Minimal CRL Gate Designs for Logical Operations with Two and Three Inputs

Weston Beebe*, J.W. Bruce*, and David Goldhirsch†

*Department of Electrical and Computer Engineering, Tennessee Tech
†Department of Electrical and Computer Engineering, Binghamton University

1. INTRODUCTION

Energy consumption and heat are crucial limiting factors in modern digital systems. Landauer [1] showed that any irreversible logic gate dissipates $kT \ln 2$ Joules of energy with each irreversible bit . Therefore, reversible logic is required for ultimate energy efficiency in computational systems. Conservative logic gates [2] – a gate whose Hamming weights for inputs and outputs are equal – are theoretically energy neutral in that output signal energy can be derived from input signal energy.

2. BACKGROUND

A well-known conservative reversible logic (CRL) gate is the Fredkin gate [2]. The Fredkin gate (FG) is a "controlled-swap" whereby one signal determines whether two other signals are passed "straight-through" or "swapped". The FG is often used as an operator in many quantum computing algorithms.
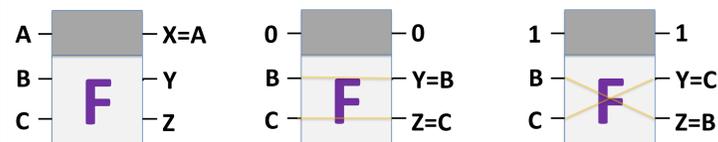

Fig. 1: Fredkin gate (L) and its two operational states

Previous research, including [2] and [3], have proposed FG implementations of common digital logic operators. Later, additional FG designs for logical primitives were proposed and used in the design of more complex logical operations [3].

CRL gates are not limited to the size of the FG. CRL gates can have many more than just three inputs and three outputs. With more inputs and outputs, CRL gates get increasingly complex. The FG, with only three inputs, is limited to permutations on the two signals that are passed "straight-through" or "swapped". With more inputs, the number of permutations on those inputs increases, leading to more gates and more complex behavior.

Many FG designs of traditional digital logic operations have been proposed, but for larger CRL gates, no systematic study of implementations of traditional digital logic operations has been undertaken.

3. OBJECTIVES

- Determine if existing FG implementations of traditional digital logical operations are unique
- Find new CRL gate implementations of traditional logical operations, if they exist
- Find CRL gate designs that are able to implement more than just one traditional logical operation

4. METHODOLOGY

To find CRL gate implementations of traditional logical operations, circuits were simulated and circuit outputs were recorded for different combinations of gate inputs and gate connections.

With a specified number of variables, gate size, and stage depth, combinations on the different placement of variables and different gate connections were generated. At least one function input is required to impinge on the first gate [4]. Additional inputs can be applied at any other location in the circuit.

CRL gates can be implemented with either an active-high or an active-low control signal [4]. This study considers both CRL gate implementations.

As new stages are added to the circuit, its inputs are taken from combinations on constants, unused function inputs, and unused gate outputs from previous stages.

In this study, combinations of variables, constants, gates, and wirings between gates were built to a specified depth of gates. The search continued in depth until specified digital functions were found. The "garbage" outputs were inspected for useful digital functions in addition to and separate from the originally specified function.

5. RESULTS AND DISCUSSION

All two-input logical functions can be implemented in one or two FGs. Four two-input logic functions require two FGs with the balance capable of being formed with a single FG. This study found that the previously published FG forms for primitive logical operation AND2 and OR2 to be optimal.
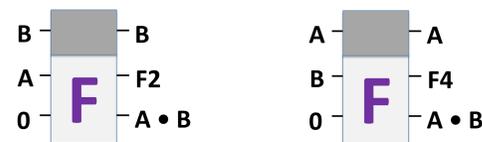

Fig. 2: FG circuits: AND2 with F2 (L) and AND2 with F4 (R)

For the two-input logical functions F2 and F4, one unique FG design was discovered. These functions show up in conjunction with AND2, and shown in Fig. 2.

For two-input logical functions F11 and F13, one unique FG design was discovered. Functions F11 and F13 are created simultaneously with OR2, and shown in Fig. 3.
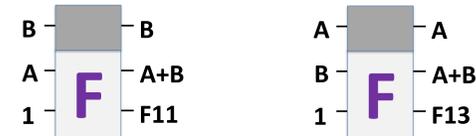

Fig. 3: FG circuits: OR2 with F11 (L), and OR2 with F13 (R)

The search of all possible FG implementations for all two-input logical functions revealed that all these functions could be implemented in two or fewer gates. Table 1 shows the number of unique FG designs for the non-trivial two-input logical operations. The results of the same search on CRL gates of size four are shown in Table 2.

Function	Fn	Number	Min. Gates		Function	Fn	Number	Min. Gates
AND2	F1	2	1		XNOR2	F9	1	2
NAND2	F14	5	2		Implication	F11	1	1
OR2	F7	2	1			F13	1	1
NOR2	F8	5	2		Inhibition	F2	1	1
XOR2	F6	1	2			F4	1	1

Table 1: Number of distinct FG implementations of the two-input logical operations. Trivial and 1-input functions F0 (null), F15 (identity), F3 (A), F5 (B), F10 (NOT B) and F12 (NOT A) are not considered.

Function	Fn	Number	Min. Gates		Function	Fn	Number	Min. Gates
AND2	F1	6	1		XNOR2	F9	30	2
NAND2	F14	45	2		Implication	F11	3	1
OR2	F7	6	1			F13	3	1
NOR2	F8	45	2		Inhibition	F2	3	1
XOR2	F6	30	2			F4	3	1

Table 2: Number of distinct size 4 CRL gate implementations of the two-input logical operations. Trivial and 1-input functions F0 (null), F15 (identity), F3 (A), F5 (B), F10 (NOT B) and F12 (NOT A) are not considered.
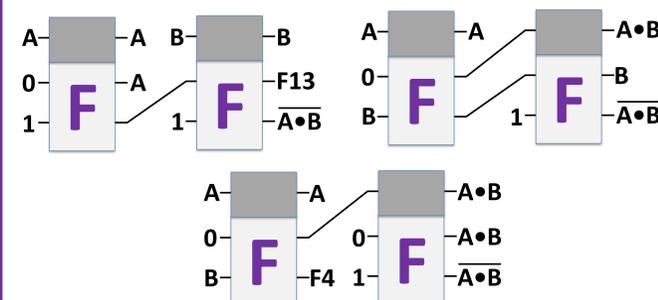

Fig. 4: Many of the possible FG NAND2 implementations. The two not shown here are the same as the first and last, but with A and B's placement swapped. The difference in output is F11 instead of F13 and F2 instead of F4.
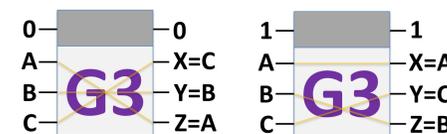

Fig. 5: The two operational states of G3, one of the CRL gates of size 4.

Fig. 5: CRL gate size 4 circuit: AND2 (F1) and OR2 (F7)

There are 256 possible logical functions over three inputs. This work also found all possible FG implementations of three-input functions. Table 3 shows the number of FG designs for common three-input logical operators.

Function	Fn	Number	Min. Gates		Function	Fn	Number	Min. Gates
AND3	F1	12	2		MAJ3	F23	30	3
NAND3	F254	69	3		FA_CARRY3	F23	30	3
OR3	F127	12	2		FA_SUM3	F105	1	3
NOR3	F128	69	3		OAI21	F234	27	3
XOR3	F105	1	3		AOI21	F168	27	3
XNOR3	F150	1	3		FA_PROP3	F60	1	3

Table 3: Number of distinct FG implementations of the common three-input operations.

On the search of three input functions with CRL gates of size four, it was discovered the F23 and F60 can be implemented in just two gates.
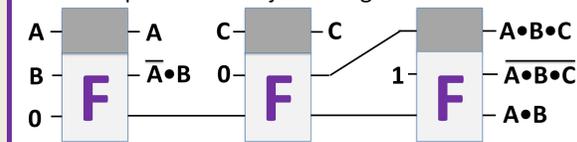

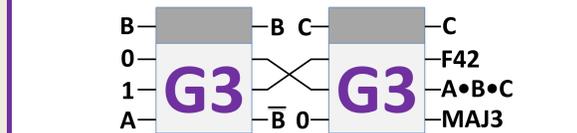Fig. 6: FG circuit: AND3 (F1), NAND3 (F254), and AND2

Fig. 7: CRL gate size 4 circuit: MAJ3 (F23) implemented in two stages and AND3 (F1)

6. CONCLUSIONS

The work describes the systematic study of CRL gate implementations of two-input and three-input logical operators. All possible FG circuit designs with one through three FGs were derived, and all possible CRL gate size 4 circuit designs with two gates were derived.

The study has identified optimal designs for all two-input functions and many useful three-input functions with CRL gates of size 3 and 4. The study also identified circuit designs which implemented more than just one of the commonly used logical operators.

7. REFERENCES

[1] R. Landauer, "Irreversibility and Heat Generation in the Computing Process", *IBM J. Research & Development*, v.3, pp. 183-191, July 1961.
[2] E. Fredkin and T. Toffoli, "Conservative Logic", *Intl. J. Theoretical Physics*, v.21, n.3-4, pp. 219-253, 1982.
[3] J.W. Bruce et al, "Efficient Adder Circuits Based on a Conservative Reversible Logic Gate", *Proc. IEEE ISVLSI 2002*, pp.83-88, 2002.
[4] W. Beebe and J.W. Bruce, "Optimal Fredkin Gate Designs for Logical Operations with Two and Three Inputs", *Proc. Student Research and Creative Inquiry Day 2021*, p. 102, 2021.