The Simeck Family of Lightweight Block Ciphers

Gangqiang Yang, Bo Zhu, Valentin Suder, Mark D. Aagaard, and Guang Gong

Electrical and Computer Engineering,
University of Waterloo

Sept 15, 2015

Outline

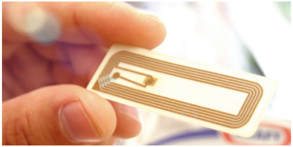
1. Simeck's Design Goals
2. Design Specifications and Rationales
3. Hardware Implementations Results
4. Results Comparison between Simeck and SIMON
5. Security Analysis
6. Conclusions

Outline

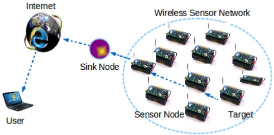
1. Simeck's Design Goals
2. Design Specifications and Rationales
3. Hardware Implementations Results
4. Results Comparison between Simeck and SIMON
5. Security Analysis
6. Conclusions

Lightweight Cryptography

- Lightweight cryptography is devised to provide suitable, secure, and compact ciphers (less than 2000 GEs) that fit into the resource constrained devices, such as passive RFID tags and wireless sensor network nodes.



RFID tags

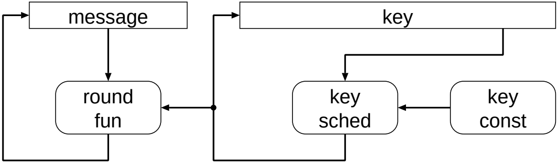


Wireless sensor network nodes

- Block ciphers: TEA, XTEA, PRESENT, KATAN, LED, EPCBC, KLEIN, LBlock, Piccolo, Twine, SIMON, and SPECK.
- Stream ciphers: Trivium, Grain, WG (WG-5, WG-7, WG-8).

A Smaller Block Cipher than SIMON

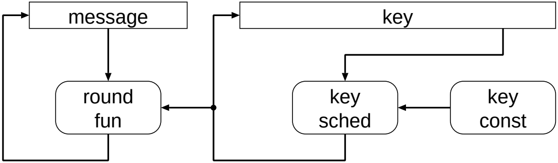
- SIMON is optimized for hardware and SPECK is optimized for software [Beaulieu *et al.*, 2013].



- How to design a smaller cipher family than SIMON?
  - The registers cannot be changed.
  - We can reduce the areas of only the round function, key schedule, and key constant.

A Smaller Block Cipher than SIMON

- SIMON is optimized for hardware and SPECK is optimized for software [Beaulieu *et al.*, 2013].



- How to design a smaller cipher family than SIMON?
  - The registers cannot be changed.
  - We can reduce the areas of only the round function, key schedule, and key constant.
- **Simeck**

Simeck: A Family of Lightweight Block Ciphers

- Simeck is designed to have similar security levels as SIMON but with smaller area.
- Simeck is designed by combining the best features of SIMON and SPECK.
  - Round function.
    - Use a modified version of SIMON's round function.
  - Key schedule.
    - Use round function for key schedule, similar to SPECK.
  - Key constant.
    - Use LFSR-based constant for key schedule, similar to SIMON, but simpler.

Simeck: A Family of Lightweight Block Ciphers

- Simeck is designed to have similar security levels as SIMON but with smaller area.
- Simeck is designed by combining the best features of SIMON and SPECK.
  - Round function.
    - Use a modified version of SIMON's round function.
  - Key schedule.
    - Use round function for key schedule, similar to SPECK.
  - Key constant.
    - Use LFSR-based constant for key schedule, similar to SIMON, but simpler.
- Simeck has three instances.
  - Simeck32/64, Simeck48/96, Simeck64/128.
  - The number of rounds for Simeck are identical with the corresponding SIMON.

Outline

1 Simeck's Design Goals

2 Design Specifications and Rationales

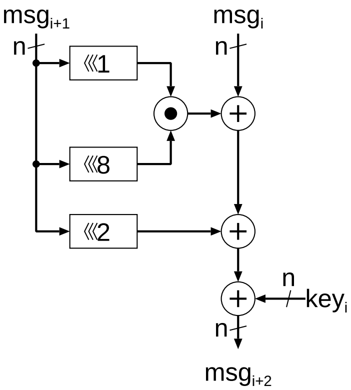
3 Hardware Implementations Results

4 Results Comparison between Simeck and SIMON

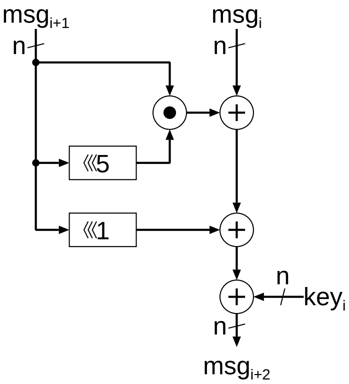
5 Security Analysis

6 Conclusions

Round Function



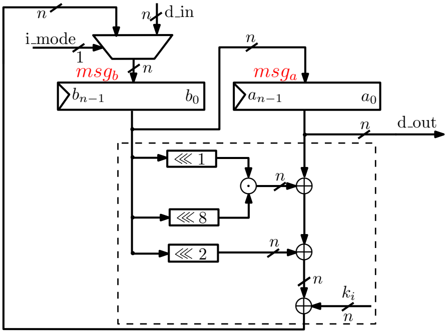
SIMON



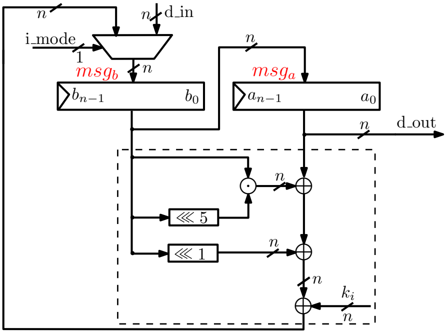
Simeck

- *n* is the word size (16, 24, 32).

Round Function in the Parallel Architecture



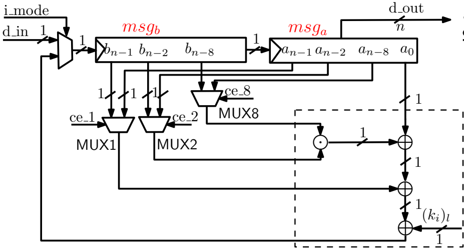
SIMON



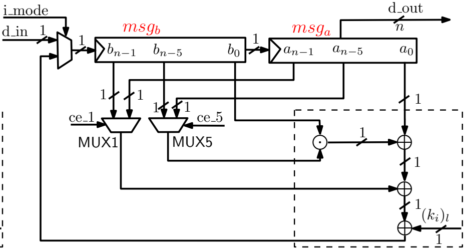
Simeck

- The parallel architecture processes 1 round per clock cycle and the datapath is *n*-bit width.
- Different shift numbers do not affect the area in parallel architecture.

Round Function in the Fully Serialized Architecture



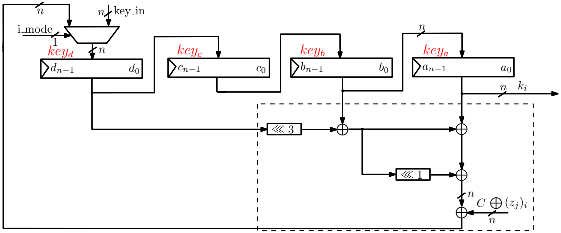
SIMON



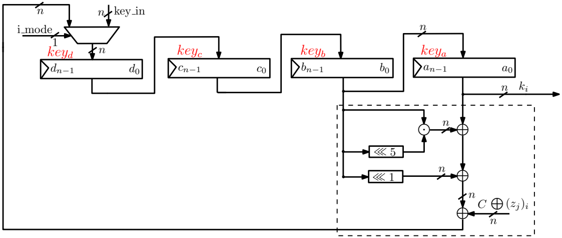
Simeck

- The fully serialized architecture processes 1 bit per clock cycle and the datapath is 1-bit width.
- Different shift numbers affect the area in the partially serialized architecture in hardware.
  - Reduce 1 MUX (multiplexer) for the fully serialized architecure.
  - Simplify logic to select the MUXes.

Key Schedule in the Parallel Architecture



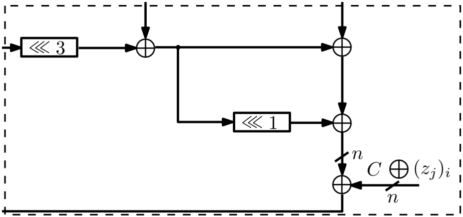
SIMON



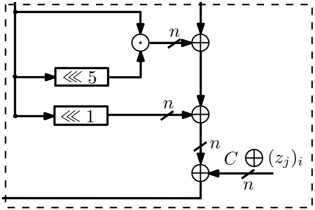
Simeck

- Similar as the round function, the parallel architecture processes 1 round per clock cycle and the datapath is *n*-bit width.

Simplified Key Schedule



SIMON



Simeck

- The combinational circuit (dashed box in above) in the key schedule of SIMON and Simeck in the parallel architecture are shown as follows:

SIMON	$(2n + 1)$ XOR + $(n - 1)$ XNOR
Simeck	$(n + 1)$ XOR + $(n - 1)$ XNOR + n AND

- In general, one XOR gate is larger than one AND gate. Thus, Simeck's key schedule is smaller than SIMON.

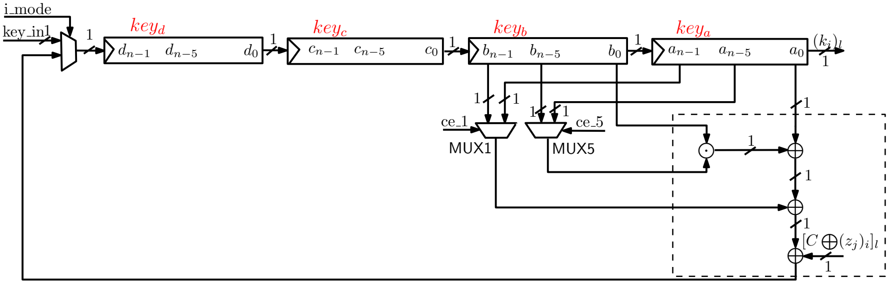
Simplified Key Constant

- The primitive polynomials for the LFSRs to generate the key constants for Simeck and SIMON.

	Simeck	SIMON
32/64	$X^5 + X^2 + 1$	$X^5 + X^4 + X^2 + X + 1$
48/96	$X^5 + X^2 + 1$	$X^5 + X^3 + X^2 + X + 1$
64/128	$X^6 + X + 1$	$X^5 + X^3 + X^2 + X + 1$

- Simeck's are all 2 XOR gates (4 GEs) less than the ones used in SIMON.

Key Schedule in the Fully Serialized Architecture


i_mode
key_in1
1
key_d
d_{n-1} d_{n-5} d_0
key_c
c_{n-1} c_{n-5} c_0
key_b
b_{n-1} b_{n-5} b_0
key_a
a_{n-1} a_{n-5} a_0
$(k_i)_l$
ce_1
MUX1
ce_5
MUX5
$[C \oplus (z_j)_i]_l$

Simeck

- Similar as the round function, the fully serialized architecture processes 1 bit per clock cycle and the datapath is 1-bit width.
- Different shift numbers affect the area in the fully serialized architecture, as round function does.
  - Reduce 1 MUX.
  - Simplify logic to select the MUXes.
- The combinational circuit (dashed box) is also decreased.

Outline

1. Simeck's Design Goals
2. Design Specifications and Rationales
3. Hardware Implementations Results
4. Results Comparison between Simeck and SIMON
5. Security Analysis
6. Conclusions

Our Implementation Results of Simeck32/64, 48/96, 64/128 in 130nm

Simeck	Partial	CMOS 130nm					
		Area (GEs)		Max Frequency (MHz)	Throughput @100 KHz (Kbps)	Total Power @100 KHz (μW)	Total Power @2 MHz (μW)
	serial	Before P&R	After P&R				
Simeck32/64	1-bit	505*	549*	292	5.6	0.417	8.3
	2-bit	510†	555†	288	11.1	0.431	8.5
	4-bit	533†	579†	312	22.2	0.463	9.2
	8-bit	591†	642†	289	44.4	0.523	10.4
	16-bit	695*	756*	526	88.9	0.606	11.9
Simeck48/96	1-bit	715†	778†	299	5.0	0.576	11.4
	2-bit	722†	785†	294	10.0	0.593	11.8
	3-bit	731†	794†	268	15.0	0.611	12.1
	4-bit	748†	813†	284	20.0	0.628	12.5
	6-bit	770†	837†	287	30.0	0.651	12.9
	8-bit	801†	871†	284	40.0	0.688	13.6
	12-bit	858†	933†	283	60.0	0.742	14.7
	24-bit	1027*	1117*	512	120.0	0.875	17.3
Simeck64/128	1-bit	924*	1005*	288	4.2	0.754	14.9
	2-bit	933†	1015†	303	8.3	0.778	15.4
	4-bit	958†	1041†	271	16.7	0.803	15.9
	8-bit	1013†	1101†	280	33.3	0.834	16.6
	16-bit	1132†	1231†	301	66.7	0.977	19.4
	32-bit	1365*	1484*	512	133.3	1.162	23.0

* Area obtained by using synthesis option compile ultra only.

† Area obtained by using synthesis option compile ultra and clock gating.

Our Implementation Results of SIMON32/64, 48/96, 64/128 in 130nm

SIMON	Partial serial	CMOS 130nm						
		Area (GEs)			Max Frequency (MHz)	Throughput @100 KHz (Kbps)	Total Power @100 KHz (μW)	Total Power @2 MHz (μW)
		Before P&R	After P&R	NSA Before P&R				
SIMON32/64	1-bit	517†	562†	523	331	5.6	0.421	8.3
	2-bit	532*	578*	535	306	11.1	0.439	8.7
	4-bit	563†	612†	566	283	22.2	0.479	9.5
	8-bit	623*	677*	627	367	44.4	0.540	10.7
	16-bit	715*	778*	722	456	88.9	0.645	12.8
SIMON48/96	1-bit	733†	796†	739	258	5.0	0.579	11.5
	2-bit	745†	810†	750	289	10.0	0.601	11.9
	3-bit	756†	822†	763	291	15.0	0.615	12.2
	4-bit	778†	846†	781	287	20.0	0.642	12.7
	6-bit	800†	869†	804	289	30.0	0.670	13.3
	8-bit	833†	905†	839	238	40.0	0.706	13.9
	12-bit	895†	973†	898	307	60.0	0.777	15.4
	24-bit	1055*	1147*	1062	467	120.0	0.929	18.4
SIMON64/128	1-bit	944†	1026†	958	225	4.2	0.762	15.1
	2-bit	955†	1038†	968	244	8.3	0.780	15.4
	4-bit	988†	1074†	1000	290	16.7	0.818	16.2
	8-bit	1043†	1134†	1057	296	33.3	0.866	17.2
	16-bit	1174†	1276†	1185	293	66.7	1.024	20.3
	32-bit	1403*	1524*	1417	465	133.3	1.239	24.6

* Area obtained by using synthesis option compile ultra only.

† Area obtained by using synthesis option compile ultra and clock gating.

Outline

1 Simeck's Design Goals

2 Design Specifications and Rationales

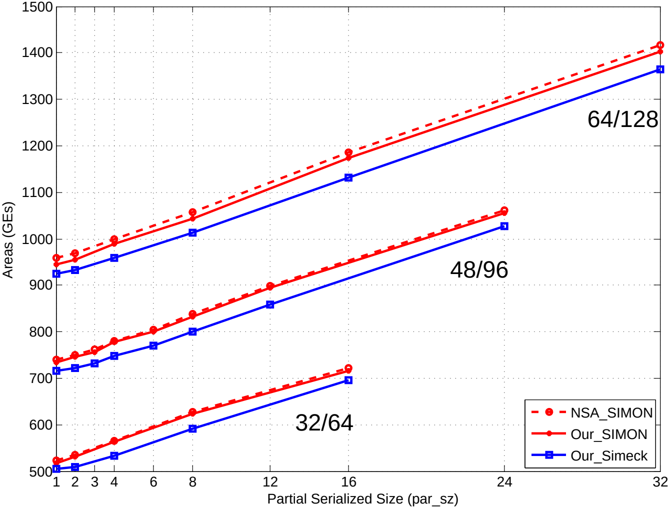
3 Hardware Implementations Results

4 Results Comparison between Simeck and SIMON

5 Security Analysis

6 Conclusions

Area (before the Place and Route) Comparisons in CMOS 130nm



Area Comparisons between Simeck32/64 and SIMON32/64

Breakdown of the Results (before the Place and Route) in CMOS 130nm

Components		Parallel (GEs)			Fully Serialized (GEs)		
		Simeck	SIMON*	Difference	Simeck	SIMON*	Difference
Control		31	35	4	71	75	4
Datapath	Round (comb)	112	112	0	7	7	0
	Key (comb)	80	96	16	5	8	3
	Regs + MUXes	474	474	0	434	443	9
Totals	Compile simple†	697	717	20	517	533	16
	Compile ultra†	695	717	-	505	520	-
	Compile ultra + clock gating†	695	715	-	506	517	-

* Our own SIMON results.

† Synthesis options.

Results Summary

- Fully serialized architecture.
  - The round function, key schedule and key constant modules of SIMON32/64 account for only 6.4% of the total area.
  - Simeck32/64 reduces this by 46%, which leads to 2.3% smaller total area in comparison to our implementations of SIMON32/64 and 3.4% smaller than the original results in 130nm.
  - Similarly, Simeck48/96, Simeck64/128 are 3.3%, 3.5% smaller than the original results in 130nm.
- Parallel architecture.
  - Simeck32/64, 48/96, 64/128 are 3.7%, 3.3%, 3.7% respectively smaller than the original results in 130nm.

Outline

1 Simeck's Design Goals

2 Design Specifications and Rationales

3 Hardware Implementations Results

4 Results Comparison between Simeck and SIMON

5 Security Analysis

6 Conclusions

Security Analysis

- Changing the shift numbers of the round function influences the security [Kölbl *et al.*, CRYPTO 15].
  - Linear and differential diffusion.
- We made a trade-off between security and area for Simeck.
- Simeck benefits from Simon/Speck's security analysis due to the similarity between Simon/Speck and Simeck [Kölbl and Roy, eprint 2015/706], [Bagheri, eprint 2015/716].
- Security analysis summary.

Cipher	Simon* attacked rounds/total rounds			Simeck attacked rounds/total rounds		
32/64	23/32	72%	(linear hull)	20/32	62.5%	(impossible differential)
48/96	25/36	69%	(linear hull)	26/36	72%	(differential)
64/128	31/44	70%	(linear hull)	33/44	75%	(differential)

* [Beaulieu *et al.*, eprint 2015/585].

Outline

1 Simeck's Design Goals

2 Design Specifications and Rationales

3 Hardware Implementations Results

4 Results Comparison between Simeck and SIMON

5 Security Analysis

6 Conclusions

Conclusions

- We have presented Simeck: a new family of lightweight block ciphers.
- We have provided an extensive exploration for different hardware architectures in order to make a balance between area, throughput, and power consumption for SIMON and Simeck in both CMOS 130nm and 65nm ASICs.
- We have shown that it is possible to design a smaller cipher than SIMON in terms of area and power consumption.
- Simeck is slightly more vulnerable than SIMON to reduced round attacks, but still has sufficient margin for real-world applications.

Appendix I: Our Implementation Results of Simeck32/64, 48/96, 64/128 in 65nm

Simeck	Partial	CMOS 65nm					
		Area (GEs)		Max Frequency (MHz)	Throughput @100 KHz (Kbps)	Total Power @100 KHz (μW)	Total Power @2 MHz (μW)
	Serial	Before P&R	After P&R				
Simeck32/64	1-bit	454*	488*	1754	5.6	1.292	5.5
	2-bit	465†	500†	1428	11.1	1.311	5.6
	4-bit	494†	531†	1388	22.2	1.376	5.9
	8-bit	550*	592*	1250	44.4	1.512	6.4
	16-bit	644*	692*	1428	88.9	1.716	6.8
Simeck48/96	1-bit	645†	693†	1562	5.0	1.805	7.8
	2-bit	656†	706†	1538	10.0	1.825	8.0
	3-bit	663†	712†	1282	15.0	1.857	8.4
	4-bit	686†	738†	1333	20.0	1.886	8.2
	6-bit	701†	753†	1282	30.0	1.919	8.4
	8-bit	732†	787†	1388	40.0	2.009	8.8
	12-bit	794*	854*	1219	60.0	2.212	9.3
	24-bit	951*	1022*	2325	120.0	2.44	9.6
Simeck64/128	1-bit	828*	891*	1369	4.2	2.304	10.2
	2-bit	838†	901†	1408	8.3	2.325	10.3
	4-bit	869†	935†	1098	16.7	2.372	10.5
	8-bit	918†	987†	1190	33.3	2.492	10.9
	16-bit	1042*	1121*	1086	66.7	2.869	12.3
	32-bit	1263*	1358*	1282	133.3	3.316	13.1

* Area obtained by using synthesis option compile ultra only.

† Area obtained by using synthesis option compile ultra and clock gating.

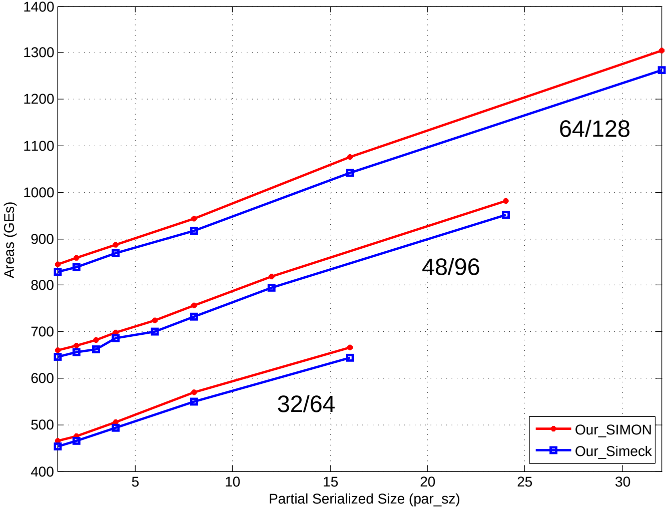
Appendix II: Our Implementation Results of Simon32/64, 48/96, 64/128 in 65nm

Simon	Partial Serial	CMOS 65nm					
		Area (GEs)		Max Frequency (MHz)	Throughput @100 KHz (Kbps)	Total Power @100 KHz (μW)	Total Power @2 MHz (μW)
		Before P&R	After P&R				
Simon32/64	1-bit	466*	501*	1428	5.6	1.311	5.6
	2-bit	476*	512*	1562	11.1	1.331	5.7
	4-bit	506*	544*	1408	22.2	1.381	5.9
	8-bit	570*	613*	1075	44.4	1.585	6.8
	16-bit	666*	716*	2222	88.9	1.751	6.8
Simon48/96	1-bit	661†	711†	1204	5.0	1.812	7.9
	2-bit	670†	720†	1136	10.0	1.889	9.5
	3-bit	682†	733†	1086	15.0	1.86	8.1
	4-bit	699†	752†	1041	20.0	1.915	8.3
	6-bit	724†	779†	1369	30.0	1.962	8.5
	8-bit	757†	814†	1282	40.0	2.122	9.0
	12-bit	819*	881*	1176	60.0	2.305	9.7
	24-bit	982*	1056*	2222	120.0	2.542	9.9
Simon64/128	1-bit	845†	908†	1282	4.2	2.336	10.2
	2-bit	858†	922†	1265	8.3	2.366	10.4
	4-bit	887†	954†	1250	16.7	2.423	10.6
	8-bit	944†	1015†	1265	33.3	2.577	11.2
	16-bit	1076*	1156*	1176	66.7	3.068	12.8
	32-bit	1305*	1403*	1694	133.3	3.398	13.4

* Area obtained by using synthesis option compile ultra only.

† Area obtained by using synthesis option compile ultra and clock gating.

Area (before the Place and Route) Comparisons in CMOS 65nm


Areas (GEs)
Partial Serialized Size (par_sz)
64/128
48/96
32/64
Our_SIMON
Our_Simeck