Carry-Select Block Size and Time

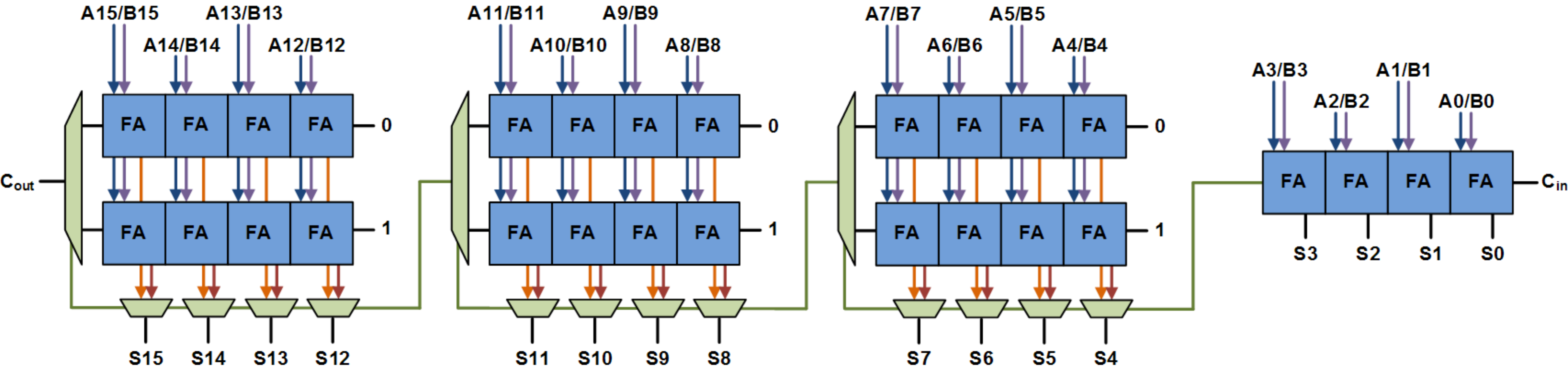
Timing of Carry-Select Adder

For 16-bit adder with 4-bit blocks:

$T_{CSA} = 4T_{FA} + 3T_{MUX}$

which is approximately $7T_{FA}$

What happens if we vary block size?



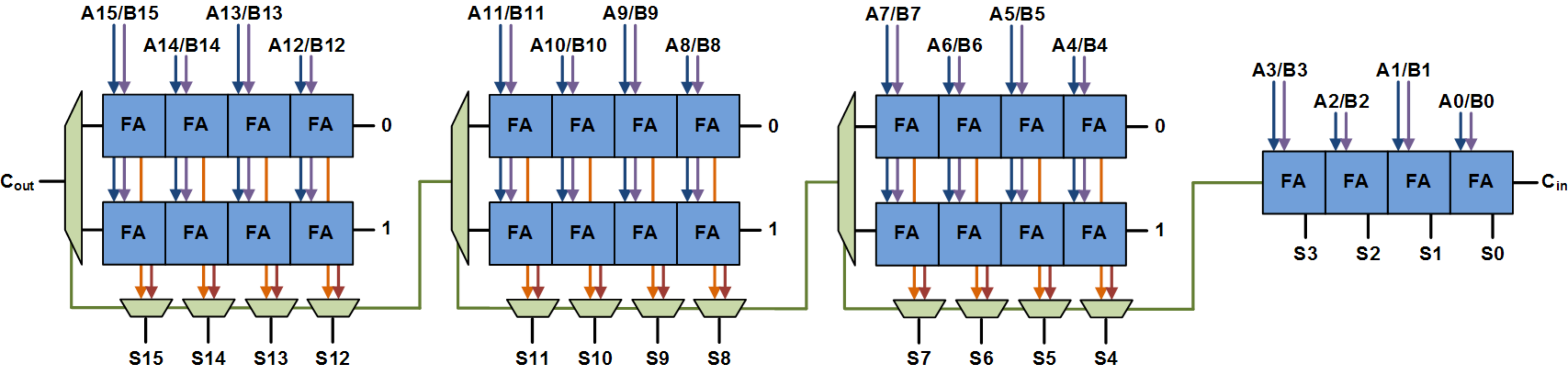
A15/B15
A14/B14
A13/B13
A12/B12
A11/B11
A10/B10
A9/B9
A8/B8
A7/B7
A6/B6
A5/B5
A4/B4
A3/B3
A2/B2
A1/B1
A0/B0
FA
0
1
C_{out}
C_{in}
S15
S14
S13
S12
S11
S10
S9
S8
S7
S6
S5
S4
S3
S2
S1
S0

Timing of Carry-Select Adder

For N-bit adder with k-bit blocks:

$$T_{CSA} = kT_{FA} + [(N/k) - 1]T_{MUX}$$

How to determine block size k?


A15/B15
A14/B14
A13/B13
A12/B12
A11/B11
A10/B10
A9/B9
A8/B8
A7/B7
A6/B6
A5/B5
A4/B4
A3/B3
A2/B2
A1/B1
A0/B0
FA
0
1
C_{out}
C_{in}
S15
S14
S13
S12
S11
S10
S9
S8
S7
S6
S5
S4
S3
S2
S1
S0

Timing of Carry-Select Adder

For N-bit adder with k-bit blocks:

$$T_{CSA} = kT_{FA} + [(N/k) - 1]T_{MUX}$$

If k is large, delay due to adders is large

If k is small, delay due to muxes is large

Timing of Carry-Select Adder

$T_{CSA} = kT_{FA} + [(N/k) - 1]T_{MUX}$

For fixed block size k, time still grows linearly with N

Make block size k function of number of bits N

$$k = \mathrm{sqrt}(N)$$

Timing of Carry-Select Adder

$T_{CSA} = \text{sqrt}(N)T_{FA} + [(N/\text{sqrt}(N)) - 1]T_{MUX}$

$T_{CSA} = \text{sqrt}(N)T_{FA} + [\text{sqrt}(N) - 1]T_{MUX}$

which, given our assumption about T_{MUX}, is approximately

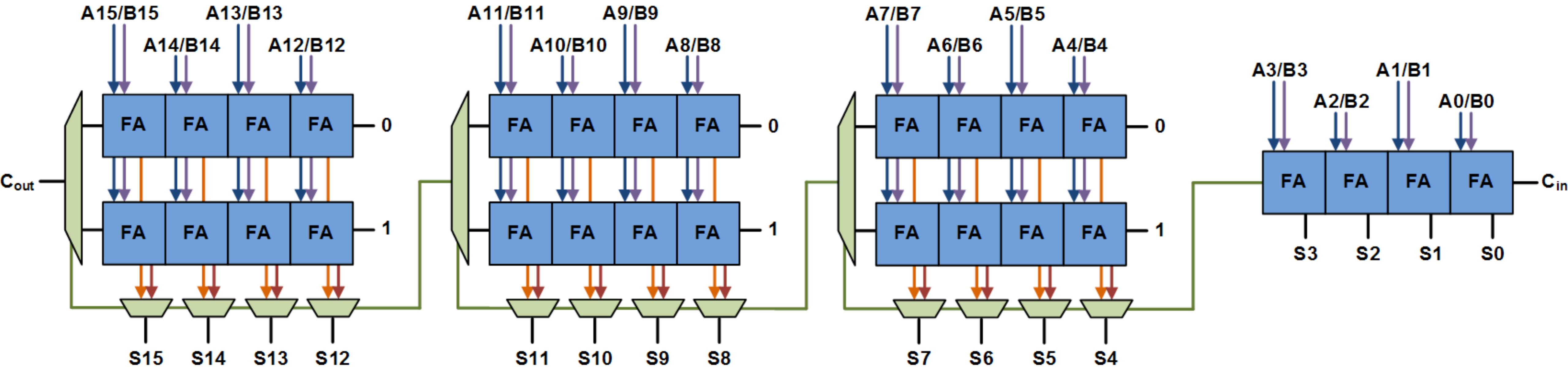
$$2\text{sqrt}(N)T_{FA}$$

Carry-select is not just faster than ripple-carry; it is *asymptotically* faster [O(sqrt(N))]

Further Improvement

For 16-bit adder, we can do slightly better still

Note that some of the hardware is “sitting idle” part of the time, just waiting for carry-in to arrive at mux



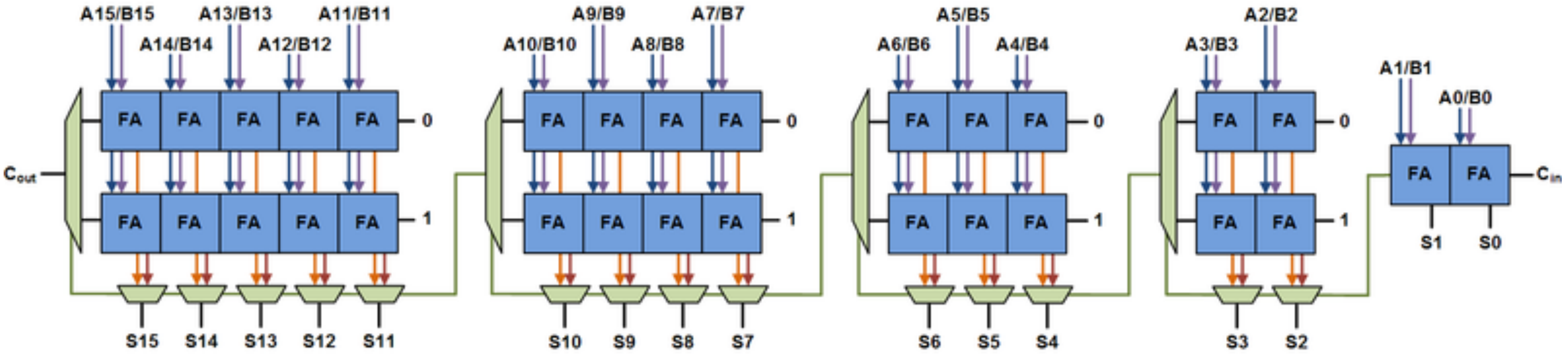
A15/B15
A14/B14
A13/B13
A12/B12
A11/B11
A10/B10
A9/B9
A8/B8
A7/B7
A6/B6
A5/B5
A4/B4
A3/B3
A2/B2
A1/B1
A0/B0
FA
0
1
C_{out}
C_{in}
S15
S14
S13
S12
S11
S10
S9
S8
S7
S6
S5
S4
S3
S2
S1
S0

Further Improvement

For 16-bit adder, we can do slightly better still

Note that some of the hardware is “sitting idle” part of the time, just waiting for carry-in to arrive at mux

By varying block size, we can ensure that no block is ever waiting idly


A15/B15
A14/B14
A13/B13
A12/B12
A11/B11
A10/B10
A9/B9
A8/B8
A7/B7
A6/B6
A5/B5
A4/B4
A3/B3
A2/B2
A1/B1
A0/B0
FA
0
1
Cout
Cin
S15
S14
S13
S12
S11
S10
S9
S8
S7
S6
S5
S4
S3
S2
S1
S0

Further Improvement

This takes time

$2T_{FA}$ (for the first two adders) + $4T_{MUX}$

which is approximately $6T_{FA}$

Not a big improvement over $7T_{FA}$, but an improvement

Varying block size does not result in asymptotic speedup

Credit

Images taken from “Carry-select adder” article on Wikipedia. Retrieved Feb. 3, 2021. Images in public domain.