http://ansinet.com/itj

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 6 (6): 843-850, 2007
ISSN 1812-5638


# Synthesis of Finite State Machine Adopting the Controller-data Path Approach: Performance Evaluation of Different Methods

Maria Rizzi, Matteo D'Aloia and Beniamino Castagnolo
Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari,
V.E. Orabona, 4-70125 Bari, Italy

**Abstract:** In the design procedure of complex digital sequential machines, it is useful to adopt the controller-data path approach more than to implement a very high complexity sequential circuit. In this approach, all the functional and memory options are concentrated in an operative unit (data path) while the control signals are generated by a simpler sequential machine (controller). In order to compare this approach with the traditional one, the design of a control unit implemented in CMOS AMS 0.35 technology, for accessing one memory shared among different processors is performed. To specify the control sequences and data processing tasks of the designed digital system, a hardware algorithm has been adopted and the corresponding algorithm state machine chart has been indicated. Moreover, various specialized synthesis methods are studied and their behaviour has been simulated with orcad software tool. Both the circuit speed and the power dissipation for the different implementations have been evaluated and compared.

**Key words:** Sequential machine, hardware algorithm, algorithm state machine, flip-flop, complex machine synthesis

## INTRODUCTION

Specifications of a modern digital system are often so complex to need a detailed characterization of both the functionality sets (such as arithmetic, decoding functionalities, etc.) and the memories between which data are transferred, according to a well defined synchronization.

In describing the behaviour of a complex sequential machine, the designer can detail the functional characteristics following the next steps (Morris Mano, 1991; Fumi *et al*., 2002; Tinder, 2000; Harmanani and Saliba, 2000; Altaf-Ul-Amin *et al*., 2002; Bhatia and Jha, 1998):

- To define the operation set in accordance with the machine specifications and to determine the necessary functional blocks;
- To characterize the operation and the storage sequencing;
- To design the data path section that manipulates the informations according to system requirements;
- To define the control signals to guarantee both the functional block activation, the data stream and the memory storage of the obtained results;
- To design the controller that provides command signals to supervise the various operations inside the data section in order to accomplish the desired data-processing tasks.

The distribution of tasks between data path and controller can differ according to both the functional blocks disposable in the design libraries and the design requirements such as area occupancy, data throughput, cost, latency, low power dissipation (Fumi *et al*., 2002; Harmanani and Saliba, 2000). Therefore, designers have to achieve the target function respecting the assigned constrains. If, for example, the target is the latency and the constrain is the implementation cost, the system has to provide the results after the shortest possible period of time has elapsed. Amongst the various solutions with the minimum latency, the optimum circuit must have the lowest cost.

In this study after a brief general description of the data path and the controller, their relationship in digital systems is indicated. The design procedure is specified by means of a digital hardware algorithm, adopting the Algorithm State Machine (ASM) chart.

To evaluate the performance of different ASM synthesis methods, the design of a control unit for access

**Corresponding Author:** Maria Rizzi, Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari V.E. Orabona, 4-70125 Bari, Italy

to a memory shared among different processors is performed. Moreover, some conclusions are drawn out.

## THE ASM CHART

The logic design of digital systems can be divided into two distinct blocks. One part is concerned with the design of digital circuits that perform the data-processing operations. The other block is concerned with the design of the control section that supervises the operations and their sequencing. A general scheme of the architecture is indicated in Fig. 1. The data path section manipulates data inside registers according to system requirements. The controller sequences the data path logic and uses status conditions coming from the data section as decision variables for determining the sequence of control signals. Therefore, the controller is a sequential circuit whose internal state are related to the control signals of the system. In fact, it provides the timing signals for starting the operations in the data path and determines the next state of the control section.

Control sequences and data processing tasks of a digital system can be specified by means of a hardware algorithm which is a procedure for implementing a specific problem taking into account the hardware equipment. To define digital hardware algorithms, the Algorithmic State Machine (ASM) chart is adopted. It is a convenient tool that translates the system specifications to an information diagram in which the sequence of operations and the conditions necessary for their execution are indicated. Differently from a conventional flow-chart, the ASM diagram describes the sequence of events as well as the time relationship between the state of the sequential machine and the events occurred during the change from one state to another.

The ASM chart is composed of three basic elements (Fig. 2):

- The state box inside which the register operations or the outputs generated by the control unit are indicated;
- The decision box that describes the effect of inputs on the controller, therefore it contains the input conditions on which depend the branching from a given state;
- The conditional box that specifies outputs associated with state transitions for a given input in a Mealy machine.

One or more ASM blocks made up an ASM chart; each of them is composed of one state box and all the condition and decision boxes connected to its exit path. Each ASM block presents in an ASM diagram, represents the state of the digital system during one clock pulse interval.

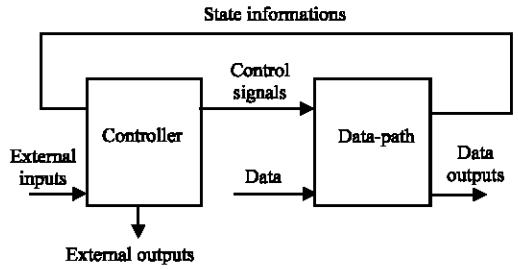


Fig. 1: Controller and data path relation

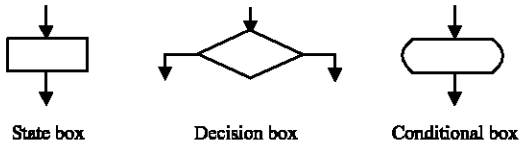


Fig. 2: Basic elements of an ASM chart

It is evident the similarity between an ASM chart and a state diagram of a synchronous sequential machine.

The operations specified inside the state and conditional boxes of each ASM block are performed in the data-path subsection while the change from one state to another is realized in the controller.

The design of data-path requires both the interpretation of the operations and their implementations adopting digital components such as registers, counters, multiplexers, adders and so on, while the synthesis of controller requires all the procedures adopted for the sequential machine.

In modern digital systems high level synthesis is necessary so machine design and implementation derive from a behavioral description. Therefore specialized methods have to be implemented.

## SYNTHESIS ADOPTING THE ASM CHART

To evaluate the performance of some specialized synthesis methods, the design of a control unit for access to a memory shared between two processors is performed. The connection is provided to one processor at time, adopting a priority procedure. In fact when a contemporary memory access request is forwarded, the processor with major priority is connected and its priority level become lower then that of the other processor whose connection request was denied.

Indicating with X and Y the Boolean variables representing the connection request on behalf of processor 1 and processor 2, respectively, the related ASM chart for the Moore automa is indicated in Fig. 3.

Analyzing the ASM chart it is evident that the digital circuit is composed of six distinguishable states. The requirements for the data-path design are specified inside

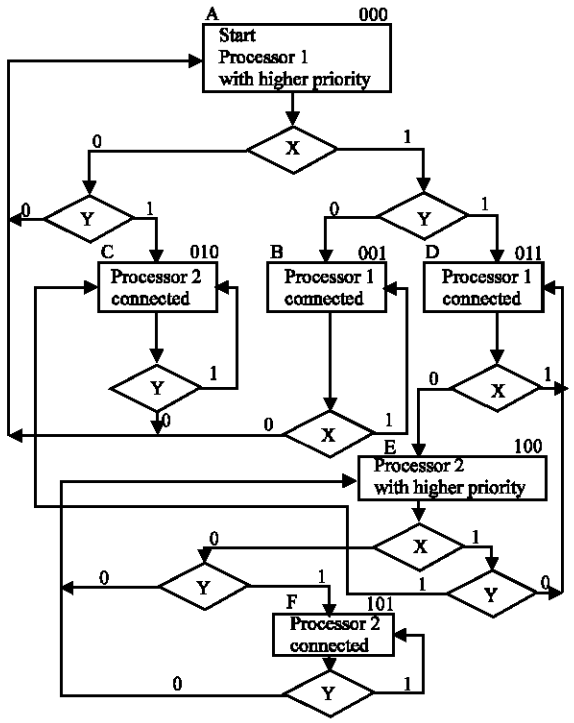


Fig. 3: Representation of the ASM Chart

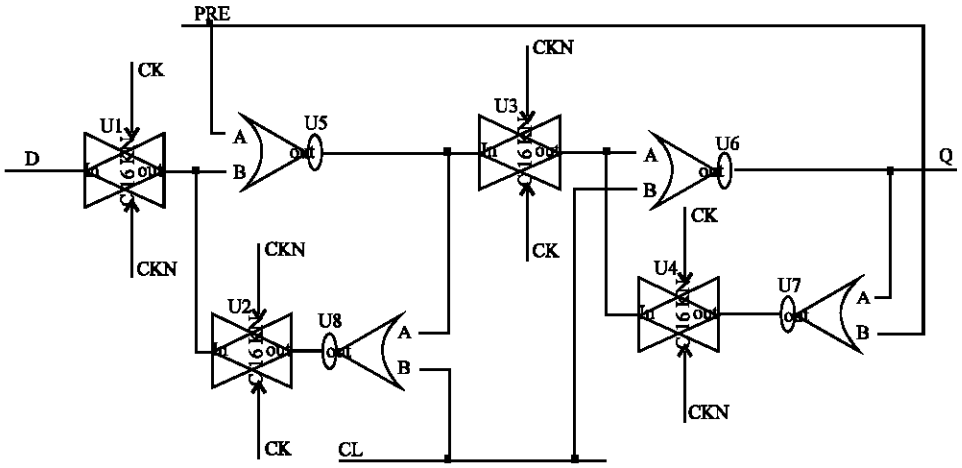


Fig. 4: D Flip-flop logical diagram

the state boxes while the control logic is determined from the decision boxes and the required state transitions.

All the logical devices necessary for the hardware implementation are realized in AMS 0.35 CMOS technology.

**Two different specialized methods are adopted:**

- The implementation with decoder and D Flip-Flops (circuit A);
- The implementation with one Flip-Flop for state (circuit B).

Dynamic type positive-edge triggered D-Flip-Flops with the Preset and Clear external asynchronous inputs are used (Fig. 4). For this reason, the transmission gates composing the memory elements are controlled by two complementary phase signals.

In Fig. 5 the circuit diagram of the adopted D-Flip-flop is shown.

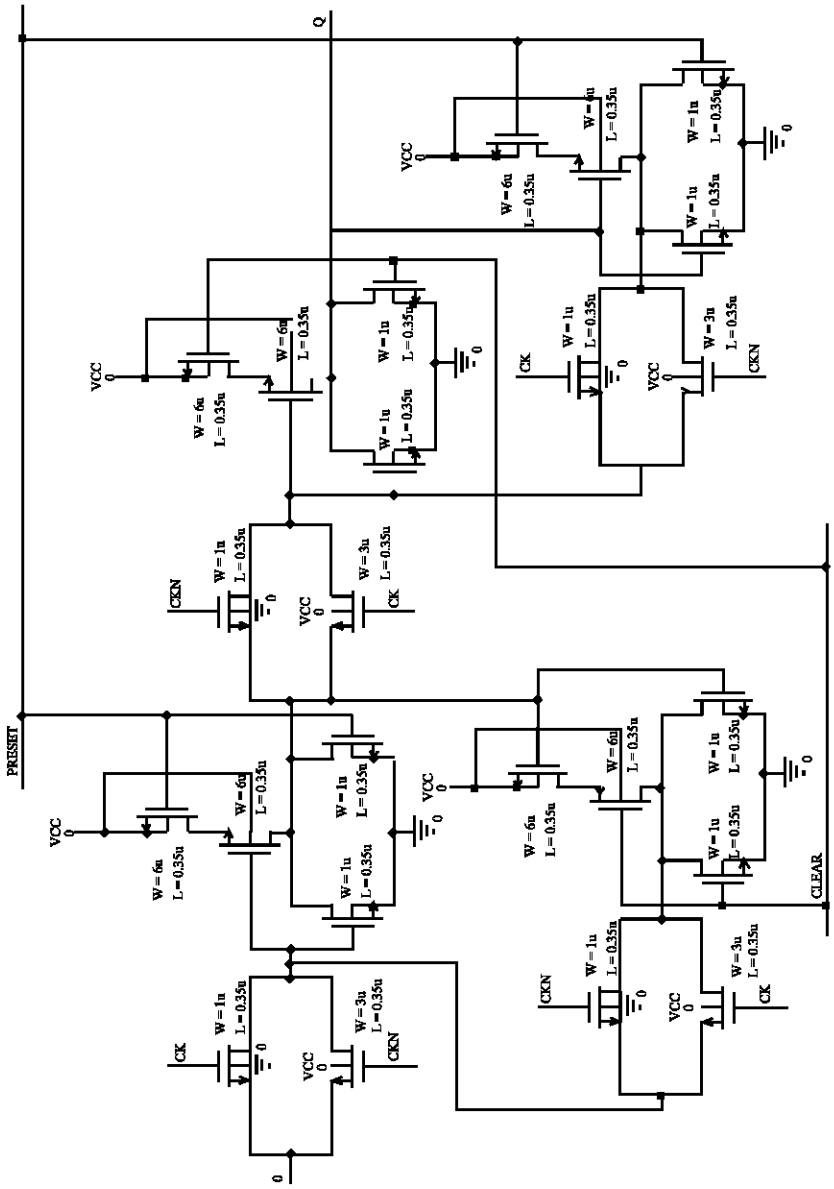


Fig. 5: D Flip-flop circuit diagram

**Synthesis with decoder and D flip-flops:** The adoption of D type memory elements simplify the design because the input functions can be obtained directly from the state table as it coincides with the excitation table.

As the ASM contains six state boxes, the logical controller has composed of a register with 3 memory cell and a 3×8 decoder.

Analyzing the ASM chart, the state table for the logical control is directly derived as indicated in Table 1 where x represents a don't care condition.

The Flip-Flop excitation functions can be derived from the state table, directly:

$$D_{G_2} = \overline{\overline{(D+E)\cdot x}\cdot F}$$

$$D_{G_1} = \overline{\overline{(A+C)\cdot y}\cdot\overline{(D+E)\cdot x}}$$

$$D_{G_0} = \overline{\overline{(A+B+D)\cdot x}\cdot\overline{E(\overline{xy}+\overline{xy})}\cdot\overline{F\cdot y}}$$

In Fig. 6 the logical schema of the controller adopting a decoder and NAND/NOR gates is indicated.

**Synthesis with one Flip-Flop for state (assignment one hot):** Adopting this method, six D Flip-Flops are necessary but only one Flip-Flop is set to the logical 1 value for every clock cycle.

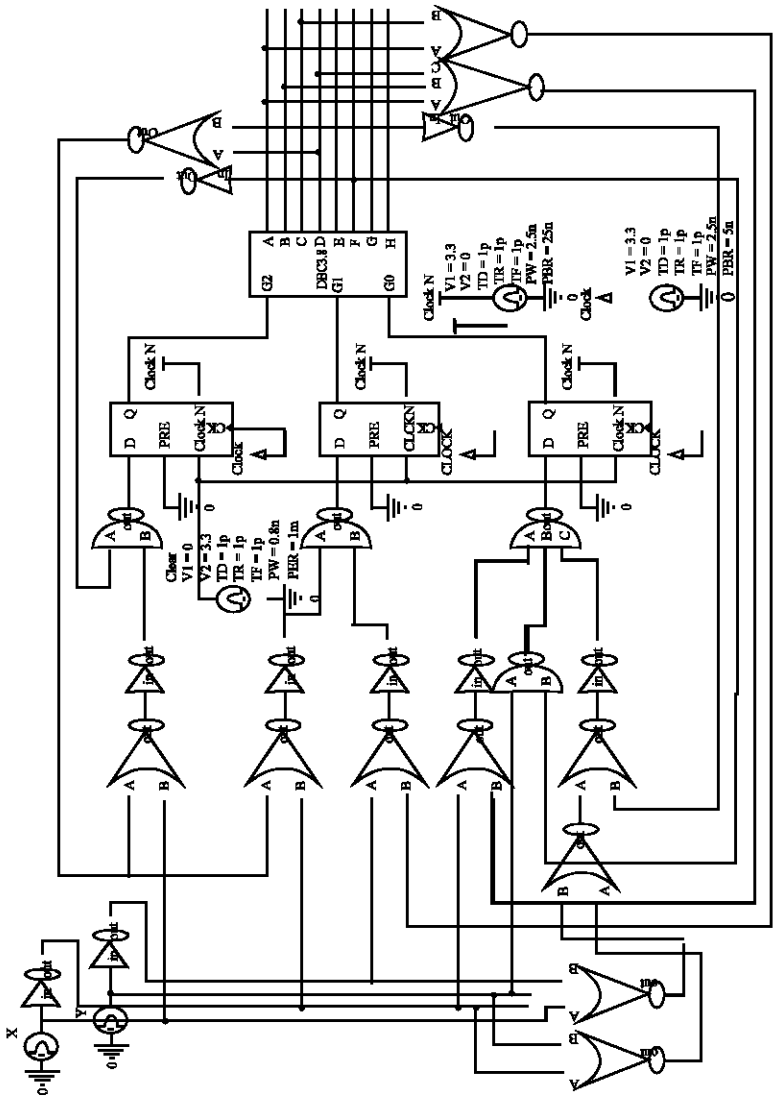

Fig. 6: Controller logical scheme with decoder

Table 1: Logical control state table

| | Inputs | | Next state | | | Outputs | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Present state | X | Y | $G_2$ | $G_1$ | $G_0$ | A | B | C | D | E | F |
| A | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| A | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| A | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| A | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| B | 0 | x | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| B | 1 | x | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| C | x | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| C | x | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| D | 0 | x | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| D | 1 | x | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| E | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| E | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| E | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| E | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| F | x | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| F | x | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |

The one-hot code is used for the state assignment phase as indicated in Table 2.

The excitation functions for all the memory elements are derived from the ASM chart, directly:

$$D_A = \overline{A \cdot \bar{x} \cdot \bar{y} \cdot B \cdot \bar{x} \cdot C \cdot \bar{y}}$$

$$D_B = \overline{A \cdot x \cdot \bar{y} \cdot \overline{B \cdot x}}$$

$$D_C = \overline{A \cdot \bar{x} \cdot y \cdot \overline{C \cdot y} \cdot \overline{E \cdot x \cdot y}}$$

$$D_D = \overline{A \cdot x \cdot y \cdot \overline{D \cdot x} \cdot \overline{E \cdot x \cdot y}}$$

$$D_E = \overline{D \cdot \bar{x} \cdot E \cdot \bar{x} \cdot \bar{y} \cdot F \cdot \bar{y}}$$

$$D_F = \overline{E \cdot \bar{x} \cdot y \cdot F \cdot \bar{y}}$$

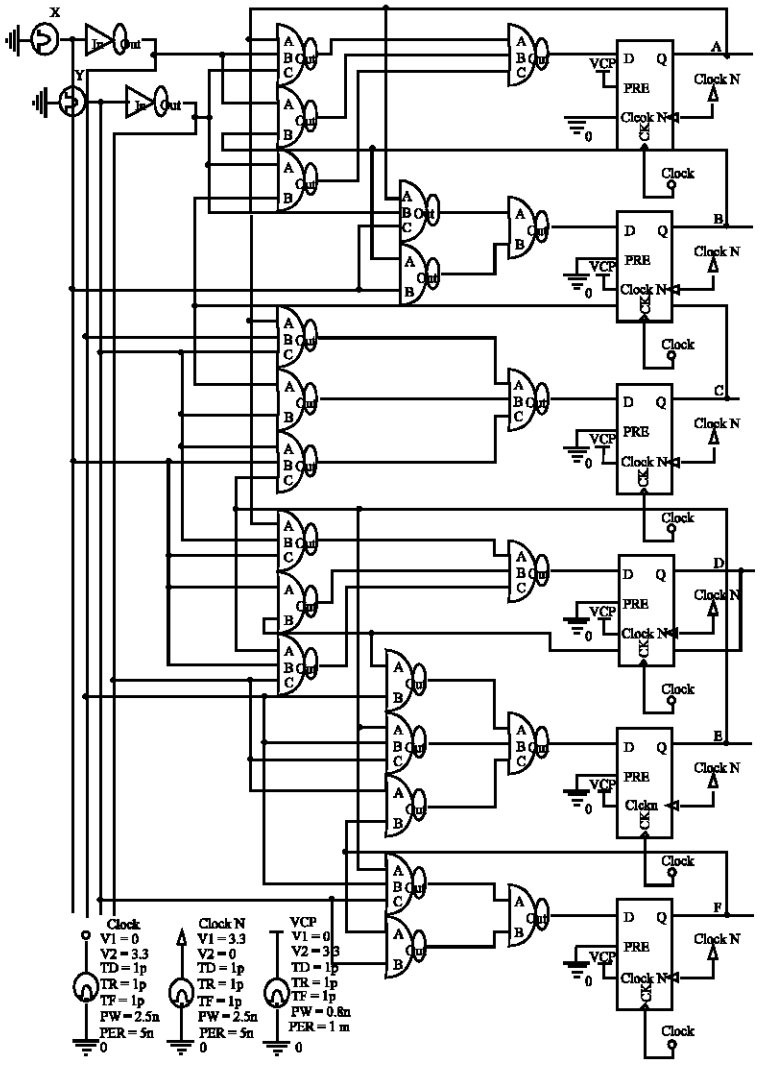


Fig. 7: Controller logical schema with the one-hot assignment

Table 2: The state assignment

| | State variables | | | | | |
|---|---|---|---|---|---|---|
| State | A | B | C | D | E | F |
| A | 1 | 0 | 0 | 0 | 0 | 0 |
| B | 0 | 1 | 0 | 0 | 0 | 0 |
| C | 0 | 0 | 1 | 0 | 0 | 0 |
| D | 0 | 0 | 0 | 1 | 0 | 0 |
| E | 0 | 0 | 0 | 0 | 1 | 0 |
| F | 0 | 0 | 0 | 0 | 0 | 1 |

In Fig. 7 the logical controller of the Finite State Machine synthesized with the one-hot code is shown.

## PERFORMANCE EVALUATION

To compare the performance of the two specialized synthesis methods, the behaviour of the previously described circuits has been simulated adopting Orcad software tool. To analyze the logical machines, various input sequences have been chosen with particular interest for the bit sequences that cause the transition of the automas to all the states composing the ASM chart.

To start the analysis, the resetting all the three D Flip-Flops in the design with decoder and the setting of the first D Flip-Flop and resetting of the others in the design with one Flip-Flop for state, has been necessary. To this aim, the Preset and Clear inputs have been used.

Adopting the CMOS AMS 0.35 technology and a 3.3V voltage supply, both the circuit speed and the power dissipation have been evaluated and compared for the two different implementations.

To evaluate the circuit speed performance, different clock frequencies have been adopted and the related output signal delays have been compared.

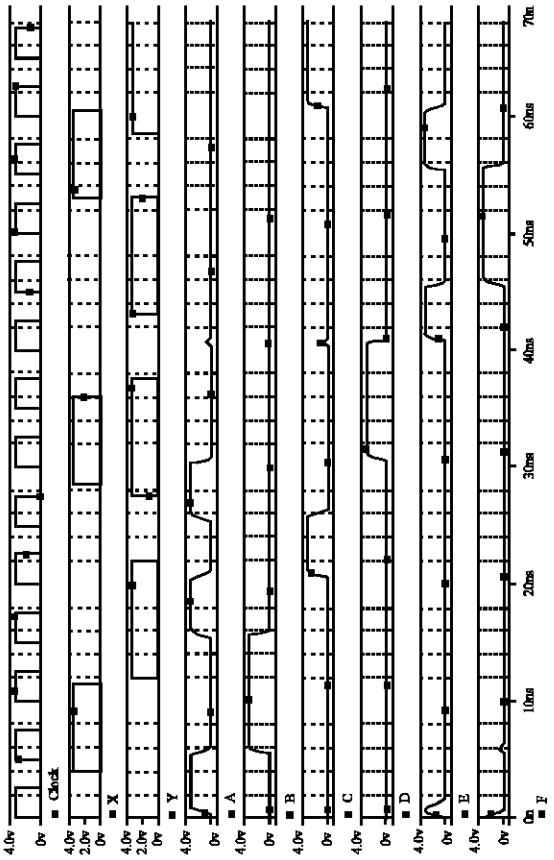


Fig.8a: Simulation results for the design with decoder and 200 MHZ clock frequency

Table 3: Expected state transition applying (X,Y) input sequence

| Clolk edge (ns) | 0 | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 | 55 | 60 | 65 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| Y | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | | 1 | 0 | 1 | 1 | 1 |
| Present state | A | B | B | A | C | A | D | D | E | F | F | E | C | C |

Applying a 200 MHZ clock signal, all the two circuits show a behaviour according to the design specs. Table 3 indicates a generic input sequence and the corresponding expected state machine transitions.

Adopting as input bits the sequence indicated in the previous table, Fig. 8a and b show the simulation results of circuit A and circuit B, respectively.

The analysis shows the correspondence between the expected state transitions and the simulated results adopting a 200 MHZ clock frequency. In fact, comparing Fig. 8a and b with Table 3 it is evident the correct working of the two circuits.

The spikes show in Fig. 8a are logical hazards due to the different number of gates that the output signals have to pass through. In circuit B, every output signal pass through an equal number of gates therefore no spike compares in the time domain analysis.

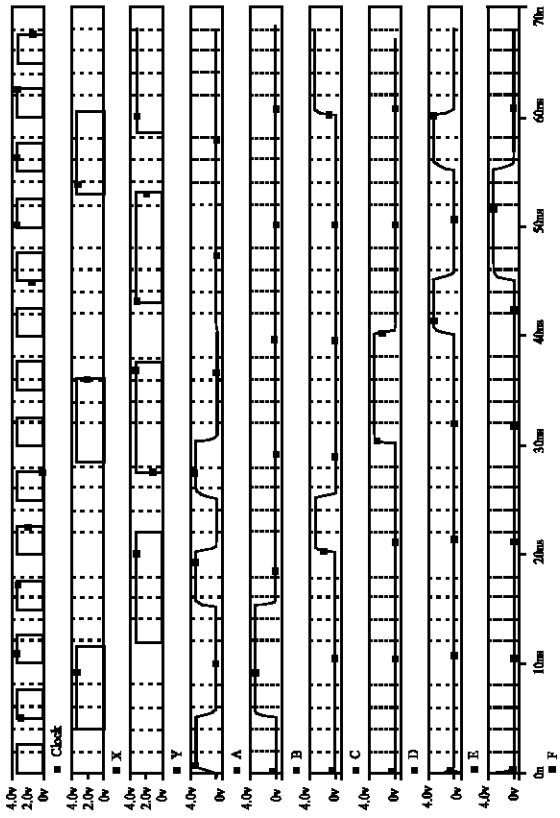


Fig. 8b: Simulation results for the design with one hot code and 200 MHZ clock frequency

Increasing the clock frequency, the studies show that circuit B meets the design requirements up to 350 MHZ while circuit A over 200 MHZ produces output signals so delayed that the obtained state transition differs from the expected one.

To better study the two synthesis methods, also the average circuit power dissipation has been evaluated as parameter of comparison. The analysis shows that circuit A has a lower power dissipation than circuit B. In fact, applying a 250 MHZ clock signal, circuit A has an average power dissipation equal to 4.34 mW while for circuit B the value is 7.15 mW. Increasing the clock frequency up to 350 MHZ, circuit B has a power dissipation equal to 9.27 mW. Comparing the two values obtained for the same circuit, it is evident the direct relation between frequency and power dissipation.

## CONCLUSION

In modern digital systems the classical synthesis of finite state machines is impractical for their complexity, therefore specialized methods have to be implemented. In

this study the design of complex digital machine is performed adopting the Algorithm State Machine procedure. Two different synthesis methods have been implemented: the method with decoder and the method with one Flip Flop for state. Comparing the obtained circuits it is evident that the circuit with one Flip flop for state operates following the specs up to 350 MHZ but has a power dissipation higher that the circuit with decoder.

## REFERENCES

Altaf-Ul-Amin, M., S. Ohtake and H. Fujiwara, 2002. Design for two-pattern testability of controller-data path circuits. In IEEE 11th Asian Test Symposium (ATS'02).

Bhatia, S. and N.K. Jha, 1998. Integration of hierarchical test generation with behavioral synthesis of controller and data path circuits. IEEE Trans. Very Large Scale Integration (VLSI) Syst., 6: 608-619.

Fumi, F., M.G. Sami and C. Silvano, 2002. Progettazione Digitale. McGraw-Hill.

Harmanani, M.H. and R. Saliba, 2000. An evolutionary approach for data path synthesis. In: 2000 Canadian Conference on Electrical and Computer Engineering, 7-10 March 2000, 1: 380-384.

Mano, M. Morrs, 1991. Digital Design. Prentice Hall International Editions.

Tinder, R.F., 2000. Engineering Digital Design. Academic Press.