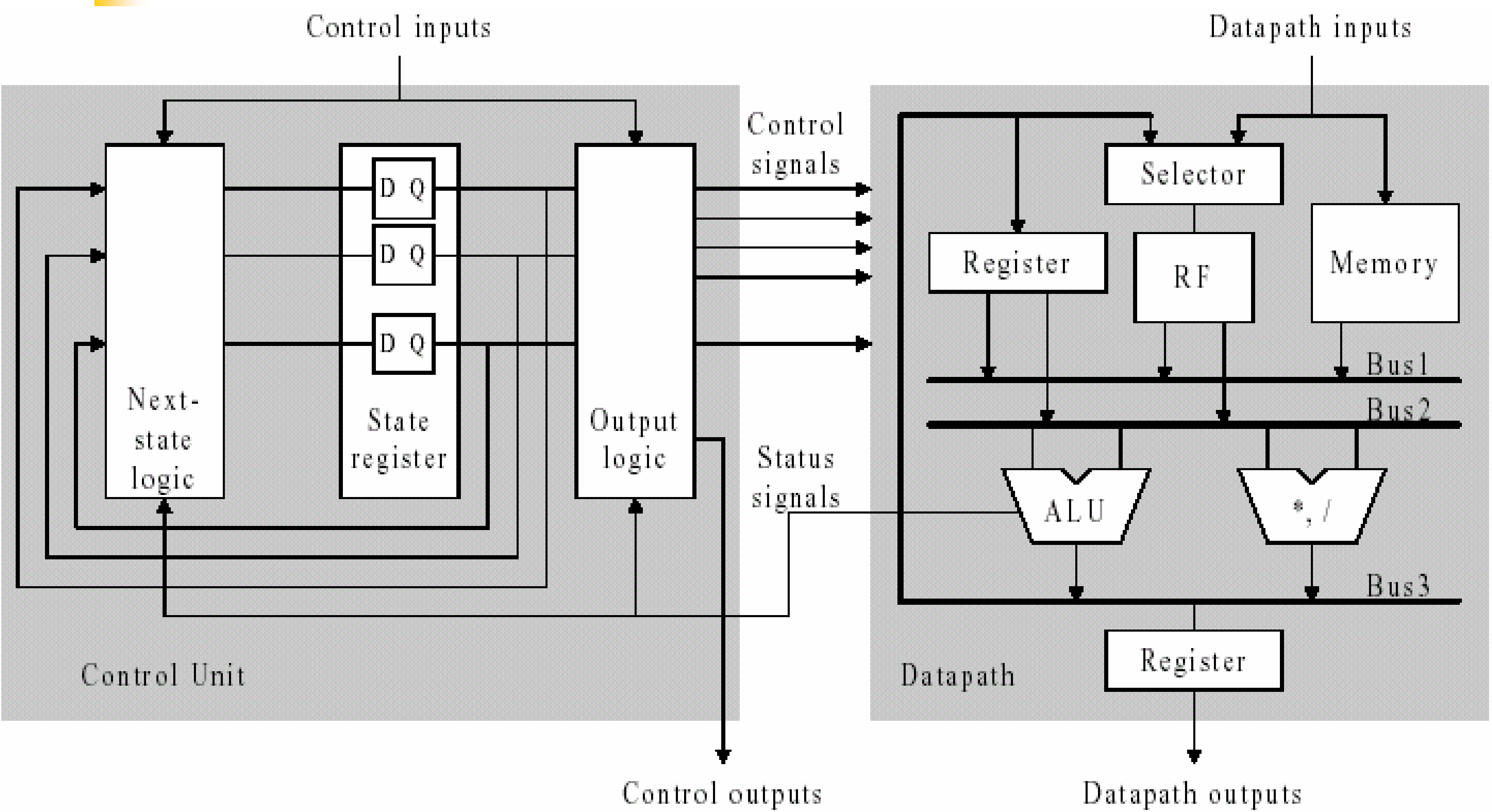
Generalized Additive Decomposition

corresponds to a given cover on the set of states of prototype FSM

RTL design model



Finite state machine with data-path (FSMD)

An FSMD is formulated as a quintuple:

$$< S,\ I \times SS,\ O \times AS,\ \delta,\ \lambda >$$

S is the set of states of the FSMD
$I \times SS$ is the set of inputs of the FSMD.
Inputs extended with status expressions
$O \times AS$ is the set of outputs of the FSMD.
Outputs extended with variable assignments

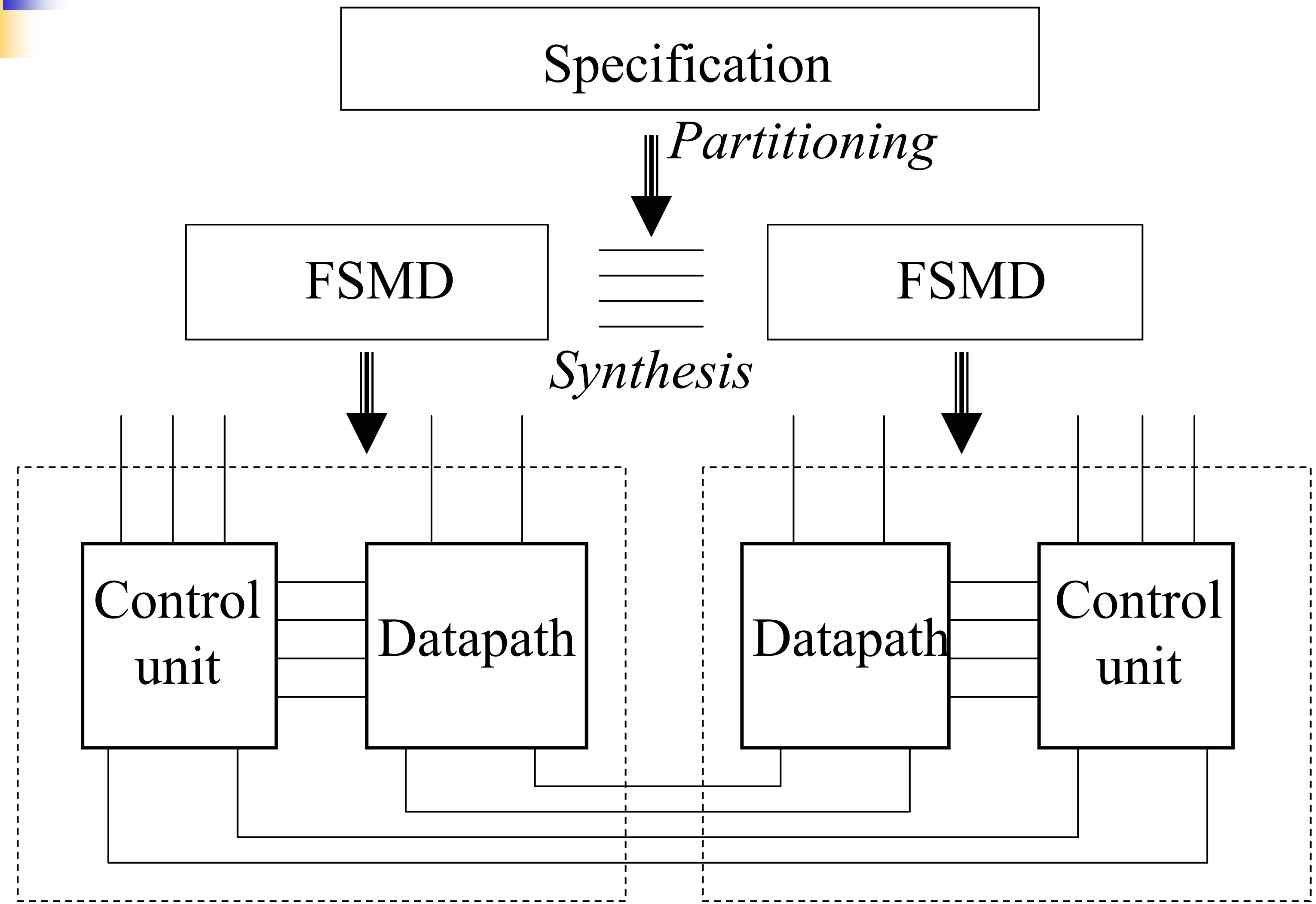
δ is the next state function, mapping

$$S \times (I \times SS) \rightarrow S$$

λ is the output function, mapping

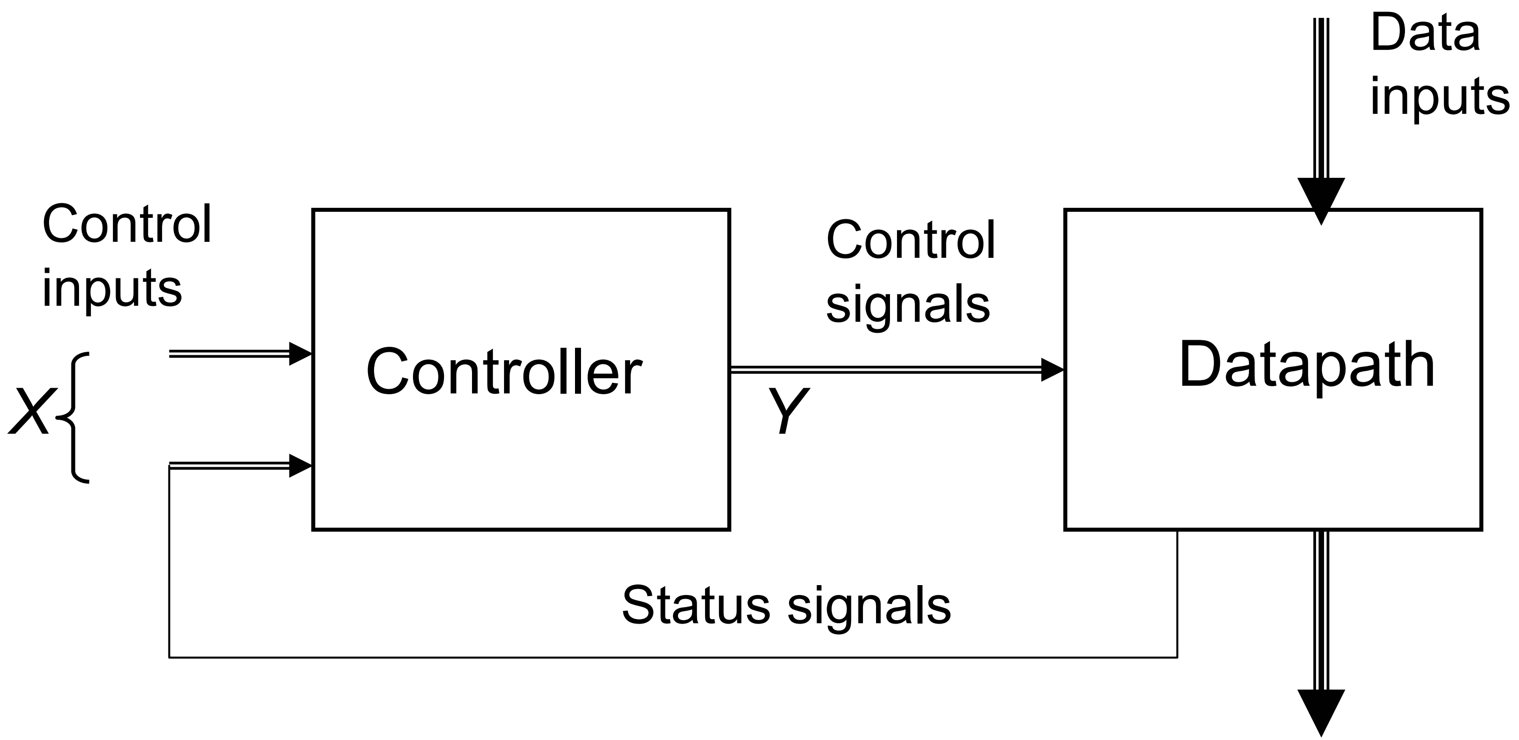
$$S \times (I \times SS) \rightarrow (O \times AS)$$

Functional partition approach



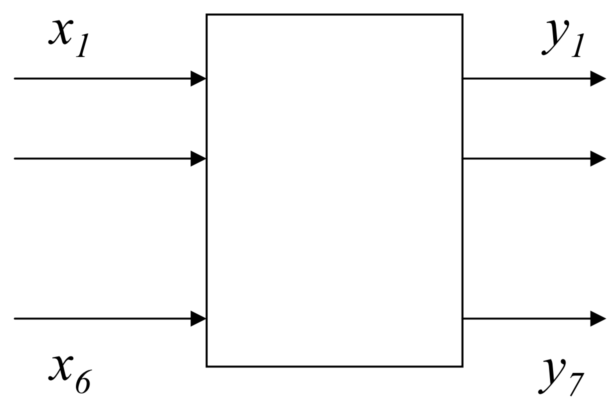
Register transfer level (to controller description)

We investigate a register transfer level design that considers both the controller and data-path simultaneously.



Example 1 controller transition table

Present state s_p	Next state s_q	Input condition $\alpha(s_p, s_q)$	Output signal Y	h
s_1	s_1	$\hat{}x_1$ & x_2	-	1
	s_2	$\hat{}x_1$ & $\hat{}x_2$	$y_1\ y_2$	2
	s_3	x_1	$y_2\ y_4$	3
s_2	s_2	x_3 & x_4	y_5	4
	s_2	$\hat{}x_3$	y_6	5
	s_4	x_3 & $\hat{}x_4$	$y_5\ y_6$	6
s_3	s_1	x_5	$y_2\ y_4$	7
	s_1	$\hat{}x_5$ & x_6	y_4	8
	s_3	$\hat{}x_5$ & $\hat{}x_6$	$y_3\ y_4$	9
s_4	s_2	x_3	y_7	10
	s_3	$\hat{}x_3$	$y_5\ y_7$	11



$$S = \{ s_1 , s_2 , s_3 , s_4 \}$$

Induced cover on the set of states of controller

Present state s_p	Next state s_q	Output signal Y	h
s_1	s_1	-	1
	s_2	y_1 y_2	2
	s_3	y_2 y_4	3
s_2	s_2	y_5	4
	s_2	y_6	5
	s_4	y_5 y_6	6
s_3	s_1	y_2 y_4	7
	s_1	y_4	8
	s_3	y_3 y_4	9
s_4	s_2	y_7	10
	s_3	y_5 y_7	11

Initial partition:

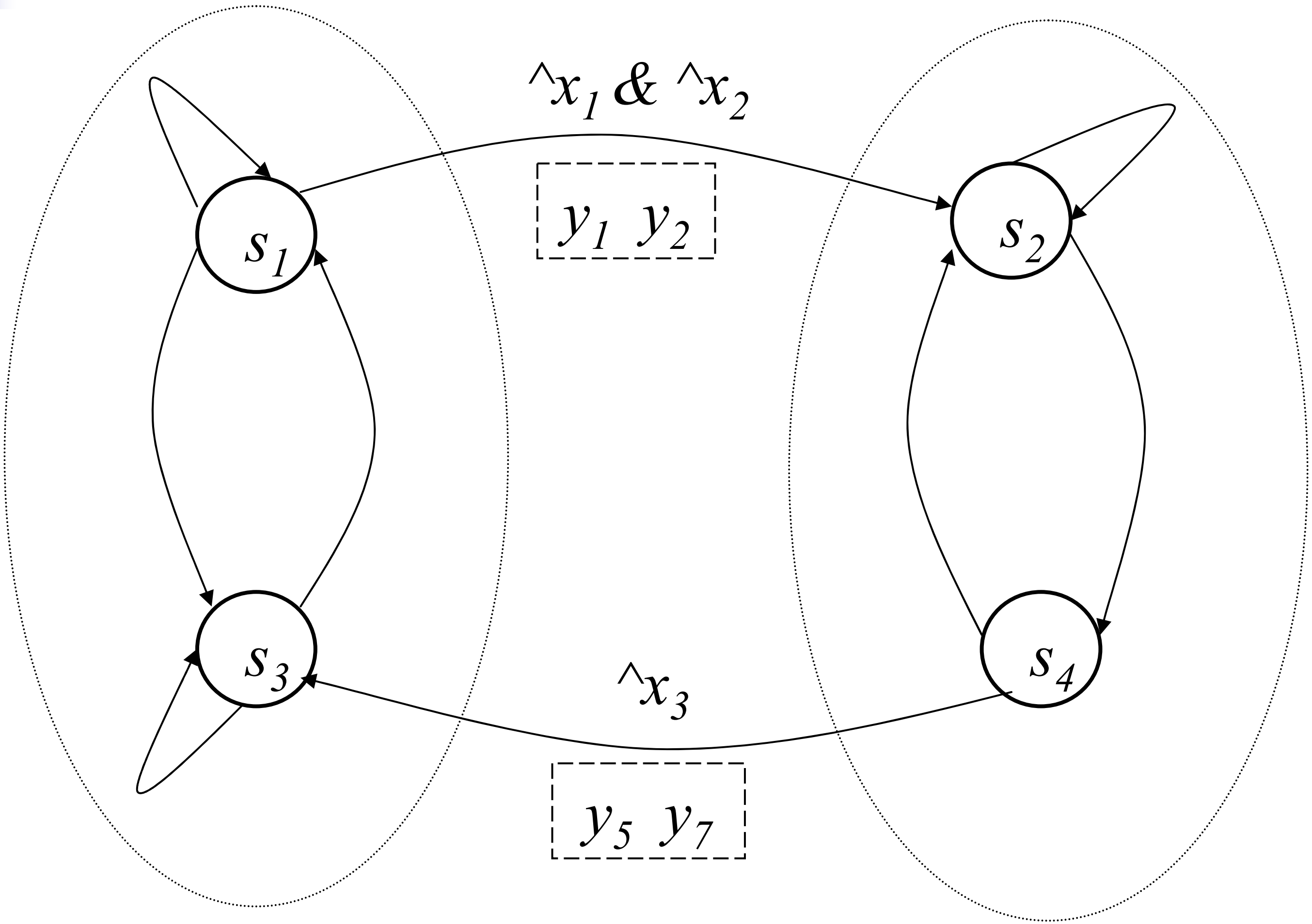
$\{ \{ y_1, y_2, y_3, y_4 \} ; \{ y_5, y_6, y_7 \} \}$

Induced partition (cover):

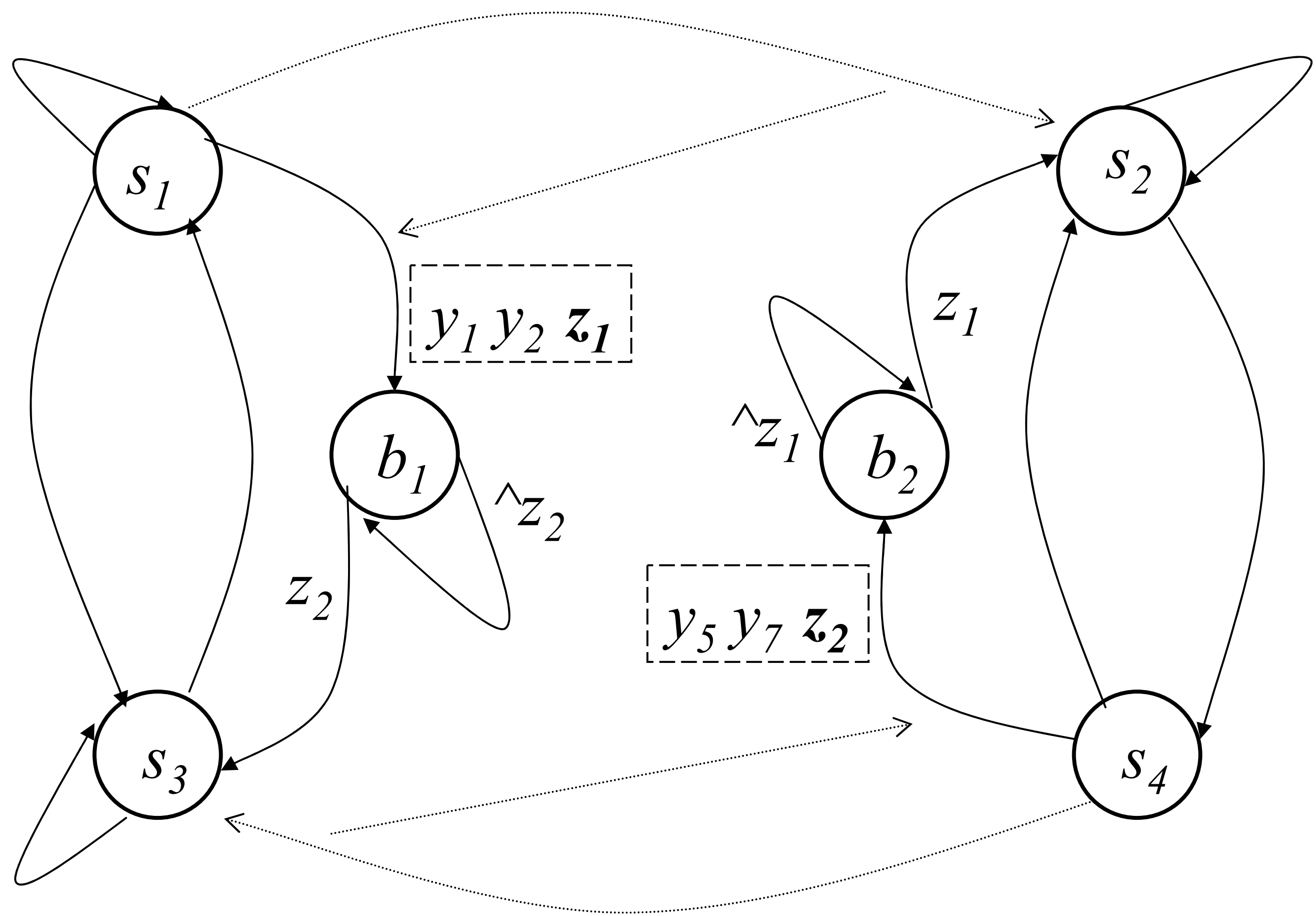
$\{ \{ s_1, s_3 \}; \{s_2, s_4 \} \}$

A collection φ of nonempty subsets of a set S whose union is S is called a ***cover*** on S. The notion of cover s generalization of a partition, that is a collection of *disjoint* subsets of S whose set union is S.

Transition graph of controller



Network controller components



The first component's transition tables

Present state	Next state	Input condition	Output signals	*h*
s_1	s_1	$\wedge x_1$ & x_2	-	1
	s_3	x_1	y_2 y_4	2
	b_1	$\wedge x_1$ & $\wedge x_2$	y_1 y_2 $\mathbf{z_1}$	3>
s_3	s_1	x_5	y_2 y_4	7
	s_1	$\wedge x_5$ & x_6	y_4	8
	s_3	$\wedge x_5$ & $\wedge x_6$	y_3 y_4	9
b_1	s_3	z_2	-	>11
	b_1	$\wedge z_2$	-	

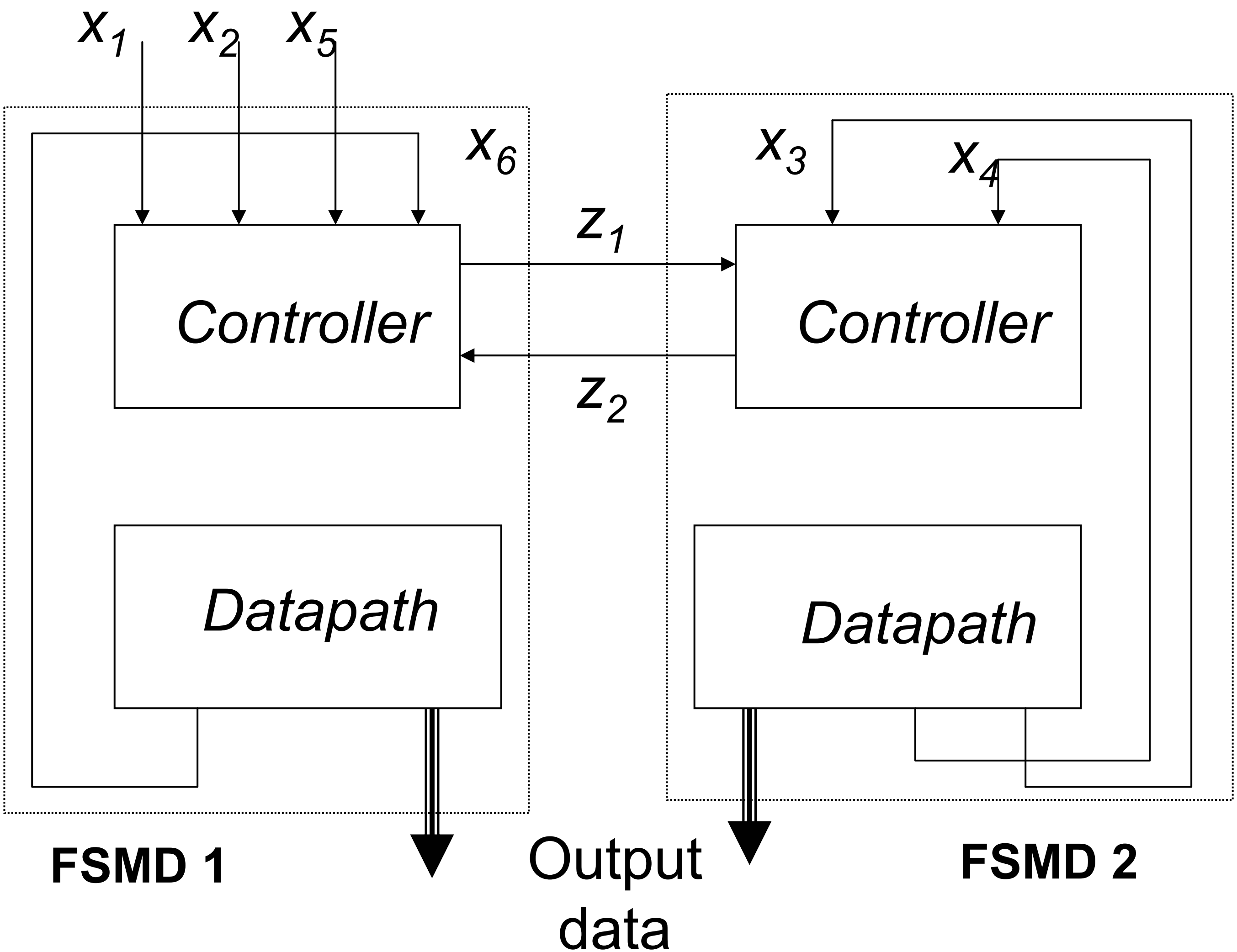
b_1 is idle (wait) state

The second component's transition tables

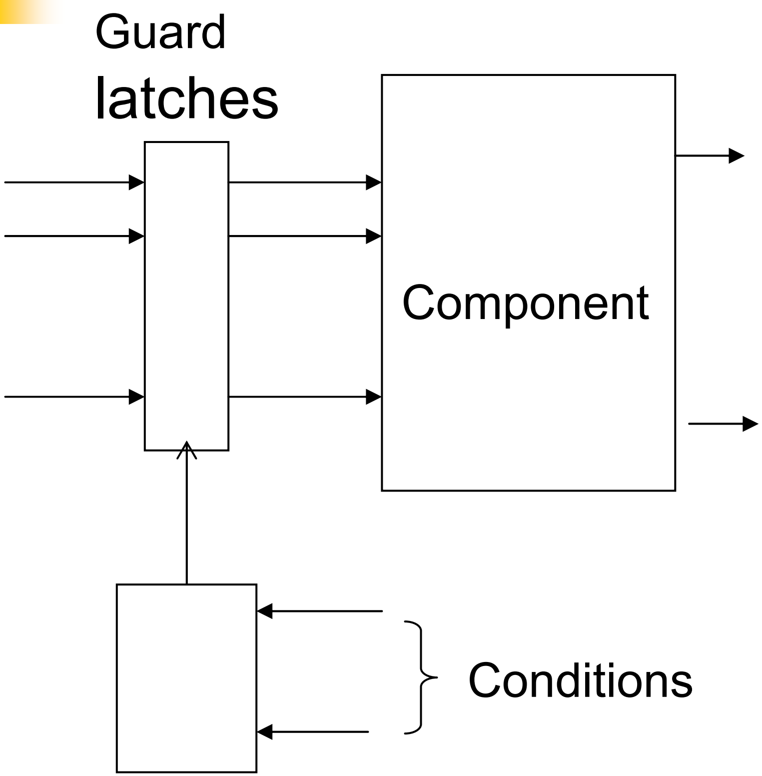
Present state	Next state	Input condition	Output signals	*h*
s_2	s_2	x_3 & x_4	y_5	4
	s_2	^x_3	y_6	5
	s_4	x_3 & ^x_4	y_5 y_6	6
s_4	s_2	x_3	y_7	10
	b_2	^x_3	$y_5 y_7$ z_2	11>
b_2	s_2	z_1	-	>2
	b_2	^z_1	-	

b_2 is idle (wait) state

FSMD network



Low power synthesis using precomputation architectures



To avoid unnecessary switching activity we partition original description into the network of components and place guards (transparent latches with an enable input) at the inputs of those parts of the network that need to be selectively turned off. If the module is to be active in a clock cycle, the enable signal makes the latch transparent, permitting normal operation. If not, the latch retains its previous state and no transitions propagate through it.

Example 2 (prototype controller behavior description)

s_1	s_1	x_1	y_7	1
	s_3	$\hat{}x_1$	-	2
s_2	s_3	1	$y_2\ y_{10}$	3
s_3	s_6	x_7	$y_2\ y_{10}$	4
	s_8	$\hat{}x_6 \ \&\ x_7$	-	5
	s_2	$\hat{}x_6 \ \&\ \hat{}x_7$	$y_2 y_5\ y_{10}$	6
s_4	s_1	$x_1 \ \&\ x_2$	$y_3\ y_4$	7
	s_3	$x_1 \ \&\ \hat{}x_2$	$y_1\ y_3\ y_4$	8
	s_5	$\hat{}x_1 \ \&\ x_4$	$y_6\ y_9$	9
	s_8	$\hat{}x_1 \ \&\ \hat{}x_4$	$y_6\ y_8\ y_9$	10
s_5	s_4	$x_3 \ \&\ x_4$	$y_6\ y_9$	11
	s_5	$x_3 \ \&\ \hat{}x_4$	$y_6\ y_9$	12
	s_8	$\hat{}x_3$	$y_6\ y_8$	13
s_6	s_2	x_5	$y_2\ y_{10}$	14
	s_3	$\hat{}x_5 \ \&\ x_7$	y_5	15
	s_8	$\hat{}x_5 \ \&\ \hat{}x_7$	$y_2\ y_{10}$	16
s_7	s_5	1	y_1	17
s_8	s_8	x_3	-	18
	s_5	$\hat{}x_3$	$y_8\ y_9$	19

$S = \{ s_1 , s_2 , \ldots , s_8 \}$

$X = \{ x_1 , x_2 , \ldots , x_7 \}$

$Y = \{ y_1 , y_2 , \ldots , y_{10} \}$

Decomposition procedure

- Generation of the initial partition
- Definition of states in component FSMs
- Definition of the set of external input variables of component FSM
- Generation of the set of the set of internal (additional) input variables
- Definition of the set of output variables of component FSMs
- Generation of transition and output functions of component FSMs
- Realization of FSM network.

The set of states of (component) sub-FSM

$\pi = \{Y^1, \dots Y^n\}$ is the partition on the set of **output variables** Y.

$G = \{g_1, \dots g_H\}$ is the set of g-transitions in the transition table (in our example every g-transition corresponds to row in table 1),

$X(g_h)$ and $Y(g_h)$ are the sets of essential input and output variables (microoperations) in the g-transition g_h $(h = 1, \dots H)$,

$X(s_i)$ and $Y(s_i)$ are the sets of input and output variables at the transitions from the state s_i .

$\varphi = \{B^1, \dots , B^n\}$; $B^p \subseteq S$, $s \in B^p \Leftrightarrow Y(s) \cap Y^p \neq \varnothing$;

$\psi = \{G^1, \dots , G^n\}$; $G^p \subseteq G$,

$g \in G^p \Leftrightarrow Y(m) \cap Y^p \neq \varnothing$.

Set of external input variables

s_1	s_1	y_7	1
	s_3	-	2
s_2	s_3	$y_2\ y_{10}$	3
s_3	s_6	$y_2\ y_{10}$	4
	s_8	-	5
	s_2	$y_2 y_5\ y_{10}$	6
s_4	s_1	$y_3\ y_4$	7
	s_3	$y_1\ y_3\ y_4$	8
	s_5	$y_6\ y_9$	9
	s_8	$y_6\ y_8\ y_9$	10
s_5	s_4	$y_6\ y_9$	11
	s_5	$y_6\ y_9$	12
	s_8	$y_6\ y_8$	13
s_6	s_2	$y_2\ y_{10}$	14
	s_3	y_5	15
	s_8	$y_2\ y_{10}$	16
s_7	s_5	y_1	17
s_8	s_8	-	18
	s_5	$y_8\ y_9$	19

Initial partition on *Y*:

$$\pi = \{ \{y_1, y_3, y_4, y_7\}, \{y_6, y_8, y_9\}, \{y_2, y_5, y_{10}\}$$

Induced cover on S:

$$\varphi = \{\{s_1, s_4, s_7\}, \{s_4, s_5, s_8\}, \{s_2, s_3, s_6\}\}$$

Induced cover on *G*:

$$\psi = \{\{1, 2, 7, 8, 17\}, \{9, 10, 11, 12, 13, 18, 19\}, \{3, 4, 5, 6, 14, 15, 16\}\}$$

Decomposition procedure

Let us put the network N with n component FSMD

$$A^p = < S^p, X^p, V^p, \delta^p, \lambda^p >, \quad p = 1, \ldots n,$$

in accordance to triplet $< A, \pi, \varphi, \psi >$.

The number of component machines is equal to the number of blocks in the partition π (or the cover φ and ψ).

(Further the steps of decomposition procedure are presented in formal way)

Decomposition procedure

$S^p = B^p \cup \{b_p\}$ is the set of states in the component controller C^p ,
where B^p is the p-th block of the cover φ, and b_p is additional state in C^p.

$X^p = X(G^p) \cup Z_x^p$ is the set of input variables in the component controller C^p.
Here $X(G^p) = \cup_{m \in G^p} X(g_h)$

$X(g_h)$ is the set of essential input variables in the g-transition g_h of the controller C of the prototype FSMD;
G^p is the p-th block of the cover ψ.
$Z_x^p = \{z_t \,/\, \delta(s_j\,,\ \alpha_h) = s_t\,;\ s_t \in S^p\,,\ s_j \notin S^p\}$.
$V^p = Y^p \cup Z_y^p$
$Z_y^p = \{z_i \,/\, \delta(s_t\,,\ \alpha_h) = s_j\,;\ s_t \in S^p\,,\ s_j \in S^k,\ s_t \notin S^k\}$.

Decomposition procedure

Assume that there is a g-transition in of prototype FSMD

$$< s_j,\ s_t,\ \alpha_h,\ \beta_h >$$

(the transition from s_j to s_t with the input condition α_h and the output β_h in the controller C):

$\delta(s_j\ ,\ \alpha_h) = s_t;\ \lambda(s_j\ ,\ \alpha_h) = \beta_h$.

Define the corresponding transitions in component controller.

Ω_j be the set of component controllers with the state s_j,

Ω_t be the set of component controllers with the state s_t,

$\Omega_{jt} = \Omega_j \cap \Omega_t$ be the set of component controllers with the states s_j and s_t

If $C^k \in \Omega_{jt}$, then in C^k $\quad \delta^k(s_j\ ,\ \alpha_h) = s_t$.

If $A^p \in \Omega_j \setminus \Omega_{jt}$, then in C^p $\delta^p(s_j\ ,\alpha_h) = b_p$,

(s_i is the state of C^p and s_t is not the state of C^p)

Decomposition procedure

The output controlling datapath of q^{th} sub-FSMD for controller $C^q \in \Omega_j$ is equal to $\beta_h \cap Y^q$
(s_i is the state of C^q and it is also possible that the next state s_t is the state of C^q).

In addition to controlling outputs, one and only one controller from Ω_j (say C^r), must generate the output signal which forces each controller from $C^u \in \Omega_t \setminus \Omega_{jt}$ (if this set is not empty) to transit from the additional (idle) state b_u to s_t and in C^q
$\lambda^r (s_j, \alpha_h) = \beta_h \cap Y^u \cup \{z_t\}$.

If $C^u \in \Omega_t \setminus \Omega_{jt}$ (s_t is the state of C^u and s_j is not the state of C^u), then in C^u
$\delta^u(b_u , z_t) = s_t$
$\lambda^u (b_u, z_t) = Y_0$.

The initial state of the component controller is equal to s_0 , if $C^p \in \Omega_1$ and b_p otherwise.

The first component transition table

The set of states of the first component is the states from the first block of cover φ plus b_1, $S^1 = \{s_1, s_4, s_7, b_1\}$.

Present state s_i	Next state s_j	Input condition α_h	Output signals β_h	h
s_1	s_1	x_1	y_7	1
	b_1	$\hat{}x_1$	z_3	2_a
s_4	s_1	x_1 & x_2	y_3 y_4	7
	b_1	x_1 & $\hat{}x_2$	y_1 y_3 y_4 z_3	8_a
	b_1	$\hat{}x_1$	-	g_1
s_7	b_1	1	y_1 z_5	17_a
b_1	s_4	z_4	-	11_b
	b_1	$\hat{}z_4$	-	d_1

The set of outputs consists of variables from Y which are in g-transitions from the first block of cover ψ plus additional output variable z_3 because there is transition from s_4 to s_3 in the original transition table, but s_3 is the state of the second component C^2.

The set of inputs consists of variables which are in g-transitions from the first block of cover ψ plus additional input variable z_4 because exists g-transition (11) not included in the first block of cover ψ with next state s_4.

The second component transition table

Present state s_i	Next state s_j	Input condition α_h	Output signals β_h	h
s_2	s_3	1	$y_2\ y_{10}$	3
s_3	s_6	x_7	$y_2\ y_{10}$	4
	b_2	$\hat{}x_6\ \&\ x_7$	z_8	5_a
	s_2	$\hat{}x_6\ \&\ \hat{}x_7$	$y_2\ y_5\ y_{10}$	6
s_6	s_2	x_5	$y_2\ y_{10}$	14
	s_3	$\hat{}x_5\ \&\ x_7$	y_5	15
	b_2	$\hat{}x_5\ \&\ \hat{}x_7$	$y_2\ y_{10}\ z_8$	16_a
b_2	s_3	z_3	-	$2_b/8_b$
	b_2	$\hat{}z_3$	-	d_3

The third component transition table

Present state s_p	Next state s_n	Input condition α_h	Output signals β_h	h
s_5	s_4	x_3 & x_4	y_6 y_9 z_4	11
	s_5	x_3 & ^x_4	y_6 y_9	12
	s_8	^x_3	y_6 y_8	13
s_8	s_8	x_3	-	18
	s_5	^x_3	y_8 y_9	19
s_4	s_5	^x_1 & x_4	y_6 y_9	9
	s_8	^x_1 & ^x_4	y_6 y_8 y_9	10
	b_3	x_1	-	g_2
b_3	s_8	z_8	-	$5_b/16_b$
	s_5	z_5	-	17_b
	b_3	^z_5 & ^z_8	-	d_2

Transitions between sub-machines

Note, that $\Omega_4 = \{ C^1, C^2 \}$ and $\Omega_3 = \{C^3\}$, but **only C^1 generates output z_3.**

Note, that when there is transition to the state s_4 both controllers C^1 and C^2 are activated but only one of them will generate outputs controlling own data-path.

The data-path of other machine will be silent (it depends on the value of input variable x_1).

FSMD network (example 2)