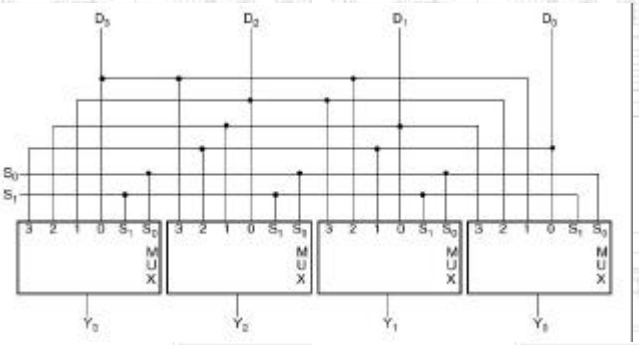
Shift Operations

- By controlling IR and IL with multiplexers it is straightforward to adapt this circuit to perform arithmetic shifts, constructive and destructive shifts, Rotates and rotate-then-carry operation

4-Bit Barrel Shifter



Multiple Shifts

- If multiple shifts are required we wire them into multiplexers that have an input for every bit on the bus to obtain a Barrel shifter (Previous slide).

S_1 S_2	Y_3 Y_2 Y_1 Y_0	Micro-ops
0 0	D_3 D_2 D_1 D_0	No Rotate
0 1	D_2 D_1 D_0 D_3	Rotate One
1 0	D_1 D_0 D_3 D_2	Rotate Two
1 1	D_0 D_3 D_2 D_1	Rotate Three

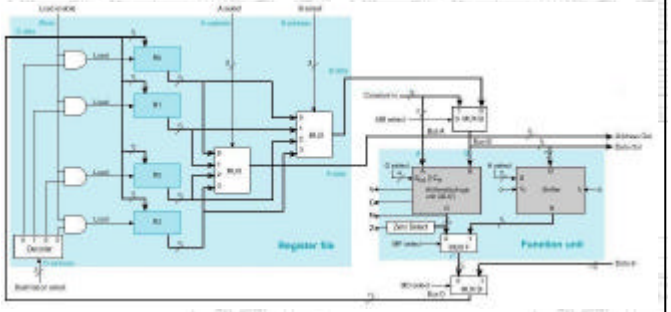
Problem 1-24

- Please do problem 1-24 at the end of chapter 7 in the Mano and Kime textbook.
- Seven out of these 24 problem will be demonstrated during the tutorial.
- Make sure you have read page 148 to 158 chapter 3 and page 224 to 231 chapter 4. The VHDL aspect was not covered in 1BA4. The rest of chapters 1-6 are prerequisites.

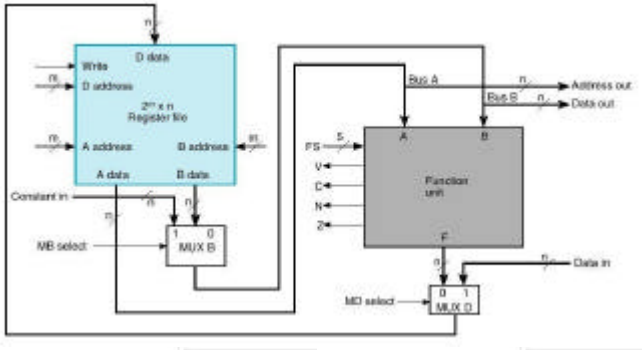
Controlling a Datapath
The Control Word

- The figure on the next slide is an updated version of our introductory datapath (4th Lecture, Page 12) where the register file has been expanded to a more realistic eight n-bit registers.
- Consequently the destination decoder and A and B bus MUX require three-bit select input.
- The Function Unit still requires five bits to select ALU/Shift micro-ops.
- Three more bits are required to control:
  - Writing to the registers (RW)
  - MUX B (MB)
  - MUX D (MD)

4th Lecture - Page 12



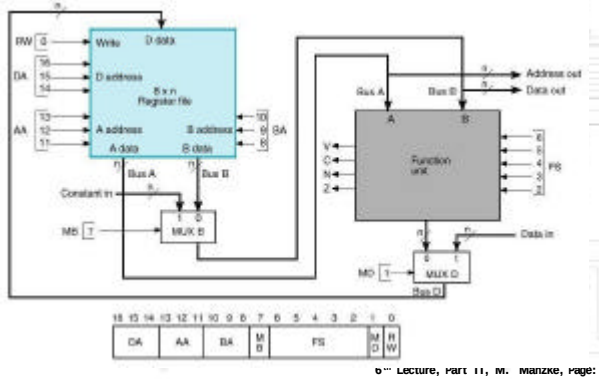
Updated Datapath



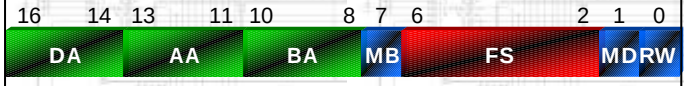
Control Word

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

Datapath and the Control Word



The Control Word Specifies One Micro-ops

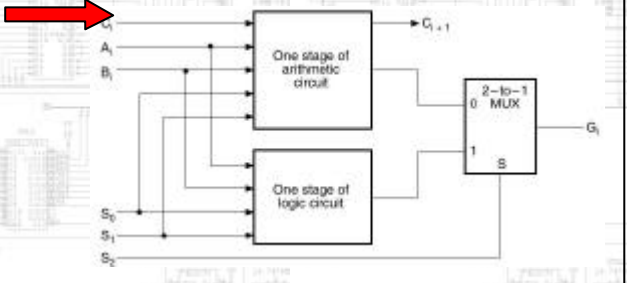


- Note that for the function select FS we need to make explicit the relationship between its value and the micro-ops.
  - See next slide for details.

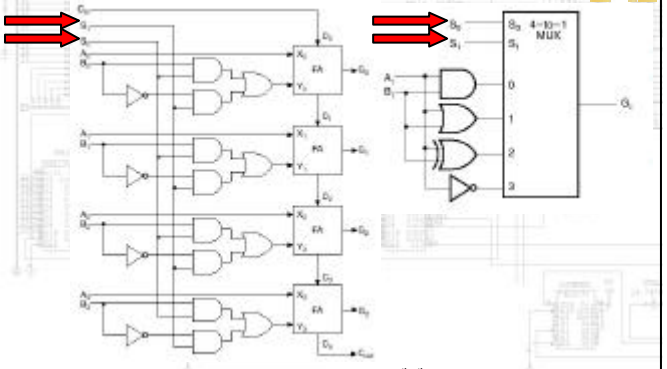
G Select, H Select and MF Select determine the FS code

FS	MF Select	G Select	H Select	Output	
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 + B‾	A plus 1's C.B
00101	0	0101	XX	G = A + B +1	SUBTRACT
00110	0	0110	XX	G = A - 1	DECREMENT
00111	0	0111	XX	G = A	TRANSFER
01000	0	1000	XX	G = A Ù B	AND
01010	0	1010	XX	G = A Ú B	OR
01100	0	1100	XX	G = A‾Å B	XOR
01110	0	1110	XX	G = A‾	NOT
10000	1	XXXX	00	G = B	TRANSFER
10100	1	XXXX	01	G = sr B	SHIFT RIGTH
11000	1	XXXX	10	G = sl B	SHIFT LEFT

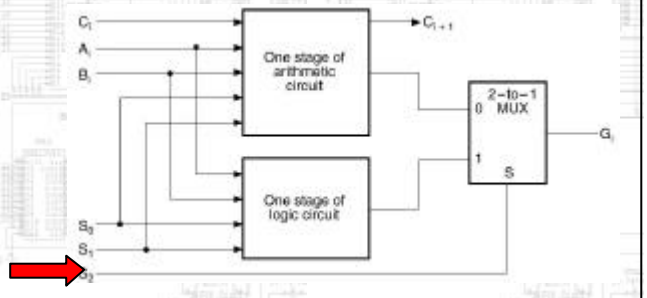
FS[2]= C_{in}



FS[3∧4]=S_0S_1



FS[5]=S_2



FS[6]=MF select