out of the frequency range of interest we need to be mindful of divide- by-4 divider (HMC362) on the board for providing the SERDES reference clock as this causes a ½ DAC clock rate spur to be shown on the ADF4355 (Internal Clock) output, i.e. there is a 2.5GHz spur on the ADF4355 output if the DAC clock rate is set to 5GHz. The spur level is around -55 dB. The ½ DAC clock rate spur can mix with the DAC output to generate two new spurs (F_{out}+/- ½ CLK) if using the ADF4355 as clock source. The user may see these two additional spurs (F_{out}+/- ½ CLK) at the DAC output. But these two spurs are not actually caused by the DAC, so they can be removed. We found that to remove these spurs, we had to use an external clock source instead, but this does add to the costs. As most spurs can in theory be predicted, our algorithms could potentially make adjustments to move the spurs away from our frequency band of interest. This can start making the algorithm quite complex and could add jitter/phase errors. The other option is to provide external filtering to remove spurs in the frequencies which are out of band.

E. BALUNS

To get the most suitable performance from the AD9164, we have evaluated various options available for the RF output transformer, known as Baluns. For either the Mini Circuits

(10 to 6000 MHz) or a Marki Microwave wideband Balun on the DAC output (500 kHz to 9 GHz). The Mini circuit's (10 to 6000 MHz) shown in Fig. 4, is the cheapest option. This pro-
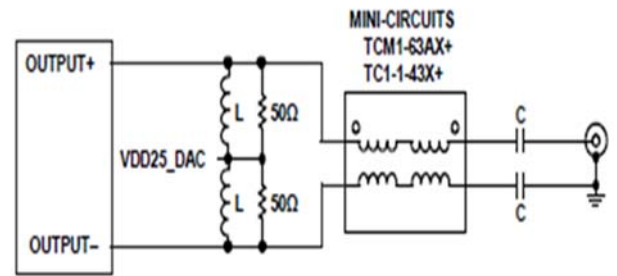

Fig. 2. Mini circuits Balun

vides a Balun covering the frequency- range of interest. The Marki Microwave device (500 kHz to 9 GHz) adds to the price but is wideband and uses the smaller 8mm X 8mm package.

IV. EXPERIMENTAL SETUP

We have proposed 5GHz chirp generator to simulate a FMCW radar signal return from an object at a set distance. Figure 5 shows the electronic test bench for the 77GHz FMCW radar that uses this chirp generator. The setup consists of the following four boards.

(a) Radar Baseband Electronic Board
This baseband normally connects to the transmitter/receiver antenna. The board then mixes the transmitted antenna signal with the returned signal (reflected from the object). In our case, we have disconnected the antenna and used the chirp generator to simulate a target at a set distance.

(b) AD9164 DDS Board
As described earlier, this board generates a high frequency signal which would match the transmitted chirp adjusted for the expected signal attenuation, and any phase difference.

(c) FPGA Board
We have used Digilent Zedboard [9] to control the AD9164 and to process baseband data from the radar electronics, which is connected using the high frequency ADC (ADC9467). The FPGA board is used for DSP processing of the input signal to generate an FFT.

(d) AD9467 Board
Since the on-board ADC available on Zedboard is limited to only 1 MSPS, we require an external high frequency ADC board to be able to process the based band signal. In our case this could be up to 250MSPS. The limiting frequency range of this board is based on the RF transformer (balun), as shown in Figure 4.
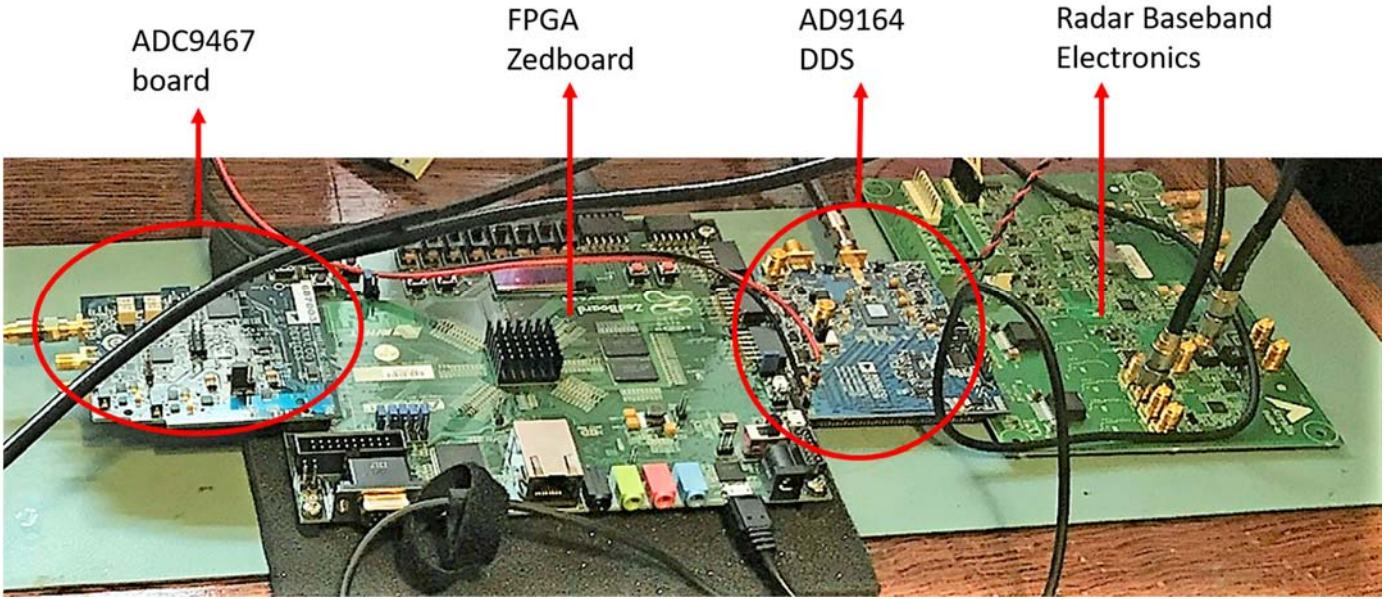
Fig. 5. Experimental bench setup for simulating 5G FMCW radar

V. RESULTS

The AD9164 board generates complicated signal pattern that needs to be tuned to match the signal profile of a FMCW radar target return signal. The LabVIEW [10] GUI is programmed to set the AD9164 to generate a specified chirp pattern that matches the techniques explained earlier in section III. LabVIEW uses the PCIe bus to transfer the data directly to the FPGA's DDR memory via DMA. In turn, the FPGA sends this pattern to the AD9164, after applying our novel algorithm which calculates the method for the reconstruction of the signature pattern to generate the RF. The GUI also allows us to specify a distance for a radar return from a target. For every sweep (at 1ms intervals, with a (frequency ramp of 1 MHz to 4 GHz), the chirp is regenerated by our LabVIEW program
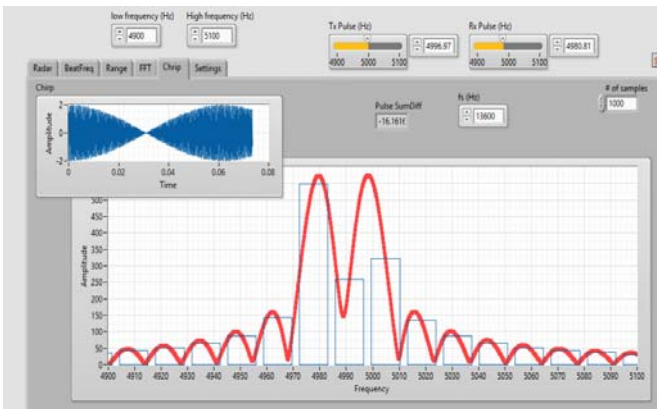

Fig. 6. Labview GUI generated radar chirp pattern for AD9164

(as shown in Fig 6) to include the effect of a return from a simulated object including any movement of the object. The movement of the object allows us to check our Doppler calculations.

On our hardware test bench setup (shown in Fig. 5), the simulated chirp signal is synced with the transmitted chirp using the AD9164 to generate the actual return signal. The RF signal is then mixed with the sum difference mixer with the TX generated chirp and the results are seen in the IF baseband. Using another Labview GUI (shown in Fig. 7), we are able to visualize the post processed Intermediate Frequency (IF) FFT. As we can see in Fig. 7, the resulting frequency of the transmitted and returned chirp signals gives us the distance to the target. In the FFT shown in Fig. 7, we can see that our return target is located at 9.5917 Mhz which in our case relates to a target distance at 357.647 metres.

VI. CONCLUSION

We have successfully demonstrated that complex frequency patterns can be generated at near to real time up to 7.5 GHz, using the AD1964 and with our novel method of combining the use of the SerDes and SPI NCO. In our example, we are using the AD9164 to simulate a FMCW radar chirp, using the 77 GHz Radar's hardware. However, this setup can be expanded to include other complex patterns such as QAM. In the case of QAM, QAM data can be generated and loaded with the less than 1us of delay. To be able to do this at 5 GHz for such a low cost would have been previously unobtainable. The novelty in our approach is to use the best method possible (SerDes/Mixed/NRZ), to generate the desired output. The software starts becoming complex but produces results that would normally be completely unobtainable for such a low-cost device.
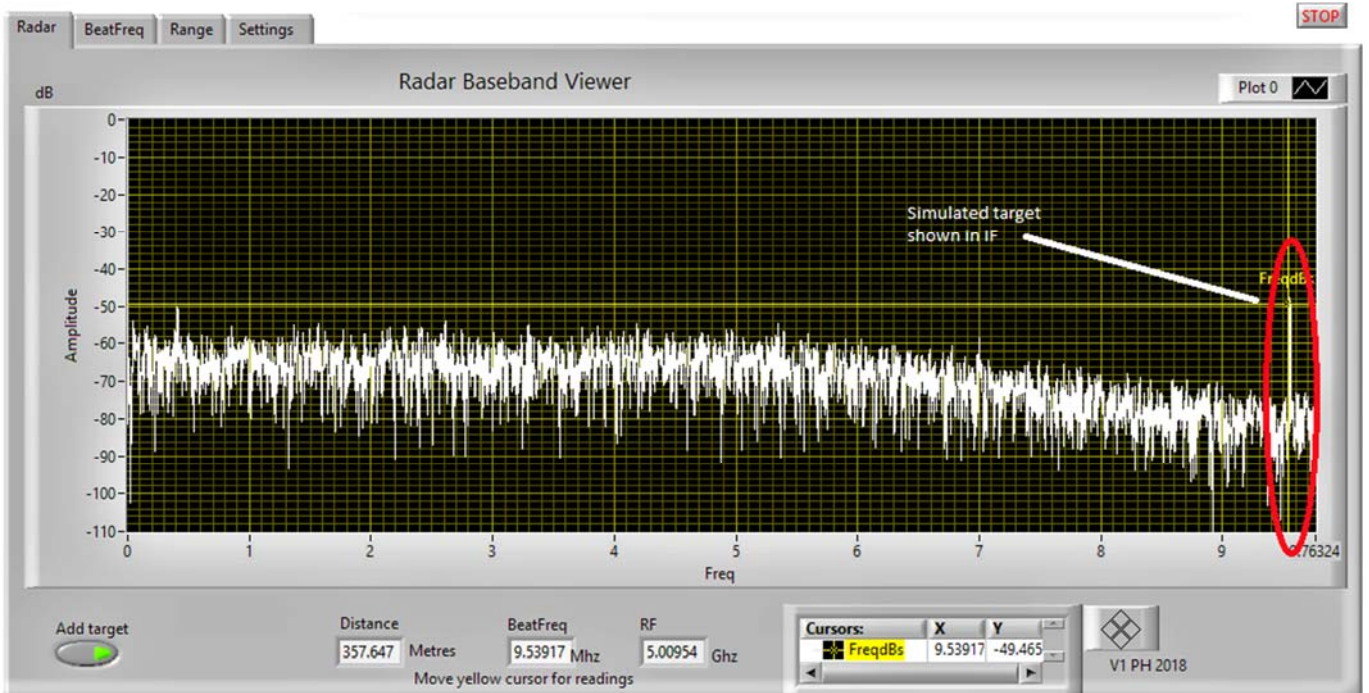
Fig. 7. Results from mixing the generated Chirp with simulated return Chirp

REFERENCES

[1] TI, "Moving from legacy 24 GHz to state-of-the-art 77 GHz radar," 2017.

[2] Xilinx, "DDS Compiler v6.0," Vivado Design Suite, 2017.

[3] E. Murphy and C. Slattery, "Ask The Application Engineer—33: All About Direct Digital Synthesis," Analog devices, AUG 2004.

[4] MOTOROLA INC., "SPI Block Guide V03.06 , Document number S12SPIV3/D," *Document number S12SPIV3/D ,* no. 04, 2003.

[5] Oudjida, A., Berrandjia, M., Tiar, R., Liacha, A., & Tahraoui, K., "FPGA Implementation of I2C & SPI Protocols:a Comparative Study," *IEEE,* 2018.

[6] F.Leens, "An Introduction to I2C and SPI Protocols," *IEEE Instrumentation & Measurement Magazine,* pp. 8-13, 2009.

[7] T. Hill, "Comprehensive JESD204B Solution Accelerates and Simplifies Development," *White Paper: All Programmable FPGAs and SoCs,* vol. WP446 , no. (v1.0.1), 2014.

[8] J. Proakis and D. Manolakis, "Algorithms, and Applications," in *Digital Signal Processing : Principles*, New York, Macmillan Publishing Company, 1992.

[9] "National Instruments - Labview," National Instruments, [Online]. Available: http://www.ni.com/en-gb/shop/labview.html.

[10] "Digilent Zedboard," Digilent, [Online]. Available: https://www.xilinx.com/support/university/boards-portfolio/xup-boards/DigilentZedBoard.html.