[65] Adrian Tang, Simha Sethumadhavan, and Salvatore Stolfo. 2017. CLKSCREW: Exposing the Perils of Security-Oblivious Energy Management. In *26th USENIX Security Symposium (USENIX Security 17)*. USENIX Association, Vancouver, BC, 1057–1074.
[66] C Josh Thomas and Nathan Keltner. 2014. Reflections on Trusting TrustZone. In *Black Hat conference*. Blackhat, Las Vegas, NV, USA, 33.
[67] Michael Tunstall, Debdeep Mukhopadhyay, and Subidh Ali. 2011. Differential Fault Analysis of the Advanced Encryption Standard Using a Single Fault. In *Information Security Theory and Practice. Security and Privacy of Mobile Devices in Wireless Communication*. Springer, Berlin, Heidelberg, 224–233.
[68] Qian Wang, An Wang, Gang Qu, and Guoshuang Zhang. 2017. New Methods of Template Attack Based on Fault Sensitivity Analysis. *IEEE Transactions on Multi-Scale Computing Systems* 3, 2 (April 2017), 113–123. https://doi.org/10.1109/TMSCS.2016.2643638
[69] Qian Wang, An Wang, Liji Wu, Gang Qu, and Guoshuang Zhang. 2015. Template attack on masking AES based on fault sensitivity analysis. In *2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, Vol. 00. Blackhat, Washington, DC, USA, 96–99. https://doi.org/10.1109/HST.2015.7140245
[70] Yuval Yarom and Katrina Falkner. 2014. FLUSH+RELOAD: A High Resolution, Low Noise, L3 Cache Side-Channel Attack. In *23rd USENIX Security Symposium (USENIX Security 14)*. USENIX Association, San Diego, CA, 719–732. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/yarom

A FAULTS INDUCED BY A HIGH VOLTAGE

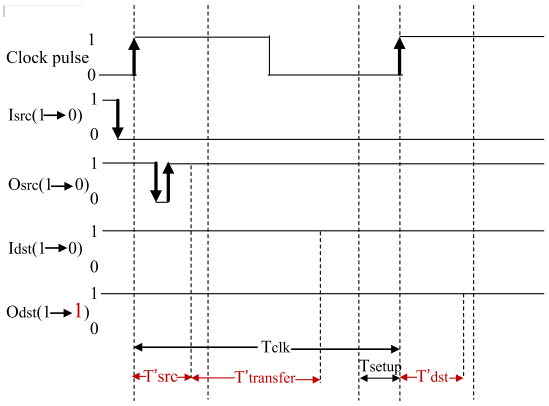

Figure 11: The bit-flip will be inserted if E_{src} becomes unstable because of a high voltage.

B THE IMPLEMENTATION OF FUNCTION ENDIANINVERSION IN ALGORITHM 1

Algorithm 2 The endian reversal algorithm

Input: The variable that needs to be endian reversed, V
Output: The reversed bytes sequence, S

```
function EndianInversion(V)
    S ← {0}
    for each i ∈ [0, bytelen(V)/4 − 1] do
        S_temp ← V[i ∗ 4] ≪ 24
        S_temp ← (V[i ∗ 4 + 1] ≪ 8) | S_temp
        S_temp ← (V[i ∗ 4 + 2] ≫ 8) | S_temp
        S_temp ← (V[i ∗ 4 + 3] ≫ 24) | S_temp
        Index ← bytelen(V) − (i + 1) ∗ 4
        S[Index, Index + 3] ← S_temp;
    end for
    return S
end function
```

C AN EXAMPLE ERROR ON THE PUBLIC MODULUS OF THE `WIDEVINE` TRUSTLET

Original N:
0xc44dc735f6682a261a0b8545a62dd13df4c646a5ede482cef858925baa1811fa
0284766b3d1d2b4d6893df4d9c045efe3e84d8c5d03631b25420f1231d8211e23
22eb7eb524da6c1e8fb4c3ae4a8f5ca13d1e0591f5c64e8e711b3726215cec59ed
0ebc6bb042b917d445288**8791**5fdf764df691d183e16f31ba1ed94c84b476e74b
488463e85551022021763af35a64ddf105c1530ef3fcf7e54233e5d3a4747bbb17
328a63e6e3384ac25ee80054bd566855e2eb59a2fd168d3643e44851acf0d118f
b03c73ebc099b4add59c39367d6c91f498d8d607af2e57cc73e3b5718435a8112
3f080267726a2a9c1cc94b9c6bb6817427b85d8c670f9a53a777511b **Corrupted N_{in}:**
0x...8cb3...
Factors of corrupted N_{in}:
0x11, 0x033b, 0x377, 0x010819f1285c6b307a82beba93d7c496488...
Private key d_m:
0x062cde999954a9ced6840f2b04ae4d4187baa01a5044c0242c70dbe...