Florida Atlantic University
Hai Pham, Rainer Steinwandt

PQSecure Technologies
Brandon Langenberg

Reducing the cost of implementing AES as a quantum circuit

Introduction

Continuous progress on quantum computers

- Allows quantum algorithms to operate and attack currently secure encryptions

Particular example of quantum algorithms

- Grover's algorithm – key search for AES

Motivation

Security baseline in a post-quantum setting

- NIST's ongoing standardization effort
- Security strength categories based on resources needed to attack symmetric primitives

Any attack that breaks the relevant security definition must require computational resources comparable to or greater than those required for key search on a block cipher with a *k*-bit key (e.g. AES-*k*)

Prior work on AES cost estimate

- Grassl et al.'s work from PQCrypto 2016
- improved quantum circuit by Almazrooie et al. 2018

Contribution

New quantum circuit for AES S-box

- Reduce the numbers for qubits and quantum gates needed (NOT, CNOT, Toffoli)

New quantum resource estimate for all three versions of AES

- Revise the cost estimate for Grover's attack against AES

Preliminary

Advanced Encryption Standard (AES)

- Operates on a state of 128 bits
- Key size (128, 192, 256 bits) and corresponding rounds (8, 10, and 12)
- Main steps:
  - SubByte
  - ShiftRows
  - MixedColumns
  - AddRoundKey

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

All of AES can be implemented by means of NOT and CNOT gates with the exception of SubByte

Algebraic Structure of SubByte

$$GF(2)^8 \to GF(2)^8$$
$$b \mapsto S(b)$$

1. Interpret input byte as element $b \in \mathbb{F}_2[x] \,/\, (x^8 + x^4 + x^3 + x + 1)$, replace b with b^{-1} (0 maps to 0)
2. Apply an affine transformation

SubByte can be expressed as a substitution table

SubByte as a Substitution Table

		y														
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Observations about SubByte structure

1. The map S

   Reduce the number of qubits by evaluating “in place”

2. The affine transformation

   Can be expressed with NOT and CNOT

SubByte as quantum circuit (prior work)

		# qubits	Toffoli	CNOT	NOT
2016	Grassl et al.	9	1385	1551	
		40	512	469	4
2018	Almazrooie et al.	56	448	494	4

Result from classical reversible circuits

Saravanan & Kalpana	Wireless Personal Communication 2018	Many “garbage outputs” Only 35 Toffoli, 152 CNOT, and 4 NOT gates
Boyar & Peralta	ePrint Archive: Report 2011/332	Depth-16 circuit for AES S-Box 34 AND gates
Boyar & Peralta (older design)	SEA 2010	Only 32 AND gates Our starting point

Proposed quantum circuit for the S-box in AES

Boyar and Peralta discuss a technique for combinational logic optimization

1. Identify non-linear circuit components and reduce the number of AND gates (saving Toffoli gates)

2. Find maximal linear components of the circuit and minimize the number of XOR gates (reduce CNOT gates)

Proposed quantum circuit for the S-box in AES

A representation for the function $S(x) = B \cdot F(U \cdot x)$

$B \in \mathbb{F}_2^{8\times 18}$

$U \in \mathbb{F}_2^{22\times 8}$

$F: \mathbb{F}_2^{22} \rightarrow \mathbb{F}_2^{18}$

$$
\begin{array}{lll}
t_2 = y_{12} \cdot y_{15} & t_3 = y_3 \cdot y_6 & t_4 = t_3 + t_2 \\
t_5 = y_4 \cdot x_7 & t_6 = t_5 + t_2 & t_7 = y_{13} \cdot y_{16} \\
t_8 = y_5 \cdot y_1 & t_9 = t_8 + t_7 & t_{10} = y_2 \cdot y_7 \\
t_{11} = t_{10} + t_7 & t_{12} = y_9 \cdot y_{11} & t_{13} = y_{14} \cdot y_{17} \\
t_{14} = t_{13} + t_{12} & t_{15} = y_8 \cdot y_{10} & t_{16} = t_{15} + t_{12} \\
t_{17} = t_4 + t_{14} & t_{18} = t_6 + t_{16} & t_{19} = t_9 + t_{14} \\
t_{20} = t_{11} + t_{16} & t_{21} = t_{17} + y_{20} & t_{22} = t_{18} + y_{19} \\
t_{23} = t_{19} + y_{21} & t_{24} = t_{20} + y_{18} &
\end{array}
$$

$$
\begin{array}{lll}
t_{25} = t_{21} + t_{22} & t_{26} = t_{21} \cdot t_{23} & t_{27} = t_{24} + t_{26} \\
t_{28} = t_{25} \cdot t_{27} & t_{29} = t_{28} + t_{22} & t_{30} = t_{23} + t_{24} \\
t_{31} = t_{22} + t_{26} & t_{32} = t_{31} \cdot t_{30} & t_{33} = t_{32} + t_{24} \\
t_{34} = t_{23} + t_{33} & t_{35} = t_{27} + t_{33} & t_{36} = t_{24} \cdot t_{35} \\
t_{37} = t_{36} + t_{34} & t_{38} = t_{27} + t_{36} & t_{39} = t_{29} \cdot t_{38} \\
t_{40} = t_{25} + t_{39} & &
\end{array}
$$

$$
\begin{array}{lll}
t_{41} = t_{40} + t_{37} & t_{42} = t_{29} + t_{33} & t_{43} = t_{29} + t_{40} \\
t_{44} = t_{33} + t_{37} & t_{45} = t_{42} + t_{41} & z_0 = t_{44} \cdot y_{15} \\
z_1 = t_{37} \cdot y_6 & z_2 = t_{33} \cdot x_7 & z_3 = t_{43} \cdot y_{16} \\
z_4 = t_{40} \cdot y_1 & z_5 = t_{29} \cdot y_7 & z_6 = t_{42} \cdot y_{11} \\
z_7 = t_{45} \cdot y_{17} & z_8 = t_{41} \cdot y_{10} & z_9 = t_{44} \cdot y_{12} \\
z_{10} = t_{37} \cdot y_3 & z_{11} = t_{33} \cdot y_4 & z_{12} = t_{43} \cdot y_{13} \\
z_{13} = t_{40} \cdot y_5 & z_{14} = t_{29} \cdot y_2 & z_{15} = t_{42} \cdot y_9 \\
z_{16} = t_{45} \cdot y_{14} & z_{17} = t_{41} \cdot y_8 &
\end{array}
$$

Conversion to a quantum circuit

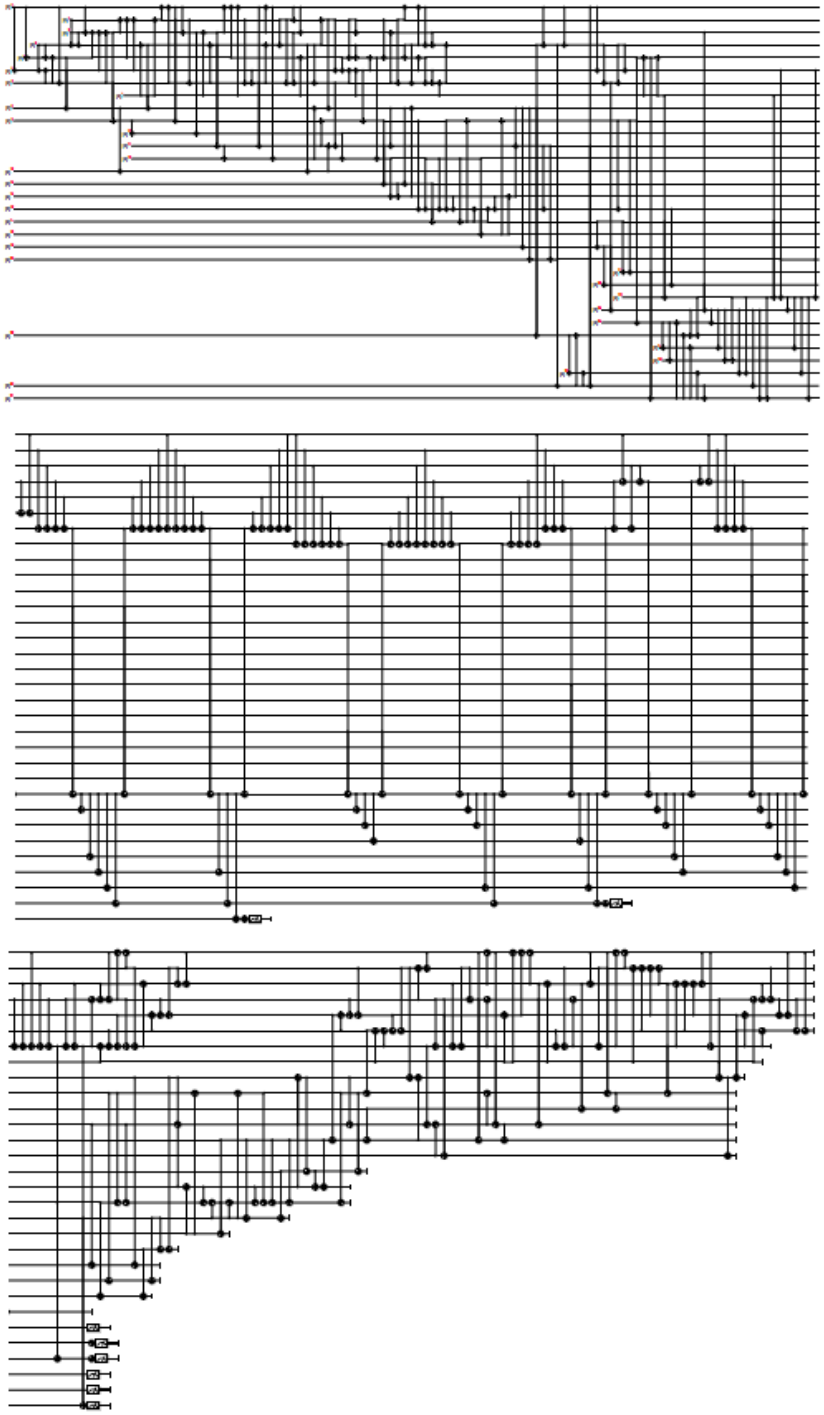
- Straightforward translation without cleaning up: 126 qubits, 32 Toffoli, 166 CNOT, and 4 NOT gates
- Observation 1: many wires remain idle after an initial use ⇒ try to “clean up” early and reuse

Conversion to a quantum circuit

- Observation 2: some Toffoli gates can be placed to write directly onto output wires ⇒ avoid uncomputing of Toffoli gates
- Observation 3: some wires just serve as “intermediate storage” ⇒ target directly the final target to save some wires

Cost estimate for the proposed S-box circuit

# qubits	32
Toffoli gates	55
CNOT gates	314
NOT gates	4
Toffoli-depth	40
Overall S-box depth	298



AES-k Cost Comparison

	AES-128	AES-192	AES-256
# Toffoli gates	16,940	19,580	23,760
# CNOT gates	107,960	125,580	151,011
# NOT gates	1,507	1,692	1,992
# qubits	864	896	1,232
S-box depth	47	41	54
Toffoli depth	1,880	1,640	2,160

Resource estimates for AES using prior design

Grassl et al.

	AES-128	AES-192	AES-256
# Toffoli gates	151,552	172,032	215,040
	16,940	19,580	23,760
# CNOT gates	166,548	166,548	233,836
	107,960	125,580	151,011
# NOT gates	1,456	1,608	1,943
	1,507	1,692	1,992
# qubits	984	1,112	1,336
	864	896	1,232
Toffoli Depth	12,672	11,088	14,976
	1,880	1,640	2,160

Resource estimates for AES using prior design

	Almazrooie et al.	This paper
	AES-128	AES-128
# Toffoli gates	150,528	16,940
# CNOT gates	192,832	107,960
# NOT gates	1,370	1,507
# qubits	976	864
Toffoli Depth	N/A	1,880

Exhaustive key search with Grover's algorithm

	Grassel et al.	This paper
AES-128		
#qubits	2,953	865
#T-gates	$1.19 \cdot 2^{86}$	$1.47 \cdot 2^{81}$
T-depth	$1.06 \cdot 2^{80}$	$1.44 \cdot 2^{77}$
#Clifford gates	$1.55 \cdot 2^{86}$	$1.46 \cdot 2^{82}$

Exhaustive key search with Grover's algorithm

	Grassel et al.	This paper
AES-192		
#qubits	4,449	1,763
#T-gates	$1.81 \cdot 2^{118}$	$1.68 \cdot 2^{114}$
T-depth	$1.21 \cdot 2^{112}$	$1.26 \cdot 2^{109}$
#Clifford gates	$1.17 \cdot 2^{119}$	$1.71 \cdot 2^{115}$

Exhaustive key search with Grover's algorithm

	Grassel et al.	This paper
AES-256		
#qubits	6,681	2,465
#T-gates	$1.41 \cdot 2^{151}$	$1.02 \cdot 2^{147}$
T-depth	$1.44 \cdot 2^{144}$	$1.66 \cdot 2^{141}$
#Clifford gates	$1.83 \cdot 2^{151}$	$1.03 \cdot 2^{148}$

References

1. Mishal Almazrooie, Azman Samsudin, Rosni Abdullah, and Kussay N. Mutter. Quantum reversible circuit of AES-128. Quantum Information Processing, 17(5):112, 2018.
2. Markus Grassl, Brandon Langenberg, Martin Roetteler, and Rainer Steinwandt. Applying Grover's Algorithm to AES: Quantum Resource Estimates. In Tsuyoshi Takagi, editor, Post-Quantum Cryptography PQCrypto 2016, volume 9606 of Lecture Notes in Computer Science, pages 29–43. Springer, 2016.
3. Joan Boyar and René Peralta. A depth-16 circuit for the AES S-box. Cryptology ePrint Archive: Report 2011/332, June 2011. Available at https://eprint.iacr.org/2011/332.

References

4. Charles H. Bennett. Logical Reversibility of Computation. IBM Journal of Research and Development, 17(6):525–532, 1973.
5. NIST. Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process, 2017. Available at https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.
6. Damian S. Steiger, Thomas Häner, and Matthias Troyer. ProjectQ: An Open Source Software Framework for Quantum Computing. CoRR, abs/1612.08091, 2016.