CSU22022 Computer Architecture I

Fiveteenth Lecture - Datapath

Michael Manzke

2023-2024

Trinity College Dublin

Controlling a Datapath

- Figure 1 on the next slide depicts a simplified datapath.
- The destination decoder and the A and B bus MUX require three-bit select inputs (addresses).
- The Function Unit requires five bits to select ALU/Shift micro-ops.
- Three more bits are required to control the datapath:
  - Writing to the registers (RW)
  - MUX B (MB)
  - MUX D (MD)

Simplified Datapath

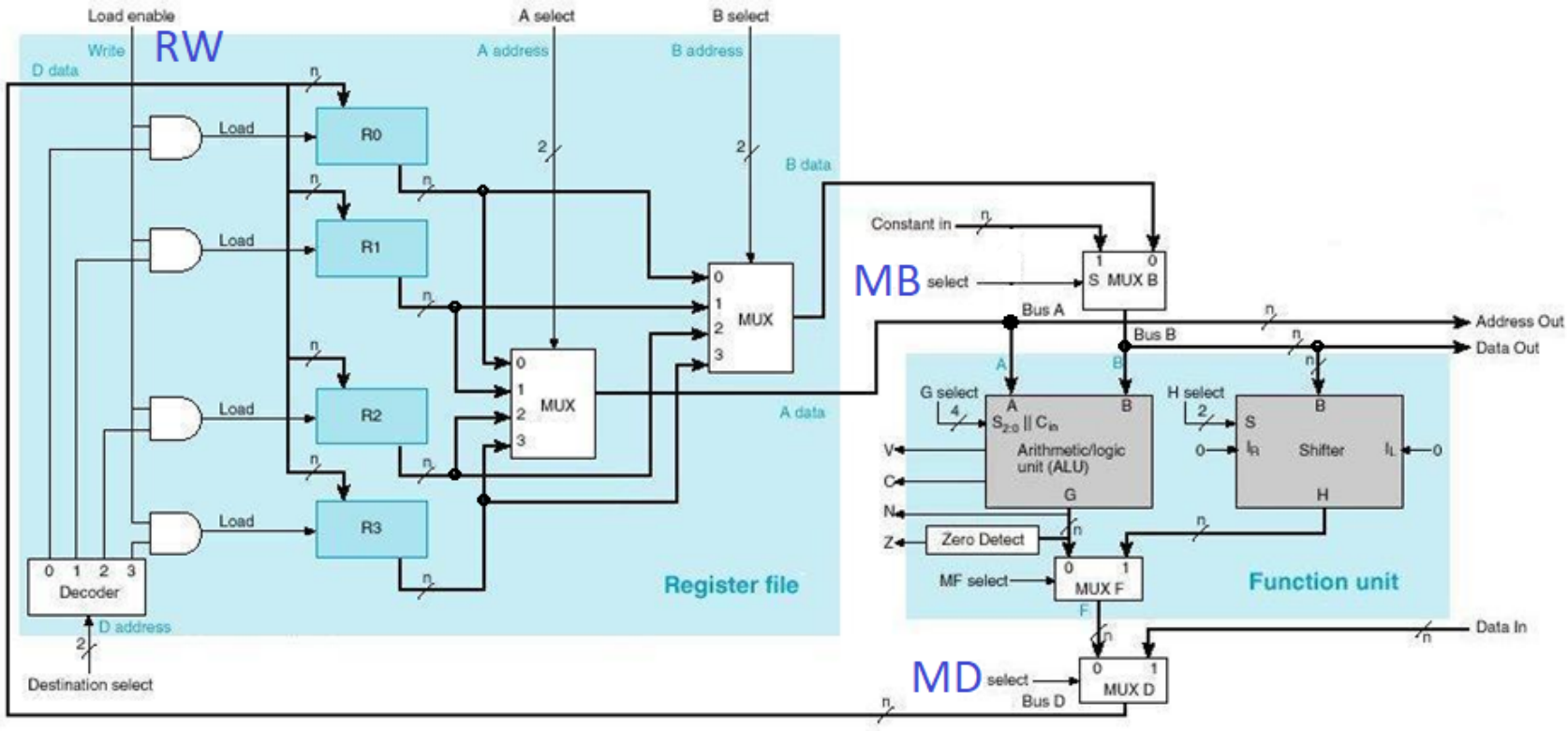


Figure 1: This simplified datapath has a register file with four registers.

Simplified Datapath with Memory

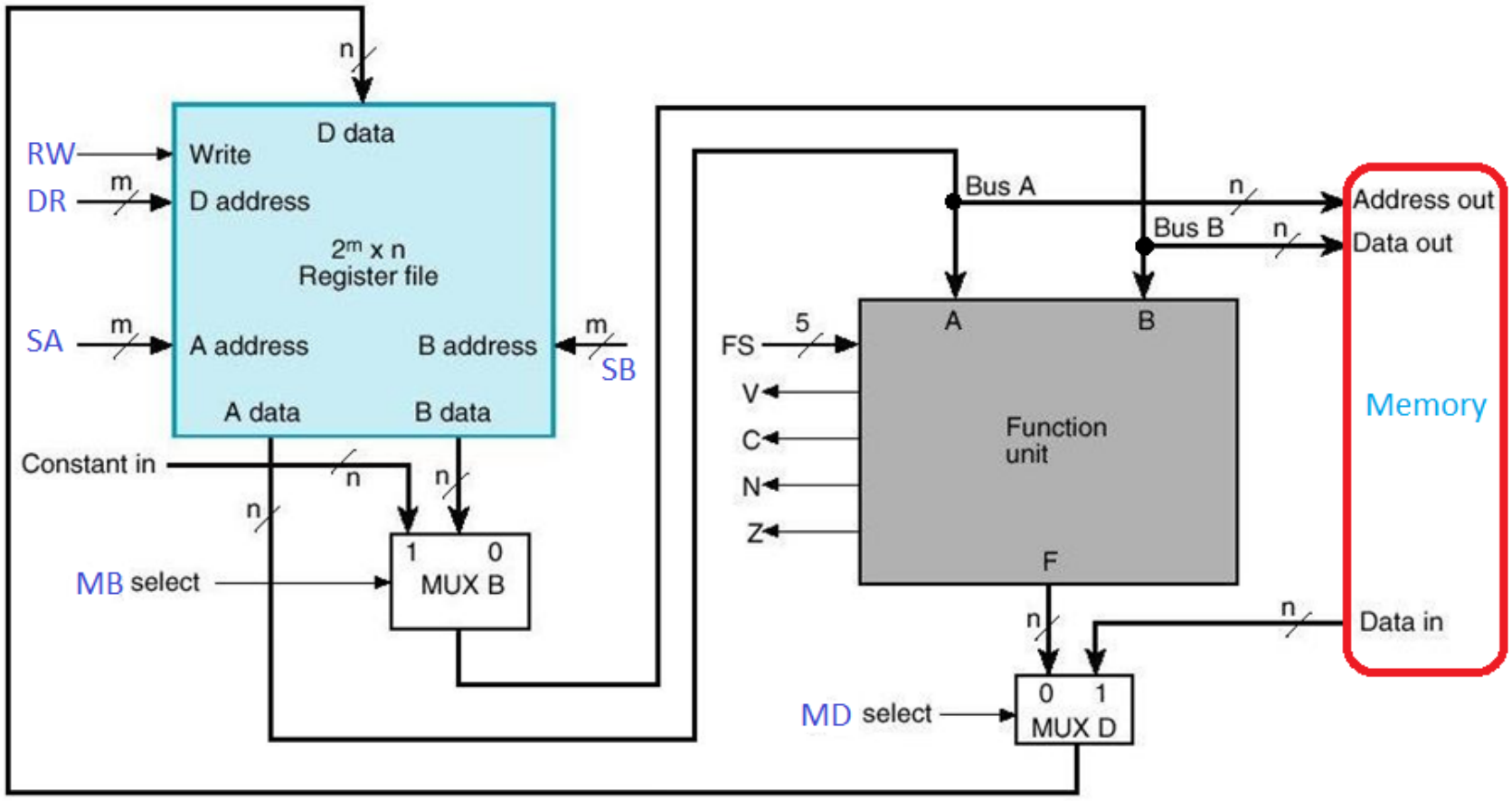


Figure 2: This figure shows how memory is connected to the simplified datapath.

Full Datapath VHDL Implementation

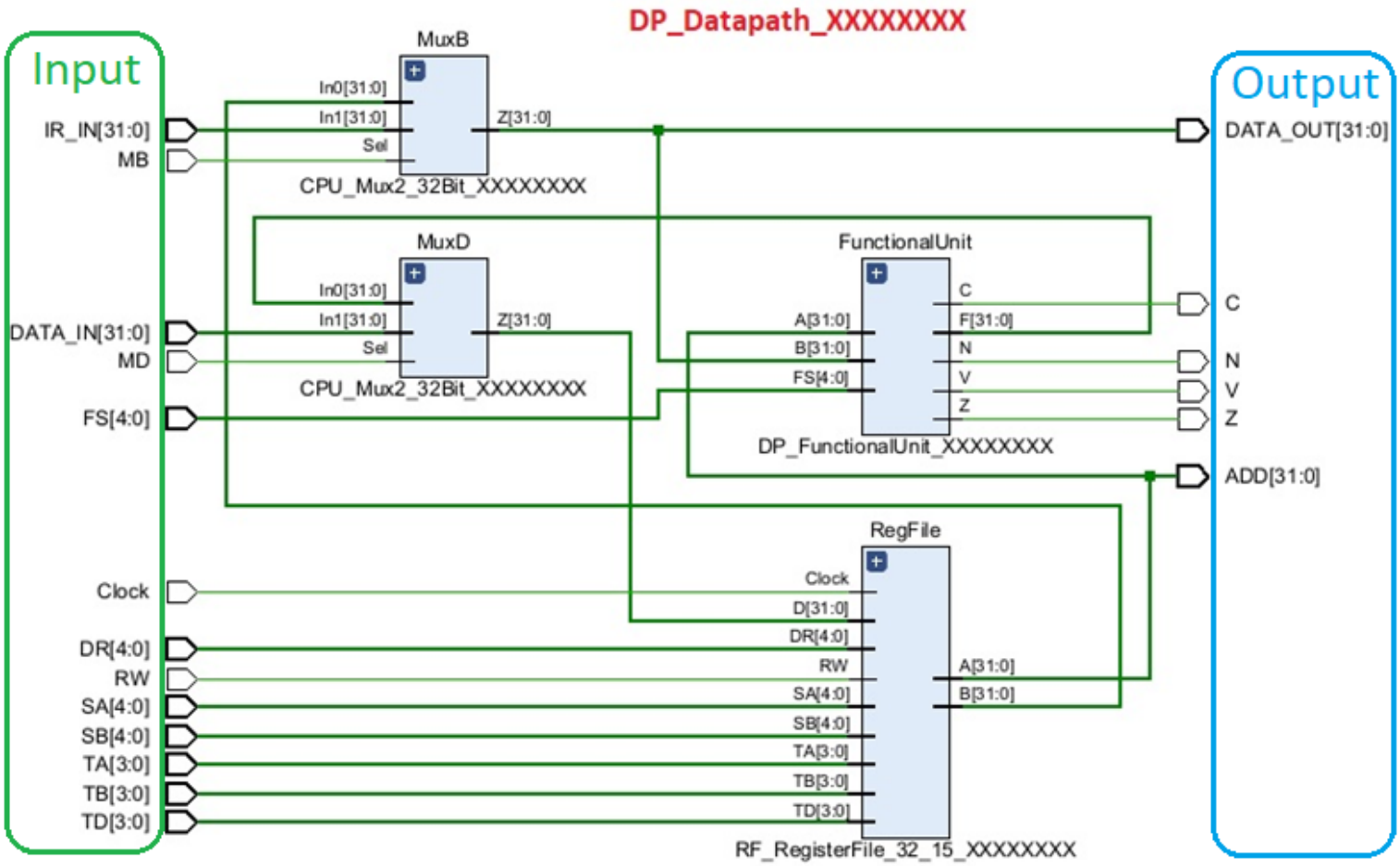
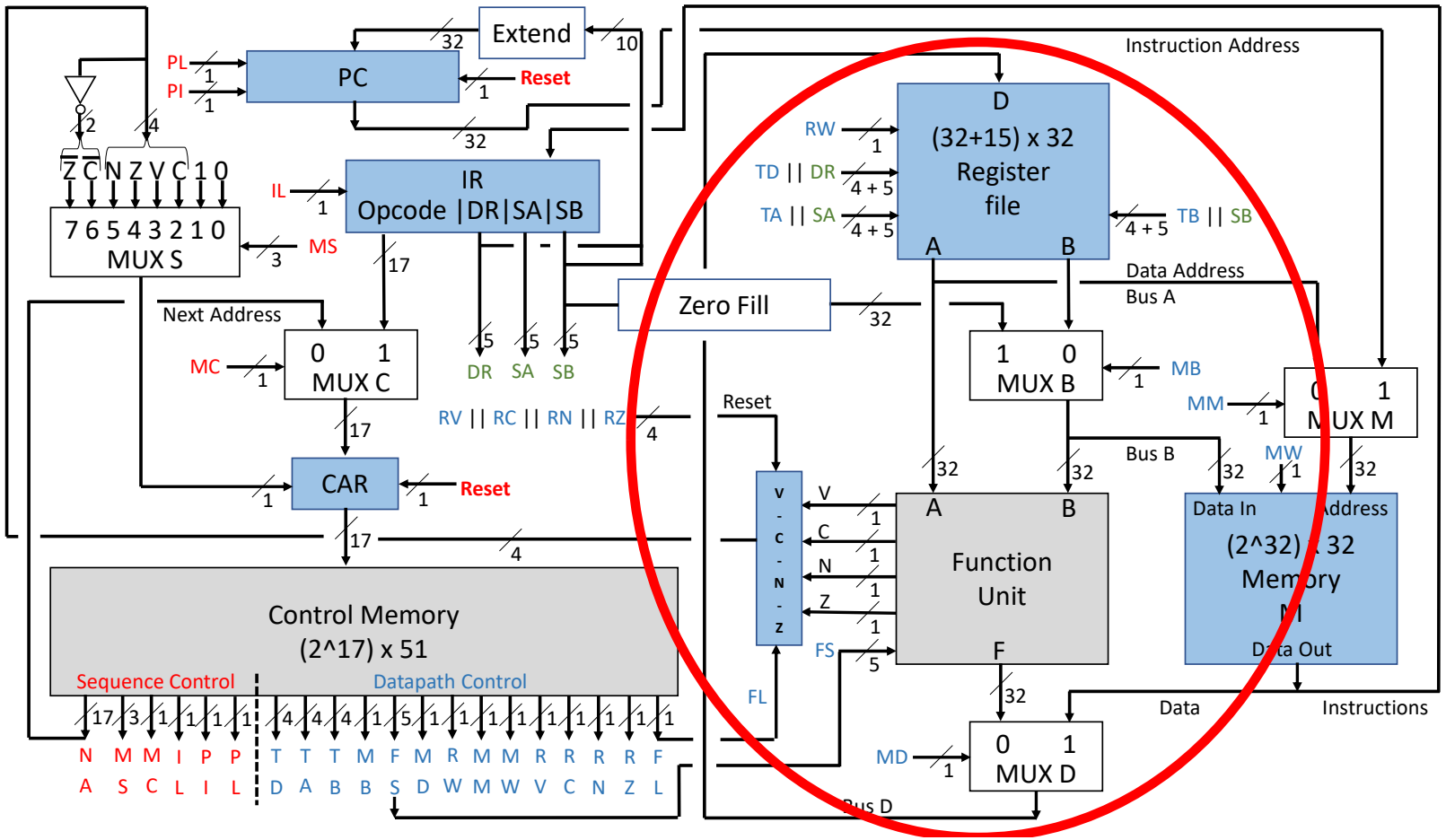


Figure 3: This figure depicts the full datapath implementation for our processor.

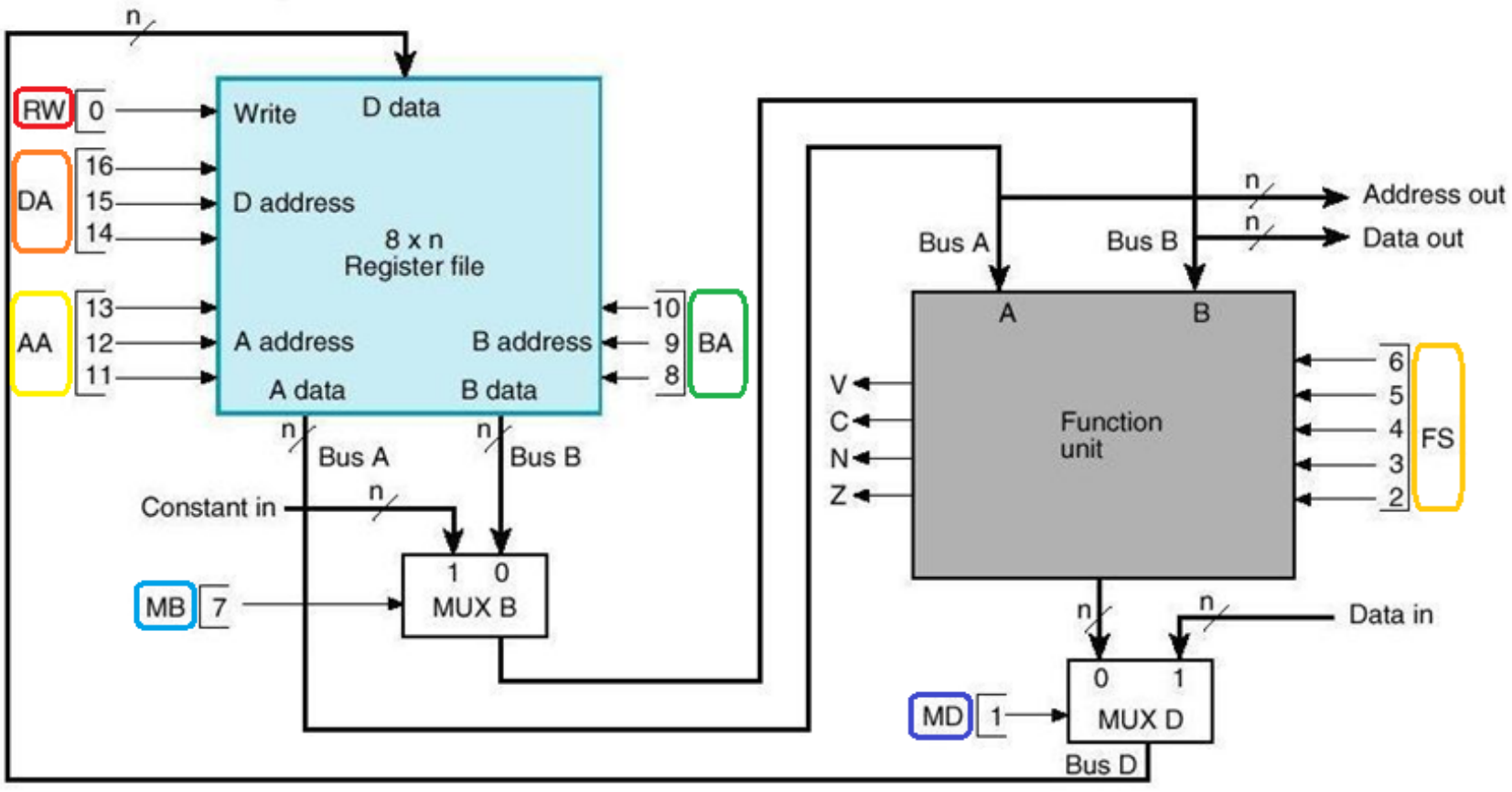
Datapath in the Context of the Processor

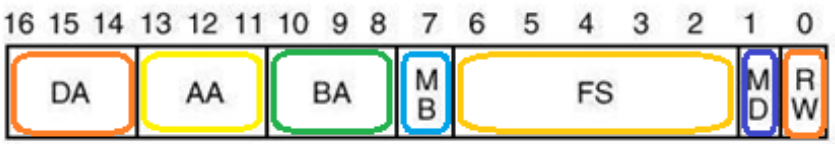


Control Word for the Datapath

The schematic on the next slide identifies all these control inputs and arranges them in a 17-bit vector called the Control Word.

Control Word for Simplified Datapath





The Control Word selects a Micro-operation

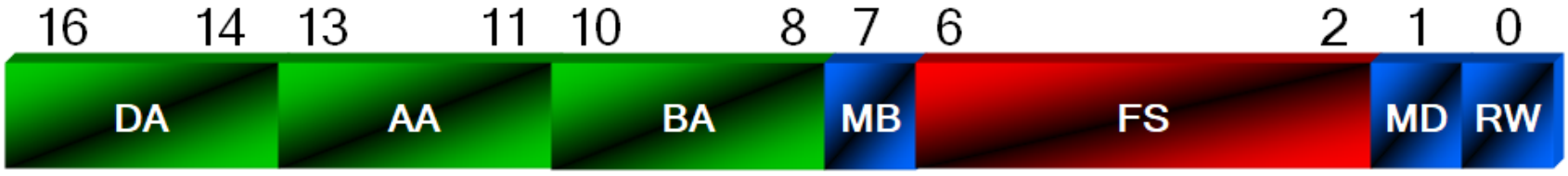


Figure 4: Note that for the Function Select FS we need to make explicit the relationship between its value and the micro-ops.

Function Unit Micro-operations

FS	MF	G	H		
00000	0	0000	XX	G=A	TRANSFER
00001	0	0001	XX	G=A+1	INCREMENT
00010	0	0010	XX	G=A+B	ADD
00011	0	0011	XX	G=A+B+1	ADD WITH C
00100	0	0100	XX	G=A$\overline{B}$	A plus 1's COMPLEMENT
00101	0	0101	XX	G=A$\overline{B}$+1	SUBTRACT
00110	0	0110	XX	G=A-1	DECREMENT
00111	0	0111	XX	G=A	TRANSFER
01000	0	1000	XX	G=A$\wedge$B	AND
01010	0	1010	XX	G=A$\vee$B	OR
01100	0	1100	XX	G=A$\oplus$B	XOR
01110	0	1110	XX	G=$\overline{A}$	NOT
10000	1	XXXX	00	G=B	TRANSFER
10100	1	XXXX	01	G=srB	SHIFT RIGHT
11000	1	XXXX	10	G=slB	SHIFT LEFT

FS, MF, G, and H

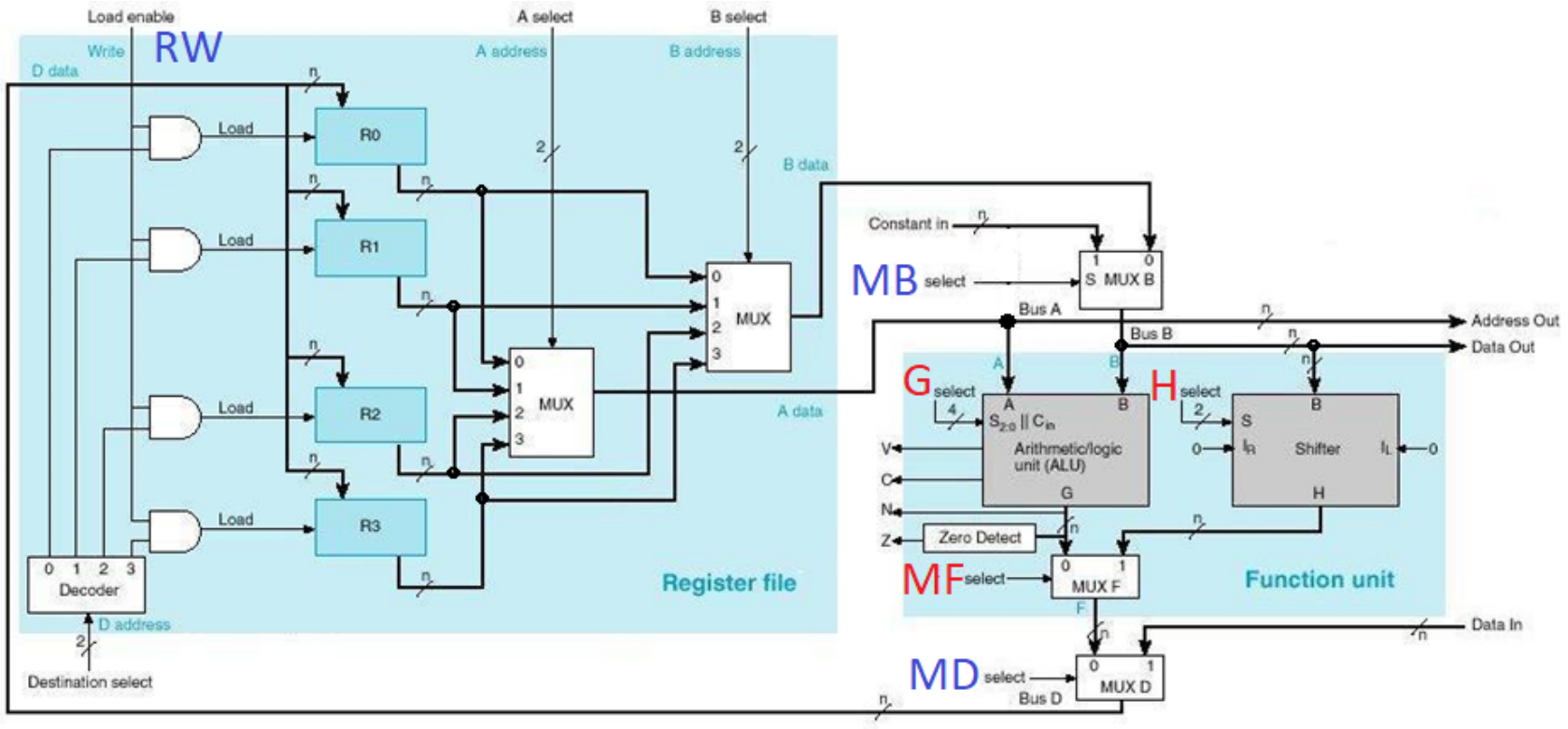
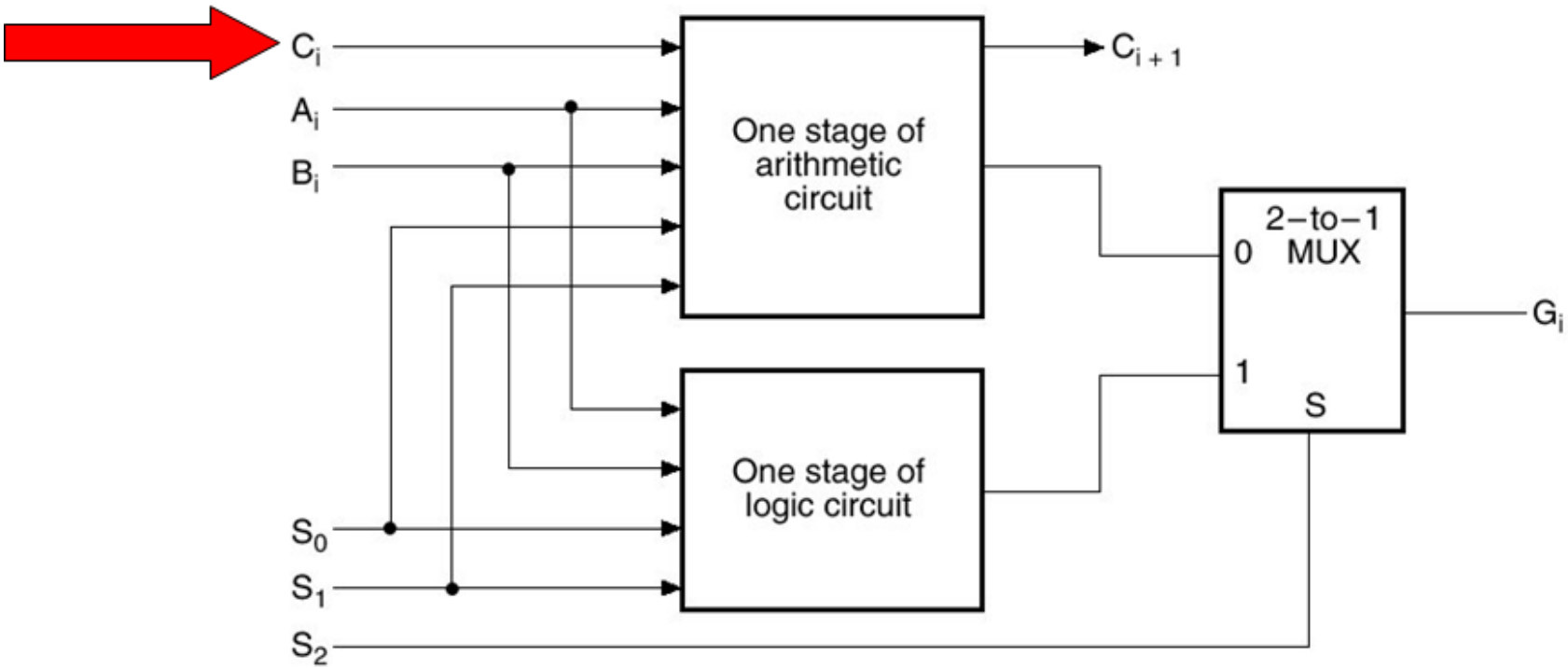
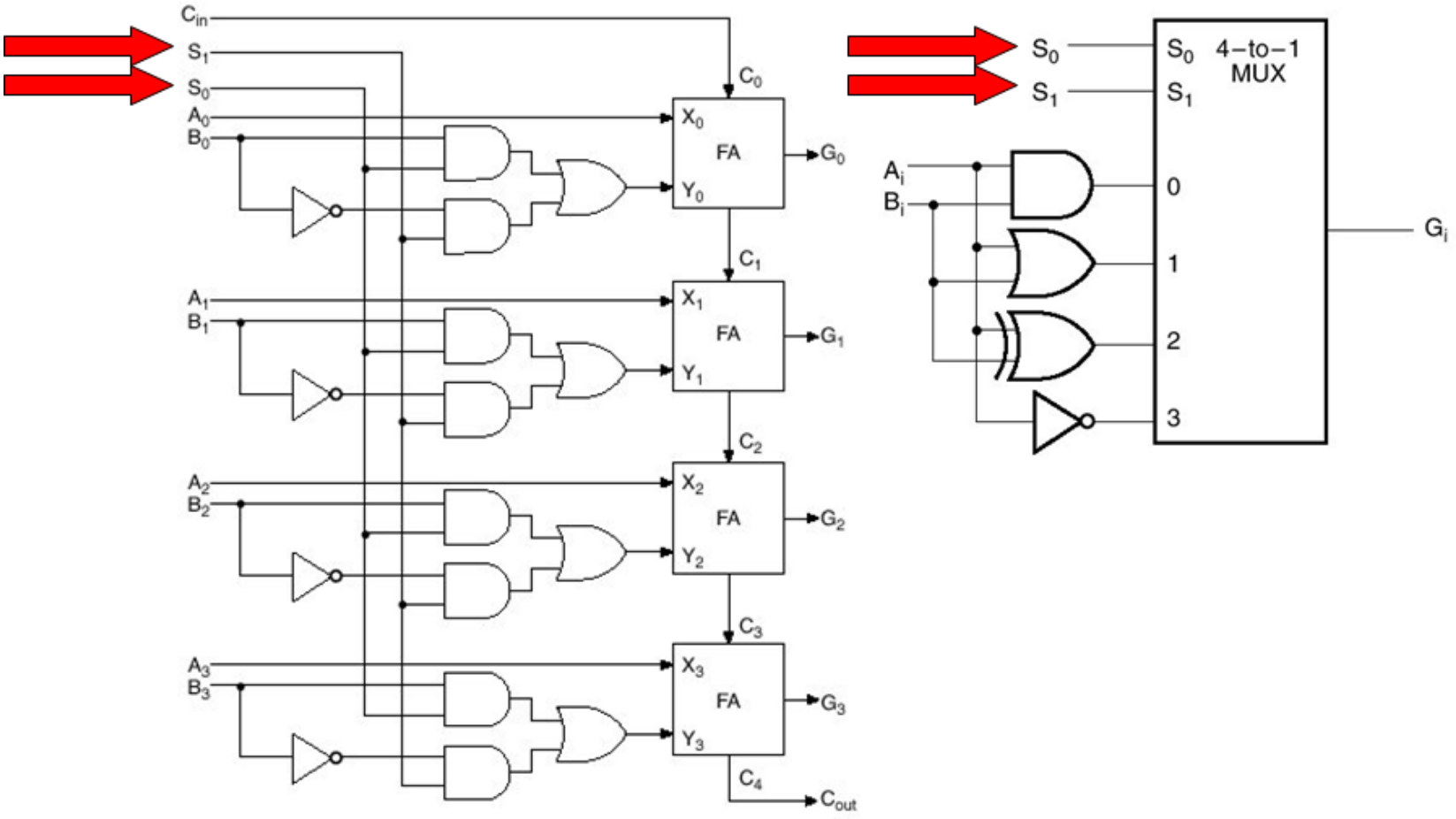


Figure 5: This figure show how FS, MF, G, and H determine the micro-ops of the Function Unit.

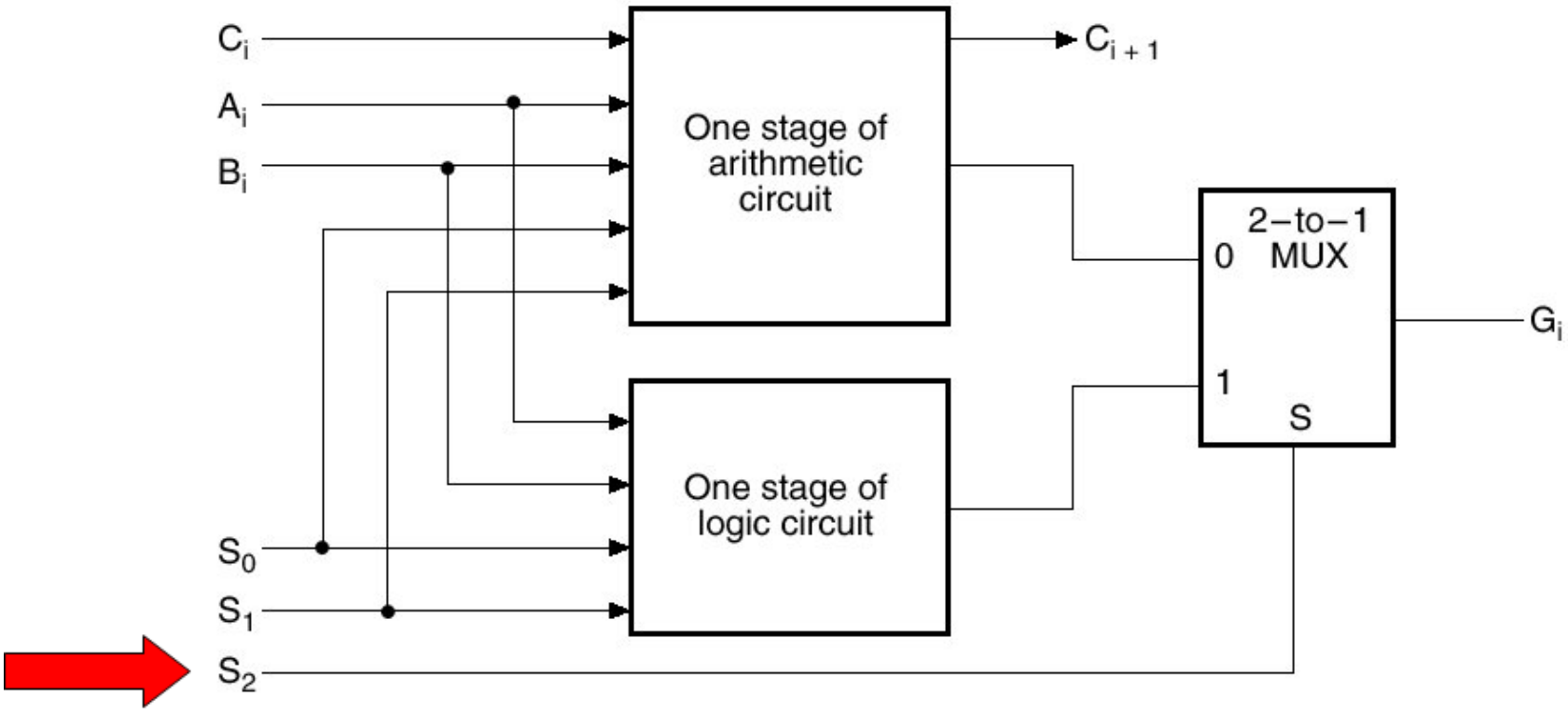
$FS(0) = C_{in}$


C_i
A_i
B_i
S_0
S_1
S_2
One stage of arithmetic circuit
One stage of logic circuit
C_{i+1}
2–to–1 MUX
0
1
S
G_i

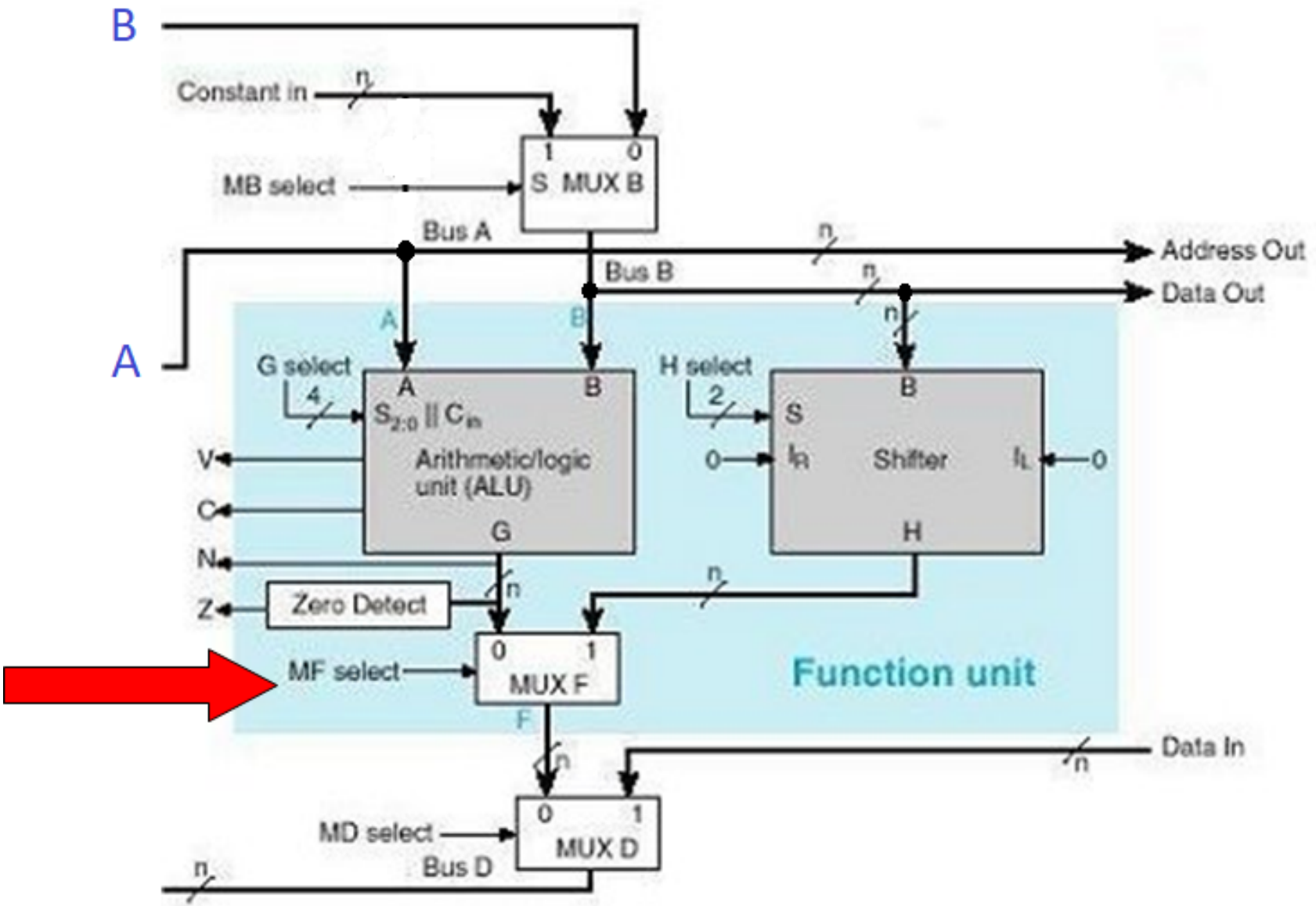
$FS(2) = S_1$, $FS(1) = S_0$


Cin
S1
S0
A0
B0
C0
X0
FA
Y0
G0
C1
A1
B1
X1
FA
Y1
G1
C2
A2
B2
X2
FA
Y2
G2
C3
A3
B3
X3
FA
Y3
G3
C4
Cout
S0
S1
S0
S1
4–to–1
MUX
Ai
Bi
0
1
2
3
Gi

$FS(3) = S_2$


C_i
A_i
B_i
S_0
S_1
S_2
One stage of arithmetic circuit
One stage of logic circuit
C_{i+1}
2–to–1 MUX
0
1
S
G_i

$FS(4) = S_3 = MF$


B
Constant in
n
1
0
S MUX B
MB select
Bus A
n
Address Out
Bus B
n
Data Out
n
A
A
B
G select
4
A
B
$S_{2:0}$ || C_{in}
Arithmetic/logic unit (ALU)
G
H select
2
B
S
0
I_R
Shifter
I_L
0
H
V
C
N
Z
Zero Detect
n
n
0
1
MF select
MUX F
F
Function unit
n
Data In
n
0
1
MD select
MUX D
Bus D
n