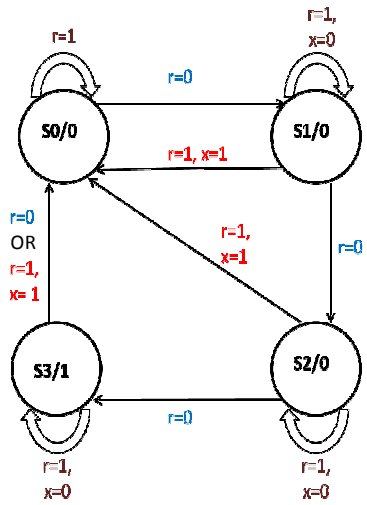
ECE 341 Midterm Exam Solution

Problem No. 1

(a) False, (b) True, (c) False, (d) False, (e) True, (f) False

Problem No. 2

(a) The state machine for the given counter is as follows:

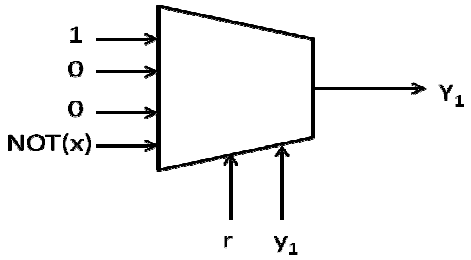


(b) The state-assigned state table is as follows:

Present State	Next State				Output z
	r = 0, x = 0	r = 0, x = 1	r = 1, x = 0	r = 1, x = 1	
y_2y_1	Y_2Y_1	Y_2Y_1	Y_2Y_1	Y_2Y_1	
00	01	01	00	00	0
01	10	10	01	00	0
10	11	11	10	00	0
11	00	00	11	00	1

(c) $Y_1 = \bar{r}\bar{x}\,\overline{y_2}\,\overline{y_1} + \bar{r}\bar{x}y_2\overline{y_1} + \bar{r}x\overline{y_2}\,\overline{y_1} + \bar{r}xy_2\overline{y_1} + r\bar{x}\,\overline{y_2}\,y_1\ r\bar{x}y_2y_1$
$= \bar{r}\bar{x}\,\overline{y_1}(\overline{y_2} + y_2) + \bar{r}x\,\overline{y_1}(\overline{y_2} + y_2) + r\bar{x}\,y_1(\overline{y_2} + y_2)$
$= \bar{r}\bar{x}\,\overline{y_1} + \bar{r}x\,\overline{y_1} + r\bar{x}\,y_1$
$= \bar{r}\,\overline{y_1}(\bar{x} + x) + r\bar{x}\,y_1$
$= \bar{r}\,\overline{y_1} + r\bar{x}\,y_1$

(d) The implementation of Y_1 with a 4-input multiplexer is as follows:



Problem No. 3

(a) X = 0101, Y = 1001, $c_0 = 1$
$G_0 = x_0$ AND y_0 = 1 AND 1 = 1
$G_1 = x_1$ AND y_1 = 0 AND 0 = 0
$G_2 = x_2$ AND y_2 = 1 AND 0 = 0
$G_3 = x_3$ AND y_3 = 0 AND 1 = 0
$P_0 = x_0$ OR y_0 = 1 OR 1 = 1
$P_1 = x_1$ OR y_1 = 0 OR 0 = 0
$P_2 = x_2$ OR y_2 = 1 OR 0 = 1
$P_3 = x_3$ OR y_3 = 0 OR 1 = 1

(b) We need to express both the numbers as 6-bit binary numbers because +16 requires 6 bits in 2's complement representation:

```
  0 1 0 0 0 0   (+16)
X 1 1 1 0 0 1   (-7)
---------------
```

⇒

```
  0 1 0 0 0 0
  0 0 -1 0 +1 -1
---------------
1 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
---------------
1 1 1 1 1 0 0 1 0 0 0 0   (- 112)
```

(c) Clock period = 0.4 nanoseconds = $0.4 * 10^{-9}$ s = $4 * 10^{-10}$ s
Clock rate "R" = 1 / Clock Period = 1 / ($4 * 10^{-10}$) = $2.5 * 10^9$ = 2.5 GHz
CPI = (10% * 3) + (25% * 5) + (15% * 4) + (50% * 4) = 4.15
Since:
Execution time = (Number of Instructions * CPI) / Clock Rate
Therefore:

Instruction Execution Rate = Number of Instructions / Execution Time = Clock Rate / CPI
= 2.5 GHz / 4.15 = $(2.5 * 10^9) / 4.15 = 6.024 * 10^8$ instructions per second
Since 1 millisecond = 10^{-3} sec:
Number of instructions completed in 1 millisecond = $(6.024 * 10^8) * 10^{-3}$ = **$6.024 * 10^5$**

Problem No. 4

(a) (i) *B_select* has values of **1** and **0** for instructions 1 and 2, respectively.
(ii) *Y_select* has values of **01** and **00** for instructions 1 and 2, respectively.
(iii) *C_select* has values of **00**, **01** and **10** for instructions 1, 2, and 3, respectively.

(b) For instruction 1:
Contents of *RY* = Contents of memory address (6000 + 100) = **100**
The contents of register R4 become 100 after instruction 1.

For instruction 2:
Contents of *RY* = [R3] - [R4] = 300 - 100 = **200**

For instruction 3:
Contents of *RY* = Return address for Call instruction = 2000 + (4 + 4 + 4) = **2012**

(c) The "ldu" instruction requires two write-back operations: one to register Rdst and the second to register Rsrc. Therefore, the following changes are needed to implement the "ldu" instruction:
Change # 1: The register file needs to have two write ports. The address input for the second write port is connected to IR[31-27], which corresponds to the register Rsrc.
Change # 2: An additional control signal *RF_write_2* is connected to the register file so that the second register write port is activated only for the "ldu" instruction.
Change # 3: A register "RS" is added between stage-4 and stage-5 (parallel to the register "RY"). A wire connects the output of the register "RZ" to the input of the register "RS". The output of register "RS" is connected to the second write port of the register file. These connections enable the "ldu" instruction to update the register Rsrc with the new effective address during the writeback stage (stage-5) of 5-stage hardware.