Advanced technique for high accuracy tunable ring oscillator vernier TDC in FPGAs and ASICs

E. Bechetoille, C. Girerd, H. Mathez
Université de Lyon 1, Villeurbanne;
CNRS/IN2P3, Institut de Physique Nucléaire de Lyon.
Outlines

- Vernier TDC with Ring oscillator principles
- Tuneable ring oscillator architecture
- Test & results
- Limitations of cumulated jitter
- Conclusions
Ring-Oscillator based TDC Architecture

Advantages:

- Simple architecture: Low Area, low consumption, can implemented in standard cells
- The TDC resolution is given by the frequency difference between oscillators
- In theory it can be very small, as small as the frequency difference

- In practice: There are some limitations as jitter …
Ring oscillators Vernier TDC timing

Step 1  The slow oscillator is started on the START signal
Step 2  The fast oscillator is started on the STOP signal
Step 3  At each period the Fast clock get an advance of $\Delta t = (T_1 - T_0)$ over the slow clock
Step 4  The two oscillators are stopped when there are in phase and counters are latched

The resolution is given by:

$$T = (N_0 \cdot T_0) - (N_1 \cdot T_1)$$

The delay measurement is:

$$T = T_0 \cdot (N_0 - N_1) + N_1 \cdot \Delta t$$

$$T_0 - T_1 = \Delta t$$
Simple Ring oscillator

-The number of stages must be odd to allow oscillations
-The number of stages determines the oscillator period
-The OSCILLATOR is gated by and AND for start and stop.

\[ f = \frac{1}{\sum_n T_{pHL(n)} + \sum_n T_{pLH(n)}} \]
Phase detector

Slow clock sampled by fast clock

Detects when slow clock previously at state HIGH goes to LOW state

This is the commonly used phase detector in this type of TDC
FPGA IMPLEMENTATION

- Example of Ring oscillator schematic
- Ignore LCELL buffers = OFF

- For the synthesis tool it is a COMBINATIONAL LOOP which is a bad design practice
- Also without explicit LCELL instantiation the synthesis tool will optimize and reduce the inverter chain to only one inverter

- LCELL buffers prevent the synthesis tool to optimize the design

- A VHDL description is also possible

In practice: It is better to synthesize the ring oscillator in open loop in order to allow the timing analysis.
Design methodology (Export/Import of partitionation / Regions / Post compilation Edition)

Partition importation

Routing preservation constraints
Limitations of classic implementation

- This classic architecture has been used successfully in some physics applications see

Implementation of sub-nanoseconds TDC in FPGA: applications to time-of-flight analysis in muon radiography

  J. Marteau (IPNL), J. De Bremond D’ars (GR), D. Gibert (GR, IPGP), K. Jourde (IPGP), S. Gardien (IPNL), C. Girerd (IPNL), J.-C. Ianigro (IPNL)

The main problem of classic method is that is very hard to target a specific resolution for the TDC (In fact to have a specific period difference between Slow and Fast oscillator).

Several iterations and manual placement and routing can be necessary to obtain the wanted resolution.

That’s the reason why we think to an other technique to control the period difference of ring oscillators.
ADVANCED TECHNIQUE OF TUNEABLE RING OSCILLATORS IN FPGA
TUNEABLE RING OSCILLATORS

Goal:
- Modifying the frequency by a digital control, on line.
- Being able to target a specific $\Delta t$ between two oscillators

Techniques:
- Using the propagation delay variations of logic cells
- Modifying the path of the signal in the chain
- Preference for structures with low variations
TUNEABLE OSCILLATORS
Moving inverters ©

Basic tuneable cell

\[ T_{p_{\text{inv}}} \]

\[ T_{p_{\text{pass}}} \]

\[ \text{enable} \]

\[ \text{Osc}_{\text{sel}[n..0]} \]

Tp_{pass} \neq Tp_{inv}

<table>
<thead>
<tr>
<th>sel</th>
<th>Tp</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Tp_{pass}</td>
</tr>
<tr>
<td>1</td>
<td>Tp_{inv}</td>
</tr>
</tbody>
</table>

Condition: Somme Cells must be different (e.g. \( T_{p_{\text{pass}}(n)} \neq T_{p_{\text{pass}}(n+1)} \))

- Involuntary due to silicon dispersion.
- Voluntary by choosing inverters of difference strength (propagation delay)
TUNABLE OSCILLATORS
Moving inverters ©

Basic tuneable cell

\[ \begin{array}{c|c|c}
\text{Sel} & \text{In} & \text{Out} \\
\hline
0 & 0 & 0 \\
0 & 1 & 1 \\
1 & 0 & 1 \\
1 & 1 & 0 \\
\end{array} \]

The basic element can be replaced by an XOR cell

The control word can change the position and the number of inverters
Example with 8 cells

The number of selected inverters must be odd (oscillation condition)

For a chain of n Elements the number of possibilities $Y$ is:

$$Y = \sum_{p=2k+1}^{n} C_n^p$$

$$Y = C_8^1 + C_8^3 + C_8^5 + C_8^7 = 128$$

The number of TDCs (combination of 2 oscillators)

$$128^2 = 16384$$

But how to quantify the differences between all these combinations
TUNEABLE OSCILLATORS
Moving inverters ©

Enable_fast
\[ \text{sel_fast}[7..0] \]

\[ T_1 = T_{\text{pass}} \text{A} + T_{\text{pass}} \text{B} + T_{\text{inv}} \text{C} + T_{\text{pass}} \text{D} + T_{\text{pass}} \text{E} + T_{\text{pass}} \text{F} + T_{\text{inv}} \text{G} + T_{\text{inv}} \text{H} \]

We change on bit in the select word

Enable_fast
\[ \text{sel_fast}[7..0] \]

\[ T_2 = T_{\text{pass}} \text{A} + T_{\text{pass}} \text{B} + T_{\text{pass}} \text{C} + T_{\text{pass}} \text{D} + T_{\text{inv}} \text{E} + T_{\text{pass}} \text{F} + T_{\text{inv}} \text{G} + T_{\text{inv}} \text{H} \]

\[ T_2 - T_1 = (T_{\text{pass}} \text{C} - T_{\text{inv}} \text{C}) - (T_{\text{pass}} \text{E} - T_{\text{inv}} \text{E}) \]

Conditions to obtain a delay variation: \( T_{\text{pass}}(i) \neq T_{\text{inv}}(i) \) \( T_{\text{pass}}(i) \neq T_{\text{pass}}(k) \) \( T_{\text{inv}}(i) \neq T_{\text{inv}}(k) \)
More generally if we have $N$ available family of XOR gate
The list of possible differences for one change in one oscillator is $N(N-1)$

\[
(\Delta_{a,b})_{a\neq b\in[1,n]} = (T_{pass_a} - T_{inv_a}) - (T_{pass_b} - T_{inv_b})
\]

These differences depends one the XOR family (a vs b) and between function passing/inverting in the same family (pass/inv)

CONCLUSION : In order to obtain frequency variations, we must use the maximum of different XOR (propagation delays) in each oscillator

BUT How to implement XOR cells with different propagation delay in a FPGA ?
LCELL STRUCTURE
(ALTERA exemple Cyclone III)

- The combinational parts is Look Up table (65536) possibilities of equations of 4 inputs
- The oscillator stages are implemented in one of fewer LCELLs
Propagation delay characteristics of LCELL in ALTERA FPGA

We observed that in the ALTERA FPGA the XOR propagation delays depends on the input which is used. Example with 4 XOR gates using different inputs

<table>
<thead>
<tr>
<th>Entrée</th>
<th>RR (rising input / Rising output) ps</th>
<th>FF Falling input Falling output ps</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>471</td>
<td>481</td>
</tr>
<tr>
<td>B</td>
<td>494</td>
<td>496</td>
</tr>
<tr>
<td>C</td>
<td>324</td>
<td>316</td>
</tr>
<tr>
<td>D</td>
<td>177</td>
<td>155</td>
</tr>
</tbody>
</table>
Open loop synthesis of ring oscillators

- Allow to specify SDC constraints and control the first synthesis results
- Here a clock constraint is enough to obtain a minimum XOR chain propagation delay

- A partition and a region for each oscillator
Report path results (Timequest) for open loop synthesis

<table>
<thead>
<tr>
<th>Total</th>
<th>Incr</th>
<th>RF</th>
<th>Type</th>
<th>Fanout</th>
<th>Location</th>
<th>Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>FF</td>
<td>CELL</td>
<td>1</td>
<td>FF_X2_Y28_N17</td>
<td>inst</td>
</tr>
<tr>
<td>0.53</td>
<td>0.53</td>
<td>FF</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N12</td>
<td>inst1</td>
</tr>
<tr>
<td>0.846</td>
<td>0.316</td>
<td>FF</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N12</td>
<td>inst1</td>
</tr>
<tr>
<td>1.106</td>
<td>0.26</td>
<td>FF</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N18</td>
<td>inst1</td>
</tr>
<tr>
<td>1.274</td>
<td>0.168</td>
<td>FR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N18</td>
<td>inst1</td>
</tr>
<tr>
<td>1.514</td>
<td>0.24</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N16</td>
<td>inst1</td>
</tr>
<tr>
<td>1.691</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N16</td>
<td>inst1</td>
</tr>
<tr>
<td>1.931</td>
<td>0.24</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N6</td>
<td>inst1</td>
</tr>
<tr>
<td>2.108</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N6</td>
<td>inst1</td>
</tr>
<tr>
<td>2.349</td>
<td>0.241</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N24</td>
<td>inst1</td>
</tr>
<tr>
<td>2.525</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N24</td>
<td>inst1</td>
</tr>
<tr>
<td>2.768</td>
<td>0.242</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N2</td>
<td>inst1</td>
</tr>
<tr>
<td>2.945</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N2</td>
<td>inst1</td>
</tr>
<tr>
<td>3.187</td>
<td>0.242</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N20</td>
<td>inst1</td>
</tr>
<tr>
<td>3.364</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N20</td>
<td>inst1</td>
</tr>
<tr>
<td>3.604</td>
<td>0.24</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N10</td>
<td>inst1</td>
</tr>
<tr>
<td>3.781</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N10</td>
<td>inst1</td>
</tr>
<tr>
<td>4.021</td>
<td>0.24</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N0</td>
<td>inst1</td>
</tr>
<tr>
<td>4.198</td>
<td>0.177</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>LCCOMB_X3_Y28_N0</td>
<td>inst1</td>
</tr>
<tr>
<td>4.833</td>
<td>0.635</td>
<td>RR</td>
<td>IC</td>
<td>1</td>
<td>DDIOOUTCELL_X3_Y29_N32</td>
<td>inst7</td>
</tr>
<tr>
<td>5.278</td>
<td>0.445</td>
<td>RR</td>
<td>CELL</td>
<td>1</td>
<td>DDIOOUTCELL_X3_Y29_N32</td>
<td>inst7</td>
</tr>
</tbody>
</table>

- The synthesis tool choose the fastest LCELL input for all cells (input D / 177 ps)

- the next step is to manually change the input to obtain an XOR chain with a maximum of possible delay adjustment.

11-13 June 2014 Journées VLSI-PCB-FPGA-IN2P3 Marseille 2014
Post compilation editing (1) ECO with **CHIP PLANNER** and **RESSOURCE PROPERTY EDITOR**

Modification of:
- Connexions
- Equations
- LUT ...
Example of chain with 8 XORs using inputs B,C,D after Post compilation Edition (input A is used for selection)

Timing report for a chain of 8 XOR. Routing has been manually modified to use 3xC inputs, 3xD inputs and 2xB inputs.
Consequence on period variation in the ring oscillator

As seen before the list of possible variation is in theory given by

\[(\Delta_{a,b})_{a \neq b \in [1,n]} = (T_{\text{pass}_a} - T_{\text{inv}_a}) - (T_{\text{pass}_b} - T_{\text{inv}_b})\]

Remark: It is possible to have an estimation using the min/max delay as the timing analyser make a transitions analysis

\((T_{\text{max}} - T_{\text{min}})\) for each 8 cells in the chain

Then we calculate all the possible delays

\[N(N-1) = 7 \times 8 = 56\]

We can note the dynamic of variation

\(~ 150 \text{ ps}\)

and the resolution

\(~ \text{few ps}\)
TESTS & RESULTS
TDC bloc diagram

- Clk ref
- Internal trigger generator
- Trigger detect
- Calibration Enable
- DAQ interface
- CPU / Controller Ex: NIOS
- Coarse Counter
- Slow Counter
- Fast Counter
- Calibration Counters
- Calibration Enable
- Ready
- Phase detector
- Osc1
- Osc2
- Sel1
- Sel2
- Fast Osc
- Slow Osc
- Inversion
- Clear
- Start Slow
- Start fast
- trigIn
- Trigger detect
- Internal trigger generator
- $T_{\text{slow}}$ can be calibrated using an external and stable clock reference

\[
N_{\text{calib}} \cdot T_{\text{slow}} + E = N_{\text{ref}} \cdot T_{\text{ref}}
\]

\[
T_{\text{slow}} = \frac{T_{\text{ref}} \cdot N_{\text{ref}}}{N_{\text{calib}}} - \left( \frac{E}{N_{\text{calib}}} \right)
\]

\[
\text{Err}_{\text{max}} = \frac{T_{\text{slow}}}{N_{\text{calib}}}
\]

- Error on $T_{\text{slow}}$ is minimized by $N_{\text{calib}}$
Calibration of $T_{\text{fast}}$ or $\Delta t$

- If SLOW and FAST are free running they will periodically reach a minimum phase shift.
- The number of clocks periods between this coincidences gives the period difference.

Vernier definition gives

$$\Delta t = \frac{T_{\text{slow}}}{N_1}$$

We can also measure directly $T_{\text{fast}}$ with the clock reference as for $T_{\text{slow}}$. 
Calibration results on hardware

The calibration consist in sweeping all the combination of slow and fast oscillators (in this case with 3 inverters among 8) when obtained 3146 combinations.

The calibration result is a list of resolutions with the corresponding TDC selection. We can choose any TDC among these 3146.

*At least the oscillators difference can be as low as 1 ps but it doesn’t guarantee that the will work well at this resolution.
Calibration results
minimum frequency differences

We observed that period differences (between slow and fast oscillator) as low as < 1 ps can be obtained with the moving inverter method in FPGA.
Calibration results
Phase detector measurement

The consistency is checked by comparing the difference of slow and fast oscillator Frequencies (blue curve) with the measure obtained with the phase detector (red curve)
Full range (25 ns) measurement with TDC
selection slow=148 fast=22 (Res=50 ps)

TDC measurement between a 40 MHz (25 ns) clock ref (FPGA PLL) and a
synchronous trigger delayed by step of 100 ps over a range > 25 ns

σ = 34 ps
DNL histogram (50 ps TDC selected)

This DNL histogram is obtained by sweeping the delay of inputs signals from 0 to 1 ns by step of 10 ps (100 measures by step)

Mean of bin width is: 52ps (expected 50ps)

Differential non linearity:
+/- 15 ps max
< 0.3 LSB
Limitations of this architecture
Cumulated Jitter
Visualization of the cumulated jitter for a R.O. of T=3.3 ns (7x inverters)
1 ns / div period jitter= 9.4 ps  cycTocyc=12.5 ps (40GSps)
Problems and limitations
e.g. cumulated jitter on a 37 ps TDC selection

We observed that the TDC jitter increases with the delay range.
This means that for low resolutions TDC the delay range must be reduced to preserve a low jitter regarding the resolution.
Cumulated jitter vs. number of cycle periods for different ring oscillators

Conclusion: for a same number of cycles the fastest oscillator has the smallest jitter.
- The cumulated jitter does not depend on the frequency oscillator.
- Cumulated jitter increases linearly with time.
- A periodic component could exist depending on setup configuration (i.e. in our case a 40 MHz oscillator entering in the FPGA).
HYBRID Architecture: Vernier + Delay Chain

Goal: divide the measurement range into small parts
in order to reduce the impact of jitter
Hybrid II adjustable delay Line

*(preliminary results)*

The maximum oscillation cycles is about 12 (excepted for the widest bin)
Conclusions

• We presented a new technique of tuneable oscillators to be used in vernier TDC or other applications with high sensitivity in frequency adjustment.
• We found that the cumulated jitter prevent us to exploit all the potential of this oscillators
• We started to test an hybrid architecture with delay chain and multiple phase detector stage to reduce the jitter effect
• An ASIC implemention has been sent in foundry, we hope that the jitter effect will lower with a dedicated routing layout.
Références

- FPGA-Based High Area Efficient Time-To-Digital IP Design
- Performance and area tradeoffs in space-qualified FPGA-based time-of-flight systems
- An FPGA wave union TDC for time-of-flight applications
- FPGA-Based Self-Calibrating Time-to-Digital Converter for Time-of-Flight Experiments
- Upgrading of Integration of Time to Digit Converter on a Single FPGA
- Area efficient time to digital converter (TDC) architecture with double ring-oscillator technique on FPGA ...  
- FPGA based self calibrating 40 picosecond resolution, wide range Time to Digital Converter
- The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay
- Implementation of High-Resolution Time-to-Digital Converters on two different FPGA devices
- Several Key Issues On implementing delay line Based TDCs using FPGAs