A Survey of AIS-20/31 Compliant TRNG Cores Suitable for FPGA Devices

Oto PETURA, Ugo MUREDDU, Nathalie BOCHARD, Viktor FISCHER, Lilian BOSSUET

Univ Lyon, UJM-Saint-Etienne, CNRS
Laboratoire Hubert Curien UMR 5516
F-42023, SAINT-ETIENNE, France
oto.petura@univ-st-etienne.fr

FPL 2016, Lausanne, Switzerland, August 2016
Goals Methodology Implementation results Conclusions

Outline

1 Goals

2 Methodology

3 Implementation results

4 Conclusions
Goals of the TRNG evaluation

Fair comparison of different TRNG principles in terms of:

- feasibility and reproducibility
- area (cost)
- speed (bitrate)
- power consumption
- entropy
Selected TRNG principles

Based on the selection criteria:

- AIS-31 compliance
- Feasibility in FPGAs

The next TRNGs were selected and implemented:

- Elementary oscillator based TRNG (ELO-TRNG)
- Coherent sampling oscillator based TRNG (COSO-TRNG)
- Multiple ring oscillator based TRNG (MURO-TRNG)
- Phase locked loop based TRNG (PLL-TRNG)
- Transient effect ring oscillator based TRNG (TERO-TRNG)
- Self timed ring based TRNG (STR-TRNG)
Outline

1. Goals
2. Methodology
3. Implementation results
4. Conclusions
Methodology to achieve a fair comparison

- Unified external interface
  (as simple as possible)
- Reduced complexity of the design
  (just the TRNG core, no post-processing)
- All designs implemented in all the devices
  (Xilinx Spartan 6 FPGA, Altera Cyclone V FPGA, Microsemi SmartFusion2 FPGA)
- Statistical properties (entropy) evaluated using the procedure B of the AIS-20/31 statistical test suite
**Hardware configuration**

**DUT**
- FPGA module with the RNG core
- Simple serial data interface
- Two LVDS lines (data, clock/strobe)

**Acquisition card**
- Evariste motherboard and Cyclone III FPGA module
- Can store up to 4 MB of continuous data at 0 – 400 Mbits/s
A reference design is used to measure the power consumption of an FPGA with no logic inside (about 4 mW)
The power consumption of the TRNG core is computed by subtracting the consumption of the ‘empty’ project from the total power consumption.

The multiplexers are used to eliminate an impact of output drivers on the power consumption measurement.
Evaluating parameters

- **Area**
  - in terms of LUTs and registers

- **Net power consumption**

- **Output bit rate**

- **Entropy**
  - evaluated using test T8 of the AIS-20/31 test suite

**Newly defined parameters:**

- **Energy efficiency**
  - number of bits generated consuming one µWs of energy

- **Entropy & bit rate product**
  - bit rate with full entropy
Outline

1. Goals
2. Methodology
3. Implementation results
4. Conclusions
### ERO-TRNG core

![Diagram of ERO-TRNG core](image)

<table>
<thead>
<tr>
<th>Family</th>
<th>N</th>
<th>K</th>
<th>Area (LUT/L&amp;R)</th>
<th>Power cons. [mW]</th>
<th>Bit rate [Mbits/s]</th>
<th>Entropy per bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan 6</td>
<td>3</td>
<td>80</td>
<td>46/19</td>
<td>2.16</td>
<td>0.0042</td>
<td>0.999</td>
</tr>
<tr>
<td>Cyclone V</td>
<td>5</td>
<td>135</td>
<td>34/20</td>
<td>3.24</td>
<td>0.0027</td>
<td>0.990</td>
</tr>
<tr>
<td>SmartFusion 2</td>
<td>5</td>
<td>20</td>
<td>45/19</td>
<td>4</td>
<td>0.014</td>
<td>0.980</td>
</tr>
</tbody>
</table>

---

ERO-TRNG core

Observations:

- Easy to implement – no placement or routing constraints needed
- Very good reproducibility
- Based on the jitter size, the K value might be very high, the size of the counter ($\leq 20$ bits) can affect scalability
COSO-TRNG core

\[ \begin{array}{c}
\text{D} & \text{Q} \\
\text{clk} & \\
\end{array} \]

\[ \text{Beat signal} \]

\[ \begin{array}{c}
\text{D} & \text{Q} \\
\text{clk} & \\
\end{array} \]

\[ \text{Digital noise} \]

\[ \text{Clk} \]

---

<table>
<thead>
<tr>
<th>Family</th>
<th>N</th>
<th>RO freq. [MHz]</th>
<th>Area (LUT/L&amp;R)</th>
<th>Power cons. [mW]</th>
<th>Bit rate [Mbits/s]</th>
<th>Entropy per bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan 6</td>
<td>8</td>
<td>144.5</td>
<td>18/3</td>
<td>1.22</td>
<td>0.54</td>
<td>0.999</td>
</tr>
<tr>
<td>Cyclone V</td>
<td>6</td>
<td>315.5</td>
<td>13/3</td>
<td>0.9</td>
<td>1.44</td>
<td>0.999</td>
</tr>
<tr>
<td>SmartFusion 2</td>
<td>10</td>
<td>185.2</td>
<td>23/3</td>
<td>1.94</td>
<td>0.328</td>
<td>0.999</td>
</tr>
</tbody>
</table>

---

**Observations:**

- The difference in periods has to be very small – difficult to achieve
- Disadvantage: Finding a suitable configuration requires long time (several hours) and the same configuration is not guaranteed to work on another device
- Placement and routing constraints are required
### MURO-TRNG core

<table>
<thead>
<tr>
<th>Family</th>
<th>Area</th>
<th>Power cons.</th>
<th>Bit rate</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan 6</td>
<td>521/131</td>
<td>54.72</td>
<td>2.57</td>
<td>0.999</td>
</tr>
<tr>
<td>Cyclone V</td>
<td>525/130</td>
<td>34.93</td>
<td>2.2</td>
<td>0.999</td>
</tr>
<tr>
<td>SmartFusion 2</td>
<td>545/130</td>
<td>66.41</td>
<td>3.62</td>
<td>0.999</td>
</tr>
</tbody>
</table>

\[
m = 120 \\
K = 100
\]

---

Observations:

- The generator requires a large number of identical rings to be implemented.
- The rings might lock which is extremely hard to detect given their number.
- No need of manual place and route.
PLL-TRNG core

PLL-TRNG core

Observations:

- The PLL setup is not straightforward for some families (Spartan 6: PLL outputs go to different clock domains)
- Once the PLLs are setup, the results are reproducible within the same device family (type of the device)
- PLLs are very well isolated from the rest of the device
TERO-TRNG core


<table>
<thead>
<tr>
<th>Family</th>
<th>Area (LUT/L&amp;R)</th>
<th>Power cons. [mW]</th>
<th>Bit rate [Mbits/s]</th>
<th>Entropy per bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan 6</td>
<td>39/12</td>
<td>3.312</td>
<td>0.625</td>
<td>0.999</td>
</tr>
<tr>
<td>Cyclone V</td>
<td>46/12</td>
<td>9.36</td>
<td>1</td>
<td>0.987</td>
</tr>
<tr>
<td>SmartFusion 2</td>
<td>46/12</td>
<td>1.23</td>
<td>1</td>
<td>0.999</td>
</tr>
</tbody>
</table>
TERO-TRNG core

Observations:
- The placement and routing constraints must be enforced in the TERO loop design
- The two TERO branches must be well unbalanced to get between 100 and 200 oscillations
- Difficult to obtain repeatable results on different devices
\[ L = 255 \]

<table>
<thead>
<tr>
<th>Family</th>
<th>Area</th>
<th>Power cons.</th>
<th>Bit rate</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spartan 6</td>
<td>346/256</td>
<td>65.9 mW</td>
<td>154 Mbits/s</td>
<td>0.998</td>
</tr>
<tr>
<td>Cyclone V</td>
<td>352/256</td>
<td>49.4 mW</td>
<td>245 Mbits/s</td>
<td>0.999</td>
</tr>
<tr>
<td>SmartFusion 2</td>
<td>350/256</td>
<td>82.52 mW</td>
<td>188 Mbits/s</td>
<td>0.999</td>
</tr>
</tbody>
</table>

Observations:

- The ring must have a huge number of cells
- Each cell must be initialized at the beginning and number of events must be verified continuously
- The topology is important – manual placement needed
## Summary of implementation results

<table>
<thead>
<tr>
<th>TRNG type</th>
<th>FPGA device</th>
<th>Area (LUT/Reg)</th>
<th>Power cons. [mW]</th>
<th>Bit rate [Mbits/s]</th>
<th>Efficiency [bits/µWs]</th>
<th>Entropy per bit</th>
<th>Entropy * Bit rate</th>
<th>Feasib. &amp; Repeat.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERO</td>
<td>Spartan 6</td>
<td>46/19</td>
<td>2.16</td>
<td>0.0042</td>
<td>1.94</td>
<td>0.999</td>
<td>0.004</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Cyclone V</td>
<td>34/20</td>
<td>3.24</td>
<td>0.0027</td>
<td>0.83</td>
<td>0.990</td>
<td>0.003</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SmartFusion 2</td>
<td>45/19</td>
<td>4</td>
<td>0.014</td>
<td>3.5</td>
<td>0.980</td>
<td>0.013</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COSO</td>
<td>Spartan 6</td>
<td>18/3</td>
<td>1.22</td>
<td>0.54</td>
<td>442.6</td>
<td>0.999</td>
<td>0.539</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Cyclone V</td>
<td>13/3</td>
<td>0.9</td>
<td>1.44</td>
<td>1 600</td>
<td>0.999</td>
<td>1.438</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SmartFusion 2</td>
<td>23/3</td>
<td>1.94</td>
<td>0.328</td>
<td>169</td>
<td>0.999</td>
<td>0.327</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MURO</td>
<td>Spartan 6</td>
<td>521/131</td>
<td>54.72</td>
<td>2.57</td>
<td>46.9</td>
<td>0.999</td>
<td>2.567</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Cyclone V</td>
<td>525/130</td>
<td>34.93</td>
<td>2.2</td>
<td>62.9</td>
<td>0.999</td>
<td>2.197</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SmartFusion 2</td>
<td>545/130</td>
<td>66.41</td>
<td>3.62</td>
<td>54.5</td>
<td>0.999</td>
<td>3.616</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLL</td>
<td>Spartan 6</td>
<td>34/14</td>
<td>10.6</td>
<td>0.44</td>
<td>41.5</td>
<td>0.981</td>
<td>0.431</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Cyclone V</td>
<td>24/14</td>
<td>23</td>
<td>0.6</td>
<td>43.4</td>
<td>0.986</td>
<td>0.592</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SmartFusion 2</td>
<td>30/15</td>
<td>19.7</td>
<td>0.37</td>
<td>18.7</td>
<td>0.921</td>
<td>0.340</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TERO</td>
<td>Spartan 6</td>
<td>39/12</td>
<td>3.312</td>
<td>0.625</td>
<td>188.7</td>
<td>0.999</td>
<td>0.624</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Cyclone V</td>
<td>46/12</td>
<td>9.36</td>
<td>1</td>
<td>106.8</td>
<td>0.987</td>
<td>0.985</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SmartFusion 2</td>
<td>46/12</td>
<td>1.23</td>
<td>1</td>
<td>813</td>
<td>0.999</td>
<td>0.999</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STR</td>
<td>Spartan 6</td>
<td>346/256</td>
<td>65.9</td>
<td>154</td>
<td>2 343.2</td>
<td>0.998</td>
<td>154.121</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Cyclone V</td>
<td>352/256</td>
<td>49.4</td>
<td>245</td>
<td>4 959.1</td>
<td>0.999</td>
<td>244.755</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SmartFusion 2</td>
<td>350/256</td>
<td>82.52</td>
<td>188</td>
<td>2 286.7</td>
<td>0.999</td>
<td>188.522</td>
<td></td>
</tr>
</tbody>
</table>
1. Goals
2. Methodology
3. Implementation results
4. Conclusions
All the presented TRNG cores are feasible in all major FPGA families

COSO and TERO TRNGs are impractical in their current state
(They both require per device placement and routing)

Each TRNG has its pros and cons

Presented implementations are not fully optimized
(Final optimization is a question of the target application)

Quality of the TRNG design depends not only on the principle used
(Hardware used and implementation itself are very important too)

VHDL source code is available at:
https://labh-curien.univ-st-etienne.fr/cryptarchi/HECTOR_TRNG_designs
This work was performed in the framework of the project **HECTOR**

Hardware Enabled Crypto and Randomness

The HECTOR project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement number 644052 starting from March 2015

www.hector-project.eu
Thank you for your attention