

APROXIMACIÓN ESTRUCTURAL AL DISEÑO DE SISTEMAS DIGITALES COMPLEJOS

T. POLLÁN, J.M. LÓPEZ, Y B. MARTÍN

*Departamento de Ingeniería Electrónica y Comunicaciones.
Tecnología Electrónica. E.U.I.T.I. Universidad de Zaragoza*

¿Cómo abordar el diseño de sistemas digitales complejos?. Divide y vencerás, es una buena respuesta; pero dividir con "estructura" (teniendo clara la relación de cada parte con el conjunto). En esta perspectiva se incluye la comunicación procesador-"memoria" a través de buses y la separación parte operativa-parte de control.

1. Introducción

En el marco de las V Jornadas de Tecnología Electrónica, en 1994, una breve comunicación [1] se refirió al hecho de que las asignaturas de Electrónica Digital no llegan a abordar el diseño de sistemas digitales complejos; "de modo que enseñamos las piezas pero no llegamos a construir sistemas". En su opinión, la formación en diseño digital no debe contentarse con el tratamiento habitual de bloques y subsistemas digitales y con el desarrollo de sistemas típicos, relativamente simples, sino que debería ir más allá y llegar a enfrentar al estudiante con el diseño de un sistema digital complejo.

Respondiendo a este reto, que coincide con uno de los aspectos docentes que siempre nos han preocupado ("capacitar para abordar la complejidad"), la presente comunicación pretende esbozar una respuesta (o, quizás mejor, una reflexión) a la pregunta planteada, formulando una aproximación conceptual al problema de la complejidad en los sistemas digitales desde una perspectiva "práctica" que resulte útil y operativa para el diseñador.

2. Hacer manejable la complejidad

Resulta difícil enfrentarse a "la complejidad", pero es algo que los diseñadores de sistemas digitales venimos haciendo en el "día a día" y conviene transferir una pautas o "formas de actuar" que orienten y ayuden a quienes se inician en las tareas de diseño.

¿Cómo abordar un diseño que nuestra mente no puede abarcar en conjunto? [2]. "Fracccionar sin perder la globalidad", podría ser una respuesta concisa: la idea es dividir el sistema en *partes*, conservando la visión global del mismo; aún más, dando particular importancia a la *estructura*, a la relación de cada parte con el conjunto. *Por partes* (divide y vencerás) y, a la vez, *con estructura* (con perspectiva organizada del conjunto): una metodología que permita poner en relación el diseño de cada parte con los requisitos globales que se pretenden, combinando adecuadamente los aspectos de partición, jerarquía y coordinación.

Esto es lo que hacemos cuando formulamos la arquitectura de un sistema en términos de "diagrama de bloques"; cada bloque constituye una parte diferenciada del sistema y la conexión entre ellos sitúa cada parte en la perspectiva de conjunto: confiere estructura a la división. Tal conexión presenta un triple aspecto: comunicación física entre los bloques, funcionalidad de cada uno de ellos respecto a los demás y ordenación temporal de las señales entre ellos (conexión física, conexión funcional y conexión temporal).

El nombre de *arquitectura*, sumamente apropiado a nuestro entender, destaca la idea de una estructura que conecta (funcional y físicamente) partes diversas; conocida la situación de cada parte en la estructura, es posible un tratamiento individualizado de la misma.

En general, esta división en partes presenta dos peldaños o etapas iniciales: una primera división en módulos, referidos a las especificaciones o prestaciones del sistema, y posterior acomodación a bloques, en relación con los recursos booleanos (subsistemas digitales).

En la aplicación concreta de esta metodología de *división estructural* a los sistemas digitales vamos a considerar a continuación dos casos de particular interés y sumamente frecuentes.

3. Sistemas con mucha transferencia de información

En el caso de elevada transferencia de información (bien porque el sistema necesite almacenar gran cantidad de datos, bien porque su comunicación con el exterior sea muy intensa y diversa) resulta sumamente eficaz la división en dos partes, procesador/"memoria", conectadas por buses, es decir, una arquitectura basada en la comunicación a través de buses.

La utilización de buses diferenciados permite organizar tanto el almacenamiento como la transferencia de información en forma simple: todos los registros se numeran correlativamente accediendo a ellos por las mismas líneas (datos), un segundo conjunto de líneas seleccionan el registro sobre el que se opera (direcciones) y unas pocas líneas de control determinan la operación a realizar y sincronizan la transferencia de información (control).

La organización de los buses puede ser aplicada de formas muy diversas (memoria única, arquitectura Harvard, módulos especializados, ...) y, además de la división en dos partes (procesador-"memoria"), permite abordar por separado el diseño de cada elemento de la "memoria", una vez establecida la numeración de sus registros.

4. Sistemas con esquema de actuación complejo

Respecto a los sistemas o subsistemas digitales dedicados directamente al procesamiento de la información (tareas de cálculo y de control) y cuyo esquema de actuación es complejo, puede resultar sumamente útil diferenciar la *parte operativa* de la *parte de control* y abordar el diseño de cada una de ellas por separado. Es decir, identificar los recursos de cálculo y registros necesarios (parte operativa) y su secuencia de actuación (control), diferenciando las operaciones a realizar y la forma en que se realizan; ello permite el diseño individual de cada registro o recurso de cálculo y, asimismo, el diseño por separado de la parte de control.

El control corresponde a un grafo o máquina de estados que puede ser muy amplia, pero cuya estructura resulta relativamente simple: es, por lo general, del tipo de autómatas de Moore,

básicamente lineal (sucesión directa de estados) con bucles, avances y alternativas [3]. Su diseño puede ser abordado en dos formas conceptualmente muy diferentes:

- directamente como grafo de estados, reflejando las transiciones en condiciones de activación de sus biestables individuales --forma microcableada--
- recogiendo el estado sobre un registro global y codificando las transiciones de los estados a partir del estado y del vector de entradas --forma microprogramada--.

Respecto a su realización "microcableada", la codificación con "un solo uno" resulta muy apropiada para tales máquinas de estado pues permiten identificar cada estado (y el conjunto de acciones asociadas al mismo) con un biestable y trasladar directamente el grafo de estados (especialmente si se dibuja en forma de "ordinograma") a su configuración circuital.

Respecto a su configuración "microprogramada", el tamaño del codificador (que contiene las microinstrucciones) aumenta fuertemente con el número de entradas (dependencia según 2^m , siendo m el nº de entradas) y con el número de salidas (proporcional al nº de las mismas); interesa, por ello, utilizar técnicas de multiplexado o de codificación que reduzcan el número de entradas y de salidas efectivas del codificador.

La importancia y posible complejidad de la parte de control justifica dedicar tiempo y esfuerzo a considerar diferentes alternativas de ambas configuraciones, para facilitar su diseño y reducir el circuito resultante; así como posibilidades intermedias como puede ser el aprovechamiento de contadores para el estado ("jump counter") [4].

La limitación de espacio impide incluir ejemplos concretos aclaratorios de la división parte operativa-parte de control (como, por ejemplo, un multiplicador de números de 64 bits, basado en algoritmo de suma y desplazamiento [3] o, más complejo, el cálculo de B mod A).

En los procesadores que actúan bajo programa es clásica la separación entre la parte operativa (ALU y registros) y la parte de control; en esta segunda, el estado se encuentra conformado básicamente por la instrucción a ejecutar (su parte inicial referida al código de operación) y un contador de ciclos en los que se desglosa dicha ejecución. La realización "microprogramada" de dicha máquina de estados es de particular interés y el correspondiente codificador coincide con el denominado "decodificador de instrucciones" (figura 1). (Téngase en cuenta que el contador de programa, aunque es parte relevante en el control de la secuencia de instrucciones, no realimenta su estado hacia dentro de la parte de control).



Figura 1: Parte de control de un procesador (microprogramada)

5. Aplicación sucesiva de particiones

Ni qué decir tiene que muchos sistemas digitales presentan, a la vez, las dos situaciones anteriormente descritas (mucha transferencia de información y esquema de actuación complejo) y, en consecuencia, les son de aplicación las dos particiones antes indicadas. Por otra parte, el sincronismo no es sino una partición aplicada al tiempo que, al cuantificarlo en unidades, facilita el diseño y le confiere seguridad funcional [5].

El esquema de "fraccionar sin perder la globalidad" puede ser aplicado sucesivamente en forma jerárquica (iterativamente), es decir, a partir de un primer esquema de bloques, los bloques complejos pueden ser divididos en "subbloques", dando lugar a un segundo nivel de "esquemas de bloques" y así sucesivamente hasta "hacer manejable" la complejidad. La propia máquina de estado relativa a la "parte de control" puede segmentarse, caso de resultar compleja, en varias máquinas de estado comunicadas entre sí.

Conceptualmente, la idea de "particionar" se opone a la práctica "chapucera" de incorporar "parches" y hacer añadidos "al paso", incluyendo al socaire del diseño nuevas condiciones mediante biestables adicionales; toda especificación o variable de diseño debe estar integrada en el conjunto y quedar reflejada en los grafos de estado (no es buena técnica desarrollar máquinas de estado dejado biestables fuera de las mismas). De vez en cuando, en el desarrollo de prototipos, conviene volver atrás y reescribirlo todo, replanteando situaciones y estados en esquemas de conjunto, que se reflejarán en los correspondientes grafos de estado.

Nunca insistiremos demasiado en la necesidad de "escribir", de documentar el diseño, de describir cada detalle, cada variable, cada registro, ... (tanto su función propia como su interrelación con el conjunto); en el ahorro de tiempo que, en definitiva, supone el texto escrito tanto para clarificar ideas, como para transmitir las, no solamente a otras personas, sino también a uno mismo con el transcurso del tiempo.

Junto con las anteriores reflexiones conceptuales y formulaciones metodológicas, conviene tener presente la importancia de la experiencia: *Sabe más el diablo por viejo que por diablo*. La experiencia es un requisito insustituible para el "saber hacer" (*know-how*): difícilmente será posible abordar con eficacia el diseño de sistemas complejos sin el bagaje de una experiencia conseguida a través de la práctica, con la dedicación personal y el proceso temporal que ello supone.

Esta comunicación resume uno de los temas de nuestro programa de microelectrónica, cuyo texto ponemos a disposición de los interesados (correo electrónico: tpollan@posta.unizar.es).

Referencias

- [1] L. A. López Nozal. *La electrónica digital dentro de la Tecnología Electrónica en los nuevos planes de estudio*. V Jornadas de Tecnología Electrónica, Santander (1994).
- [2] G. G. Scarrott. *From computing slave to knowledgeable servant: the evolution of computers*. Proc. R. Soc. Lon. A. **369**, 1-30 (1979).
- [3] T. Pollan. *Electrónica Digital*. Prensas Universitarias Zaragoza (1994).
- [4] R. H. Katz. *Contemporary Logic Design*. The Benjamin/Cumming P. Co. (1994).
- [5] T. Pollán y B. Martín. *Sincronismo, Reloj, Tiempo y Camino*. TAEE (2000).