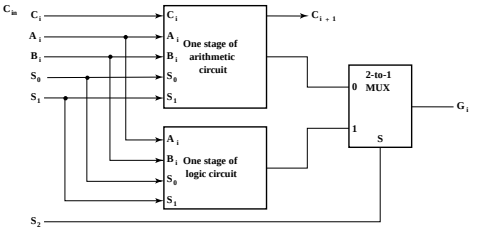
ECE 3401 Lecture 18

Microprogramming (I)

Arithmetic Logic Unit (ALU)

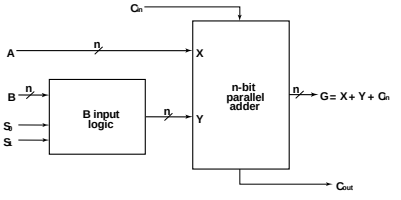
- Decompose the ALU into:
  - An arithmetic circuit & A logic circuit
  - A selector to pick between the two circuits



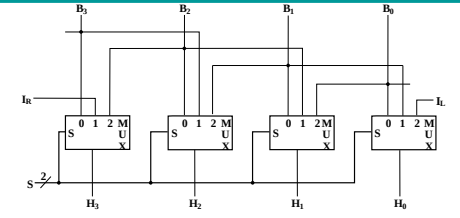
Arithmetic Circuit Design

- Arithmetic circuit design
  - Decompose the arithmetic circuit into:
    - An n-bit parallel adder
    - A block of logic that selects four choices for the B input to the adder
- There are only four functions of B to select as Y in G = A + Y + C_{in}:

	C_{in} = 0	C_{in} = 1
0	G = A	G = A + 1
B	G = A + B	G = A + B + 1
$\overline{B}$	G = A + $\overline{B}$	G = A + $\overline{B}$ + 1
1	G = A – 1	G = A

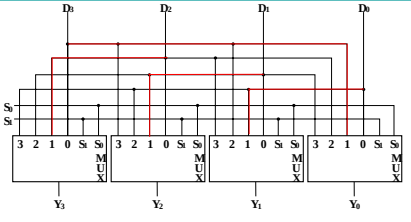


4-Bit Basic Left/Right Shifter



- Serial Inputs:
  - I_R for right shift
  - I_L for left shift
- Shift Functions:
  (S_1, S_0) = 00 Pass B unchanged
  01 Right shift
  10 Left shift
  11 Unused

Barrel Shifter

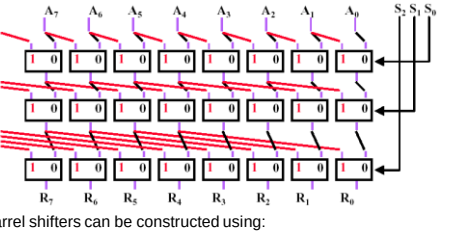


- A rotate is a shift in which the bits shifted out are inserted into the positions vacated
- The circuit rotates its contents left from 0 to 3 positions depending on S:
  S = 00 position unchanged
  S = 01 rotate left by 1 positions
  S = 10 rotate left by 2 positions
  S = 11 rotate left by 3 positions

Combinational Shifter from MUXes

Basic Building Block

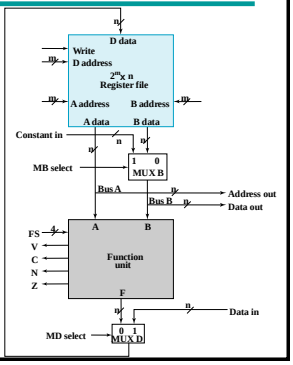
8-bit right shifter



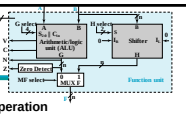
- Example 8-bit:
  - Layer 1 shifts by 0, 4
  - Layer 2 shifts by 0, 2
  - Layer 3 shifts by 0, 1
- Large barrel shifters can be constructed using:
  - Layers of multiplexers
  - 2 - dimensional array circuits designed at the electronic level

Datapath Representation

- In the register file:
  - Select inputs for multiplexers => A address & B address
  - Decoder input => D address
  - Load enable => write
  - Input data to the registers => D data
  - Multiplexer outputs => A data & B data
- The register file now appears like a memory based on clocked flip-flops
- FS?



Definition of Function Unit Select (FS) Codes



FS(3:0)	MF Select	G Select(3:0)	H Select(1:0)	Microoperation
0000	0	0000	XX	$F \leftarrow A$
0001	0	0001	XX	$F \leftarrow A + 1$
0010	0	0010	XX	$F \leftarrow A + B$
0011	0	0011	XX	$F \leftarrow A + \overline{B} + 1$
0100	0	0100	XX	$F \leftarrow A + \overline{B}$
0101	0	0101	XX	$F \leftarrow A + \overline{B} + 1$
0110	0	0110	XX	$F \leftarrow A - 1$
0111	0	0111	XX	$F \leftarrow A$
1000	0	1X00	XX	$F \leftarrow A \wedge B$
1001	0	1X01	XX	$F \leftarrow A \vee B$
1010	0	1X10	XX	$F \leftarrow \overline{A} \oplus B$
1011	0	1X11	XX	$F \leftarrow \overline{A}$
1100	1	XXXX	00	$F \leftarrow B$
1101	1	XXXX	01	$F \leftarrow$ sr B
1110	1	XXXX	10	$F \leftarrow$ sl B

Boolean Equations:

$MFS = F_3 F_2$

$GS_i = F_i$

$HS_i = F_i$

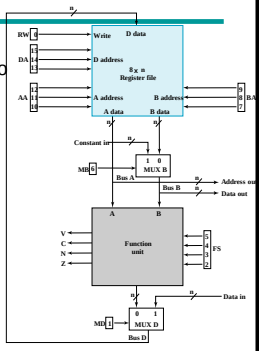
The Control Word

- The datapath has many control input signals, can be organized into a *control word*
- To execute a microinstruction, we apply control word values for a clock cycle.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DA			AA			BA			MB	FS				MD	RW

Control word

DA – D Address, AA – A Address
BA – B Address, MB – Mux B
FS – Function Select, MD – Mux D
RW – Register Write



Control Word Encoding

DA, AA, BA		MB		FS		MD		RW	
Function	Code	Function	Code	Function	Code	Function	Code	Function	Code
R0	000	Register	0	$F \leftarrow A$	0000	Function	0	No write	0
R1	001	Constant	1	$F \leftarrow A + 1$	0001	Data In	1	Write	1
R2	010			$F \leftarrow A + B$	0010				
R3	011			$F \leftarrow A + \overline{B} + 1$	0011				
R4	100			$F \leftarrow A + \overline{B}$	0100				
R5	101			$F \leftarrow A + \overline{B} + 1$	0101				
R6	110			$F \leftarrow A - 1$	0110				
R7	111			$F \leftarrow A$	0111				
				$F \leftarrow A \wedge B$	1000				
				$F \leftarrow A \vee B$	1001				
				$F \leftarrow A \oplus B$	1010				
				$F \leftarrow \overline{A}$	1011				
				$F \leftarrow B$	1100				
				$F \leftarrow$ sr B	1101				
				$F \leftarrow$ sl B	1110				

Microoperations for the Datapath – Symbolic & Binary Representation

Micro-operation	DA	AA	BA	MB	FS	MD	RW
$R1 \leftarrow R2 - R3$	R1	R2	R3	Register	$F = A + \overline{B} + 1$	Function	Write
$R4 \leftarrow$ sl R6	R4	—	R6	Register	$F =$ sl B	Function	Write
$R7 \leftarrow R7 + 1$	R7	R7	—	Register	$F = A + 1$	Function	Write
$R1 \leftarrow R0 + 2$	R1	R0	—	Constant	$F = A + B$	Function	Write
Data out $\leftarrow R3$	——		R3	Register	—	—	No Write
$R4 \leftarrow$ Data in	R4	——		—	—	Data in	Write
$R5 \leftarrow 0$	R5	R0	R0	Register	$F = A \oplus B$	Function	Write

Micro-operation	DA	AA	BA	MB	FS	MD	RW
$R1 \leftarrow R2 - R3$	001	010	011	0	0101	0	1
$R4 \leftarrow$ sl R6	100	XXX	110	0	1110	0	1
$R7 \leftarrow R7 + 1$	111	111	XXX	0	0001	0	1
$R1 \leftarrow R0 + 2$	001	000	XXX	1	0010	0	1
Data out $\leftarrow R3$	XXX	XXX	011	0	XXXX	X	0
$R4 \leftarrow$ Data in	100	XXX	XXX	X	XXXX	1	1
$R5 \leftarrow 0$	101	000	000	0	1010	0	1

Datapath Simulation