EEE262 DESIGN EXERCISE
VHF SYNTHESISER FOR WIRELESS COMMUNICATIONS

LAWRENCE CARLAKE

150166643

ABSTRACT:

This is a report into the design, simulation and testing of a 200MHz synthesized oscillator. This design makes use of systems such as a PLL with a common-base Colpitts oscillator and 10MHz crystal oscillator. It has been undertaken as part of the 2nd Year undergraduate labs, with the aim of developing RF design skills. As an outcome of this, we have successfully built a circuit that can synthesise frequencies between 195-205MHz at powers greater than -10dBm, with a minimum step size of 100KHz. Finally, the report investigates frequency hopping techniques and their applications.

OUTLINE

Introduction

Phase-locked loops are widely used in devices from communications to switch mode power supplies. From a reference wave, they are able to generate an adjustable output frequency many times faster than that of the reference. PLLs (Phase locked loops) are usually favoured because of their output stability and low cost.

During this exercise, we were tasked to design and build a VHF (Very high frequency, 30-300MHz) synthesizer that achieves the specifications set out in Table 1. The approach we will use to do this will be to complete approximate calculations in Matlab then, simulate the circuit in LTSPICE. Also, we will prototype the circuit, allowing us to re-simulate as necessary until the final specifications are achieved.

Specifications

Table 1 shows the specifications have been provided to us as baseline from which we can assess the success of our design.

10MHz Crystal Oscillator:	
Transistor Q2 collector DC bias current:	1.5mA
DC voltage across Q2 collector-emitter	3.6V
Output amplitude	greater than 3V pk-pk
Output frequency accuracy	better than +/-5kHz
200MHz Voltage Controlled Oscillator:	
Frequency tuning range of at least:	195MHz – 205MHz
Transistor Q3 collector DC bias current:	3mA
Integrated PLL system:	
Frequency step size	100kHz
Prescaler value	P = 8
PLL Open Loop Bandwidth	1kHz
PLL Open Loop Phase Margin	55 degrees
PLL max charge pump current	2.5mA
Final operational frequency	A – 197MHz

Table 1 – Specifications [1]

10MHz Frequency reference

The reference frequency for our PLL is derived from a 10MHz Pierce quartz crystal oscillator. The crystal has a very sharp transfer function so, when combined with the biasing and amplification components it provides a single output frequency.

As a signal appears in the feedback loop, the transistor provides gain until the circuit is in equilibrium. The measure of this gain is called loop gain. For the circuit to start oscillating the loop gain must be greater than 1, this is so that the small fluctuations in voltage caused by noise within the circuit will be amplified building upto the oscillation. For the loop to maintain oscillation the loop gain must be equal to 1, this is achieved in this circuit by the components coming to equilibrium at our desired 10MHz frequency.

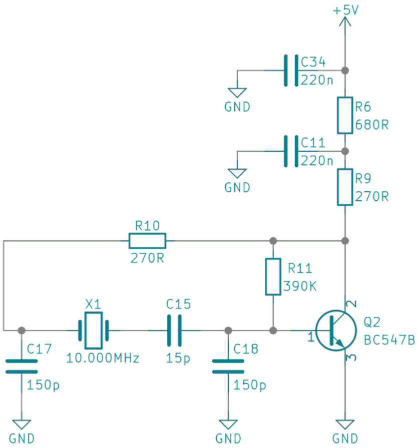


Figure 1 - 10MHz Oscillator diagram

A further requirement for oscillation is that the loop phase shift must be an integer multiple of 2π radians. In our circuit, π radians of this phase shift is achieved through the crystal and its passive components, which will be designed by us. The further π radians of phase shift is caused by the configuration of the transistor (Q2).

Understanding these constraints and the circuit specifications, we performed the required calculations (Appendix 1) summarized in the table 2.

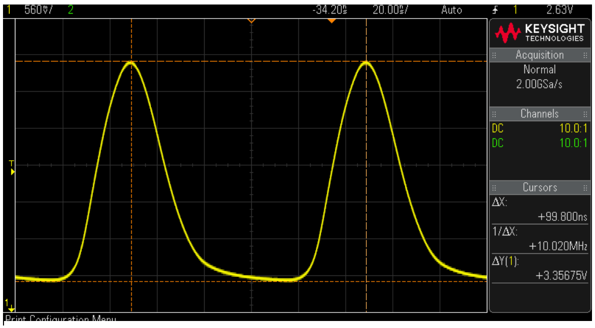
Component	Value
C17	150pF
C18	150pF
R9	270Ω
R10	270Ω
R6	680Ω
R11	390KΩ
C15	15pF

Table 2 - 10MHz Oscillator chosen components

Lab results

The first test of the frequency reference was to measure the accuracy output frequency. As seen in the screen capture (Figure 2), the frequency was 0.2% higher than the required 10MHz. The most likely reason for this is our misplacing of the cursors, as the controls only allowed for steps of 20KHz at this time base. Another source of error in this measurement could be inaccuracy of the oscilloscopes time base which would also rely on a similar circuit to the one under test. Looking into this, the accuracy of the oscilloscope is 30ppm [2, 2, 2] (0.003%), significantly smaller than the error we were measuring, meaning this should be discounted as an explanation for this error.

At this stage, we consider the 0.2% error to be small enough not to affect the operation of our PLL however if we need to adjust for this error we can increase capacitor C15 to decrease the frequency of resonance.



Acquisition
Normal
2.00GSa/s

Channels
DC 10.0:1
DC 10.0:1

Cursors
ΔX: +99.800ns
1/ΔX: +10.020MHz
ΔY(1): +3.35675V

Figure 2 - 10MHz Oscillator output

Voltage Controlled Oscillator

The next area of design was the VCO (Figure 3), this allows for the PLL controller to set and adjust the output frequency of the system. This is designed around standard circuit; a common-based Colpitts oscillator. When modelling the VCO, to assist in calculating component values, it was separated into 4 sub systems: negative resistance amplifier, frequency adjustable resonator, PCB parasitics and output coupler.

To start, we calculated the component values for the negative resistance amplifier. These calculations can be simplified by approximating the transistor using a hybrid-pi model (figure 4). Next, as an addition to this model, passive components were added, providing feedback to the transistor. Using this model, the impedance looking into the collector of the transistor (Z_t) can be calculated.

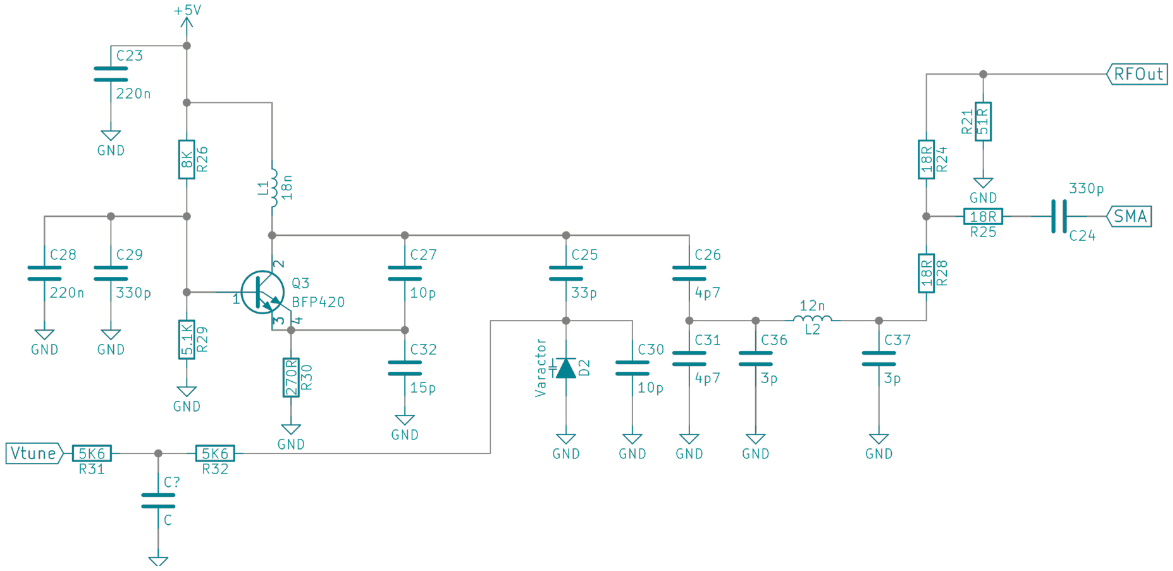


Figure 3 - VCO Schematic

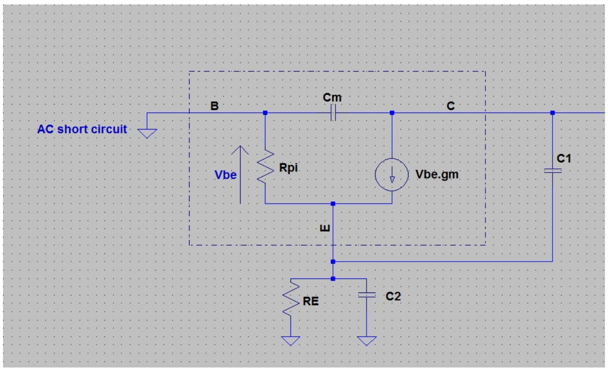


Figure 4 – Hybrid-pi model of a BJT with feedback [1]

With this equation and the provided constraints for the real and imaginary components for Zt, components C1, C2 and RE could be calculated.

Finally, to finish the negative resistance amplifier analysis, the miller capacitance must be accounted for (Figure 5). This value (0.3pF) can be found on the BFP420 datasheet [3].

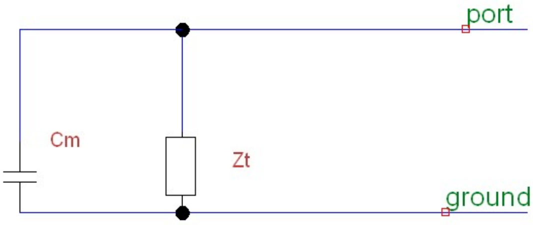


Figure 5 - Addition of Miller capacitance [1]

Excluding parasitics, the other sub-systems require no approximation and can be simply added to the model (Figure 6). We continued by summing the impedance of the transistor circuit (Zt) and the impedance of the resonator circuit (Zr) (Equation 1).

$$K = Z_t + Z_R$$

Equation 1 - Summation of reactance [1]

For this circuit to resonate, the real component of K must be negative and the imaginary component must be zero. The constraints and equations were then inputted to an equation solver which returned possible component values. An advantage of using this to find a first set of values was the initial speed however without the building an understanding of component value sensitivity, it slowed the later stages of component selection.

The initial component values, allowed us to simulate the circuit and check that the value of I_e matched the required specification. This is set as 3mA and this is what was measured.

These initial components had been designed for an output frequency of 200MHz with no regard for tuning range. Consequently, as we progressed to observing the simulated frequency range was smaller than specified. After some analysis of the inductor, the value of inductance (Lt) we had chosen so low so that the value of C_t was very large, obscuring the effect of the changing capacitance.

This is the first example of iteration within our calculations, returning to Matlab with a different value of L_t and then observing how the graphs responded to changes in Ct. Figure 7 shows how K responds to chanages in the variable part of C_t; C_{D2} (Varactor capacitance).

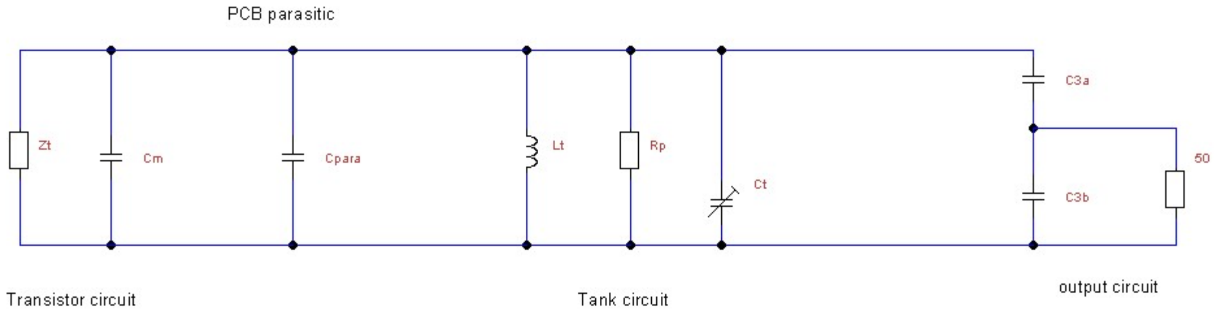


Figure 6 - Model of the VCO [1]

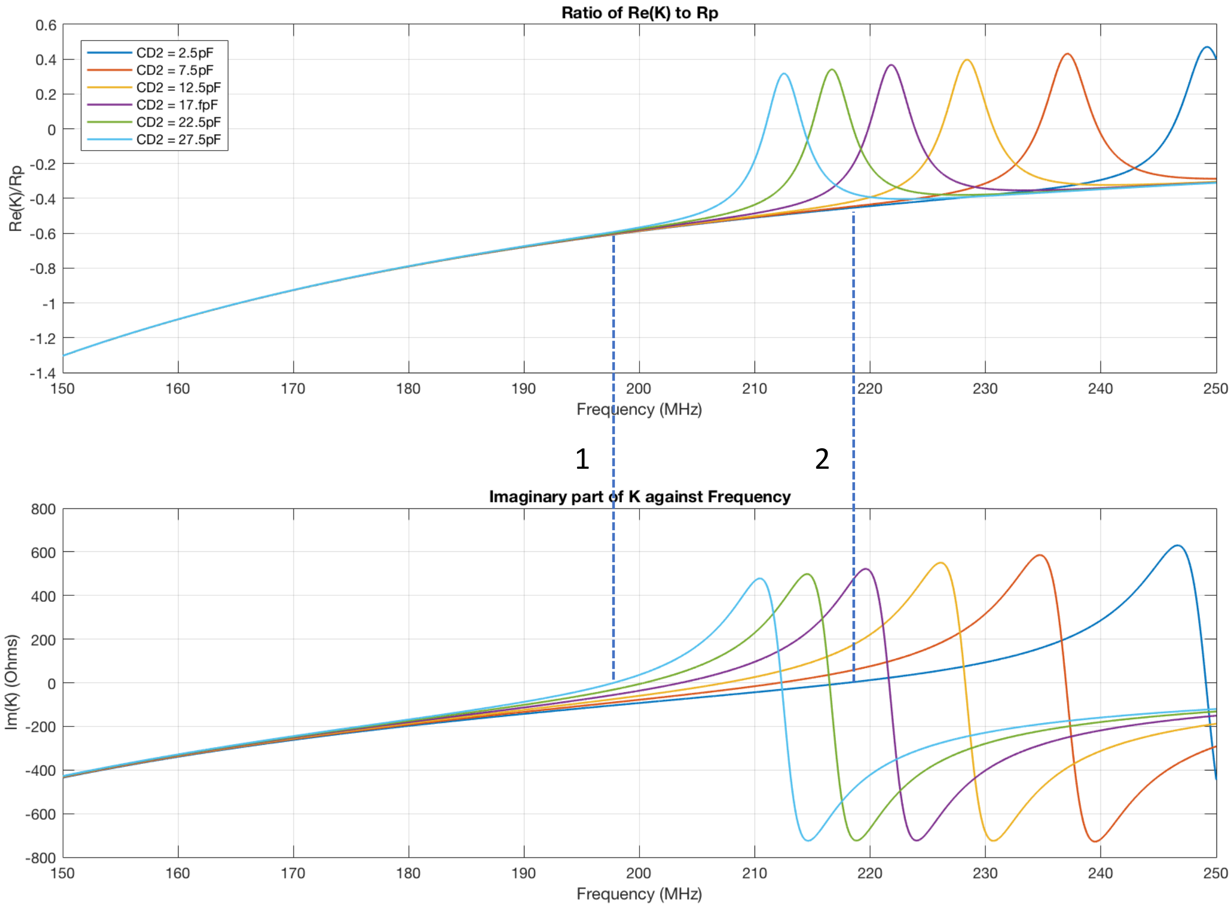


Figure 7 - K as a function of frequency for possible CD2 values

In Figure 7, each different colour, as shown in the legend, represents a different value of C_{D2}. As C_{D2} increases, this causes a decrease in output frequency. Changes in C_{D2} are caused by the varactor reacting to changes in the tuning voltage. Dotten line 1 represents the lower bound of possible frequencies achieveable, 197MHz, for the SMV1249 [4] varactor this is a capacitance of 27.5pF when a voltage of 0.5v is across it. Line 2 shows the upper bound of possible frequencies, 218MHz, at a C_{D2} of 2.5pF, with a voltage of 4.5v across the varactor.

For the stability of the VCO to be maintained it is important for us to check that for the each line on the Im(K) graph, if there is more than one zero crossing point, only one of these corresponds to a negative Re(K). Otherwise, it is possible that the VCO could oscillate at both frequencies resulting in a distored signal.

When we were designing component values, we also designed around the aim of making Re(K)/Rp as negative as possible. This is because, a lower value would result in a greater output power helping to achieved the minium required power for the PLL.

Simulation

With an suitible set of values from these calculations, we performed a simulation in LTSPICE. This allowed us to predict the response of the transitor with a more advanced model and the ability to perform a FFT on the output to observe the output frequency and power. From this, we were able to see that the tuning range was less than specified but the output power and centre frequency were sucessfully achieved.

With this insight, we returned to Matlab to recalculate a different set of components until all the required specifactions were achieved in both programs.

During this process, I noticed that matlab was useful to calculate a first approximation of components as it allowed an understanding of the effect of each component on the circuit where as LTSPICE was most useful for getting a detailed response of a set of components. Using this software greatly speed up the developemt process as we were able to quickly check many different possble component values without having to solder/desolder.

After many itterate steps, we were confident in our component choice and moved onto soldering the components onto the board.

Without the PLL fully soldered we had to inject an external tuning volatage, enabaling us to control the output frequency of the VCO, in combination with the spectrum analyzer, we could observe the real tuning range and output power.

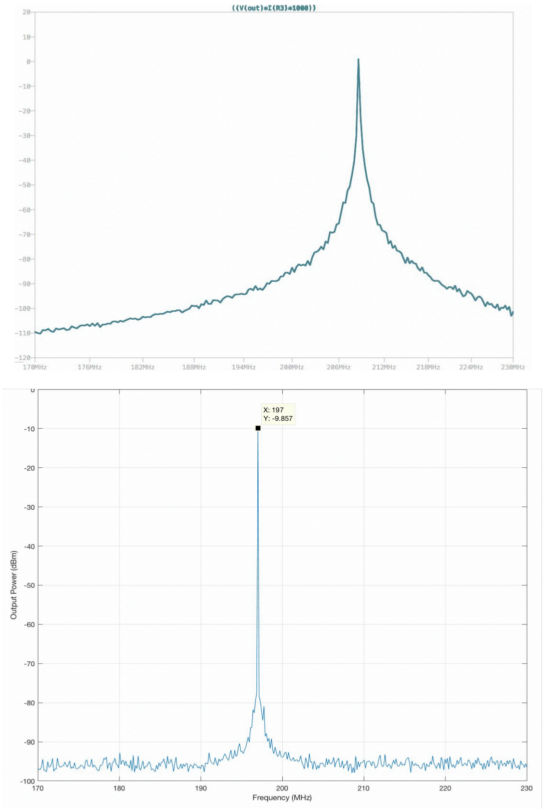


Figure 8 - VCO output for V_{tune} 2.2v in LTSPICE (top) and in reality (Bottom)

While our chosen values had performed well in simulation, during testing the real tuning range was offset by 10MHz and the output power was approximatley 10dBm lower than the simulation (Figure 8).

To correct for this we returned to LTSPICE to simulate more component values. Our approach to this stage was that we could trade off some of the tuning range (20MHz) for more output power. This aim was achieved by adjusting the ratio of C25 and C30 (Figure 3). The capacitance of C30 had a large effect on the size of the tuning range so by decreasing this value we were able to reduce the size of our tuning range. However as we had a sucessful value for Ct we must increase C25 so that the tuning range would still be centred around our desired frequency. The test for tuning range was repested sucessfully, then C25 was adjusted to centre the range and finally, the frequency response was as specified.

While these changes did result in an increased output power, the change was not enough to achieve the minimum 10dBm over our tunining range. We indentified the best method of increasing the power without changing the frequency characteristics would be to increse the base current of the transistor. While this goes against the specified I_B=3mA, the changes were checked against the the datasheet for the transistor. The higher 5.5mA was significantly under the absoloute maximum current of 60mA [3] and again under the maximum power dissipation of 210mW [3].

Component	Chosen Value
R26	8 KΩ
R29	5.1 KΩ
R30	270 Ω
L1	18 nH
C27	10 pF
C32	15 pF
C25	33 pF
C30	10 pF
C33	2.2 nF

Table 3 - Calculated values for the VCO

Lab results

The final step in developing the VCO was to charaterise its output and response to V_{tune}. Summarised in Table 4 and Table 5Table 5 is the full response of the VCO and the response over our desired range. It can be see that over our required range of 195-205MHz we have exceded the minimum required power.

	Frequncy (MHz)	Power (dBm)	Vtune (V)
Min.	186.0	-20	0.5
Mid.	199.5	-7.60	2.5
Max.	206.1	-4.84	4.5

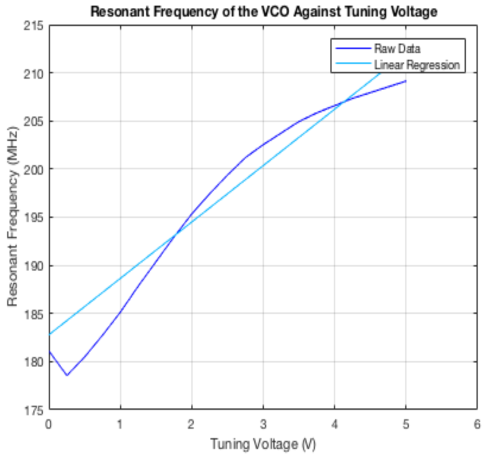
Table 4 - VCO Output over full range

	Frequncy (MHz)	Power (dBm)	Vtune (V)
Lower	195.0	-9.10	1.80
Desired	197.0	-8.50	2.07
Upper	206.1	-4.84	4.5

Table 5 - VCO output over desired range

These values were also used to calculate K_vco, the VCO voltage to frequency conversion factor. We took a linear approximation of the results providing us with a gradient that could be used in the PLL design (Figure 9Figure 7 - K as a function of frequency for possible CD2 values).

Figure 9 - K_vco linear approximation



Theory and design of PLL System

Bringing our designs for the 10MHz reference and the VCO together, is the PLL controller. The ADF4112, contains an integrated phase frequency detector, charge pump, and dividers [5].

Using these components and closed loop feedback the PLL is able to lock to a programmed output frequency. Through comparison of the output phase and frequency with the reference signal the PLL outputs pulses of current (Figure 10) which are then turned into a voltage by the loop filter tuning the VCO to the desired output frequency.

To start we took the specified value for charge pump current (2.5mA) and compared it to the table in the ADF4112 datasheet. This provided a resistor value and a setting for the internal register.

The loop filter was then designed by first calculating the time constants to achieve the desired response. This was performed by considering the open loop phase margin and bandwidth to calculate T1. T2 was then calculated using T1 and the Open loop bandwidth.

The component values could be calculated from these time constants, voltage to frequency conversion factor, charge pump current and the loop division frequency. Our final values are summarised in Table 6.

Component	Value
R2	4.7 KΩ
C19	55 nF
C20	487 nF
C33	3.3 nF
R14	1 KΩ
R31	5.6 KΩ

Table 6 - Calculated PLL and loop filter components

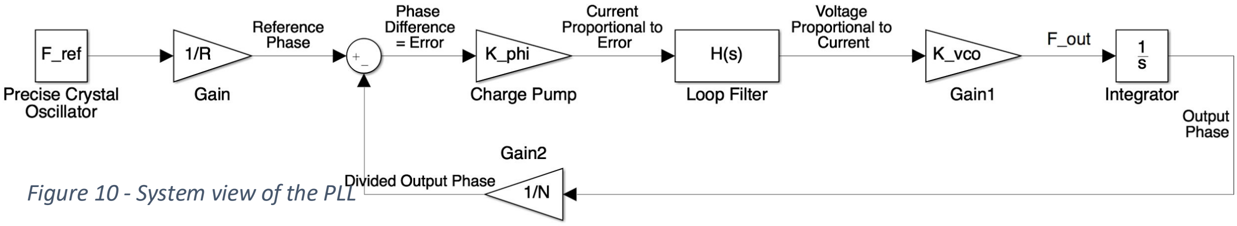


Figure 10 - System view of the PLL

The component selection section of this task has now been completed so we can move onto designing the register values to get the PLL to perform as desired.

PLL Registers

The ADF4112s internal registers are able to control all of the functions of the chip and can be programmed over an SPI bus from a microcontroller. Having been provided with the code, we calculated the required register values and converted them to the appropriate bits within the chips registers.

The A and B registers divide the output frequency down to the internal comparison frequency of our circuit. As this frequency is specified as 100KHz, we must first divide the reference frequency by a factor of 100 (R = 100) to meet this speciation. The RF signal must now also be divided to meet this comparison frequency and the values for the division registers can be calculated using Equation 2 and are summarised in Table 7.

$$F_{out} = \frac{F_{Ref}}{R}[PB + A]$$

Equation 2 - Fout from Fref division [1]

Divider/Scaler	Value
R	100
P	8
A	250
B	0

Table 7 - PLL Values for 200MHz [1]

These values were then converted to their binary equivalent and organised according to the ADF4112 datasheet. These registers were coded onto a Adafruit trinket using the Arduino IDE. This code interfaced over the SPI bus to the PLL controller uploading our set of registers.

Register	Value (HEX)
RefReg	0x00100190
FuncReg	0x000D8092
InitReg	0x000D8093
NReg_200MHz	0x0000FA01
NRed_197MHz	0c0000F0C9

Table 8 - PLL Register Values [1]

Lab Results

Using this first code we successfully confirmed our circuit locked onto the chosen frequency of 200MHz. This measurement was made using a spectrum analyser connected through a coax cable to the SMA connector, to see the frequency components of the output signal (Figure 8 - VCO output for Vtune 2.2v in LTSPICE (top) and in reality (Bottom)Figure 8). Figure shows this as there is one distinct peak over the measured frequencies.

The lock was also confirmed by the LED being lit. The LED is driven high when the chips detects 3 PD cycles with less than 15 ns deviation. From this we were able to change the registers so that the PLL now produced the 197MHZ signal assigned to our group. This was performed in the same method as before. The lock was then again confirmed with the frequency and power being recorded in Table 9. This time the frequency was measured using the SAs built in frequency counter so was recorded at a much higher precision than before.

Measure	Desired	Actual
Output power (dBm)	>-10	-8.5
Output frequency (MHz)	197	197.006555

Table 9 - PLL Output characteristics

Frequency Hopping

As a final extension to our project, we investigated the response of the PLL to changes in output frequency. To set this up, the code was changed so it would cycle though an array of register values cycling the set frequency (Appendix 2). With this code programmed into the Trinket, the circuit performed the changes as desired but due to decision made when writing the code, the time between steps was very large.

To improve the speed of the code, delays included within the bit-banging implementation of a “SPI” bus were removed and a more efficient function was chosen to toggle the output pins (Appendix 2). These improvements resulted in a bus clock speed of 8MHz over 20 times quicker than organically.

Using these improvements, we were able to experiment with step size to adjust the overshoot and speed of frequency hopping. For a jump from 195 to 205Mhz, (70-80ms Figure 12), there was be a large overshoot (0.7v) and settling time (4ms). A disadvantage of this is that it takes the VCO out of specification when exceeding the 0.5-4.5v designed range. Furthermore, the long settling time reduce the applicability of our circuit to systems that require fast frequency hopping such as GSM.

To improve on this response, we broke one large 10MHz step into 10x 1MHz steps and allowed the PLL to lock between each (10-55ms, Figure 12). The overshoot has been reduced to 0.1v however the time for the device to settle on the final frequency has increased significantly to 20ms.

The time between 55-70ms of Figure 12, represents our best attempt of switching frequency quickly while reducing overshoot. To further attempt to improve the step response of our circuit we calculated the transfer function of the loop filter as it is the loop filter that defines the response to the change in frequency.

The method we used to undertake this was to calculate the output voltage (voltage across C3) in terms of the input current from the charge pump. The equations are included in appendix 1. Using this, a graph of a frequency step response has been created (Figure 11) and compared to the measurement taken from the oscilloscope.

Further work is needed in this area to improve the response model to match the measured values as the model has a much smaller overshoot. Once this has been completed we would be able to proceed to adjust the loop filter to maximise either settling time or overshoot.

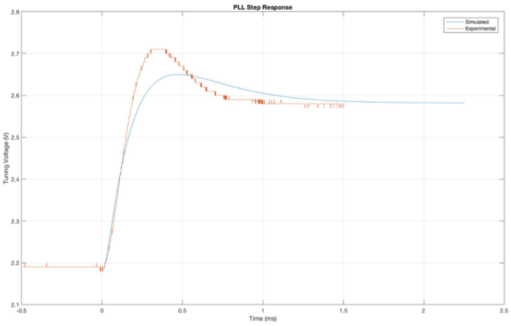


Figure 11 - Step Response Comparison

Conclusion

During this project, we have successfully designed and built a 195-205MHz synthesised oscillator using a phase locked loop that met the required specifications. This process, is an example of the iterative nature of RF circuit design, with calculations, simulations and testing having to be

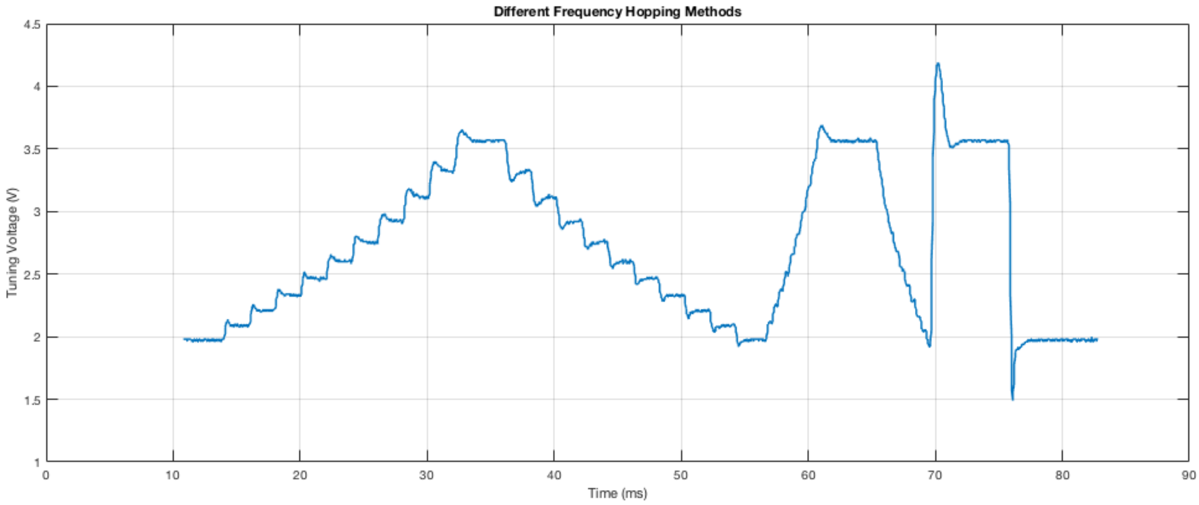


Figure 12 - Frequency hopping methods data taken from oscilloscope measurement of tp2

performed and referred back to. Understanding the concepts is also highlighted, as to be able to adjust a single characteristic of the circuit the significance of each component must be known and how changing one will have unwanted effects.

Finally, using the knowledge gained from previous steps, individual work has been carried out in an attempt to make the circuit useable in a real application such as CDMA and GSM Networks, or Bluetooth [6].

Bibliography

[1] E. Ball, "VHF Synthesiser for Wireless Communications," The University of Sheffield, Sheffield, 2017.

[3] Infineon, "Low Noise Silicon Bipolar RF Transistor," 19 09 2013. [Online]. Available: http://www.infineon.com/dgdl/Infineon-BFP420-DS-v01_01- en.pdf. [Accessed 05 2017].

[4] SKYWORKS, "SMV1247-SMV1255 Series: Hyperabrupt Junction Tuning Varactors," 18 08 2016. [Online]. Available: http://www.skyworksinc.com/uploads/documents/SMV1247_SMV1255_Series_200061U.pdf . [Accessed 05 2017].

[5] Analog Devices, "RF PLL Frequency Synthesizers," 2013. [Online]. Available: http://www.analog.com/media/en/technical-documentation/data-sheets/ADF4110_4111_4112_4113.pdf. [Accessed 05 2017].

[6] J. Haartsen, "Bluetooth - The Universal Radio Interface for Ad Hoc Wireless Connectivity," *Ericsson Review,* vol. 3, no. 10, 1998.