Appendix A. LLVM optimization passes

Figure A.2: Zoomed CPA on keeloq_tb041 generated implementations
Figure A.2: Zoomed CPA on keeloq_tb041 generated implementations
Figure A.3: Resistance against CPA of keeloq_tb041 generated implementations
Figure A.3: Resistance against CPA of keeloq_tb041 generated implementations
Figure A.4: Resistance against zoomed CPA of keeloq_tb041 generated implementations
Figure A.4: Resistance against zoomed CPA of keeloq_tb041 generated implementations
Appendix A. LLVM optimization passes

This annex provides the results of DCPA applied to the implementation groups generated in Chapter 5. The figures include the results of DCPA and the resistance evaluation against DCPA of the different implementations.
Figure A.5: DCPA on keeloq_tb041 generated implementations
Figure A.5: DCPA on keeloq_tb041 generated implementations
Figure A.6: Resistance against DCPA of keeloq_tb041 generated implementations
Appendix A. LLVM optimization passes

Figure A.6: Resistance against DCPA of keeloq_tb041 generated implementations
Appendix B

KeeLoq implementations

This appendix shows the KeeLoq source code inspired in tb041 application note [93] used in Chapters 5 and 6.

Listing B.1: tb041 implementation

```c
for (i = 16; i > 0; i--) {
    //NLF
    if (bit(data.i16.high,15)) {
        aux = KeeLoq_NLF_H;
    } else {
        aux = KeeLoq_NLF_L;
    }

    if (bit(data.i16.high,10)) {
        asm volatile("swpb %1"
                     :="r"(aux)
                       :"0"(aux)
                       :"cc");
    }

    if (bit(data.i16.high,4)) {
        aux >>= 4;
    }

    if (bit(data.i16.low,9)) {
        aux >>= 2;
    }

    if (bit(data.i16.low,1)) {
        aux >>= 1;
    }

    //linear operation
    aux ^= data.i16.high ^ data.i16.low ^ key;
    data.i32 >>= 1;
    if (aux & 0x0001) {
        data.i16.high |= 0x8000;
    }
}
```

147
Appendix B. KeeLoq implementations

Listing B.2: nlftb041 implementation

```c
for (i = 16; i > 0; i--) {
    nlf_mask = 1;
    //NLF
    if (data.i16.low & 0x0002) {
        nlf_mask *= 2;
    }
    if (data.i16.low & 0x0200) {
        nlf_mask *= 4;
    }
    if (data.i16.high & 0x0010) {
        nlf_mask *= 16;
    }
    if (data.i16.high & 0x0400) {
        nlf_mask *= 256;
    }
    aux = 0;
    if (data.i16.high & 0x8000) {
        if (KeeLoq_NLF_H & nlf_mask) {
            aux = 0x0001;
        }
    } else {
        if (KeeLoq_NLF_L & nlf_mask) {
            aux = 0x0001;
        }
    }
    //linear operation
    aux ^= data.i16.high ^ data.i16.low ^ key;
    data.i32 >>= 1;
    if (aux & 0x0001) {
        data.i16.high |= 0x8000;
    }
    key >>= 1;
}
```

Listing B.3: gentb041 implementation

```c
size_t nlf_lut[] =
    {0,1,1,1, 0,1,0,0, 0,0,1,0, 1,1,1,0,
     0,0,1,1, 1,0,1,0, 0,1,0,1, 1,1,0,0};
size_t nlf_mask;
...
for (i = 16; i > 0; i--) {
    nlf_mask = 0;
    //NLF
    if (data.i16.low & 0x0002) {
        nlf_mask |= 1;
    }
    if (data.i16.low & 0x0200) {
        nlf_mask |= 2;
    }
    if (data.i16.high & 0x0010) {
        nlf_mask |= 4;
    }
    if (data.i16.high & 0x0400) {
        nlf_mask |= 8;
    }
    if (data.i16.high & 0x8000) {
        nlf_mask |= 16;
    }
    aux = nlf_lut[nlf_mask];
    //linear operation
    aux ^= data.i16.high ^ data.i16.low ^ key;
    data.i32 >>= 1;
    if (aux & 0x0001) {
        data.i16.high |= 0x8000;
    }
    key >>= 1;
}
Bibliography


Acronyms

AES  Advanced Encryption Standard. 4, 21, 23, 26, 44, 48
ALU  Arithmetic Logic Unit. 64, 67, 112–114, 120
ANF  Algebraic Normal Form. ii, 38, 111–113, 117, 118, 121, 125, 126, 128–130, 133
ASIC Application-Specific Integrated Circuit. 2, 7, 26, 32, 53
AWDDL Asynchronous WDDL. 35
BCDL Balanced Cell-based Dual-rail Logic. 34, 38, 81
BRU Basic Reconfigurable Unit. 77, 78, 83–85, 87, 132, 134
CFG Control Flow Graph. 64, 94, 141
CPA Correlation Power Analysis. vii, 20, 26–28, 53, 60, 63, 68–70, 84, 85, 87, 96, 98–100, 103, 104, 106, 112, 115, 117, 118, 122, 125, 126, 130, 132, 133, 143
DAG Directed Acyclic Graph. 64
DCPA Differential CPA. 70, 84, 98–100, 103, 104, 132, 133, 152
DES Data Encryption Standard. 21–23, 46, 47, 69
DFA Differential Frequency Analysis. 51
DFG Data Flow Graph. 49
DPA Differential Power Analysis. 20, 22, 24–26, 28, 29, 33, 34, 37, 40, 42, 46, 51, 52, 54, 55, 68, 69, 89, 106
DPL Dual-rail Precharge Logic. 32–35, 38, 42, 44, 81
DRSL Dual-Rail Random Switching Logic. 32, 37, 38
DSCA Differential Side-Channel Attack. 4, 5, 20, 24, 26–29, 44, 46, 48, 52, 53, 55, 60, 92, 134
DSP Digital Signal Processor. 2
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDPA</td>
<td>Euclidean DPA. 28, 70</td>
</tr>
<tr>
<td>EMA</td>
<td>Electro-Magnetic Analysis. 5, 19, 35, 51</td>
</tr>
<tr>
<td>EPDU</td>
<td>Evaluation-Precharge Detection Unit. 37</td>
</tr>
<tr>
<td>EPE</td>
<td>Early Propagation Effect. 34–38</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform. 51</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Array. 2, 7, 32, 34, 35, 38, 42, 43, 53, 77, 81, 89, 134</td>
</tr>
<tr>
<td>FSR</td>
<td>Feedback Shift Register. i–iv, 2, 3, 9–13, 17, 19, 51–53, 60, 65, 66, 69, 73, 74, 83, 92, 96, 111, 113, 120, 131, 132, 134, 135</td>
</tr>
<tr>
<td>GF</td>
<td>Galois Field. 113</td>
</tr>
<tr>
<td>HD</td>
<td>Hamming Distance. 63, 69, 87</td>
</tr>
<tr>
<td>HO-DPA</td>
<td>Higher Order DPA. 47, 50</td>
</tr>
<tr>
<td>HW</td>
<td>Hamming Weight. 20, 26, 63, 68, 112, 115, 128</td>
</tr>
<tr>
<td>IPA</td>
<td>Inferential Power Analysis. 22</td>
</tr>
<tr>
<td>IR</td>
<td>Intermediate Representation. 63, 64, 92</td>
</tr>
<tr>
<td>IV</td>
<td>initial value. 11, 52, 53</td>
</tr>
<tr>
<td>KDE</td>
<td>kernel density estimator. 28</td>
</tr>
<tr>
<td>kNN</td>
<td>k-nearest neighbours. 28</td>
</tr>
<tr>
<td>LFSR</td>
<td>Linear Feedback Shift Register. 10, 52–54, 111, 113</td>
</tr>
<tr>
<td>LLVM</td>
<td>Low-Level Virtual Machine. 49, 63–65, 79, 80, 84, 91–93, 106, 109, 128, 134</td>
</tr>
<tr>
<td>LSB</td>
<td>Least Significant Bit. 87, 112, 120</td>
</tr>
<tr>
<td>LUT</td>
<td>Look-Up Tables. 13, 34, 35, 38, 46, 48, 77, 94, 111, 112, 114, 115, 118, 122, 125, 126, 128, 130, 133</td>
</tr>
<tr>
<td>MCU</td>
<td>Microcontroller Unit. 65</td>
</tr>
<tr>
<td>MDCA</td>
<td>Multiple-Differential Side-Channel Collision Attack. 23</td>
</tr>
<tr>
<td>MDPL</td>
<td>Masked Dual-rail Precharge Logic. 32, 36–38</td>
</tr>
<tr>
<td>MIA</td>
<td>Mutual Information Analysis. 20, 27–29, 39, 43, 54, 57, 68</td>
</tr>
<tr>
<td>ML</td>
<td>maximum likelihood. 28</td>
</tr>
<tr>
<td>Acronyms</td>
<td>Definitions</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>MSB</strong></td>
<td>Most Significant Bit. 87, 112, 115, 120</td>
</tr>
<tr>
<td><strong>NLF</strong></td>
<td>Non-Linear Function. 111, 112, 114, 117, 128, 129</td>
</tr>
<tr>
<td><strong>NLFSR</strong></td>
<td>Non-Linear Feedback Shift Register. 10, 53, 111–113, 130</td>
</tr>
<tr>
<td><strong>PA-DPL</strong></td>
<td>Precharge-Absorbed Dual-rail Precharge Logic. 35, 38, 81</td>
</tr>
<tr>
<td><strong>PAA</strong></td>
<td>Power Analysis Attack. 19, 59</td>
</tr>
<tr>
<td><strong>pdf</strong></td>
<td>probability density function. 27, 37</td>
</tr>
<tr>
<td><strong>PUF</strong></td>
<td>Physically Unclonable Function. 43</td>
</tr>
<tr>
<td><strong>QDI</strong></td>
<td>Quasi-Delay Insensitive. 40, 41</td>
</tr>
<tr>
<td><strong>RFID</strong></td>
<td>Radio Frequency Identification Device. 5, 50</td>
</tr>
<tr>
<td><strong>RFU</strong></td>
<td>Reconfigurable Functional Unit. 73–75, 78</td>
</tr>
<tr>
<td><strong>RISC</strong></td>
<td>Reduced Instruction Set Computing. 48, 89</td>
</tr>
<tr>
<td><strong>RKE</strong></td>
<td>Remote Keyless Entry. i, 3, 9, 10, 53, 61, 130, 132</td>
</tr>
<tr>
<td><strong>RPI</strong></td>
<td>Random Process Interrupt. 44, 45</td>
</tr>
<tr>
<td><strong>RSL</strong></td>
<td>Random Switching Logic. 32, 37</td>
</tr>
<tr>
<td><strong>SABL</strong></td>
<td>Sense Amplifier Based Logic. 32–34, 53</td>
</tr>
<tr>
<td><strong>SCARE</strong></td>
<td>SCA Reverse Engineering. 22, 53, 100, 123, 134</td>
</tr>
<tr>
<td><strong>SIMD</strong></td>
<td>Single Instruction Multiple Data. 47, 48, 74, 78, 84, 113, 130, 133</td>
</tr>
<tr>
<td><strong>SNR</strong></td>
<td>Signal-Noise Ratio. 19, 39, 51, 57, 84</td>
</tr>
<tr>
<td><strong>SoC</strong></td>
<td>System on Chip. 1, 2</td>
</tr>
<tr>
<td><strong>SORU</strong></td>
<td>Stream Oriented Reconfigurable Unit. 73</td>
</tr>
<tr>
<td><strong>SPA</strong></td>
<td>Simple Power Analysis. 19–24, 44, 53–55, 63, 67, 92, 100, 101, 130</td>
</tr>
<tr>
<td><strong>TDPL</strong></td>
<td>Three-phase Dual-rail Precharge Logic. 32, 34</td>
</tr>
<tr>
<td><strong>VLIW</strong></td>
<td>Very Long Instruction Word. 43, 89</td>
</tr>
<tr>
<td><strong>VLSU</strong></td>
<td>Vector Load and Store Unit. 73, 78, 81</td>
</tr>
</tbody>
</table>
**Acronyms**

**VLU**  Vector Load Unit. 81

**VSU**  Vector Store Unit. 81

**WDDL**  Wave Dynamic Differential Logic. 32–36, 38

**WDE**  wavelet density estimator. 28

**WSN**  Wireless Sensor Networks. 10, 14, 65, 66, 131, 132