Arithmetic Function Extraction using Network Flow Model

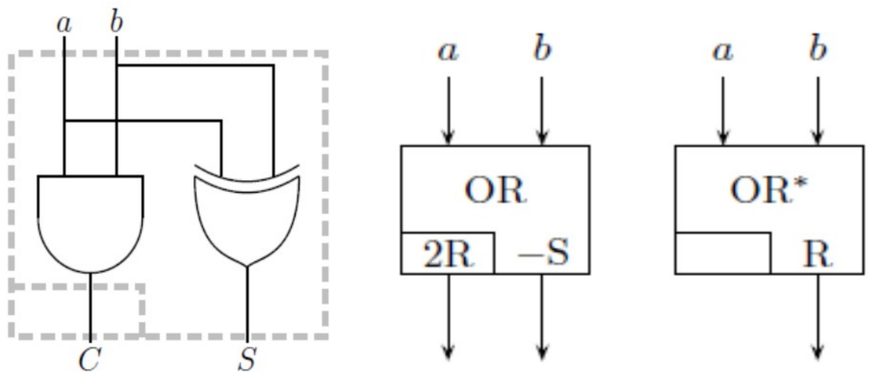
Maciej Ciesielski, Walter Brown, Duo Liu, Andre Rossi

Abstract

The paper describes a method to derive a polynomial function computed by an arithmetic bit-level circuit. The circuit is modeled as a bit-level network composed of adders and logic gates and computation performed by the circuit is viewed as a flow of binary data through the network. The problem is cast as Network Flow problem and solved using standard algebraic techniques. Extraction of the arithmetic function from the circuit is accomplished by transforming the expression at the primary outputs into an expression at the primary inputs. Experimental results show application of the method to certain classes of large arithmetic circuits.

Algebraic Model

Arithmetic circuit is represented as ABL network
- Composed of HA, FA, INV, OR, and OR*
- XOR and AND use the HA model, with unused outputs (floating signals)



$a + b = 2C + S$ $a + b = 2R - S$ $a + b = R$

Method Overview

Computation is modeled as data flow of binary data
- Data flow is expressed as algebraic expression

Basic Vocabulary

Input signature: Linear combination of input variables defining the unique function of the circuit (specification)
Ex: 3-bit adder: 8a3+8b3+4a2+4b2+2a1+2b1+a0+b0+c0

Output signature: Binary encoding of outputs (given)
16Cout + 8S3 + 4S2 + 2S1 + S0

Floating signals: signals that do not propagate further
{ S7, S6, S11, C10 }

Weights: coefficients associated with signal variables; determine flow value contributed by the signal.

Network Flow Conservation
- Flow at PI = flow at PO
- For each module: flow into the module = flow out of the module
- Flow at two consecutive cuts must be equal, taking into account data added by fanouts (Δ fanout) and data lost in floating signals (Σ floats):

SigPI + Δ(fanout) = SigPO + Σ(floats)

For signature to be trusted (functional correctness condition):

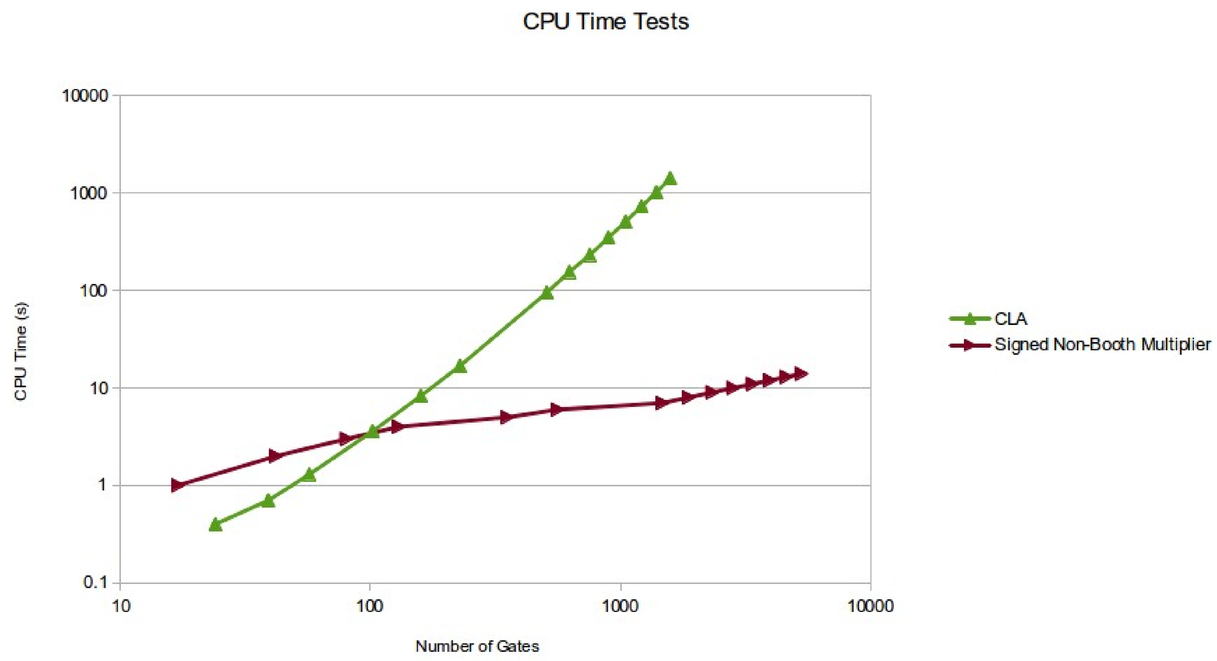
Δ(fanout) - Σ(floats) = 0

Extract function computed by the circuit:
Transform output signature into input signature through cut rewriting

Functional Verification
Analyze the input signature, compare with the specification (if known)

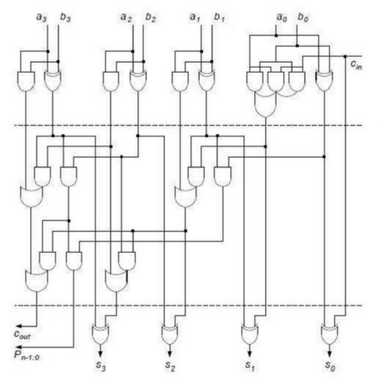
Debugging
Difference between specification and the computed input signature determines the bug; can be used as bug trace.

Experimental Results

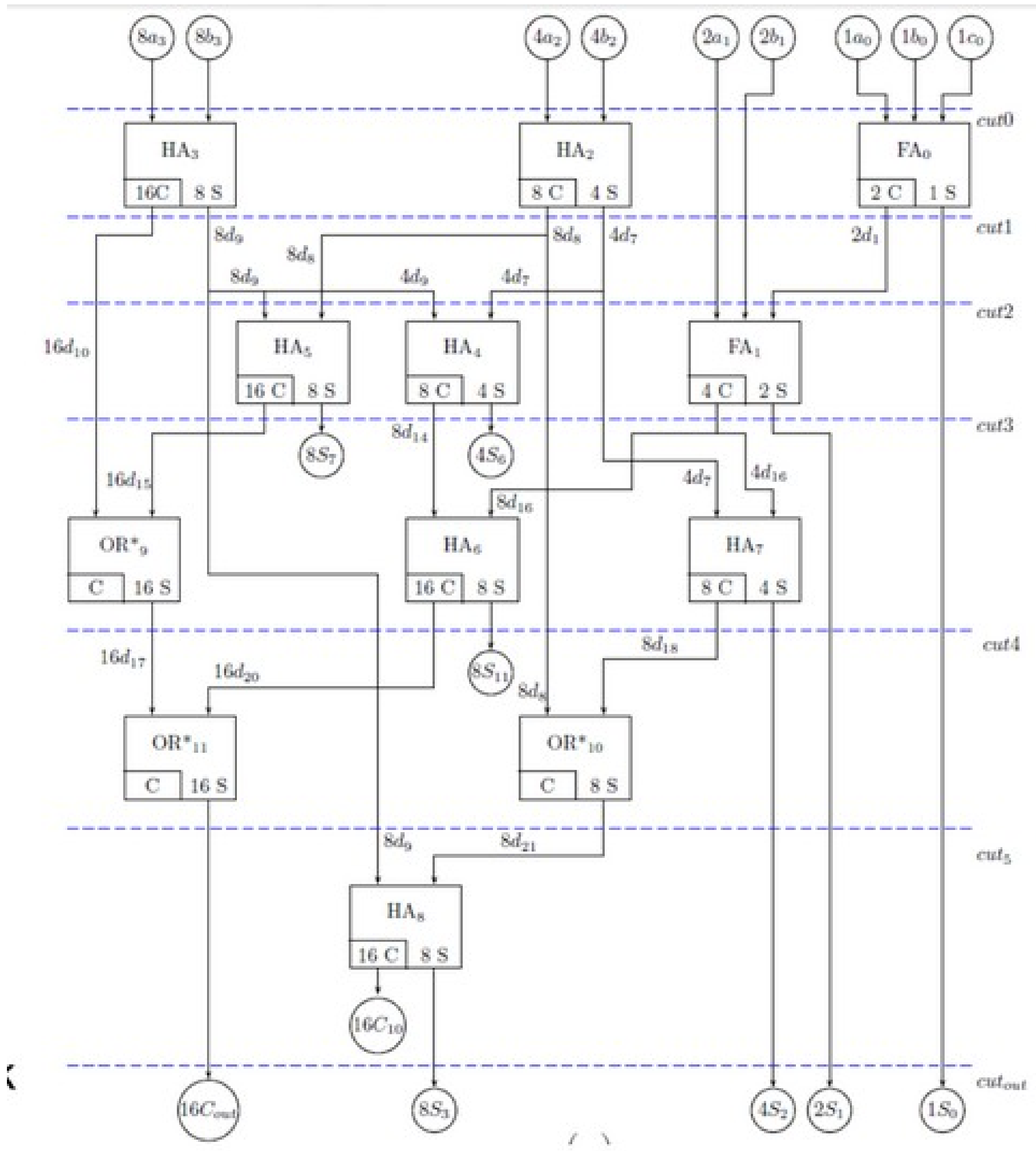


Comments

- Applicable to linear circuits or linear subsets of larger circuit
- ABL network must be derived from gate-level circuit.



Gate Level to ABL



Department of Electrical and Computer Engineering

This work was supported by a grant from the National Science Foundation, awards CCF-1319496 and CCF-1419657 (REU)

College of Engineering - University of Massachusetts Amherst

DAC14