Model: Circuit Shield Against Physical Attacks (RV, state)
Type: P/T Net
Origin: Industrial (Tiempo Secure)

Wendelin Serwe and Hubert Garavel
wendelin.serwe@inria.fr

since
MCC 2020

This form is a summary description of the model entitled “Circuit Shield Against Physical Attacks (RV, state)” proposed for the Model Checking Contest @ Petri Nets. Models can be given in several instances parameterized by scaling parameters. Colored nets can be accompanied by one or many equivalent, unfolded P/T nets. Models are given together with property files (possibly, one per model instance) giving a set of properties to be checked on the model.

Description

Physical attacks to an integrated circuit are generally meant to allow the attacker to probe for the sensitive information that is stored in the internal registers/wires of the circuit during its ordinary operation. To protect a circuit against such attacks, the patent [1] proposes to add over the circuit, as a top layer, an additional circuit, called *shield*.

Any physical attack or probing procedures will inevitably tamper with the shield, which constitutes the top layer of the accessible part of the circuit. Hence, proving that the shield is able to flag any tampering attempts amounts to proving that the circuit itself is able to detect a physical attack, stop its normal operation and, take an appropriate countermeasure (e.g., completely deactivate the circuit).

The shield is a serial composition of $N + 1$ *sequencers*, or even a parallel composition of several series of sequencers. The gate-level design for a sequencer can be found in Figures 4–8 of [1]. Each sequencer transmits a first signal, called *request* to its successor; when the last sequencer outputs a request, a second signal, called *acknowledgment* is transmitted through the series of sequencers in the opposite direction. If a sequencer is designed in such a way that any physical modification on the connections for the transmission of the request (respectively, the acknowledgment) blocks the transmission of the acknowledgment (respectively, the request), a physical attack can be detected by the absence of an acknowledgment for a request sent into the shield.

This collection of P/T nets was obtained from the formal description in LNT of the shield given in [2], extended to series of more than two sequencers. With respect to the terminology of [2], the LNT descriptions implement the state-based approach for the RV modelling variant. Each LNT description was translated to LOTOS, and then to an interpreted Petri net using the CADP toolbox. Finally, a P/T net was obtained by stripping out all data-related information (variables, types, assignments, guards, etc.) from the interpreted Petri net, leading to a NUPN (Nested-Unit Petri Net) model translated to PNML using the CÆSAR.BDD tool.

Each instance of the model is parameterized by N, which is equal to the number of sequencers minus one. Each instance is also parameterized by its version V, which specifies how the NUPN has been produced from the LOTOS specification. V is either equal to “*a*” if the NUPN has been generated *after* applying all the structural and data-flow optimizations of the CÆSAR compiler for LOTOS, or to “*b*” if the NUPN has been generated *before* these optimizations.

References

[1] Marc Renaudin, Bertrand Folco, and Boulahia Boubkar. *Circuit intégré protégé.* European Patent Office, Fascicule de Brevet Europeen EP 3 276 656 B1, February 13, 2019.

[2] Radu Mateescu, Wendelin Serwe, Aymane Bouzafour, and Marc Renaudin. *Modeling an Asynchronous Circuit Dedicated to the Protection Against Physical Attacks.* Proceedings of the 4th Workshop on Models for Formal Analysis of Real Systems (MARS 2020). Electronic Proceedings in Theoretical Computer Science, April 2020.

Scaling parameter

Parameter name	**Parameter description**	**Chosen parameter values**
(N, V)	N is the number of sequencers minus one, and V is the version defined above	$\{1, 2, 3, 4, 5, 10, 20, 30, 40, 50, 100\} \times \{a, b\}$

Size of the model

Parameter	Number of places	Number of transitions	Number of arcs	Number of units	HWB code
$N = 1, V = a$	17	22	92	6	2–5–14
$N = 1, V = b$	43	48	144	9	5–5–21
$N = 2, V = a$	31	41	184	10	2–9–25
$N = 2, V = b$	83	93	288	17	6–9–40
$N = 3, V = a$	45	60	276	14	2–13–36
$N = 3, V = b$	123	138	432	25	7–13–59
$N = 4, V = a$	59	79	368	18	2–17–47
$N = 4, V = b$	163	183	576	33	8–17–78
$N = 5, V = a$	73	98	460	22	2–21–58
$N = 5, V = b$	203	228	720	41	9–21–97
$N = 10, V = a$	143	193	920	42	2–41–113
$N = 10, V = b$	403	453	1440	81	14–41–192
$N = 20, V = a$	283	383	1840	82	2–81–223
$N = 20, V = b$	803	903	2880	161	24–81–382
$N = 30, V = a$	423	573	2760	122	2–121–333
$N = 30, V = b$	1203	1353	4320	241	34–121–572
$N = 40, V = a$	563	763	3680	162	2–161–443
$N = 40, V = b$	1603	1803	5760	321	44–161–762
$N = 50, V = a$	703	953	4600	202	2–201–553
$N = 50, V = b$	2003	2253	7200	401	54–201–952
$N = 100, V = a$	1403	1903	9200	402	2–401–1103
$N = 100, V = b$	4003	4503	14400	801	104–401–1902

Structural properties

ordinary — *all arcs have multiplicity one* .... yes

simple free choice — *all transitions sharing a common input place have no other input place* .... no [a]

extended free choice — *all transitions sharing a common input place have the same input places* .... no [b]

state machine — *every transition has exactly one input place and exactly one output place* .... no [c]

marked graph — *every place has exactly one input transition and exactly one output transition* .... no [d]

connected — *there is an undirected path between every two nodes (places or transitions)* .... yes [e]

strongly connected — *there is a directed path between every two nodes (places or transitions)* .... no [f]

source place(s) — *one or more places have no input transitions* .... yes [g]

sink place(s) — *one or more places have no output transitions* .... no [h]

source transition(s) — *one or more transitions have no input places* .... no [i]

sink transitions(s) — *one or more transitions have no output places* .... no [j]

loop-free — *no transition has an input place that is also an output place* .... yes [k]

conservative — *for each transition, the number of input arcs equals the number of output arcs* .... no [l]

subconservative — *for each transition, the number of input arcs equals or exceeds the number of output arcs* .... no [m]

nested units — *places are structured into hierarchically nested sequential units* [n] .... yes

[a] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[b] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[c] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[d] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[e] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[f] from place 1 one cannot reach place 0.
[g] place 0 is a source place.
[h] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[i] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[j] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[k] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[l] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[m] stated by CÆSAR.BDD version 3.3 on all 22 instances (10 values of N × 2 values of V).
[n] the definition of Nested-Unit Petri Nets (NUPN) is available from `http://mcc.lip6.fr/nupn.php`

Behavioural properties

safe — *in every reachable marking, there is no more than one token on a place* . . . . yes [o]
dead place(s) — *one or more places have no token in any reachable marking* . . . . ? [p]
dead transition(s) — *one or more transitions cannot fire from any reachable marking* . . . . ? [q]
deadlock — *there exists a reachable marking from which no transition can be fired* . . . . ? [r]
reversible — *from every reachable marking, there is a transition path going back to the initial marking* . . . . ? [s]
live — *for every transition t, from every reachable marking, one can reach a marking in which t can fire* . . . . ? [t]

Size of the marking graphs

Parameter	Number of reachable markings	Number of transition firings	Max. number of tokens per place	Max. number of tokens per marking
$N = 1, V = a$	171 [u]	?	1	5
$N = 1, V = b$	3596 [v]	?	1	5
$N = 2, V = a$	16021 [w]	?	1	9
$N = 2, V = b$	4.26146e+06 [x]	?	1	9
$N = 3, V = a$	1.5305e+06 [y]	?	1	13
$N = 3, V = b$	$\geq$ 2.4422e+09 [z]	?	1 [aa]	13
$N = 4, V = a$	$\geq$ 1.46336e+08 [ab]	?	1 [ac]	17
$N = 4, V = b$	$\geq$ 7.53004e+11 [ad]	?	1 [ae]	17
$N = 5, V = a$	$\geq$ 7.0032e+09 [af]	?	1 [ag]	21
$N = 5, V = b$	$\geq$ 1.69436e+14 [ah]	?	1 [ai]	21
$N = 10, V = a$	$\geq$ 7.10995e+15 [aj]	?	1 [ak]	41
$N = 10, V = b$	$\geq$ 9.4182e+24 [al]	?	1 [am]	41
$N = 20, V = a$	$\geq$ 7.85377e+11 [an]	?	1 [ao]	81
$N = 20, V = b$	?	?	1 [ap]	81
$N = 30, V = a$	$\geq$ 7.85377e+11 [aq]	?	1 [ar]	121
$N = 30, V = b$	?	?	1 [as]	121
$N = 40, V = a$	$\geq$ 7.43409e+13 [at]	?	1 [au]	161
$N = 40, V = b$	?	?	1 [av]	161
$N = 50, V = a$	$\geq$ 7.43409e+13 [aw]	?	1 [ax]	201
$N = 50, V = b$	$\geq$ 1.02304e+112 [ay]	?	1 [az]	201
$N = 100, V = a$	$\geq$ 7.85377e+11 [ba]	?	1 [bb]	401
$N = 100, V = b$	$\geq$ 4.20627e+219 [bc]	?	1 [bd]	401

[o] safe by construction – stated by the CÆSAR compiler.
[p] stated by CÆSAR.BDD version 3.3 to be false on 10 instance(s) out of 22, and unknown on the remaining 12 instance(s).
[q] stated by CÆSAR.BDD version 3.3 to be false on 10 instance(s) out of 22, and unknown on the remaining 12 instance(s).
[r] stated by CÆSAR.BDD version 3.3 to be true on 5 instance(s) out of 22, and unknown on the remaining 17 instance(s).
[s] stated by CÆSAR.BDD version 3.3 to be false on 5 instance(s) out of 22, and unknown on the remaining 17 instance(s).
[t] stated by CÆSAR.BDD version 3.3 to be false on 5 instance(s) out of 22, and unknown on the remaining 17 instance(s).
[u] stated by CÆSAR.BDD version 3.3.
[v] stated by CÆSAR.BDD version 3.3.
[w] stated by CÆSAR.BDD version 3.3.
[x] stated by CÆSAR.BDD version 3.3.
[y] stated by CÆSAR.BDD version 3.3.
[z] stated by CÆSAR.BDD version 3.3.
[aa] stated by the CÆSAR compiler.
[ab] stated by CÆSAR.BDD version 3.3.
[ac] stated by the CÆSAR compiler.
[ad] stated by CÆSAR.BDD version 3.3.
[ae] stated by the CÆSAR compiler.
[af] stated by CÆSAR.BDD version 3.3.
[ag] stated by the CÆSAR compiler.

[ah] stated by CÆSAR.BDD version 3.3.
[ai] stated by the CÆSAR compiler.
[aj] stated by CÆSAR.BDD version 3.3.
[ak] stated by the CÆSAR compiler.
[al] stated by CÆSAR.BDD version 3.3.
[am] stated by the CÆSAR compiler.
[an] stated by CÆSAR.BDD version 3.3.
[ao] stated by the CÆSAR compiler.
[ap] stated by the CÆSAR compiler.
[aq] stated by CÆSAR.BDD version 3.3.
[ar] stated by the CÆSAR compiler.
[as] stated by the CÆSAR compiler.
[at] stated by CÆSAR.BDD version 3.3.
[au] stated by the CÆSAR compiler.
[av] stated by the CÆSAR compiler.
[aw] stated by CÆSAR.BDD version 3.3.
[ax] stated by the CÆSAR compiler.
[ay] stated by CÆSAR.BDD version 3.3.
[az] stated by the CÆSAR compiler.
[ba] stated by CÆSAR.BDD version 3.3.
[bb] stated by the CÆSAR compiler.
[bc] stated by CÆSAR.BDD version 3.3.
[bd] stated by the CÆSAR compiler.