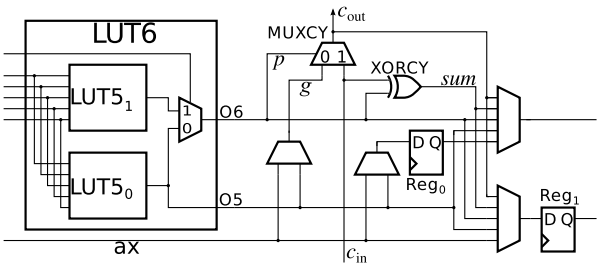
Fast and Area Efficient Adder for Wide Data in Recent Xilinx FPGAs

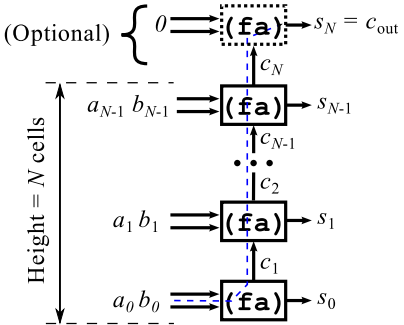
Petter Källström and Oscar Gustafsson

Linköping University
INSTITUTE OF TECHNOLOGY
Linköping University
Department of Computer Engineering

Introduction



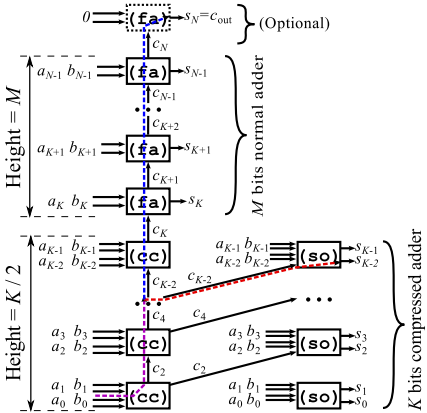
The Virtex 6 basic cell.



The normal adder.

Large N gives long critical path. Each LUT6 controls the *propagate* and *generate* signal.

Proposed Adder



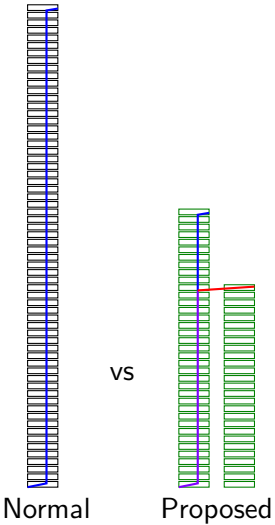
The proposed approach.

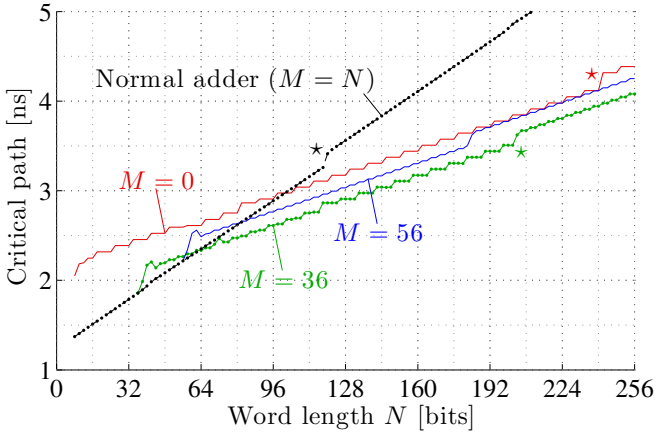
Decompose N into K LSBs and M MSBs.

- `(cc)` – $K/2$ carry compr. cells
- `(so)` – $K/2$ sum out cells
- `(fa)` – M normal full adder cells

In total $2K/2 + M = N$ LUTs, i.e. no more than a normal adder.
Optimum $M = 36$.

Result


vs
Normal
Proposed


Normal adder ($M = N$)
$M = 0$
$M = 56$
$M = 36$
Critical path [ns]
Word length N [bits]
1
2
3
4
5
0
32
64
96
128
160
192
224
256

Summary

The proposed adder architecture

- Twice the carry speed in parts of the adder without pipelining
- Same LUT count as a normal adder
- Efficient for long word lengths

Thank you!