EE10a HW2

Steven Lei

Counting Up + Carry En

	Q3	Q2	Q1	Q0	DOWN	CEN1	D3	D2	D1	D0
0	0	0	0	0	0	1	0	0	0	1
1	0	0	0	1	0	1	0	0	1	0
2	0	0	1	0	0	1	0	0	1	1
3	0	0	1	1	0	1	0	1	0	0
4	0	1	0	0	0	1	0	1	0	1
5	0	1	0	1	0	1	0	1	1	0
6	0	1	1	0	0	1	0	1	1	1
7	0	1	1	1	0	1	1	0	0	0
8	1	0	0	0	0	1	1	0	0	1
9	1	0	0	1	0	1	0	0	0	0

Counting Down + Carry Enabled

	Q3	Q2	Q1	Q0	DOWN	CEN	D3	D2	D1	D0
0	0	0	0	0	1	1	1	0	0	1
9	1	0	0	1	1	1	1	0	0	0
8	1	0	0	0	1	1	0	1	1	1
7	0	1	1	1	1	1	0	1	1	0
6	0	1	1	0	1	1	0	1	0	1
5	0	1	0	1	1	1	0	1	0	0
4	0	1	0	0	1	1	0	0	1	1
3	0	0	1	1	1	1	1	0	1	0
2	0	0	1	0	1	1	1	0	0	1
1	0	0	0	1	1	1	0	0	0	0

Counting Down + Carry Disabled

	Q3	Q2	Q1	Q0	DOWN	CEN1	D3	D2	D1	D0
0	0	0	0	0	1	0	0	0	0	0
9	1	0	0	1	1	0	1	0	0	1
8	1	0	0	0	1	0	1	0	0	0
7	0	1	1	1	1	0	0	1	1	1
6	0	1	1	0	1	0	0	1	1	0
5	0	1	0	1	1	0	0	1	0	1
4	0	1	0	0	1	0	0	1	0	0
3	0	0	1	1	1	0	1	0	1	1
2	0	0	1	0	1	0	1	0	1	0
1	0	0	0	1	1	0	0	0	0	1

Counting up + carry Disabled

	Q3	Q2	Q1	Q0	DOWN	CEN	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	0	1
2	0	0	1	0	0	0	0	0	1	0
3	0	0	1	1	0	0	0	0	1	1
4	0	1	0	0	0	0	0	1	0	0
5	0	1	0	1	0	0	0	1	0	1
6	0	1	1	0	0	0	0	1	1	0
7	0	1	1	1	0	0	0	1	1	0
8	1	0	0	0	0	0	1	0	0	0
9	1	0	0	1	0	0	1	0	0	1

D_0

DOWN = FALSE $\bar{D}$

CEN = TRUE C

D0-0	Q3/Q2			
Q1/Q0	00	01	11	10
00	T	T	X	T
01	F	F	X	F
11	F	F	X	X
10	T	T	X	X

DOWN = TRUE D

D0-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	T	T	X	T
01	F	F	X	F
11	F	F	X	X
10	T	T	X	X

$C\bar{Q}_0 + \bar{C}Q_0 = 0$

CEN = FALSE $\bar{C}$

D0-0	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	F
01	T	T	X	T
11	T	T	X	X
10	F	F	X	X

D0-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	F
01	T	T	X	T
11	T	T	X	X
10	F	F	X	X

D_2

CEN = TRUE C

D1-0	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	F
01	T	T	X	F
11	F	F	X	X
10	T	T	X	X

D1-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	T	X	T
01	F	F	X	F
11	T	T	X	X
10	F	F	X	X

$C\bar{D}(\bar{Q}_3\bar{Q}_1Q_0 + Q_1\bar{Q}_0) + CD(Q_1Q_0 + Q_3\bar{Q}_0 + Q_2\bar{Q}\bar{Q}_0) + \bar{C}Q_1$

$= C(\bar{D}(\bar{Q}_3\bar{Q}_1Q_0 + Q_1\bar{Q}_0) + D(Q_1Q_0 + Q_3\bar{Q}_0 + Q_2\bar{Q}_1\bar{Q}_0) + \bar{C}Q_1$

CEN = FALSE $\bar{C}$

D1-0	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	F
01	F	F	X	F
11	T	T	X	X
10	T	T	X	X

D1-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	F
01	F	F	X	F
11	T	T	X	X
10	F	F	X	X

D_2

CEN = TRUE C

D2-0	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	T	X	F
01	F	T	X	F
11	T	F	X	X
10	F	T	X	X

D2-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	F
01	F	T	X	F
11	F	T	X	X
10	F	T	X	X

$C\bar{D}(Q_2\bar{Q}_1\bar{Q}_0 + \bar{Q}_2Q_1Q_0) + CD(\bar{Q}_2Q_1Q_0 + Q_3\bar{Q}_2\bar{Q}_0) + C(\bar{Q}_2Q_1\bar{Q}_0 + CQ_2\bar{Q}_1Q_0) + \bar{C}CQ_2 =$

$C(\bar{D}(\bar{Q}_2\bar{Q}_1\bar{Q}_0 + \bar{Q}_2Q_1Q_0) + D(Q_2Q_1Q_0 + Q_3\bar{Q}_2\bar{Q}_0) + Q_2(\bar{Q}_1\bar{Q}_0 + Q_1Q_0)) + \bar{C}Q_2$

CEN = FALSE $\bar{C}$

D2-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	T	X	F
01	F	T	X	F
11	F	T	X	X
10	F	T	X	X

D2-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	T	X	F
01	F	T	X	F
11	F	T	X	X
10	F	T	X	X

D_3

$Q_2Q_1Q_0 + Q_3\bar{Q}_0$

CEN = TRUE C

D3-0	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	T
01	F	F	X	F
11	F	T	X	X
10	F	F	X	X

D3-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	T	F	X	F
01	F	F	X	T
11	F	F	X	X
10	F	F	X	X

$(C\bar{D}(Q_2Q_1Q_0 + Q_3\bar{Q}_0)) + (CD(\bar{Q}_3\bar{Q}_2\bar{Q}_1\bar{Q}_0 + Q_3\bar{Q}_1Q_0)) + \bar{C}Q_3$

$= C(\bar{D}(Q_2Q_1Q_0 + Q_3\bar{Q}_0) + D(\bar{Q}_3\bar{Q}_2Q_1\bar{Q}_0 + Q_3\bar{Q}_1Q_0)) + \bar{C}Q_3$

CEN = FALSE $\bar{C}$

D3-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	T
01	F	F	X	T
11	F	F	X	X
10	F	F	X	X

D3-1	Q3/Q2			
Q1/Q0	00	01	11	10
00	F	F	X	T
01	F	F	X	T
11	F	F	X	X
10	F	F	X	X

C = when q and up or q and down

$C = (\bar{D}(Q_3\bar{Q}_2\bar{Q}_1Q_0) + D(\bar{Q}_3\bar{Q}_2\bar{Q}_1Q_0))$

```
MODULE      deccntr
TITLE      'Up/Down Decade Counter'


" deccntr       DEVICE  'MACH4032'


" Description:  This is the template for a 3-Digit Up/Down Decade Counter
"               for EE/CS 10a, Homework #2.


" Revision History:
" 04/17/07   Glen George  Initial Revision
" 10/13/07   Glen George  Changed CountUp pin to be called CountDown (its
"                         active high meaning)
" 10/13/07   Glen George  Updated comments
" 01/12/18   Glen George  Updated pinout to match new PCB (v2.0)
" 01/08/20   Glen George  Updated comments
" 01/04/21   Glen George  Updated comments
" 01/21/23   Steven Lei   Added program for HW2


" Pins

" Inputs

Mode            pin 41;                 "input   Mode (extra credit)
CountDown       pin 14;                 "input   Up/Down input (high for Down)


" Counter Outputs

Q11             pin 31   ISTYPE 'REG';  "output BCD digit 2, bit 3
Q10             pin 35   ISTYPE 'REG';  "output BCD digit 2, bit 2
Q9              pin 36   ISTYPE 'REG';  "output BCD digit 2, bit 1
Q8              pin 37   ISTYPE 'REG';  "output BCD digit 2, bit 0
Q7              pin 24   ISTYPE 'REG';  "output BCD digit 1, bit 3
Q6              pin 25   ISTYPE 'REG';  "output BCD digit 1, bit 2
Q5              pin 26   ISTYPE 'REG';  "output BCD digit 1, bit 1
Q4              pin 29   ISTYPE 'REG';  "output BCD digit 1, bit 0
Q3              pin 18   ISTYPE 'REG';  "output BCD digit 0, bit 3
Q2              pin 19   ISTYPE 'REG';  "output BCD digit 0, bit 2
Q1              pin 20   ISTYPE 'REG';  "output BCD digit 0, bit 1
Q0              pin 21   ISTYPE 'REG';  "output BCD digit 0, bit 0


" Clock Input

Clock           pin 39;                 "input   clock
```

```
" Unconnected Pins

"IOA2           pin 42                  input/output
"IOA3           pin 43                  input/output
"IOA4           pin 44                  input/output
"IOA5           pin  2                  input/output
"IOA6           pin  3                  input/output
"IOA7           pin  4                  input/output
"IOA8           pin  7                  input/output
"IOA9           pin  8                  input/output
"IOA10          pin  9                  input/output
"IOA12          pin 13                  input/output
"IOA14          pin 15                  input/output
"IOA15          pin 16                  input/output
"IOB4           pin 22                  input/output
"IOB9           pin 30                  input/output

"IOGOE0         pin 40                  input/output/output enable
"IOGOE1         pin 38                  input/output/output enable
"ICLK2          pin 17                  input/clock


" Programming Pins (not available for use in the design)

"TCK            pin 10                  pgm     programming interface TCK
"TDI            pin  1                  pgm     programming interface TDI
"TDO            pin 32                  pgm     programming interface TDO
"TMS            pin 23                  pgm     programming interface TMS


" Power Pins

"GND            pin  5                  supply  power ground
"GND            pin 12                  supply  power ground
"GND            pin 27                  supply  power ground
"GND            pin 34                  supply  power ground
"VCC            pin 11                  supply  power Vcc
"VCC            pin 33                  supply  power Vcc
"VCCIO          pin 28                  supply  power Vcc I/O
"VCCIO          pin  6                  supply  power Vcc I/O


" Intermediate Terms

CountEn1        node;           " enable out of digit 0 into digit 1
CountEn2        node;           " enable out of digit 1 into digit 2

EQUATIONS
```

```
" clocks for all of the BCD counter bits

Q11.CLK  =  Clock;                    " use the global clock pin
Q10.CLK  =  Clock;                    " use the global clock pin
Q9.CLK   =  Clock;                    " use the global clock pin
Q8.CLK   =  Clock;                    " use the global clock pin
Q7.CLK   =  Clock;                    " use the global clock pin
Q6.CLK   =  Clock;                    " use the global clock pin
Q5.CLK   =  Clock;                    " use the global clock pin
Q4.CLK   =  Clock;                    " use the global clock pin
Q3.CLK   =  Clock;                    " use the global clock pin
Q2.CLK   =  Clock;                    " use the global clock pin
Q1.CLK   =  Clock;                    " use the global clock pin
Q0.CLK   =  Clock;                    " use the global clock pin


" This is the code for a 12 bit 3 digit decimal counter. The counter can count in
decimal 000->999 and wrap counting both up and down.
" To build the logic for the counter we need to consider the variables involved.
Each digit has 4 bits, there is a count up/down switch, and a carry on for
tens/hundreds digits (allows the tens/hundreds to count only when necessary).
" This means there are 6 variable inputs (4 bits, CountEn, CountDown) involved. A 6
variable K-Map was constructed (attached with this paper). For the first digit, only
5 variables control the digit, but for the other two digits, 6 variables control the
input.
" The simplified equations for each bit were generated using the 6 variable K-Map.
However, the comments explain the intuitive process we can use to think about
generating each equation per bit.



" LOW ORDER DIGIT
        " The low order digit is always counting, so no CountEn input is involved.

        " Q0 is always alternating, regardless whether counting up or down.

Q0  := !Q0;


        " For Q1 and above we need to consider cases when counting up vs. down.
        " Counting up: Q1 will be high because of increment from 0X01 or XX10.
        " Counting down: Q1 will be high because of decrement from XX11 or X110 or
10X0.

Q1  := (!CountDown & ((!Q3 & !Q1 & Q0) # (Q1 & !Q0)))
        # (CountDown & ((Q1 & Q0) # (Q2 & !Q1 & !Q0) # (Q3 & !Q0)));


        " Counting up: Q2 will be high because of increment from 0011 or from any
```

```
X1XX except X111
        " Counting down: Q2 will be high because of decrement from 1000 or any X1XX
except X100

Q2  := (!CountDown & ((Q2 & !Q1 & !Q0) # (!Q2 & Q1 & Q0)))
        # (CountDown & ((Q2 & Q1 & Q0) # (Q3 & !Q2 & !Q0)))
        # (Q2 & ((!Q1 & Q0) # (Q1 & !Q0)));


        "Counting up: Q3 will be high because of increment from 0111 or 1XXX except
1001 (since 9 needs to wrap to 0 and the rest are don't care values)
        "Counting down: Q3 will be high because of decrement from 0000 or 1XX1
(since 0 needs to wrap to 9)

Q3  := (!CountDown & ((Q2 & Q1 & Q0) # (Q3 & !Q0)))
        # (CountDown & ((!Q3 & !Q2 & !Q1 & !Q0) # (Q3 & !Q1 & Q0)));


        "CountEn1 will allow the middle digit to count. It is high in two cases:
        "Counting up: high when the lowest digit is 9
        "Counting down: high when the lowest digit is 0 (allows for wrap around)

CountEn1 = (!CountDown & (Q3 & !Q2 & !Q1 & Q0))
           # (CountDown & (!Q3 & !Q2 & !Q1 & !Q0));


" MIDDLE DIGIT

        " The middle and high order digits follow the same COUNTING logic as the
first digit.
        " However they must account for the CountEn inputs since we want the values
to hold while the previous digit is counting.
        " E.g. if we're counting from 150 to 159, we want the hundreds and tens
place to hold their value while the ones is counting.
        " This is what makes the K-Map a 6 variable K-map.

        " For example, Q4 will count if CountEn1 is high but will retain its value
if CountEn1 is low

Q4  := (CountEn1 & !Q4)
        # (!CountEn1 & Q4);

Q5  := (CountEn1
        & ((!CountDown & ((!Q7 & !Q5 & Q4) # (Q5 & !Q4)))
        # (CountDown & ((Q5 & Q4) # (Q7 & !Q4) # (Q6 & !Q5 & !Q4)))))
        # (!CountEn1 & Q5);

Q6  := (CountEn1
        & ((!CountDown & ((Q6 & !Q5 & !Q4) # (!Q6 & Q5 & Q4)))
        # (CountDown & ((Q6 & Q5 & Q4) # (Q7 & !Q6 & !Q4)))
```

```
        # (Q6 & ((!Q5 & Q4) # (Q5 & !Q4)))))
        # (!CountEn1 & Q6);

Q7  := (CountEn1
        & ((!CountDown & ((Q6 & Q5 & Q4) # (Q7 & !Q4)))
        # (CountDown & ((!Q7 & !Q6 & !Q5 & !Q4) # (Q7 & !Q5 & Q4)))))
        # (!CountEn1 & Q7);


        " CountEn2 follows the same logic as CountEn1, however it must also account
for when CountEn1 is high;
        " The last digit can only count if the previous digit is also going to count
(e.g. 199 -> 200).

CountEn2 = ((!CountDown & (Q7 & !Q6 & !Q5 & Q4))
           # (CountDown & (!Q7 & !Q6 & !Q5 & !Q4))) & CountEn1;


" HIGH ORDER DIGIT

        " Same logic as Q4-Q7

Q8  := (CountEn2 & !Q8)
        # (!CountEn2 & Q8);

Q9  := (CountEn2
        & ((!CountDown & ((!Q11 & !Q9 & Q8) # (Q9 & !Q8)))
        # (CountDown & ((Q9 & Q8) # (Q11 & !Q8) # (Q10 & !Q9 & !Q8)))))
        # (!CountEn2 & Q9);

Q10  := (CountEn2
        & ((!CountDown & ((Q10 & !Q9 & !Q8) # (!Q10 & Q9 & Q8)))
        # (CountDown & ((Q10 & Q9 & Q8) # (Q11 & !Q10 & !Q8)))
        # (Q10 & ((!Q9 & Q8) # (Q9 & !Q8)))))
        # (!CountEn2 & Q10);

Q11  := (CountEn2
        & ((!CountDown & ((Q10 & Q9 & Q8) # (Q11 & !Q8)))
        # (CountDown & ((!Q11 & !Q10 & !Q9 & !Q8) # (Q11 & !Q9 & Q8)))))
        # (!CountEn2 & Q11);

END  deccntr
```