

dce 2009

Digital Logic Design 1

Counters and Registers

dce 2009

Asynchronous (Ripple) Counters

- Review of four bit counter operation (refer to next slide)
 - Clock is applied only to FF A. J and K are high in all FFs to toggle on every clock pulse.
 - Output of FF A is CLK of FF B and so forth.
 - FF outputs D, C, B, and A are a 4 bit binary number with D as the MSB.
 - After the negative transition of the 15th clock pulse the counter recycles to 0000.
- This is an asynchronous counter because state is not changed in exact synchronism with the clock.

dce 2009

Four-bit asynchronous (ripple) counter

*All J and K inputs assumed to be 1.

□ MOD = the number of states

dce 2009

Frequency division

- The output frequency of each FF = the clock frequency of input / 2.
- The output frequency of the last FF = the clock frequency / MOD.

dce 2009

Propagation Delay in Ripple Counters

- Ripple counters are simple, but the cumulative propagation delay can cause problems at high frequencies.
- For proper operation the following apply:
 - $T_{clock} \geq N \times t_{pd}$
 - $F_{max} = 1/(N \times t_{pd})$

dce 2009

Ripple Counter Propagation Delay

Counters with MOD Number $< 2^N$

- Find the smallest MOD required so that 2^N is less than or equal to the requirement.
- Connect a NAND gate to the asynchronous CLEAR inputs of all FFs.
- Determine which FFs are HIGH at the desired count and connect the outputs of these FFs to the NAND gate inputs.

State transition diagram for the MOD-6 counter

MOD-6 Counter

MOD-6 counter produced by clearing a MOD-8 counter when a count of six (110) occurs.

Counters with MOD Number $< 2^N$

- General Procedures Counter Design**
 - Find the smallest number of FF
 - Connect a NAND gate to the Asynchronous CLEAR inputs of all the FFs
 - Determine which FFs will be in the HIGH state at a count = X; then connect the normal outputs of these FFs to the NAND gate inputs

Decade counters/BCD counters

- Decade counters/BCD counters
 - A decade counter is any counter with 10 distinct states, regardless of the sequence. Any MOD-10 counter is a decade counter.
 - A BCD counter is a decade counter that counts from binary 0000 to 1001.
- Decade counters are widely used for counting events and displaying results in decimal form.

Asynchronous Down Counter

- All of the counters we have looked were up counters.
- Down counter counts number downward e.g: 111 -> 000

	CBA
(7)	111
(6)	110
(5)	101
(4)	100
(3)	011
(2)	010
(1)	001
(0)	000

Recycles

Asynchronous Down Counter

- Each FF, except the first must toggle when the preceding FF goes from LOW to HIGH
- If the FFs have CLK inputs that respond to negative transition (HIGH to LOW), then an inverter can be placed in front of each CLK input; however the same effect can be accomplished by driving each FF CLK input from the inverted output of the preceding FF.
- Input pulses are applied to A. The A' output serves as the CLK input for B; the B' output serves as the CLK input for the C.
- The waveforms at A, B and C show that B toggles whenever A goes LOW to HIGH and C toggles whenever B goes LOW to HIGH.

Asynchronous Down Counter

Count (CBA) 000 111 110 101 100 011 010 001 000 Recycles

IC Asynchronous counter

RESET INPUTS		OUTPUTS		
MR ₁	MR ₂	Q ₀	Q ₁	Q ₂
H	H	L	L	L
L	H	L	L	L
L	L	L	L	L
L	L	L	L	L

Example

- Show how to wire the 74LS293 as a MOD-16, MOD-10 counter with a 10-kHz clock input. Determine the frequency at Q3.

$f = 10 \text{ kHz} / 16 = 625 \text{ Hz}$

$f = 10 \text{ kHz} / 10 = 1 \text{ kHz}$

Synchronous (Parallel) Counters

- All FFs are triggered by CLK **simultaneously**
- Mod-16 counter.
 - Each FF has J and K inputs connected so they are **HIGH only when the outputs of all lower-order FFs are HIGH.**
 - The total propagation delay will be the same for any number of FFs.
- Synchronous counters can operate at much higher frequencies than asynchronous counters.

Count	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
0	0	0	0	0

Synchronous (Parallel) Counters

- Circuit Operation**
 - On a given NGT of the clock, only those FFs that are supposed to toggle on that NGT should have $J=K=1$ when that NGT occurs.
 - FF A must change states at each NGT. Its J and K inputs are permanently HIGH so that it will toggle on each NGT of the CLK input.
 - FF B must change states on each NGT that occurs while $A=1$.
 - FF C must change states on each NGT that occurs while $A=B=1$
 - FF D must change states on each NGT that occurs while $A=B=C=1$

Synchronous (Parallel) Counters

- Each FF should have its J&K inputs connected such that they are HIGH only when the outputs of all lower-order FFs are in the HIGH state.
- Advantages over asynchronous:
 - FFs will change states simultaneously; synchronized to the NGTs of the input clock pulses.
 - Propagation delays of the FFs do not add together to produce the overall delay.
 - The total response time is the time it takes one FF to toggle plus the time for the new logic levels to propagate through a single AND gate to reach the J, K inputs.
- total delay = FF tpd + AND gate tpd**

Counters for MOD 2^N

MOD-14 counter resets when count 14 is reached.

MOD-10 (decade) counter. Resets when count 10 is reached.

Example: MOD-60 Counter

Resets when count 60 is reached.

Synchronous Down and Up/Down Counters

- The synchronous counter can be converted to a down counter by using the inverted FF outputs to drive the JK inputs.

Up/Down

CLOCK

A

B

C

Count (CBA) 000 001 010 011 100 101 100 011 001 000

Up

Down

MOD-8 synchronous up/down counter

The counter counts up when the control input Up/Down = 1; it counts down when the control input Up/Down = 0.

MOD-8 synchronous up/down counter

The counter counts up when the control input Up/Down = 1; A and B signals are passed. It counts down when the control input Up/Down = 0; inverted A and B signals are passed.

Synchronous, MOD-16, down counter

Input

A

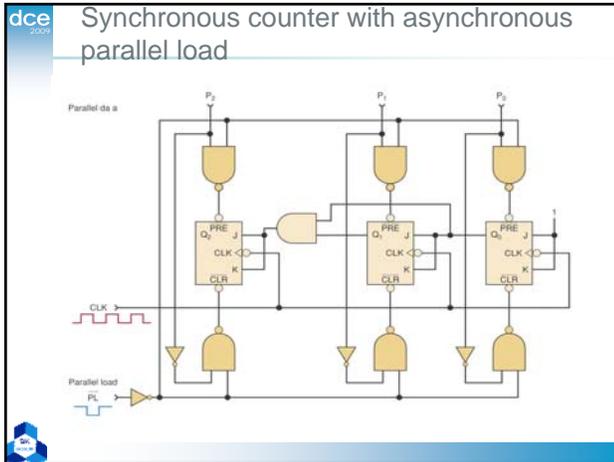
B

C

D

Presettable Counters

- A presettable counter can be set to any desired starting point either asynchronously or synchronously.
- The preset operation is also called parallel loading the counter.



IC Synchronous Counters

(a)

Part Number	Modulus
74ALS160	10
74ALS161	16
74ALS162	10
74ALS163	16

(b)

- 4 FFs,
- PGT at the CLK input,
- The counter can be preset to any value (applied to the A, B, C, and D inputs) by applying an active-low LOAD input.

74ALS160-74ALS163 Function Table

CLR	LOAD	ENP	ENT	CLK	Function	Part Numbers
L	X	X	X	X	Asynch. Clear	74ALS160 & 74ALS161
L	X	X	X	↑	Synch. Clear	74ALS162 & 74ALS163
H	L	X	X	↑	Synch. Load	All
H	H	H	H	↑	Count up	All
H	H	L	X	X	No change	All
H	H	X	L	X	No change	All

(c)

Synchronous Counter Example

- start counting at t1
- synchronous clear at t2
- synchronous load at t3
- stop counting at t4 (ENT low)
- no counting at t5 (ENP low)
- resume counting at t6 (ENT low)
- terminal state sets RCO (ripple carry out) high automatic reset at t7

Synchronous Counter Example

- start counting at t1
- asynchronous clear at t2
- asynchronous clear at t3
- stop counting at t4 (ENT low)
- synchronous load at t5
- stop counting at t6 (ENT low)
- continue counting at t7 terminal state of 1001 sets RCO
- stop counting at t8 (ENP)
- RCO goes low at t9 due to low ENT (ENP does not affect RCO)

74ALS190-74ALS191 series synchronous counters (up/down)

(a)

Part Number	Modulus
74ALS190	10
74ALS191	16

(b)

74ALS190-74ALS191 Function Table

LOAD	CTEN	D/U	CLK	Function
L	X	X	X	Asynch. Load
H	L	L	↑	Count up
H	L	H	↑	Count down
H	H	X	X	No change

(c)

MOD-10 Counter

•Maximum state is 1001

•Max/min is high when state is 1001 and up-counting; or 0000 and down-counting

•Max/min low at other times

MOD-12 & MOD-11 Counters

Using 74ALS163 (syn clear) and 74ALS191 (async clear) MOD-16 counters for other MODs

Synchronous load
0001-1100
mod-12 counter

asynchronous load
0001-1011
mod-11 counter (in 1100 state for a short period of time)

Extending Maximum Counting Range

To higher-order counter stages

To higher-order counter stages

Decoding a Counter

- Decoding is the conversion of a binary output to a decimal value.
- The active high decoder could be used to light an LED representing each decimal number 0 to 7.
- Active low decoding is obtained by replacing the AND gates with NAND gates.

Decoding a Counter

Using AND Gates to Decode a MOD-8 Counter (produce pulse at specific count)

C, B, A

0, 1, 2, 3, 4, 5, 6, 7

Decoding a Counter

Circuit to Make X High Between Counts of 8 and 14 (sets FF at count 8, then clears at count 14)

decodes 1000

decodes 1110

Analyzing Synchronous Counters

- Example of a synchronous up counter.
 - The control inputs are as follows: $J_C = A \cdot B$, $K_C = C$, $J_B = K_B = A$, $J_A = K_A = \bar{C}$

Analyzing Synchronous Counters

• State transition diagram and timing diagram for synchronous counter

• unused states not in timing diagram

CLK

C

B

A

Synchronous Counter Design

- Determine desired number of bits and desired counting sequence
- Draw the state transition diagram showing all possible states
- Use the diagram to create a table listing all PRESENT states and their NEXT states
- Add a column for each JK input (or other inputs). Indicate the level required at each J and K in order to produce transition to the NEXT state.
- Design the logic circuits to generate levels required at each JK input.
- Implement the final expressions.

Choose a type of FF – JK in this example

State transition diagram for the synchronous counter design

unused states

Present State	Next State	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

State table of counter example

	PRESENT State			NEXT State			J _c	K _c	J _b	K _b	J _a	K _a
	C	B	A	C	B	A						
Line 1	0	0	0	0	0	1	0	x	0	x	1	x
2	0	0	1	0	1	0	0	x	1	x	x	1
3	0	1	0	0	1	1	0	x	x	0	1	x
4	0	1	1	1	0	0	1	x	x	1	x	1
5	1	0	0	0	0	0	x	1	0	x	0	x
6	1	0	1	0	0	0	x	1	0	x	x	1
7	1	1	0	0	0	0	x	1	x	1	0	x
8	1	1	1	0	0	0	x	1	x	1	x	1

K maps for the J and K logic circuits

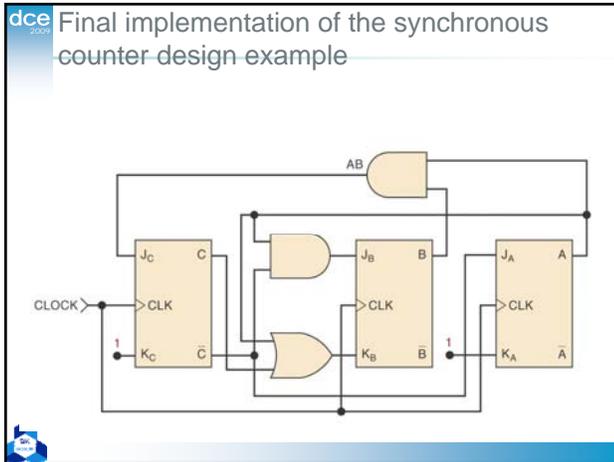
K map used to obtain the simplified expression for J_A ; from the state table

Line	PRESENT State			NEXT State			J _c	K _c	J _b	K _b	J _a	K _a
	C	B	A	C	B	A						
1	0	0	0	0	0	1	0	x	0	x	1	x
2	0	0	1	0	1	0	0	x	1	x	x	1
3	0	1	0	0	1	1	0	x	x	0	1	x
4	0	1	1	1	0	0	1	x	x	1	x	1
5	1	0	0	0	0	0	x	1	0	x	0	x
6	1	0	1	0	0	0	x	1	0	x	x	1
7	1	1	0	0	0	0	x	1	x	1	0	x
8	1	1	1	0	0	0	x	1	x	1	x	1

$$J_A = \bar{C}$$

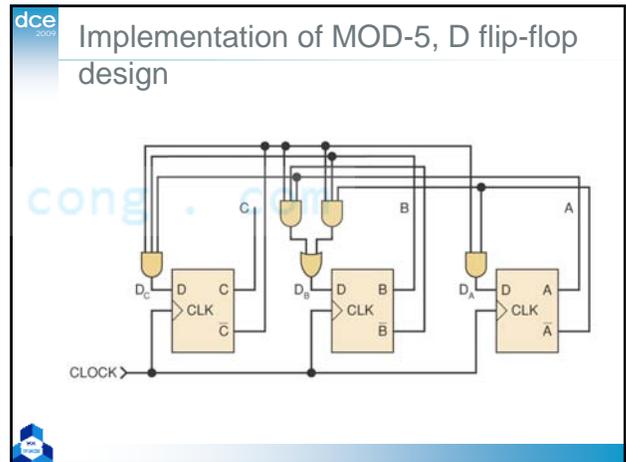
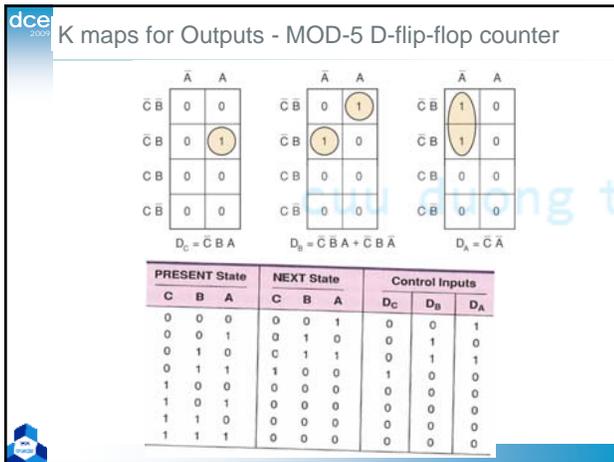
K maps for the J and K logic circuits

Line	PRESENT State			NEXT State			J _c	K _c	J _b	K _b	J _a	K _a
	C	B	A	C	B	A						
1	0	0	0	0	0	1	0	x	0	x	1	x
2	0	0	1	0	1	0	0	x	1	x	x	1
3	0	1	0	0	1	1	0	x	x	0	1	x
4	0	1	1	1	0	0	1	x	x	1	x	1
5	1	0	0	0	0	0	x	1	0	x	0	x
6	1	0	1	0	0	0	x	1	0	x	x	1
7	1	1	0	0	0	0	x	1	x	1	0	x
8	1	1	1	0	0	0	x	1	x	1	x	1



State Table for Example: MOD-5 Counter Using D-type Flip-Flops

PRESENT State			NEXT State			Control Inputs		
C	B	A	C	B	A	D _C	D _B	D _A
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	1	0	0	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0



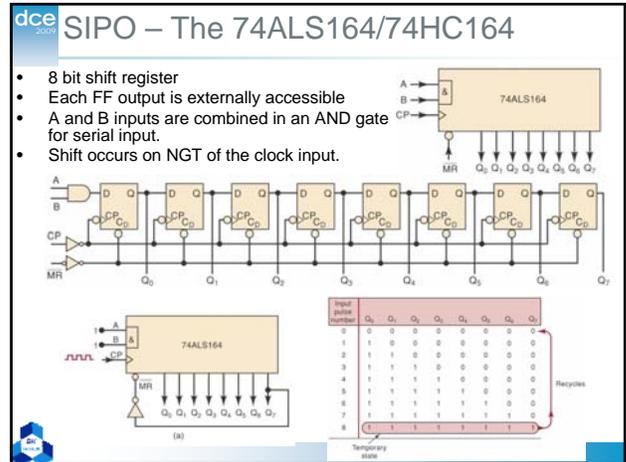
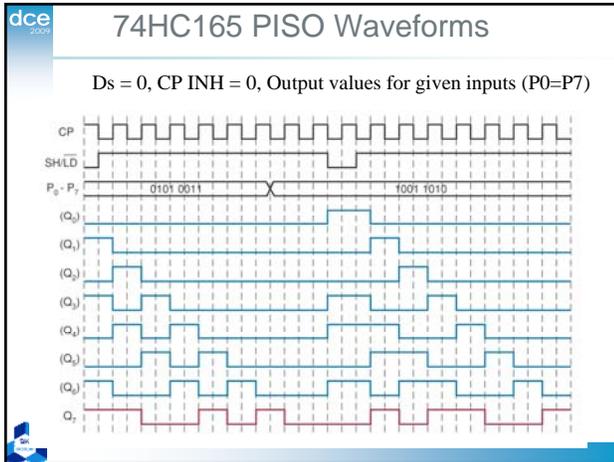
- Integrated-Circuit Registers
- Registers can be classified by the way data is entered for storage, and by the way data is outputted from the register.
 - Parallel in/parallel out (PIPO)
 - Serial in/serial out (SISO)
 - Parallel in/serial out (PISO)
 - Serial in/parallel out (SIPO)

PISO – The 74ALS165/74HC165

- 8 bit register
 - Serial data entry via D_S
 - Asynchronous parallel data entry P₀ through P₇
 - Only the outputs of Q₇ are accessible
- CP is clock input for shifting
- Clock inhibit input
- Shift load input

Inputs			Operation
SHLD	CP	CPINH	
L	X	X	Parallel load
H	H	X	No change
H	X	H	No change
H	L	L	Shifting
H	L	F	Shifting

H = high level
L = low level
X = immaterial
F = PGT



- ### Other similar devices
- 74194/ALS194/HC194
 - 4 bit bi-directional universal shift register
 - Performs shift left, shift right, parallel in and parallel out.
 - 74373/ALS373/HC373/HCT373
 - 8 bit PIPO with 8 D latches
 - Tristate outputs
 - 74374/ALS374/HC374
 - 8 bit PIPO with 8 edge triggered D FFs, Tristate outputs

- ### Shift Register Counters
- Ring Counter
 - Last FF shifts its value to first FF
 - Uses D-type FFs (JK FFs can also be used)
 - Must start with only one FF in the 1 state and all others in the 0 state.

