Efficient Power and Timing Side Channels for Physical Unclonable Functions

CHES, September 26, 2014

U. Rührmair (a), (*) , X. Xu (b), (*), J. Sölter (c), A. Mahmoud (a), M. Majzoobi (d), F. Koushanfar (d), W. Burleson (b)

(a) TU München, (b) University of Massachusetts at Amherst
(c) Freie Universität Berlin, (d) Rice University
(*) These authors contributed equally
Outline

1. **Background: The Arbiter PUF Family, Pure Modeling Attacks, and Their Limitations**

2. Power and Timing Side Channels on XOR Arbiter PUFs

3. Combining Side Channels with Modeling Attacks

4. Our Results

5. Summary
Physical Unclonable Functions (PUFs)
Physical Unclonable Functions (PUFs)

PUF
(= (partly) disordered, unclonable physical system S)

Responses $R_i$
($R_i$ is a function of the applied challenge $C_i$ and the specific disorder in S)

$(C_i, R_i)$: Challenge-response pairs (CRPs) of the PUF

External Stimuli/Challenges $C_i$
Physical Unclonable Functions (PUFs)

Strong PUFs:

PUF

(= (partly) disordered, unclonable physical system $S$)

Responses $R_i$

($R_i$ is a function of the applied challenge $C_i$ and the specific disorder in $S$)

$(C_i, R_i)$: Challenge-response pairs (CRPs) of the PUF
Physical Unclonable Functions (PUFs)

**Strong PUFs:**

- Challenge-response interface is publicly accessible
  - **Everyone** who holds physical possession of the Strong PUF can freely apply challenges and read out responses

**PUF**

(= (partly) disordered, unclonable physical system $S$)

**Responses $R_i$**

($R_i$ is a function of the applied challenge $C_i$ and the specific disorder in $S$)

($C_i, R_i$): Challenge-response pairs (CRPs) of the PUF
Physical Unclonable Functions (PUFs)

**Strong PUFs:**

- Challenge-response interface is publicly accessible
  - **Everyone** who holds physical possession of the Strong PUF can freely apply challenges and read out responses
- Very many possible challenges (*ideally exponentially many*)

**PUF**

(= (partly) disordered, unclonable physical system S)

**Responses R_i**

(R_i is a function of the applied challenge C_i and the specific disorder in S)

(C_i, R_i): Challenge-response pairs (CRPs) of the PUF

External Stimuli/Challenges C_i
Physical Unclonable Functions (PUFs)

**Strong PUFs:**

- Challenge-response interface is publicly accessible
  - *Everyone* who holds physical possession of the Strong PUF can freely apply challenges and read out responses
- Very many possible challenges (*ideally exponentially many*)
- Complex: No numerical prediction of unknown responses

\[
(\text{External Stimuli/Challenges } C_i, \text{Responses } R_i) \text{: Challenge-response pairs (CRPs) of the PUF}
\]

\[
\text{PUF} = (\text{partly) disordered, unclonable physical system } S)
\]
The most widespread electrical Strong PUF: 

**Arbiter PUFs** (1)

(1) B. Gassend et al, CCS 2002

(2) D. Lim, MIT, 2004, and elsewhere
The most widespread electrical Strong PUF: Arbiter PUFs \(^{(1)}\)

\(1\) B. Gassend et al, CCS 2002 \hspace{1cm} \(2\) D. Lim, MIT, 2004, and elsewhere

---

**Diagram:**

- **Arbiter PUF** circuit diagram showing inputs \(X[0], X[1], \ldots, X[126], X[127]\) and output \(Y\).
- The diagram includes logic gates and a latch symbolizing the output logic operation.
The most widespread electrical Strong PUF: Arbiter PUFs \(^{(1)}\)

\(^{(1)}\) B. Gassend et al, CCS 2002  \(^{(2)}\) D. Lim, MIT, 2004, and elsewhere
The most widespread electrical Strong PUF: Arbiter PUFs\textsuperscript{(1)}

\[
X[1] = 0
\]

\textsuperscript{(1)} B. Gassend et al, CCS 2002 \quad \textsuperscript{(2)} D. Lim, MIT, 2004, and elsewhere
The most widespread electrical Strong PUF: Arbiter PUFs $^{(1)}$

X[1] = 0
X[126] = 1

(1) B. Gassend et al, CCS 2002  (2) D. Lim, MIT, 2004, and elsewhere
The most widespread electrical Strong PUF: Arbiter PUFs \(^{(1)}\)

- **But: Linear!**

\[ X[1] = 0 \quad X[126] = 1 \]

- B. Gassend et al, CCS 2002
- D. Lim, MIT, 2004, and elsewhere
The most widespread electrical Strong PUF: Arbiter PUFs \(^{(1)}\)

- **But:** *Linear!*

- Adversaries can derive the internal delays via machine learning techniques (in so-called „*modeling attacks*“) \(^{(2)}\)
  - **Complexity of attacks:** *Linear* no. of CRPs, *quadratic* runtime

\[ X[1] = 0 \quad X[126] = 1 \]

---

\(^{(1)}\) B. Gassend et al, CCS 2002 \quad \(^{(2)}\) D. Lim, MIT, 2004, and elsewhere
Enhanced Designs of the Arbiter PUF Family

(1) Rührmair et al., CCS 2010. (2) Rührmair et al., T-IFS 2013
Enhanced Designs of the Arbiter PUF Family

**k-XOR Arbiter PUF**
G. Suh et al, DAC 2007

(1) Rührmair et al., CCS 2010. (2) Rührmair et al., T-IFS 2013
Enhanced Designs of the Arbiter PUF Family

**k-XOR Arbiter PUF**
G. Suh et al, DAC 2007

**Lightweight PUF (LW PUF)**
M. Majzoobi et al, ICCAD 2008

---

(1) Rührmair et al., CCS 2010. (2) Rührmair et al., T-IFS 2013
Enhanced Designs of the Arbiter PUF Family

k-XOR Arbiter PUF
G. Suh et al, DAC 2007

Lightweight PUF (LW PUF)
M. Majzoobi et al, ICCAD 2008

- Both XOR-based... (Also output network of LW PUF is XOR-based)

(1) Rührmair et al., CCS 2010.  (2) Rührmair et al., T-IFS 2013
Enhanced Designs of the Arbiter PUF Family

k-XOR Arbiter PUF
G. Suh et al, DAC 2007

Lightweight PUF (LW PUF)
M. Majzoobi et al, ICCAD 2008

• Both XOR-based... (Also output network of LW PUF is XOR-based)
• „Most secure“ members of the Arbiter PUF family! (1,2)
  — All others have been broken (1,2)

(1) Rührmair et al., CCS 2010. (2) Rührmair et al., T-IFS 2013
Enhanced Designs of the Arbiter PUF Family

**k-XOR Arbiter PUF**
G. Suh et al, DAC 2007

**Lightweight PUF (LW PUF)**
M. Majzoobi et al, ICCAD 2008

---

(1) Rührmair et al., CCS 2010.  (2) Rührmair et al., T-IFS 2013
Enhanced Designs of the Arbiter PUF Family

k-XOR Arbiter PUF
G. Suh et al, DAC 2007

Lightweight PUF (LW PUF)
M. Majzoobi et al, ICCAD 2008

• How secure?
  – Modeling attacks have exponential complexity (in no. of XORs) \(^{(1,2)}\)
  – Downside: Also exponentially bad stability (in no. of XORs)...
  – 8 XORs explicitly recommended as secure in literature \(^{(1,2)}\)

\(^{(1)}\) Rührmair et al., CCS 2010. \(^{(2)}\) Rührmair et al., T-IFS 2013
Outline

1. Background: Arbiter PUF Variants, Pure Modeling Attacks, and Their Limitations

2. **Power and Timing Side Channels on XOR Arbiter PUFs and LW PUFs**

3. Combining Side Channels with Modeling Attacks

4. Our Results

5. Summary
Basic Idea of the Side Channels
Basic Idea of the Side Channels

Ideal, but difficult!!!
Basic Idea of the Side Channels

Since direct access is difficult, we measure a global parameter instead:

The cumulative number of ones (and zeros) in the individual outputs of the parallel Arbiter PUFs!

For example: In an 8 XOR Arbiter PUF, 5 individual outputs are one, 3 are zero (but unknown which are 0/1)
Basic Idea of the Side Channels

Since direct access is difficult, we measure a global parameter instead:

The cumulative number of ones (and zeros) in the individual outputs of the parallel Arbiter PUFs!

For example: In an 8 XOR Arbiter PUF, 5 individual outputs are one, 3 are zero (but unknown which are 0/1)

- Either by power analysis or by timing analysis...
Power Side Channel (PSC)
Power Side Channel (PSC)

- **Basic idea:** Transition in the latches *from zero to one* draws power...
Power Side Channel (PSC)

• **Basic idea:** Transition in the latches from zero to one draws power...
• More power consumption *means* more transitions *means* more ones!
  – Provides **cumulative** number of ones/zeros in single Arb PUF outputs
**Power Side Channel (PSC)**

- **Basic idea:** Transition in the latches from zero to one draws power...

- More power consumption means more transitions means more ones!
  - Provides cumulative number of ones/zeros in single Arb PUF outputs

![Diagram of XOR gate with latches and power consumption graph](image-url)

Measure „global“ power consumption
Power Side Channel (PSC) and Noise

- The PUF embedding device has other parts that draw power
- Can we isolate the effect of the latches?
  - Develop **specialized statistical technique** in the paper:
    Repeat measurements, analyze probability distribution

Power trace of the whole design
Timing Side-Channel (TSC)

(1) M. Majzoobi et al., T-IFS 2011
Timing Side-Channel (TSC)

TSC extraction schematic \(^{(1)}\)

(1) M. Majzoobi et al., T-IFS 2011
Timing Side-Channel (TSC)

- Sweep clock to approximate the timing of XOR inputs
- Toggle will be created by changes from individual Arbiter PUFs
- Estimate the number of flipping XOR inputs with a good probability

TSC extraction schematic (1)

(1) M. Majzoobi et al., T-IFS 2011
Overview: Power and Timing Side Channels
Overview: Power and Timing Side Channels

- Both provide the **cumulative** number of zeros and ones in the $k$ individual Arbiter PUF outputs within a $k$-XOR Arbiter PUF or LW PUF
Overview: Power and Timing Side Channels

- Both provide the **cumulative** number of zeros and ones in the $k$ individual Arbiter PUF outputs within a $k$-XOR Arbiter PUF or LW PUF

- **Non-invasive, non-destructive, inexpensive**
Overview: Power and Timing Side Channels

- Both provide the **cumulative** number of zeros and ones in the $k$ individual Arbiter PUF outputs within a $k$-XOR Arbiter PUF or LW PUF

- **Non-invasive, non-destructive, inexpensive**

- **Timing SC**: Requires only an FPGA board, measurement of one CRP and side channel info takes about 1ms.
Overview: Power and Timing Side Channels

• Both provide the cumulative number of zeros and ones in the $k$ individual Arbiter PUF outputs within a $k$-XOR Arbiter PUF or LW PUF

• Non-invasive, non-destructive, inexpensive

• **Timing SC:** Requires only an FPGA board, measurement of one CRP and side channel info takes about 1ms.

• **Power SC:** Requires only an FPGA board and an oscilloscope, measurement of one CRP and side channel info takes about 1ms.
Outline

1. Background: Arbiter PUF Variants, Pure Modeling Attacks, and Their Limitations
2. Power and Timing Side Channels on XOR Arbiter PUFs and LW PUFs
3. Combining Side Channels with Modeling Attacks
4. Our Results
5. Summary
Are the Side Channels Useful At All?
Are the Side Channels Useful At All?

- At first sight, the **cumulative** number of zeros/ones appears **useless**...
  - No straightforward relevance for the underlying machine learning (ML) problem...
Are the Side Channels Useful At All?

- At first sight, the **cumulative** number of zeros/ones appears **useless**...
  - No straightforward relevance for the underlying machine learning (ML) problem...

- *It requires a „tailormade“ ML approach to exploit this info*
  - Quite non-trivial...
  - One of the main contributions of the paper
  - Summary over next two slides
  - Details: See paper
Machine Learning and Side Channels

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

• General model for i-th Arbiter PUF within k-XOR Arbiter PUF \(^{(1,2)}\):
Machine Learning and Side Channels

- General model for i-th Arbiter PUF within k-XOR Arbiter PUF $^{(1,2)}$:

\[ \hat{R}_i = \theta(\overline{w}_i^T \varphi_i) \]

binary response of ArbPUF
Heavyside step function
Delay difference parameter for all stages
challenge parity vector

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

- General model for i-th Arbiter PUF within k-XOR Arbiter PUF \(^{(1,2)}\):  

\[
\hat{R}_i = \theta(\vec{w}_i^T \varphi_i)
\]

- Model the cumulative number of ones as:

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

- General model for i-th Arbiter PUF within k-XOR Arbiter PUF \(^{(1,2)}\):

\[
\hat{R}_i = \theta(\mathbf{w}_i^T \varphi_i)
\]

- Model the cumulative number of ones as:

\[
\hat{n} = \sum_i \hat{R}_i = \sum_i \theta(\mathbf{w}_i^T \varphi_i)
\]

- **Optimize** PUF-model \( \mathbf{w} \) and minimize prediction error \( l \):

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

• General model for i-th Arbiter PUF within k-XOR Arbiter PUF \(^{(1,2)}\):

\[ \hat{R}_i = \theta(\vec{w}_i^T \varphi_i) \]

- binary response of ArbPUF
- Heaviside step function
- Delay difference parameter for all stages
- challenge parity vector

• Model the cumulative number of ones as:

\[ \hat{n} = \sum_i \hat{R}_i = \sum_i \theta(\vec{w}_i^T \varphi_i) \]

• **Optimize** PUF-model \( \vec{w} \) and minimize prediction error \( l \):

\[ l(\vec{w}, \text{CRPs}) = \sum_{(C,n) \in \text{CRPs}} (\hat{n}(\vec{w}) - n)^2 \]

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

• Use the following gradient in the optimization of $w$:

$$
\nabla_{\vec{w}_i} l = \sum_{(C,n) \in CRPs} 2(\hat{n} - n) \sigma(\vec{w}_i^T \varphi_i)(1 - \sigma(\vec{w}_i^T \varphi_i)) \varphi_i
$$

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

- Use the following gradient in the optimization of $w$:

$$\nabla_{\vec{w}_i} l = \sum_{(C,n) \in CRPs} 2(\hat{n} - n) \sigma (\vec{w}_i^T \varphi_i)(1 - \sigma (\vec{w}_i^T \varphi_i)) \varphi_i$$

- In each summand, only terms with index „$i$“ appear...

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

- Use the following gradient in the optimization of $w$:

$$
\nabla_{\tilde{w}_i} l = \sum_{(C,n) \in \text{CRPs}} 2(\hat{n} - n) \sigma(\tilde{w}_i^T \varphi_i)(1 - \sigma(\tilde{w}_i^T \varphi_i)) \varphi_i
$$

- In each summand, only terms with index „$i$“ appear...

- Contrary to case $w/o$ side channels $(1,2)$:

$$
\nabla_{\tilde{w}_i} l = \sum_{(C,n) \in \text{CRPs}} 2(\hat{r} - r) \varphi_i \prod_{j \neq i} \tilde{w}_j^T \varphi_j
$$

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Machine Learning and Side Channels

• Use the following gradient in the optimization of $\mathbf{w}$:

$$
\nabla_{\mathbf{w}_i} l = \sum_{(C,n) \in \text{CRPs}} 2(n - \hat{n}) \sigma (\mathbf{w}_i^T \varphi_i) \left(1 - \sigma (\mathbf{w}_i^T \varphi_i)\right) \varphi_i
$$

• In each summand, only terms with index „$i$“ appear…

• Contrary to case w/o side channels $^{(1,2)}$:

$$
\nabla_{\mathbf{w}_i} l = \sum_{(C,n) \in \text{CRPs}} 2(\hat{r} - r) \varphi_i \prod_{j \neq i} \mathbf{w}_j^T \varphi_j
$$

• This leads to a strong (exponential!) efficiency improvement

(1) Rührmair et al., CCS 2010  (2) Rührmair et al., T-IFS 2013
Outline

1. Background: Arbiter PUF Variants, Pure Modeling Attacks, and Their Limitations
2. Power and Timing Side Channels on XOR Arbiter PUFs and LW PUFs
3. Combining Side Channels with Modeling Attacks
4. Our Results
5. Summary
Attack Results on Silicon CRP Data (from FPGAs)
## Attack Results on Silicon CRP Data (from FPGAs)

### Timing SC:

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.5%</td>
<td>2 min</td>
<td>98.5%</td>
<td>1 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>97.5%</td>
<td>12 min</td>
<td>98.2%</td>
<td>9 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>103</td>
<td>97.7%</td>
<td>1:35 hrs</td>
<td>97.8%</td>
<td>1:00 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>205</td>
<td>97.4%</td>
<td>16:50 hrs</td>
<td>97.5%</td>
<td>3:30 hrs</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.1%</td>
<td>16.5 min</td>
<td>98.5%</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.4%</td>
<td>38.5 min</td>
<td>97.9%</td>
<td>24.1 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>154.5</td>
<td>97.1%</td>
<td>3.8 hrs</td>
<td>97.3%</td>
<td>1.75 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>308</td>
<td>96.92%</td>
<td>56.25 hrs</td>
<td>97.11%</td>
<td>9.55 hrs</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>37 min</td>
<td>98%</td>
<td>7 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2 hrs</td>
<td>97.5%</td>
<td>51.7 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>206</td>
<td>97.3%</td>
<td>15.1 hrs</td>
<td>96.9%</td>
<td>4.8 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>410</td>
<td>96.5%</td>
<td>102 hrs</td>
<td>96.7%</td>
<td>20.2 hrs</td>
</tr>
</tbody>
</table>
### Attack Results on Silicon CRP Data (from FPGAs)

#### Timing SC:

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.5%</td>
<td>2 min</td>
<td>98.5%</td>
<td>1 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>97.5%</td>
<td>12 min</td>
<td>98.2%</td>
<td>9 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>103</td>
<td>97.7%</td>
<td>1:35 hrs</td>
<td>97.8%</td>
<td>1:00 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>205</td>
<td>97.4%</td>
<td>16:30 hrs</td>
<td>97.5%</td>
<td>3:30 hrs</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.1%</td>
<td>16.5 min</td>
<td>98.5%</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.4%</td>
<td>38.5 min</td>
<td>97.9%</td>
<td>24.1 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>154.5</td>
<td>97.1%</td>
<td>3.8 hrs</td>
<td>97.3%</td>
<td>1.75 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>308</td>
<td>96.92%</td>
<td>56.25 hrs</td>
<td>97.11%</td>
<td>9.55 hrs</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>37 min</td>
<td>98%</td>
<td>7 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2 hrs</td>
<td>97.5%</td>
<td>51.7 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>206</td>
<td>97.3%</td>
<td>15.1 hrs</td>
<td>96.9%</td>
<td>4.8 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>410</td>
<td>96.5%</td>
<td>102 hrs</td>
<td>96.7%</td>
<td>20.2 hrs</td>
</tr>
</tbody>
</table>

#### Power SC:

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.1%</td>
<td>3 min</td>
<td>98.4%</td>
<td>1.25 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>98%</td>
<td>13 min</td>
<td>98.1%</td>
<td>9.25 min</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.3%</td>
<td>11 min</td>
<td>98.2%</td>
<td>3.5 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.3%</td>
<td>47 min</td>
<td>97.8%</td>
<td>25 min</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>38 min</td>
<td>98%</td>
<td>6.5 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2:28 hrs</td>
<td>97.5%</td>
<td>46.5 min</td>
</tr>
</tbody>
</table>
Attack Results on Silicon CRP Data (from FPGAs)

- **Timing SC:**

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.5%</td>
<td>2 min</td>
<td>98.5%</td>
<td>1 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>97.5%</td>
<td>12 min</td>
<td>98.2%</td>
<td>9 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>103</td>
<td>97.7%</td>
<td>1:35 hrs</td>
<td>97.8%</td>
<td>1:00 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>205</td>
<td>97.4%</td>
<td>16:50 hrs</td>
<td>97.5%</td>
<td>3:30 hrs</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.1%</td>
<td>16.5 min</td>
<td>98.5%</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.4%</td>
<td>38.5 min</td>
<td>97.9%</td>
<td>24.1 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>154.5</td>
<td>97.1%</td>
<td>3.8 hrs</td>
<td>97.3%</td>
<td>1.75 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>308</td>
<td>96.92%</td>
<td>56.25 hrs</td>
<td>97.11%</td>
<td>9.55 hrs</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>37 min</td>
<td>98%</td>
<td>7 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2 hrs</td>
<td>97.5%</td>
<td>51.7 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>206</td>
<td>97.3%</td>
<td>15.1 hrs</td>
<td>96.9%</td>
<td>4.8 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>410</td>
<td>96.5%</td>
<td>102 hrs</td>
<td>96.7%</td>
<td>20.2 hrs</td>
</tr>
</tbody>
</table>

- **Power SC:**

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.1%</td>
<td>3 min</td>
<td>98.4%</td>
<td>1.25 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>98%</td>
<td>13 min</td>
<td>98.1%</td>
<td>9.25 min</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.3%</td>
<td>11 min</td>
<td>98.2%</td>
<td>3.5 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.3%</td>
<td>47 min</td>
<td>97.8%</td>
<td>25 min</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>38 min</td>
<td>98%</td>
<td>6.5 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2:28 hrs</td>
<td>97.5%</td>
<td>46.5 min</td>
</tr>
</tbody>
</table>

*Stronger noise in the power SC for large bitlengths!*
Attack Results on Silicon CRP Data (from FPGAs)

- Timing SC:

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.5%</td>
<td>2 min</td>
<td>98.5%</td>
<td>1 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>97.5%</td>
<td>12 min</td>
<td>98.2%</td>
<td>9 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>103</td>
<td>97.7%</td>
<td>1:35 hrs</td>
<td>97.8%</td>
<td>1:00 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>205</td>
<td>97.4%</td>
<td>16:50 hrs</td>
<td>97.5%</td>
<td>3:30 hrs</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.1%</td>
<td>16.5 min</td>
<td>98.5%</td>
<td>2 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.4%</td>
<td>38.5 min</td>
<td>97.9%</td>
<td>24.1 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>154.5</td>
<td>97.1%</td>
<td>3.8 hrs</td>
<td>97.3%</td>
<td>1.75 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>308</td>
<td>96.92%</td>
<td>56.25 hrs</td>
<td>97.11%</td>
<td>9.55 hrs</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>37 min</td>
<td>98%</td>
<td>7 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2 hrs</td>
<td>97.5%</td>
<td>51.7 min</td>
</tr>
<tr>
<td></td>
<td>256</td>
<td>206</td>
<td>97.3%</td>
<td>15.1 hrs</td>
<td>96.9%</td>
<td>4.8 hrs</td>
</tr>
<tr>
<td></td>
<td>512</td>
<td>410</td>
<td>96.5%</td>
<td>102 hrs</td>
<td>96.7%</td>
<td>20.2 hrs</td>
</tr>
</tbody>
</table>

- Power SC:

<table>
<thead>
<tr>
<th>No. of XORs</th>
<th>Bit Length</th>
<th>CRPs ($\times 10^3$)</th>
<th>Prediction Rate XOR Arb. PUF</th>
<th>Training Time XOR Arb. PUF</th>
<th>Predict. Rate LW PUF</th>
<th>Training Time LW PUF</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>64</td>
<td>26</td>
<td>98.1%</td>
<td>3 min</td>
<td>98.4%</td>
<td>1.25 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>51.6</td>
<td>98%</td>
<td>13 min</td>
<td>98.1%</td>
<td>9.25 min</td>
</tr>
<tr>
<td>12</td>
<td>64</td>
<td>39</td>
<td>98.3%</td>
<td>11 min</td>
<td>98.2%</td>
<td>3.5 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>77.4</td>
<td>97.3%</td>
<td>47 min</td>
<td>97.8%</td>
<td>25 min</td>
</tr>
<tr>
<td>16</td>
<td>64</td>
<td>52</td>
<td>98%</td>
<td>38 min</td>
<td>98%</td>
<td>6.5 min</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>103.2</td>
<td>97.5%</td>
<td>2:28 hrs</td>
<td>97.5%</td>
<td>46.5 min</td>
</tr>
</tbody>
</table>

Stronger noise in the power SC for large bitlengths!

Recall: 8 XORs had explicitly been suggested as secure...
Asymptotic Performance Analysis on Simulated CRP Data
Asymptotic Performance Analysis on Simulated CRP Data

![Graph showing time vs. number of free parameters for different bit lengths and XORs]
Asymptotic Performance Analysis on Simulated CRP Data

- Only \textit{cubic} runtime and \textit{linear} no. of CRPs required!
  - \textbf{Compare:} Quadratic runtime complexity and \textit{linear} no. of CRPs of pure modeling attacks on standard Arb PUFs (i.e., without XORs)
Outline

1. Background: Arbiter PUF Variants, Pure Modeling Attacks, and Their Limitations
2. Power and Timing Side Channels on XOR Arbiter PUFs and LW PUFs
3. Combining Side Channels with Modeling Attacks
4. Our Results
5. Summary
Summary

Summary

• New attack strategy on XOR-based Arbiter PUFs: Combined modeling and side channel attacks
  – Non-invasive, non-destructive, inexpensive, very efficient...

Summary

• New attack strategy on XOR-based Arbiter PUFs:
  Combined modeling and side channel attacks
  – Non-invasive, non-destructive, inexpensive, very efficient...

• Presented side channels are:
  – The first power and timing side channels on PUFs
  – The first direct side channels on Strong PUFs
    that can notably increase attack performance (compare \(^{(1,2,3)}\))

Summary

- New attack strategy on XOR-based Arbiter PUFs: Combined modeling and side channel attacks
  - **Non-invasive, non-destructive**, inexpensive, very efficient...
- Presented side channels are:
  - The first **power** and **timing** side channels on PUFs
  - The first **direct** side channels on Strong PUFs that can notably **increase** attack performance (compare\(^{(1,2,3)}\))
- Enables low-degree polynomial attacks for LW PUFs and XOR Arbiter PUFs
  - These were considered the most secure members of the Arbiter PUF family prior to our attacks
  - Only **linear** no. of CRPs and **cubic** runtime required

---

Summary
Summary

• *As long as no countermeasures are developed and put in place,* no existing member of the Arbiter PUF remains secure
  – Some countermeasures are sketched in our paper, but this topic is mainly *ongoing work*
Summary

• As long as no countermeasures are *developed and put in place*, no existing member of the Arbiter PUF remains secure
  – Some countermeasures are sketched in our paper, but this topic is mainly *ongoing work*

• *Arms race between codemakers and codebreakers on Strong PUFs continues!*
Summary

• As long as no countermeasures are developed and put in place, no existing member of the Arbiter PUF remains secure
  – Some countermeasures are sketched in our paper, but this topic is mainly ongoing work

• Arms race between codemakers and codebreakers on Strong PUFs continues!

• Watch this space, there’s more to come! 😊