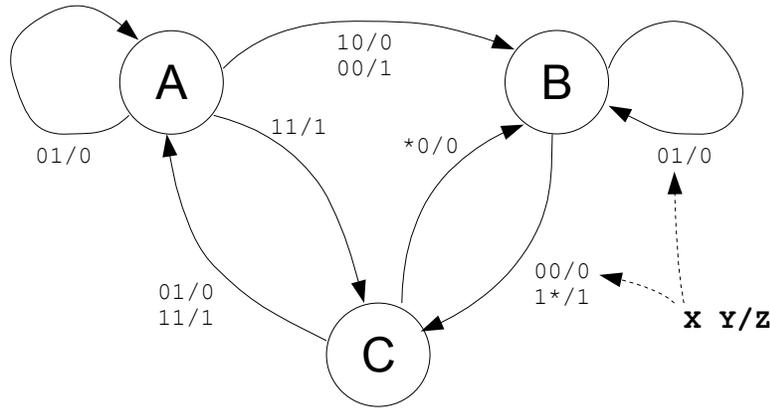


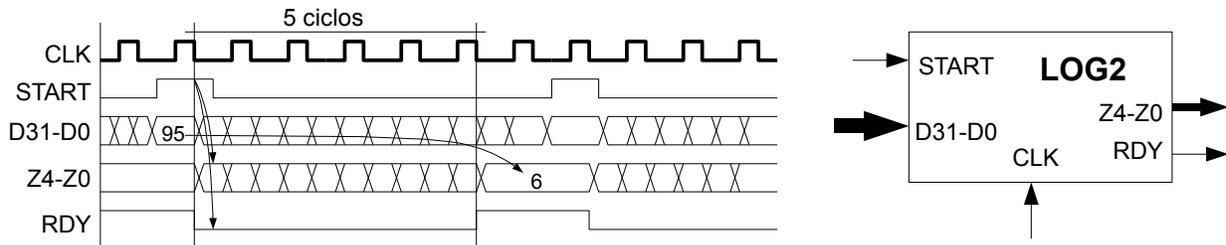
**CC41C Introducción al Hardware**  
**Control 1 - Otoño 2008**  
**Profesor: Luis Mateu**

**Pregunta 1**

Un circuito secuencial tiene entradas **X**, **Y**, y salida **Z**. El siguiente diagrama de estados especifica su comportamiento. Utilice la metodología vista en clases para implementar este circuito. Haga la codificación de estados, tablas de verdad, mapas de karnaugh y obtenga las fórmulas algebraicas.



**Pregunta 2**



El circuito LOG2 de la figura calcula la parte entera del logaritmo en base 2 de un número entero binario de 32 bits distinto de 0. Esto equivale al peso (o posición) del bit más significativo de ese número. Por ejemplo  $\log_2(1)=0$ ,  $\log_2(2)=1$ ,  $\log_2(3)=1$ ,  $\log_2(4)=2$ , ...,  $\log_2(95)=6$ , ..., etc. Cuando START está en 1 durante el pulso de bajada del reloj, el circuito LOG2 lee el número binario que aparece en D31-D0, y comienza a calcular el logaritmo. Después de exactamente 5 ciclos se indica el término del cálculo colocando un 1 en RDY y el resultado aparece en Z4-Z0. Esto se explica en el diagrama de tiempo adjunto.

Implemente el circuito LOG2 usando diseño modular, recurriendo a las componentes vistas en clases, como multiplexores, registros, sumadores, desplazadores, etc. Si necesita un circuito secuencial o combinacional, solo especifique el diagrama de estados o la tabla de verdad, sin implementarlo. Por razones de eficiencia, para calcular el resultado Ud. *debe* usar el siguiente algoritmo:

```

int log2(unsigned n) { /* n es de 32 bits */
    int log= 0, s= 16, i;
    for (i= 0; i<5; i++) { /* Cada iteración es un ciclo del reloj */
        unsigned m= n>>s; /* desplazamiento a la der. en s bits */
        log= log<<1;
        if (m!=0) {
            log = log | 1; /* or */
            n= m;
        }
        s= s>>1;
    }
    return log;
}

```