A Robust Programable Frequency Divider for Fractional Synthesizers

Victor R. Gonzalez-Diaz†, Guillermo Espinosa Flores-Verdad††, Miguel A. Garcia-Andrade††† and Franco Maloberti†

† University of Pavia, Integrated Microsystems Laboratory. Pavia, Italia.
†† National Institute for Astrophysics Optics and Electornics (INAOE), Puebla, Mexico.
††† Universidad Autonoma de Baja California, Electronics Department, Mexico.

E-mail : victor.gonzalez@unipv.it, gespino@inaoep.mx, miguel.grc336@gmail.com, franco.maloberti@unipv.it

ABSTRACT
The phase selection technique is a good solution to design programable dividers widely used in fractional synthesizers. However, as the process parameters vary, the delays in the phase selection logic change, and they erroneously trigger the phase selection. This paper presents an improvement that consists on making the phase selection logic in a semi sequential fashion. In this way if there is any error due to process, temperature or voltage variations; the phase selection logic will correct and pass correctly the pulses in the next clock cycle. The proposed technique is verified with the fabrication of a programable frequency divider in a 0.35μm CMOS process.

Keywords: Frequency divider, Programable, Frequency synthesizers.

1. Introduction
The good performance of a frequency synthesizer in a phase locked loop arrangement strongly depends on the Voltage Controlled Oscillator (VCO) and the frequency divider design. Indeed, a fractional frequency synthesizer needs a frequency divider whose modulus must change during time with a pseudorandom form [1]. The latter characteristic increases the programable divider’s complexity and reliability of the circuit. In addition, the programable divider is be very sensitive to the process, voltage and temperature variations and thereby affecting its performance. This problem rises as the frequency divider processes the output from the VCO to obtain by itself the necessary signals to change the programmable division ratio. Therefore, if a stage in the programable divider has a timing issue, the error is propagated and the frequency divider fails the count.

The phase selection technique [2] is a good option to design a wide-range programable divider. The technique is still the base of many new programable frequency dividers [3]. Nevertheless, it suffers from the hazardous glitches due to the error timing propagations. For illustration purposes Fig. 1 shows the basic concept of the phase selection technique.

For the case in Fig. 1 the VCO is consecutively divided by 2 to obtain 4 signals with a 90° phase difference between each one and they are processed by a Multiplexer (phase selector). In this programable divider a state machine selects a signal in the MUX to obtain the phase switching. For instance, in Fig. 2 at time $t_0$ the MUX’s output ($F_{4q}$) is following the signal $F_{4l}$ and the state machine changes the selection to the next following phase $F_{4q}$. This action is traduced in a miss count of a complete cycle in the VCO and the overall division modulus increases by 1. If during an entire cycle of the programable divider’s output of Fig. 1 ($F_{out}$), the MUX changes $2^M$ times to the next following signal, the total division is ($N + 2^M$). N is the total entire division modulus from the VCO to the output signal ($N = 64$ for Fig. 1). The multiplexer’s output generates the signals that are used in the state machine to obtain the desired phase changes number M. The phase selection is smooth as long as the state machine is capable to change the multiplexer’s value for the time in which the following phases have the same logic level. This can be achieved by designing the state
2. The semi sequential programable divider

One of the principal issues of the phase selection technique are the glitches in the multiplexer’s output as they trigger the subsequent stages of the divider and the overall division fails. The other source of glitches is the state machine as it takes the output from the multiplexer to change the selected phase. Several solutions have been focused to avoid the undesirable glitches only from the former source but we focus in both of them. The conventional approach is the location of subsequent phases as to avoid a critical glitch when the phase selection is held [4]. Fig. 3 shows the case when two following phases ($F_{4i}$, $F_{4q}$) are allocated in a forward configuration and when they are allocated in a backward configuration. For the case of a forward allocation, the phase selection can be obtained in a smooth way only for a delay time $t_x$ within the $(90-180)°$ and in the $(270-360)°$ ranges. On the other hand, when the following phases ($F_{4i}$, $F_{4q}$) are in a backward configuration a smooth phase selection can be obtained for the $(0-270)°$ range. The former case has a higher probability to fail the count as the time issues are within the whole period in the multiplexer’s output. For the latter case a disadvantage comes as the phase selection logic must be faster and thus increases the power consumption. An alternative solution could be to lower the frequency in the state’s machine control logic but the overall division modulus is changed [4].

In this work it is proposed to design the state machine in the programable divider in a pseudo sequential fashion. Fig. 4 shows the proposed architecture for a 4-bit programable divider that takes the multiplexer’s output to generate a semi sequential phase selection logic. The proposed solution not only divides the signal $F_4$ to obtain the necessary signals for the phase selection logic but also takes it as the clock signal for a registers chain to obtain the signals $F_8$, $F_{16}$, $F_{32}$, $F_{64}$ (and their complements). With this registers chain, the divided signals $F_4-F_{64}$ switch at the same time and the phase selection logic yields the necessary pulses for the multiplexer with no glitches. If there is a variation on the delay signals for the chain of divided signals, the synchronization network corrects and transfers them in the next clock pulse. Also as the phase selector is a simple static multiplexer the glitches may appear at the output. Therefore, a glitch filter is set at the phase selector’s output as is also shown in Fig. 4 to enhance the performance.
Fig. 5. Improvement of the high frequency divide by 2 circuit.

Fig. 6. Phase logic for the programmable divider.

The first stage divide by two cells are designed as a master-slave based divide by two improved topology. The improvement is on the buffered outputs of the master-slave architecture that needed to change the phase signs with two digital inverters that limit the maximum speed [5]. We propose to change the phases with a simple crossing of the paths as is shown in Fig. 5. This also increases the speed of the architecture and also can be used for the medium frequency divide by 2 circuits to obtain the $F_{4i} - F_{4q}$ signals.

With this improvements in the phase-selection based programmable divider, the selection logic of figure 4 can be even designed with the static cells in Fig. 6. The number of pulses in the selection logic depends on the 4-bit M word that controls the number of times the multiplexer will change the phase with a simple counter.

To test the programmable divider an integrated LC tank VCO was designed to obtain on-chip the input signal of the circuit. The VCO has a LC-tank topology as it has the best noise performance and has a tuning range from $(1.49 - 1.64)$GHz.

The post layout simulations of the programmable divider are respectively shown. The simulations were executed for the worst case operation conditions when the programmable divider triggers all the phases in a complete output period (that is for $N = 79$). The result for the worst case speed is shown in the left side of Fig. 7, has a VCO output frequency of 1.6010GHz and divided by 79 to obtain a frequency of 20.27MHz. The same occurs for the worst power post simulation result with an output frequency of 21.39MHz (shown in the right side in Fig. 7). The post layout simulations were run for all the corner cases with voltage and temperature variations and the circuit worked as expected for all cases, without glitches in the pulses for the phase selection logic and obtaining correctly the desired division modulus. The programmable divider is therefore robust to those parameters variations.

3. Experimental results

To prove the proposed improvement of the phase-selection technique a 4-bit programmable frequency divider was fabricated in a 0.35μm CMOS process. Fig. 8 shows the microphotograph of the chip. The VCO is also integrated to generate on-chip the signal that will be divided by the High Frequency divide by two cells. Thereafter, the four signals $F_{4i} - F_{4q}$ are used in the improved phase selection logic. The chip was prepared to change the VCO’s control voltage to change the programmable divider input. Also, a 4-bit digital control signal is used to change the modulus division from $N = (64..79)$.

The experimental results are shown in the figure 9 from the capture of an Agilent 54622D Oscilloscope. Also, in the plots, the output signal from the output’s half frequency is shown. The plot on the left side of Fig. 9 shows the experimental results when the VCO
is tuned at 0V and the 4-bit digital input is in 0 (for that case the multi-modulus frequency divider is at $N = 64$). In this conditions the phase-select logic is deactivated and the division is surely $N = 64$. From this discussion the VCO output frequency can be known if the programable divider’s output frequency is multiplied by 64. In the left side plot of Fig. 9 it can be seen that the divide by 64 circuit output frequency is $f_{64} = 25.6MHz$. Therefore, the VCO output frequency is for sure 1.64GHz.

If the tuning voltage of the VCO is maintained at 0V, and so the VCO output frequency is at 1.64GHz, the modulus division can be changed and the divider output frequency will change. In the right side plot of Fig. 9 it is shown the output waveform from the programable divider when $N = 79$. It can be seen that the frequency at the divider’s output is now $f_{79} = 20.7MHz$ which is congruent with the division modulus of $N = 79$, for a VCO output frequency = 1.64GHz.

The programable divider was tested when the VCO was tuned at 3.3V. The left side plot of Fig. 10 shows the case when $N = 64$ and the output frequency in this case is $f_{64} = 23.4MHz$. The VCO output frequency is $64 \times 23.4MHz = 1.49GHz$. When the division modulus is programmed at $N = 79$, the expected output frequency is $f_{79} = 1.49GHz / 79$. Form the right side plot of Fig 10 the measured frequency is 18.9MHz it match wells with the estimated frequency.

### Table 1. Programable divider characteristics

<table>
<thead>
<tr>
<th>Size (including I/O pads)</th>
<th>(611 x 490) (μm)$^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{DD}$</td>
<td>3.3V</td>
</tr>
<tr>
<td>Prog. Div. power</td>
<td>130mW</td>
</tr>
<tr>
<td>VCO power</td>
<td>86mW</td>
</tr>
<tr>
<td>Frequency range</td>
<td>1.49 – 1.64GHz</td>
</tr>
<tr>
<td>Programable range</td>
<td>$N = 64 – 79$</td>
</tr>
</tbody>
</table>

**Fig. 7. Post simulation results of the programable divider.**

**Fig. 9. Experimental result of the programable divider.**
Table 1 resumes the characteristics of the programmable divider. The power consumption is even more than the VCO being both of them the most power consuming circuits.

4. Conclusion

An improved phase selection-based programmable frequency divider was proposed. The improved architecture uses the same output from the phase-selector (multiplexer) to make the phase selection logic in a semi-sequential fashion. In this way, if the process, voltage and temperature variations make to appear the glitches in the selection logic, the synchronization network makes to trigger all the signal in the following clock pulse. The method was verified with experimental results of a frequency divider with an integrated LC tank VCO.

Acknowledgment

This work was supported by CONACyt Mexico with grant #131617 and also to the National Program Italian FIRB with project number #RBAP06L4S5.

References


