CHAPTER 5

IMPLEMENTATION OF
DIGITAL FREQUENCY SYNTHESIZER

5.1 INTRODUCTION

We use a Digital Frequency Synthesizer in our system to generate a sampled sinusoidal wave of frequency 71 KHz ± estimated carrier frequency offset. The major advantage of Digital Frequency Synthesizer (DFS) is that its output frequency, phase and amplitude can be precisely and rapidly manipulated under the control of a DSP. Other inherent DFS attributes include the ability to tune with extremely fine frequency and phase resolution and to rapidly “hop” between the frequencies. These combined characteristics have made this technology popular in military, radar and communications systems. The digital circuits used to implement signal processing functions do not suffer the effects of thermal drifts, aging and component variations associated with their analog counterparts. The implementation of digital functional blocks makes it possible to achieve a high degree of system integration. Recent advances in IC fabrication technology, particularly the CMOS technology coupled with advanced DSP algorithms and architectures provide possible single chip solutions to complex communication and signal processing sub-systems such as modulators, demodulators, local oscillators, programmable clock generators, cellular base stations and wireless local loop base stations.
5.2 BASIC IDEA OF DIGITAL FREQUENCY SYNTHESIZER

The DFS is shown in a simplified form in Fig. 5.1. It consists of a phase accumulator and a phase to amplitude converter (conventionally a sine ROM) [84]. The phase accumulator consists of a \( j \) bit frequency register, which stores a digital phase increment word followed by a \( j \) bit full adder and a phase register. The digital input phase increment word is entered in the frequency register. At each clock pulse, this data is added to the data previously held in the phase register. The phase increment word represents a phase angle step that is added to the previous value at each \((1/f_{\text{clk}})\) second to produce a linearly increasing phase value. The phase is generated by modulo \(2^j\) overflowing property of a phase accumulator. The rate of overflow is the output frequency, expressed as

\[
f_{\text{out}} = \frac{\Delta P f_{\text{clk}}}{2^j} \quad \forall f_{\text{out}} \leq \frac{f_{\text{clk}}}{2}.
\]

where \(\Delta P\) is the phase increment word, \(j\) is the number of phase accumulator bits, \(f_{\text{clk}}\) is the clock frequency and \(f_{\text{out}}\) is the output frequency. The constraint in the above equation comes from the sampling theorem. As the phase increment word is an integer, the frequency resolution is found by setting \(\Delta P=1\) in the following expression

\[
\Delta f = \frac{f_{\text{clk}}}{2^j}
\]

The read only memory (ROM) is a look-up table, which converts the digital phase information into the values of a sine wave.
5.3 MODULATION CAPABILITY OF DFS

It is simple to add modulation capabilities to the DFS because DFS is a digital signal processing device. In a DFS, it is possible to modulate numerically all three waveform parameters as

\[ s(n) = A(n) \sin(2\pi(\Delta P(n) + P(n))), \]  

(5.1)

![DFS Block Diagram and Wave Shapes](image)

Figure 5.1 DFS Block Diagram and Wave Shapes
where $A(n)$ is the amplitude modulation and $\Delta P(n)$ is the phase modulation. All modulation techniques use one, two or all three basic modulation types simultaneously. Consequently, any known waveform can be synthesized from these three basic types within the Nyquist bandwidth limitations of the DFS. Fig. 5.2 shows the block diagram of the basic DFS system with all three modulations in place [85]. The frequency modulation is made possible by placing an adder before the phase accumulator. The phase modulation requires an adder between the phase accumulator and the phase to amplitude converter. The amplitude modulation is implemented by placing a multiplier after the phase to amplitude converter. The multiplier adjusts the output digital amplitude word.

5.4 BLOCKS OF DIGITAL FREQUENCY SYNTHESIZER

5.4.1 Phase Accumulator

The phase accumulator comprises of a frequency register, an adder and a phase register as shown in Fig. 5.1. The output of the phase register is fed back to the adder as one of its inputs. The other input of the adder comes from the
frequency register. This second input, also called the frequency control word, is added to the previous phase sum on every clock cycle. A clock with frequency $f_{clk}$ is the synthesizer’s only time reference. The phase accumulator’s output is thus a ramp, as it overflows to 0 periodically. Assuming an N-bit accumulator, the frequency of the ramp is given by

$$f_{out} = f_{clk} \times \frac{\text{frequency control word}}{2^N} \quad (5.2)$$

Every value at the output of the phase accumulator is converted to approximated sine amplitude by a phase-to-sine amplitude converter.

### 5.4.2 Phase to Amplitude Converter

The spectral purity of the DFS is also determined by the values stored in the ‘sine’ table ROM. Therefore, it is desirable to increase the resolution of the ROM. Unfortunately, a larger ROM storage means higher power consumption, lower speed and greatly increased costs. The most elementary technique of compression is to store only $\pi/2$ radians of sine wave information and to generate the ROM samples for the full range of $2\pi$ by exploiting the quarter wave symmetry of the sine function. Methods of quarter wave symmetry include trigonometric identity, Nicholas’ method and the Taylor series. A different approach to the phase-to-sine amplitude mapping is the CORDIC algorithm, which uses an iterative computation method. The costs of various methods are increased circuit complexity and distortions that will be generated, when the methods of memory compression are employed. Because the possible number of generated frequencies is large, it is impossible to simulate all of them to find the worst case situation. If the least significant bit of the phase
accumulator input is forced to one, then only one simulation is needed to determine the worst case carrier-to-spur level [106].

5.5 EXPLOITATION OF SINE FUNCTION SYMMETRY

A well-known technique is to store only $\pi/2$ radians of sine information and to generate the sine look-up table samples for the full range of $2\pi$ by exploiting the quarter-wave symmetry of the sine function, as mentioned earlier. The decrease in the look-up table capacity is paid for by the additional logic necessary to generate the complements of the accumulator and the look-up table output. The details of this method are shown in Fig. 5.3. The two most significant bits are used to decode the quadrant, while the remaining (k-2) bits are used to address a one-quadrant sine look-up table. The most significant bit determines whether the amplitude is increasing or decreasing. The accumulator
output is used “as is” for the first and the third quadrants. The bits must be complemented so that the slope of the saw-tooth is inverted for the second and fourth quadrants. As shown in Fig. 5.3, the sampled waveform at the output of the look-up table is a full wave rectified version of the desired sine wave. The final output sine wave is then generated by multiplying the full wave rectified version by -1 when the phase is between \( \pi \) and \( 2 \pi \).

In most practical DFS digital implementations, the numbers are represented in 2’s complement format. Therefore 2’s complement must be used to invert the phase and multiply the output of the look-up table by -1. However, it can be shown that if a \( \frac{1}{2} \) LSB offset is introduced into a number that is to be complemented, then a 1’s complement may be used in the place of the 2’s complement without introducing error [106]. This provides savings in hardware since a 1’s complement may be implemented as a set of simple Exclusive-OR gates. This \( \frac{1}{2} \) LSB offset is provided by choosing look-up table samples such that there is a \( \frac{1}{2} \) LSB offset in both the phase and the amplitude of the samples [106] as shown in Fig. 5.4. In that figure, the phase offset has been used to reduce the address bits by two. If there is no phase offset, 0 and \( \pi / 2 \) have the same phase address and, one more address bit is needed to distinguish between these two values.

5.6 ANALYSIS

Various techniques are available in the literature for quarter wave memory compression, such as Sine-phase difference algorithm, Taylor series expansion, Modified Sunderland Architecture, Nicholas’ Architecture,
CORDIC (Coordinate Rotation Digital Computer) Algorithm. The implicit goal of these phase-to-sine conversion techniques is to reduce the maximum amplitude error for any phase angle, in effect mimicking the behavior of a LUT. In pursuing this goal, all architectures become complex in one way or the other. Also, the ROM size becomes fairly large as it grows exponentially with the width of the phase accumulator whereas a large phase accumulator width is desirable in order to achieve fine frequency tuning. Truncating the phase accumulator output, on the other hand, introduces spurious harmonics.

5.7 CONCEPT OF THE ARCHITECTURE USED

Instead of a ROM LUT, a hardware-optimized phase-to-sine amplitude converter approximates the first quadrant of the sine function with eight equal-length piecewise linear segments [111]. The main goal is to maintain low system complexity and reduce power consumption and chip area requirements. The second aim is to achieve a specified spectral purity, which is defined as the ratio of the power in the desired frequency to the power in the greatest harmonic, across the synthesizer’s tuning bandwidth. Spectral purity is an essential design parameter for synthesizer used in communication systems,
ensuring that undesired in-band signals remain below a given threshold and are not detected.

In order to achieve the first goal, we approximate a sinusoid as a series of eight equal-length piecewise continuous linear segments \( s_i \), where

\[
s_i(x) = m_i \times (x - \frac{i}{8}) + y_i, \quad i \in [0,7] \tag{5.3}
\]

is the slope of each segment and is carefully selected to eliminate the requirement for multiplication by representing each one as a sum of at the most two powers of two. This is well known and often used technique [113]. We also restrict the precision of slope representation, i.e., the difference between the smallest and the largest powers of two used, in effect putting an upper bound on the adder’s width. Equal length segments are selected to reduce the control system circuitry costs. In order to achieve a desired spectral purity, different sets of \( m_i \) and \( y_i \) coefficients are evaluated and the best one meeting the requirements is selected.

### 5.8 DESCRIPTION OF THE ARCHITECTURE

The new DSFS architecture is shown in Fig. 5.5. It corresponds to a set of coefficients yielding 60dB purity. The coefficients are given in Table 5.1. The phase to sine amplitude converter block includes a 1’s complement to exploit quarter wave symmetry as previously seen in other structures. Clearly, this architecture is significantly less complex than those of the other methods discussed previously. It does not include a ROM. No multipliers or squaring circuits are required. Equal length segments are used to simplify the control circuitry. Only three integers need to be added and multiplexers shown in Fig. 5.5 have been optimized by combining similar inputs and implemented in combinational logic.
The phase accumulator is 20 bits wide, truncated to 12 bits. The two Most Signifying Bit (MSBs) are used for quadrant symmetry. The next three bits identify the segment. The remaining seven bits identify different sub-angles. The two upper multiplexers shift these remaining seven bits according to the slopes $m_i$, listed in Table 5.1.

In Fig. 5.5, the notation \{\gg n\} signifies a right shift by n bits, or equivalently, division by $2^n$. The lower multiplexer selects the appropriate $y_i$ approximation listed in the table. The output from the multiplexers is 13 bits wide, to account for the whole dynamic range of possible values. The three-operand adder sums the multiplexer outputs together and rounds the result to 7 bits.

**Table 5.1: Linear Segment Coefficients for 60 dB Purity**

<table>
<thead>
<tr>
<th>$i$</th>
<th>$m_i$</th>
<th>$y_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$1 + \frac{1}{2}$</td>
<td>2/1024</td>
</tr>
<tr>
<td>1</td>
<td>$1 + \frac{1}{2}$</td>
<td>191/1024</td>
</tr>
<tr>
<td>2</td>
<td>$1 + \frac{1}{4}$</td>
<td>384/1024</td>
</tr>
<tr>
<td>3</td>
<td>$1 + \frac{1}{8}$</td>
<td>552/1024</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>697/1024</td>
</tr>
<tr>
<td>5</td>
<td>$\frac{1}{2} + \frac{1}{4}$</td>
<td>819/1024</td>
</tr>
<tr>
<td>6</td>
<td>$\frac{1}{2}$</td>
<td>909/1024</td>
</tr>
<tr>
<td>7</td>
<td>$\frac{1}{8}$</td>
<td>971/1024</td>
</tr>
</tbody>
</table>
5.9 RESULTS

In order to verify the architectural design of DFS, it was coded in Verilog conforming to RTL coding guidelines. The spectrum was obtained by taking the DFT of one grand repetition of the system output data. It can be easily seen that all the spurs are at least 60 dB below the fundamental. Spectra for other odd frequency control words are similar with spurs no greater than those shown here. It has been shown in [106] that the spectrum corresponding to two frequency control words that are relatively prime to the phase accumulator’s overflow values are permutations of each other. The amplitudes shown in the spectra of Fig. 5.6 are, therefore, exact in amplitude to the spurs for any other odd frequency control word. Only their locations change.

**Figure 5.5 DFS Architecture**
Architecture for digital frequency synthesizer was presented. The main advantage of this architecture is that it does not depend upon the extensive use of ROM, as is normally the case with other commonly available architectures. Hence, it fits into a very small area on the chip. Also, the spectral purity of its output is sufficient for the present system requirement.