I completed this assignment entirely on my own, except for discussions with Jiamin Xu

Ethan Wong
CSMS1A

Homework #5

Display : Hours Minutes Seconds
HH M1 M0 S1 S0

Definitions

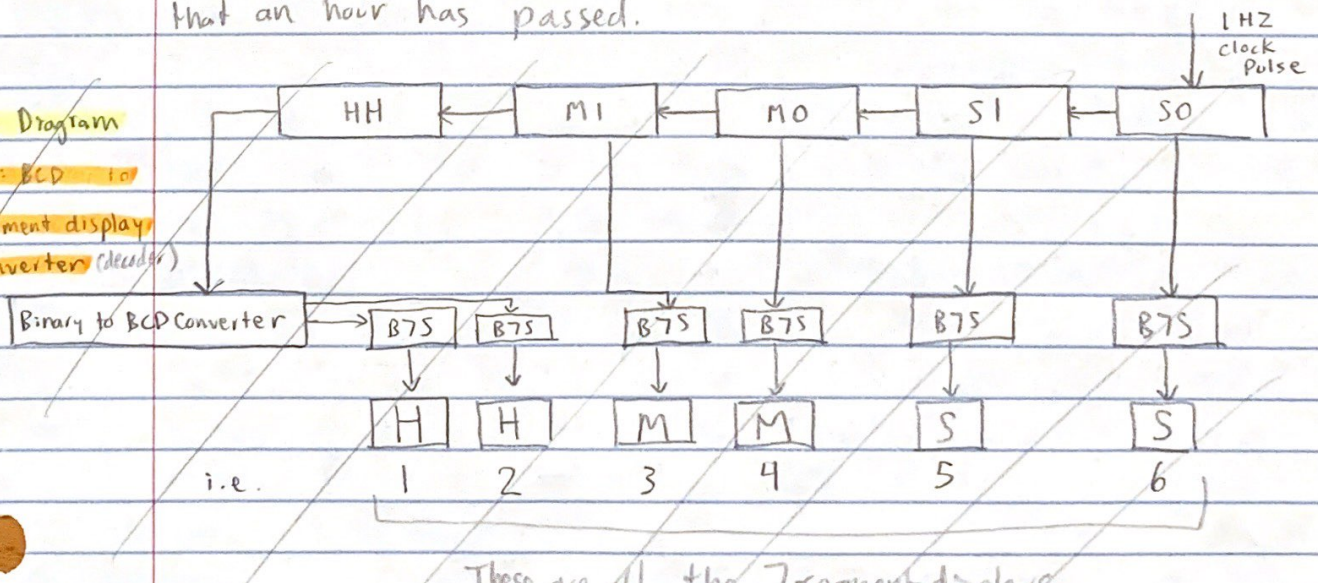
HH : 4-bit counter that counts from 1 - 12 -> 4 input 7 segment display
M1 : 3-bit counter that counts from 0 - 5
M0 : 4-bit counter that counts from 0 - 9
S1 : 3-bit counter that counts from 0 - 5
S0 : 4-bit counter that counts from 0 - 9

Logic

The idea is to have a 1 Hz clock pulse that gets applied to the S0 counter once after every second. S0 will count up to 9 (1001_2) and then reset to 0 (0000_2). When S0 becomes 0, a clock signal is generated to S1, which counts from 0 - 5. When S1 becomes 5 (101_2) and S0 becomes 9, the next clock pulse sets S1 and S0 back to zero. After this, the clock signal is generated and applied to M0. M0 and M1 basically work in the same fashion as S0 and S1 since seconds and minutes both count from 0 - 59. When M1 is set back to zero, a clock signal is generated to HH to signify that an hour has passed.

Block Diagram

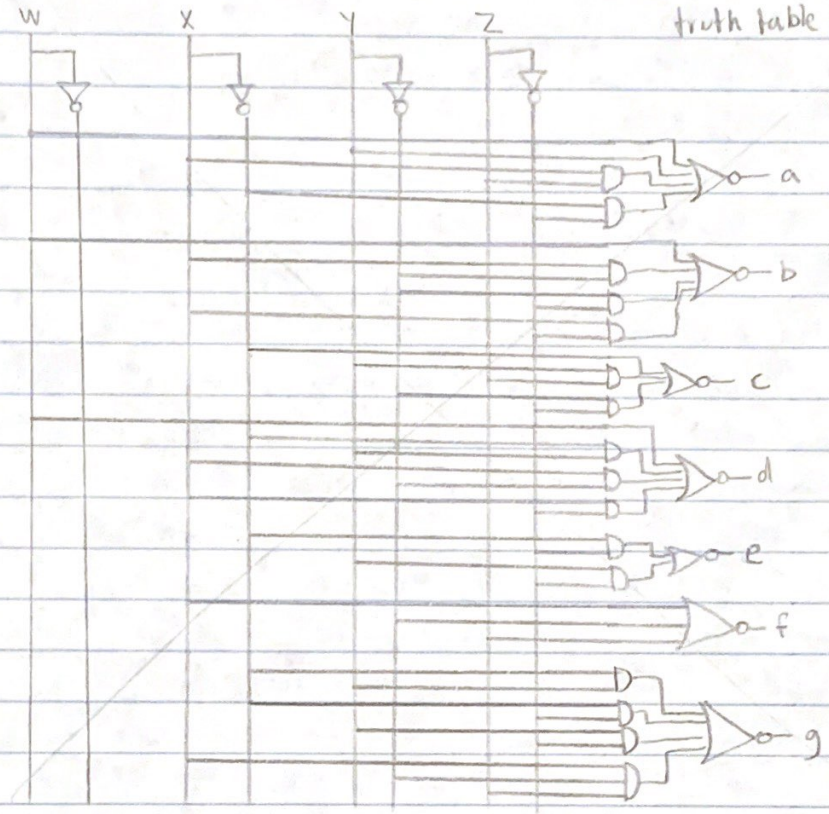
B7S : BCD to 7 segment display converter (decoder)



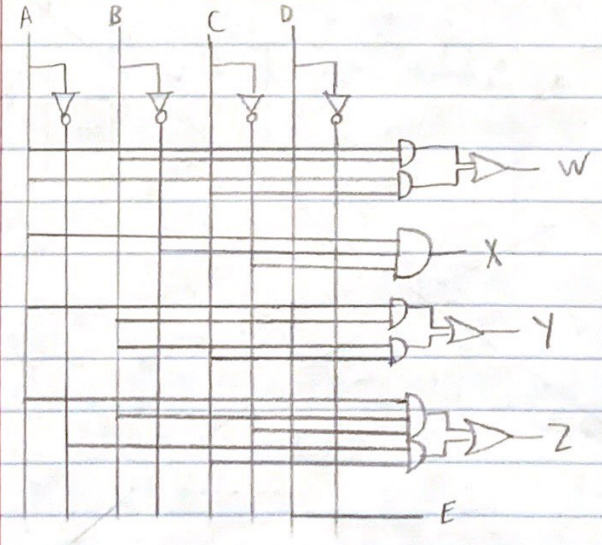
These are all the 7-segment displays (created in assignment 2)

Circuit for BCD to 7-segment display decoder

This is based off the truth table from HW #2, so I won't rewrite the truth table / K-maps/expressions



Circuit for Binary to BCD



$W = (A \cdot B) + (A \cdot C)$

$X = A \cdot B' \cdot C'$

$Y = (A' \cdot B) + (B \cdot C)$

$Z = (A \cdot B \cdot C') + (A' \cdot C)$

$E = D$

Truth tables and K-maps are on the next page.

(3)

Truth Table, Karnaugh Maps for Binary to BCD

Input: Binary Code				Output: BCD Code				
A	B	C	D	W	X	Y	Z	E
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	1	1	0	0	0	1	1
0	1	0	0	0	0	1	0	0
0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1
1	0	0	0	0	1	0	0	0
1	0	0	1	0	1	0	0	1
1	0	1	0	1	0	0	0	0
1	0	1	1	1	0	0	0	1
1	1	0	0	1	0	0	1	0
1	1	0	1	1	0	0	1	1
1	1	1	0	1	0	1	0	0
1	1	1	1	1	0	1	0	1

W:

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	1	1

$W = (A \cdot B) + (A \cdot C)$

X:

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	0	0	0
10	1	1	0	0

$X = A \cdot B' \cdot C'$

Y:

AB \ CD	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	1	1
10	0	0	0	0

$Y = (A' \cdot B) + (B \cdot C)$

Z:

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	1	1	0	0
10	0	0	0	0

$Z = (A \cdot B \cdot C') + (A' \cdot C)$

E:

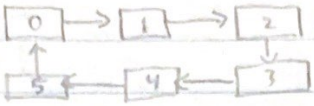
AB \ CD	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

$E = D$

(4)

0-5 counter (used for M1, S1)

(1)



(2) state table

current state	0	1	2	3	4	5
next state	1	2	3	4	5	0

(3) state reduction: none

(4) # of flip flops: 3

(5) state assignment: 0-000 1-001 2-010 3-011 4-100 5-101

(6) Type of flip flop: JK flip flop (most flexible)

(7) Full state table with flip-flop excitations, K-maps

	current state A B C	next state A B C	flip flop excitations JA KA JB KB JC KC
0	0 0 0	0 0 1	0 X 0 X 1 X
1	0 0 1	0 1 0	0 X 1 X X 1
2	0 1 0	0 1 1	0 X X 0 1 X
3	0 1 1	1 0 0	1 X X 1 X 1
4	1 0 0	1 0 1	X 0 0 X 1 X
5	1 0 1	0 0 0	X 1 0 X X 1

JA

a \ bc	00	01	11	10
0	0	0	1	0
1	X	X	X	X

JA = B·C

KA

a \ bc	00	01	11	10
0	X	X	X	X
1	0	1	X	X

KA = C

JB

a \ bc	00	01	11	10
0	0	1	X	X
1	0	0	X	X

JB = A'·C

KB

a \ bc	00	01	11	10
0	X	X	1	0
1	X	X	X	X

kB = C

JC

a \ bc	00	01	11	10
0	1	X	X	1
1	1	X	X	X

JC = 1

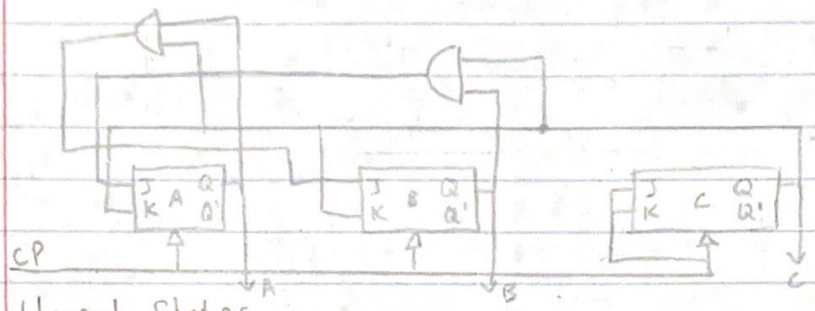
KC

a \ bc	00	01	11	10
0	X	1	1	X
1	X	1	X	X

KC = 1

(5)

(8) Circuit Diagram

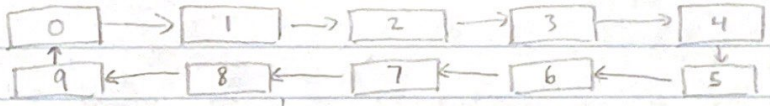


(9) Unused States

Current State			Flip-Flop Excitations						Next State			Stable?
A	B	C	JA	KA	JB	KB	JC	KC	A	B	C	Yes / No
1	1	0	0	0	0	0	1	1	1	1	1	Yes
1	1	1	1	1	0	1	1	1	0	0	0	Yes

0-9 counter (used for M0, S0)

(1)



(2)

State table	current state	0	1	2	3	4	5	6	7	8	9
	next state	1	2	3	4	5	6	7	8	9	10

(3) State reduction: none

(4) # of flip-flops : 4

(5) State assignment : 0 - 0000 1 - 0001 2 - 0010 3 - 0011
4 - 0100 5 - 0101 6 - 0110 7 - 0111 8 - 1000 9 - 1001
Unused states : 10 - 1010 11 - 1011 12 - 1100 13 - 1101
14 - 1110 15 - 1111

(6) Type of flip flop : JK flip flop (most flexible)

(6)

(7) Full state table with flip-flop excitations, K-maps

	current state	next state	flip-flop excitations					
	A B C D	A B C D	JA KA	JB KB	JC KC	JD KD		
0	0 0 0 0	0 0 0 1	0 X	0 X	0 X	1 X		
1	0 0 0 1	0 0 1 0	0 X	0 X	1 X	X 1		
2	0 0 1 0	0 0 1 1	0 X	0 X	X 0	1 X		
3	0 0 1 1	0 1 0 0	0 X	1 X	X 1	X 1		
4	0 1 0 0	0 1 0 1	0 X	X 0	0 X	1 X		
5	0 1 0 1	0 1 1 0	0 X	X 0	1 X	X 1		
6	0 1 1 0	0 1 1 1	0 X	X 0	X 0	1 X		
7	0 1 1 1	1 0 0 0	1 X	X 1	X 1	X 1		
8	1 0 0 0	1 0 0 1	X 0	0 X	0 X	1 X		
9	1 0 0 1	0 0 0 0	X 1	0 X	0 X	X 1		

JA

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	X	X	X	X
10	X	X	X	X

JA = B · C · D

KA

AB \ CD	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	X	X	X	X
10	0	1	X	X

KA = D

JB

AB \ CD	00	01	11	10
00	0	0	1	0
01	X	X	X	X
11	X	X	X	X
10	0	0	X	X

JB = C · D

KB

AB \ CD	00	01	11	10
00	X	X	X	X
01	0	0	1	0
11	X	X	X	X
10	X	X	X	X

KB = C · D

JC

AB \ CD	00	01	11	10
00	0	1	X	X
01	0	1	X	X
11	X	X	X	X
10	0	0	X	X

JC = A' · D

KC

AB \ CD	00	01	11	10
00	X	X	1	0
01	X	X	1	0
11	X	X	X	X
10	X	X	X	X

KC = D

(7)

JD

AB \ CD	00	01	11	10
00	1	X	X	1
01	1	X	X	1
11	X	X	X	X
10	1	X	X	X

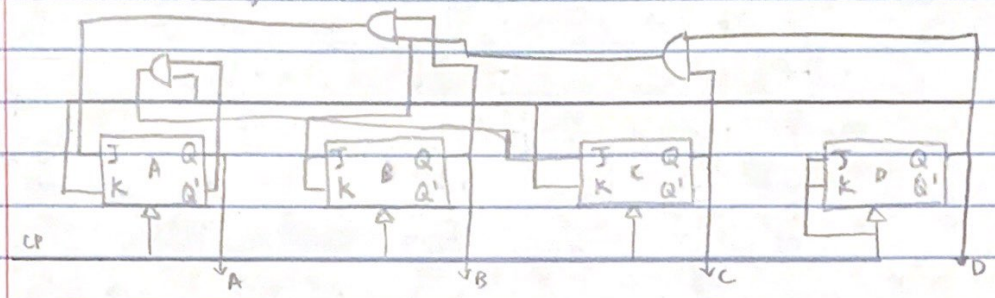
JD = 1

KD

AB \ CD	00	01	11	10
00	X	1	1	X
01	X	1	1	X
11	X	X	X	X
10	X	1	X	X

KD = 1

(8) Circuit Diagram

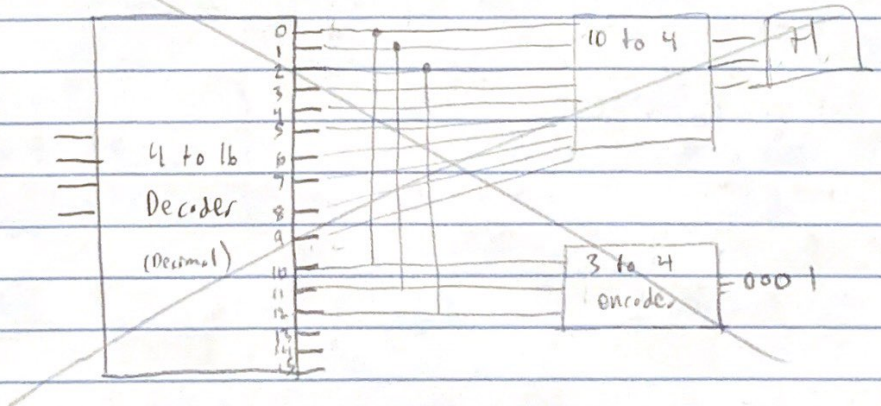


(9)

Unused States				Flip-Flop Excitations								Next State				Stable
A	B	C	D	JA	KA	JB	KB	JC	KC	JD	KD	A	B	C	D	Yes/No
1	0	1	0	0	0	0	0	0	0	1	1	1	0	1	1	Yes
1	0	1	1	0	1	1	1	0	1	1	1	0	1	0	0	Yes
1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	1	Yes
1	1	0	1	0	1	0	0	0	1	1	1	0	1	0	0	Yes
1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	Yes
1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	0	Yes

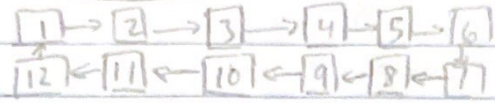
ABC Circuit

1100



(8)

1-12 State Counter

(1) 

(2) state table

current state	1	2	3	4	5	6	7	8	9	10	11	12
next state	2	3	4	5	6	7	8	9	10	11	12	1

(3) state reduction : none

(4) # of flip flops 3

(5) state assignments : 1 - 0001 2 - 0010
3 - 0011 4 - 0100 5 - 0101 6 - 0110 7 - 0111 8 - 1000 9 - 1001
10 - 1010 11 - 1011 12 - 1100
unused states 13 - 1101 14 - 1110 15 - 1111 0 - 0000

(6) Type of flip flop : JK flip flop (most flexible)

(7) Full state table with flip-flop excitations, k-maps

	current state	next state	flip-flop excitations					
	A B C D	A B C D	JA KA	JB KB	JC KC	JD KD
1	0 0 0 1	0 0 1 0	0 X	0 X	1 X	X 1
2	0 0 1 0	0 0 1 1	0 X	0 X	X 0	1 X
3	0 0 1 1	0 1 0 0	0 X	1 X	X 1	X 1
4	0 1 0 0	0 1 0 1	0 X	X 0	0 X	1 X
5	0 1 0 1	0 1 1 0	0 X	X 0	1 X	X 1
6	0 1 1 0	0 1 1 1	0 X	X 0	X 0	1 X
7	0 1 1 1	1 0 0 0	1 X	X 1	X 1	X 1
8	1 0 0 0	1 0 0 1	X 0	0 X	0 X	1 X
9	1 0 0 1	1 0 1 0	X 0	0 X	1 X	X 1
10	1 0 1 0	1 0 1 1	X 0	0 X	X 0	1 X
11	1 0 1 1	1 1 0 0	X 0	1 X	X 1	X 1
12	1 1 0 0	0 0 1 0	X 1	X 1	1 X	0 X

(9).

(8) JA

AB \ CD	00	01	11	10
00	X	0	0	0
01	0	0	1	0
11	X	X	X	X
10	X	X	X	X

$JA = B \cdot C \cdot D$

KA:

AB \ CD	00	01	11	10
00	X	X	X	X
01	X	X	X	X
11	1	X	X	X
10	0	0	0	0

$KA = B$

JB

AB \ CD	00	01	11	10
00	X	0	1	0
01	X	X	X	X
11	X	X	X	X
10	0	0	1	0

$JB = C \cdot D$

KB

AB \ CD	00	01	11	10
00	X	X	X	X
01	0	0	1	0
11	1	X	X	X
10	X	X	X	X

$KB = (C \cdot D) + (A)$

JC

AB \ CD	00	01	11	10
00	X	1	X	X
01	0	1	X	X
11	1	X	X	X
10	0	1	X	X

$JC = D + (A \cdot B)$

KC

AB \ CD	00	01	11	10
00	X	X	1	0
01	X	X	1	0
11	X	X	X	X
10	X	X	1	0

$KC = D$

JD

AB \ CD	00	01	11	10
00	X	X	X	1
01	1	X	X	1
11	0	X	X	X
10	1	X	X	1

$JD = A' \cdot B'$

KD

AB \ CD	00	01	11	10
00	X	1	1	X
01	X	1	1	X
11	X	X	X	X
10	X	1	1	X

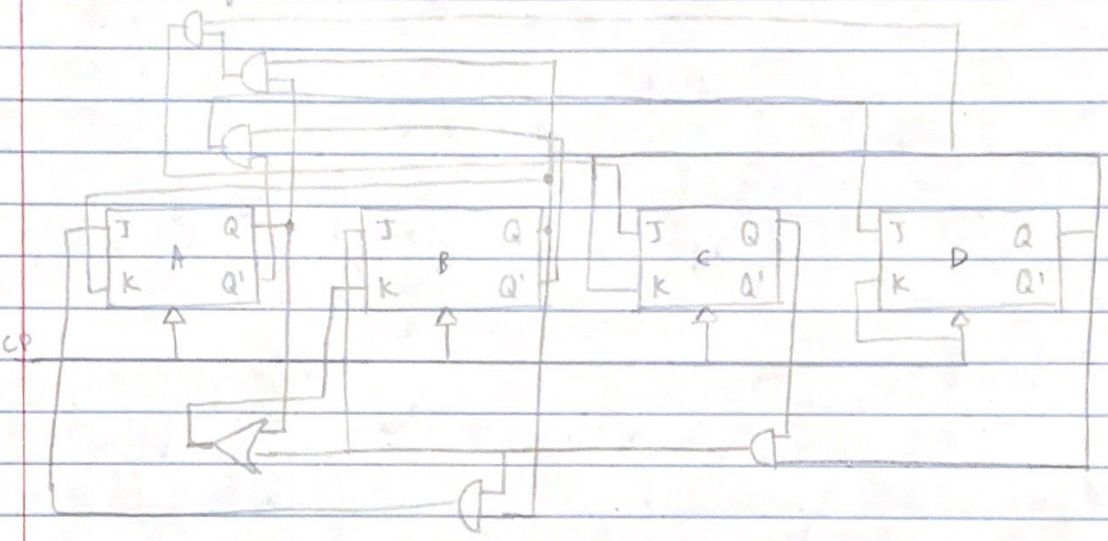
$KD = 1$

(10)

(9)

	Unused States				Flip Flop Excitations								Next State				Stable
	A	B	C	D	JA	KA	JB	KB	JC	KC	JD	KD	A	B	C	D	Yes/No
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	Yes
13	1	1	0	1	0	1	0	1	1	1	0	1	0	0	1	0	Yes
14	1	1	1	0	0	1	0	1	1	0	0	1	0	0	1	0	Yes
15	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0	0	Yes

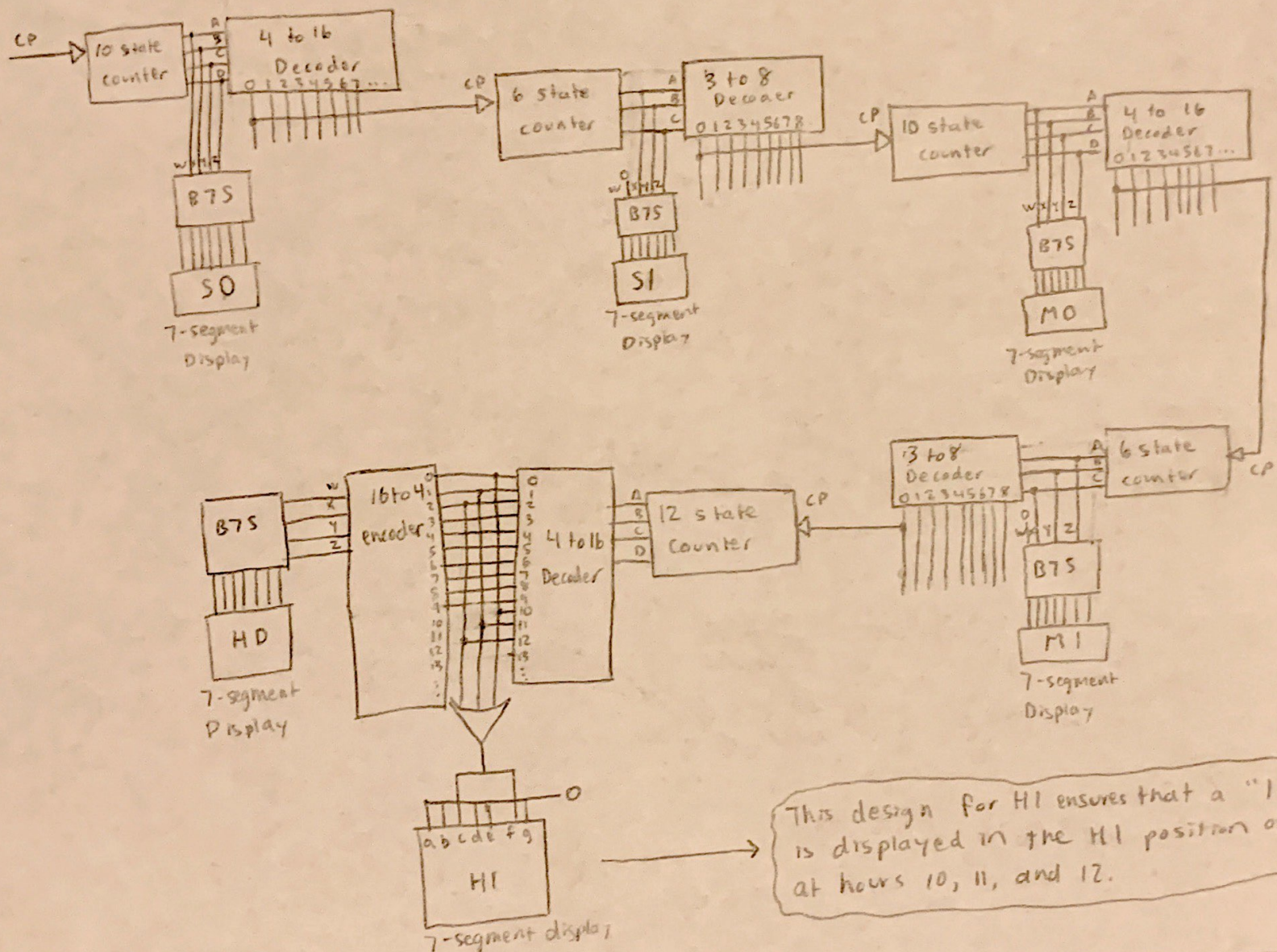
Circuit Diagram



(Sorry it's a bit messy, this one was complicated)

Note:
My final block diagram uses encoders and decoders that we designed as a class in discussion, so I didn't include their implementation.

Block Diagram for Entire Clock



Displays As

H1 H0 M1 M0 S1 S0

HH Hour | MM Minute | SS Second

B7S: Binary Decoder used to convert binary to input for 7-segment display