State Assignment Using Partition Pairs

- Discuss hypercube method, add slides later on

State Assignment Using Partition Pairs

- This method allows for finding high quality solutions but is slow and complicated
- Only computer approach is practical
- **Definition of Partition.**
  - Set of blocks B_i is a partition of set S if the union of all these blocks forms set S and any two of them are disjoint
  - B1 u B2 u B3 … = S
  - B1 /\ B2={}, B2 /\ B3 … = {}, etc
  - Example 1: {12,45,36}, {{1,2},{4,5},{3,6}}
  - Example 2: {123,345} not a partition but a **set cover**

State Assignment Using Partition Pairs

- **Definition of X-successor of state S_a**
  - The state to which the machine goes from state S_a using input X
- **Definition of Partition Pair**
  - P1=> P2 is a partition pair if for every two elements S_a and S_b from any block in P1 and every input symbol X_i the X_i successors of states S_a and S_b are in the same block of P2

State Assignment Using Partition Pairs

Methods of calculation of Partition Pairs

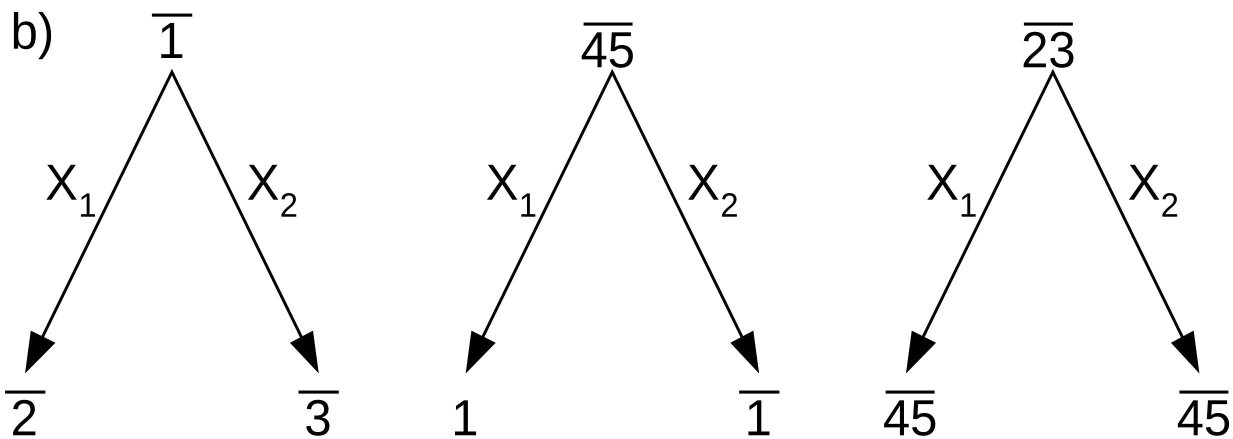
- Partition pair P1 => P2 calculated with known partition P1
- Partition Pair P1 => P2 calculated with known partition P2

Multi-line method
Multi-line method
Multi-line method

Calculation of successor partition from the predecessor partition in the partition pair

{1,23,45} => ???

	X_1	X_2
1	2	3
2	4	5
3	5	4
4	1	1
5	1	--



{1,23,45}=>{1,2,3,45}

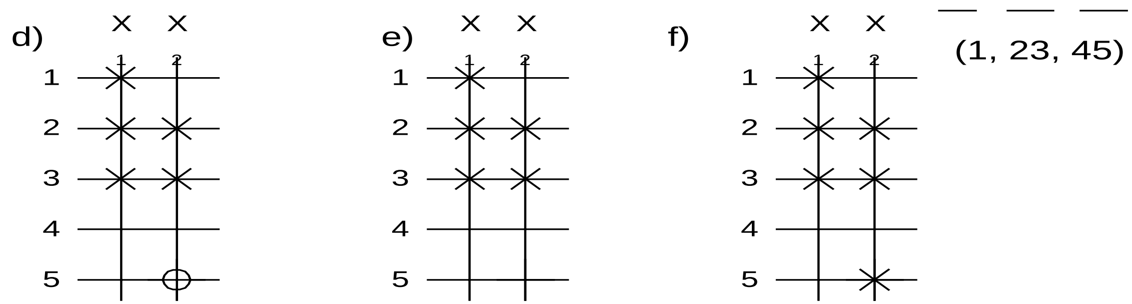
Calculation of successor partition from the successor partition in the partition pair

0 1

{WHAT??} => {13,245}

	X_1	X_2
1	2	3
2	4	5
3	5	4
4	1	1
5	1	--

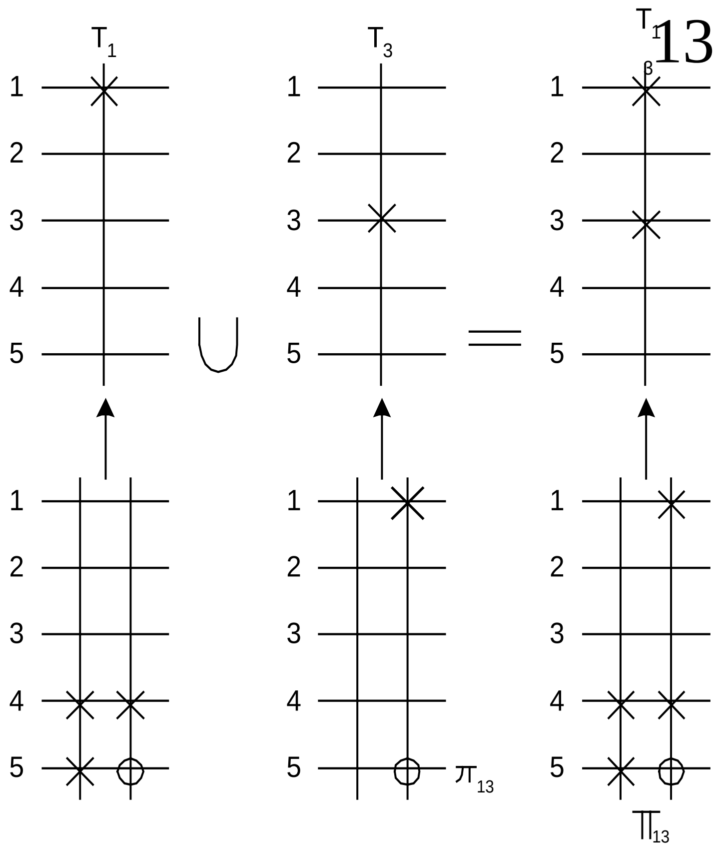
	X_1	X_2
1	1	0
2	1	1
3	1	1
4	0	0
5	0	--



{1,23,45} => {13,245}

Machine M1

Operations on Partitions represented as Multi-lines



{{1}, {2,3,4,5}} u {{3},{1,2,4,5}}={{1,3},{2,4,5}}

{1, 2345} u {3,1245}={13,245}

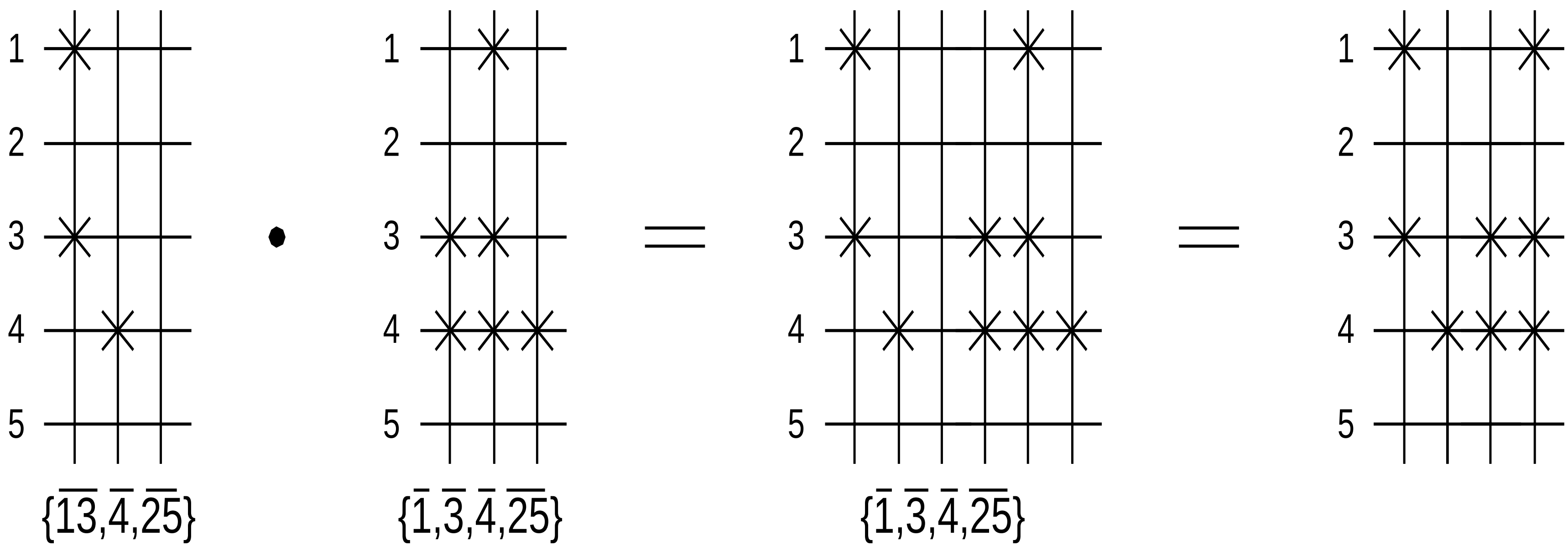
Union of images of predecessors

THIS IS **NOT** A SUM OF PARTITIONS OPERATION

Operations on Partitions represented as Multi-lines

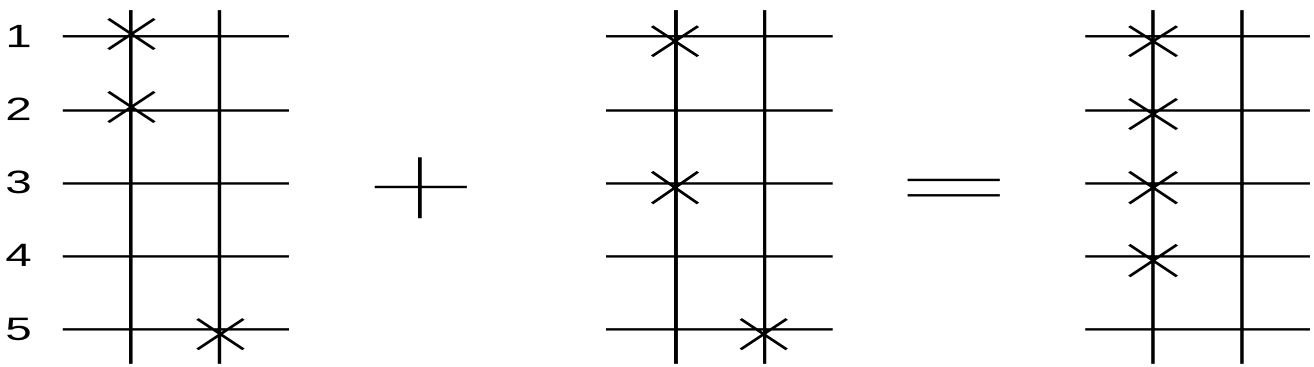
Intersection (called also a product) of partitions

b)



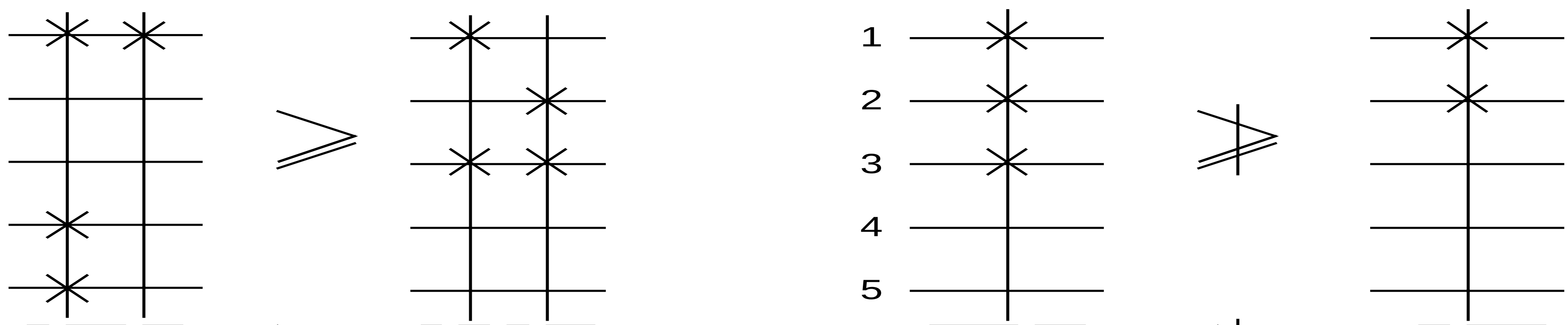
Operations on Partitions represented as Multi-lines

This is SUM of partitions. Every two states that are in one block in any of the arguments must be in one block of the result.



$\{\overline{12},\overline{34},\overline{5}\} + \{\overline{13},\overline{24},\overline{5}\} = \{\overline{1234},\overline{5}\}$

Combine Overlapping Blocks



$\{\overline{1},\overline{23},\overline{45}\} \geqslant \{\overline{1},\overline{2},\overline{3},\overline{45}\}$

$\{\overline{123},\overline{45}\} \ngeqslant \{\overline{12},\overline{345}\}$

- These methods are used to find a good state assignment.

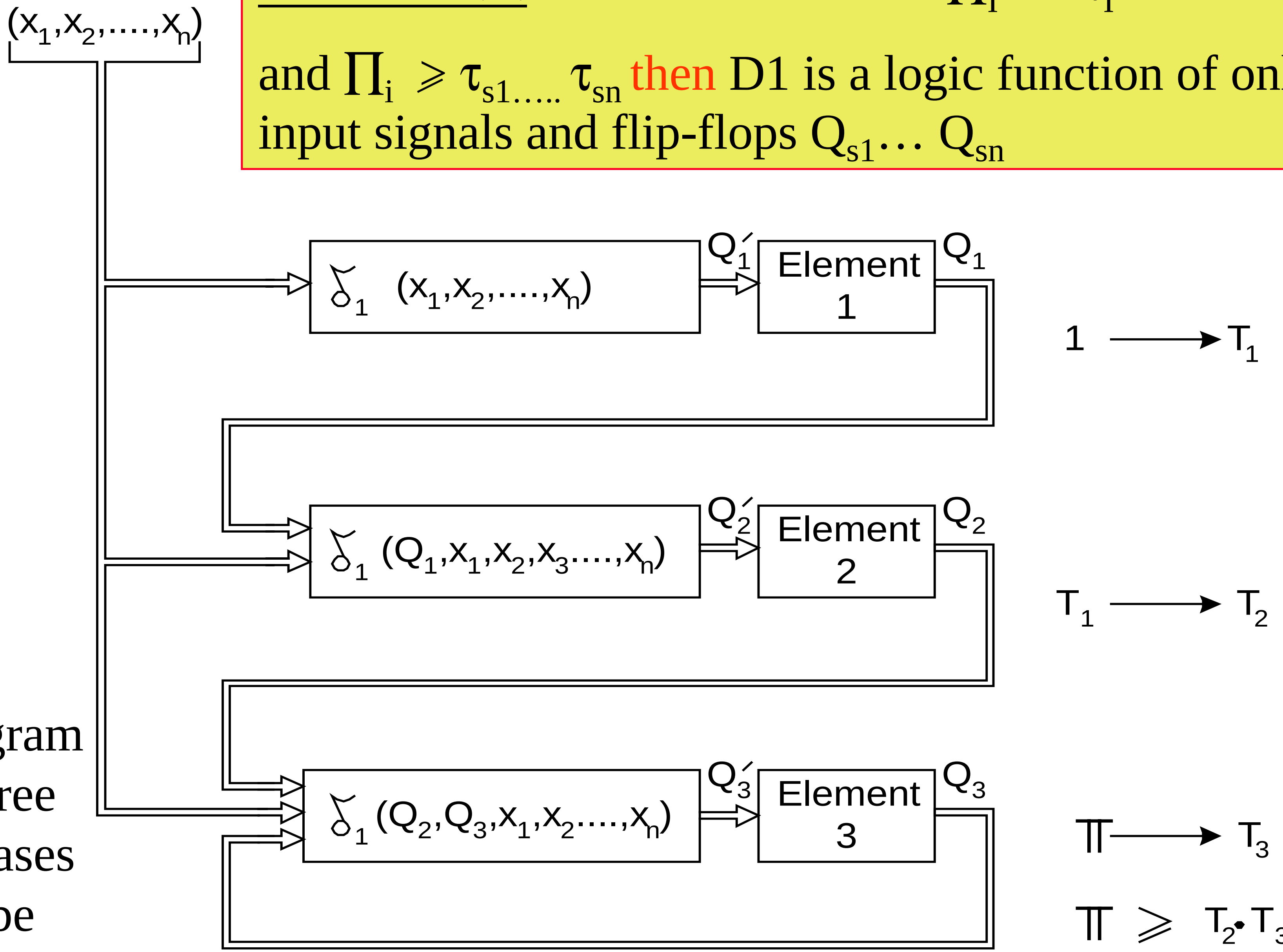
- This means, the assignment that minimizes the total number of variables as arguments of excitation (and output) functions.

- There is a **correspondence** between the structure of the set of all partition pairs for all two-block (proper) partitions of a machine and the realization (decomposition) structure of this machine

- Simple pairs lead to simple **submachines**

Theorem 5.3. If there is transition $\prod_I$ --> τ_I

and $\prod_i \geqslant \tau_{s1}.....\ \tau_{sn}$ then D1 is a logic function of only input signals and flip-flops $Q_{s1}\dots\ Q_{sn}$



This diagram shows three special cases that can be used in hand design

Fig.5.37. Structure of automaton illustrating application of Theorem 5.3

Let us assume D type Flip - Flops

What are good partitions for outputs of the machine?

a)

A \ X	X_1	X_2	$y_1 y_2$
1	1	2	0 0
2	1	3	0 0
3	4	3	- 0
4	5	3	1 1
5	1	3	0 1

b)

A \ X	X_1	X_2
1	1/1	2/1
2	1/0	3/0
3	4/1	3/0
4	5/0	3/0
5	1/1	3/1

c)

$Q_1Q_2Q_3$ \ X	X_1	X_2
0 0 0	000/1	010/1
0 1 0	000/0	110/0
1 1 0	101/1	110/0
1 0 1	001/0	110/0
0 0 1	000/1	110/1

d)

A \ X	X_1	X_2
1	1/1	2/1
2	1/0	3/0
3	4/1	3/-
4	5/-	3/0
5	1/-	3/1

$\Pi_{x_1}(y_1) = \{\overline{125}, \overline{4}, (3)\}$

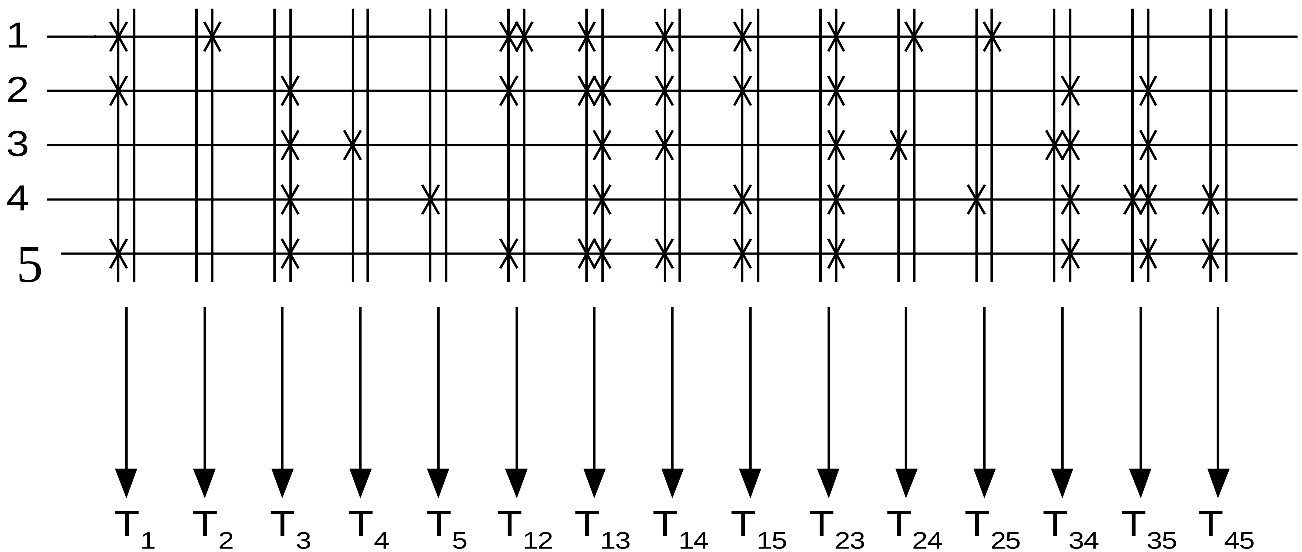
Machine M2

For machine M2 partitions (1235,4) $=T_4$ and (125, 34)$=T_{34}$ are good for y_1

For machine M2 partition (123,45)$=T_{45}$ is good for y_2

Calculation of all partition pairs for Machine M2

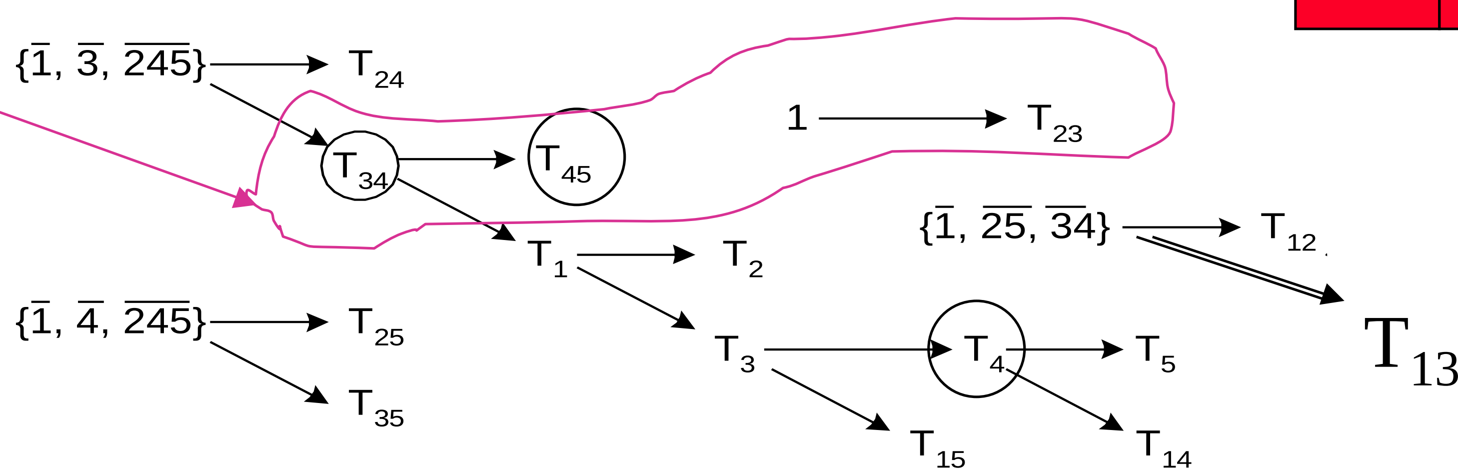
a)



	X_1	X_2
1	1	2
2	1	3
3	4	3
4	5	3
5	1	3

Selection of partitions for Encoding

b)



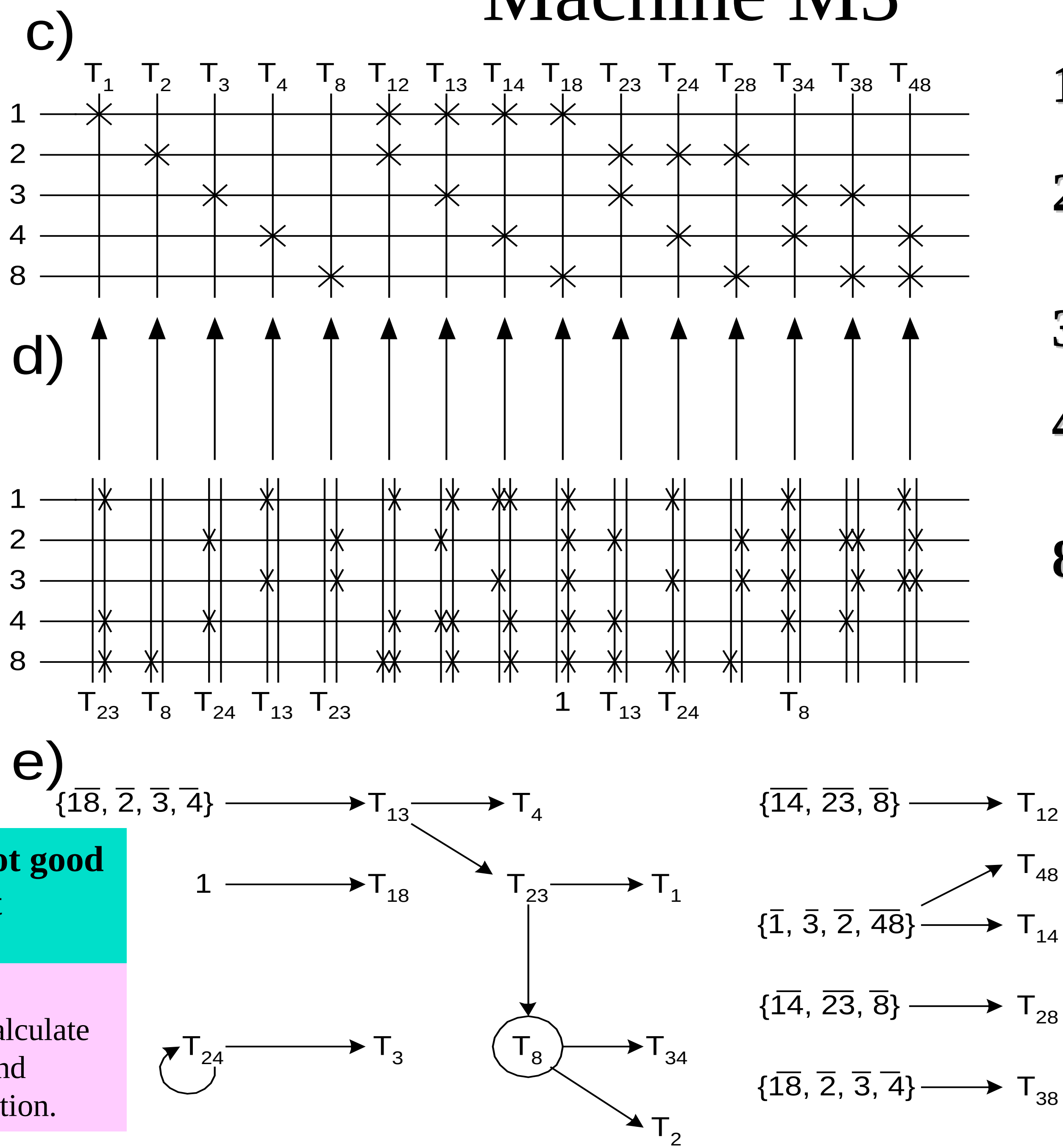
Partitions good for output are circled

Selected Partitions

- T_{23} is always good since it has a predecessor of 1
- Out of many pairs of proper partitions from the graph we select partitions T_{34} and T_{45} because they are both good for outputs
- So now we know from the main theorem that the (logic) excitation function of the Flip-flop encoded with partition T_{23} will depend only on input signals and not on outputs of other flip-flops
- We know also from the main theorem that the excitation function of flip-flop encoded with T_{45} will depend only on input signals and flip-flop encoded with partition T_{34}
- The question remains how good is partition T_{34}. It is good for output but how complex is its excitation function? This function depends either on two or three flip-flops. Not one flip-flop, because it would be seen in the graph. Definitely it depends on at most three, because the product of partitions T_{23} T_{34} T_{45} is a zero partition.
- In class we have done calculations following main theorem to evaluate complexity and the result was that it depends on three.
- Please be ready to understand these evaluation calculations and be able to use them for new examples.

Machine M3

Calculation of partition pair graph from multi-line for machine



GRAPH OF PARTITION PAIRS

	X_1	X_2
1	4	1
2	3	8
3	4	8
4	3	1
8	2	1

T_8 is a good output partition

Select T_{18}, T_{24} and T_8

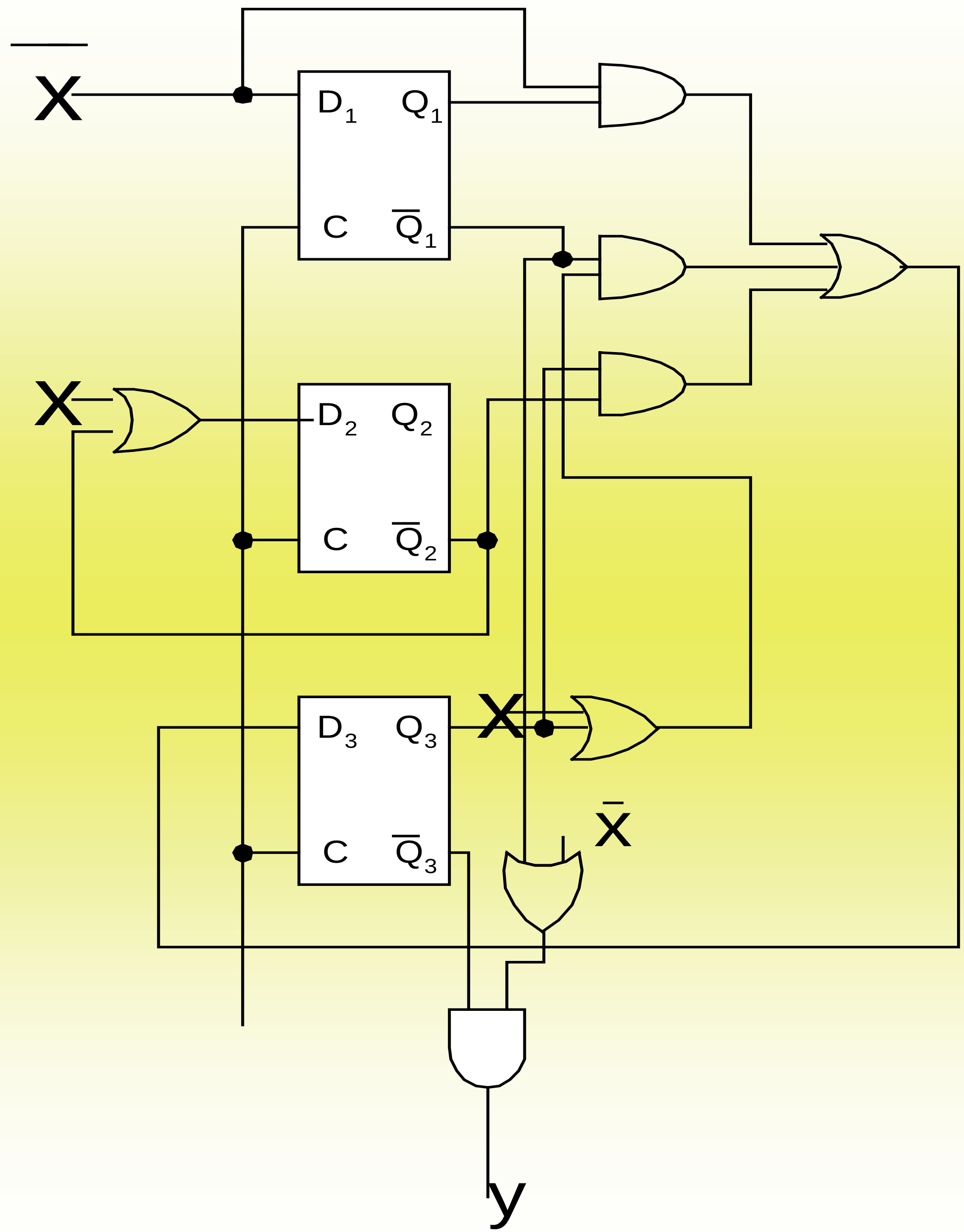
Explain why this is a not good choice - because 2,4 not separated

Evaluate complexities of all excitation functions. Next calculate the functions from Kmaps and compare. Give final explanation.

Complete this example to the end as an exercise

Problem for homework. What can you tell about the partitions from this schematic of a machine?

What can you tell about the partition pairs?



JK flip-flops are very important since they include D and T as special cases - you have to know how to prove it

Relation between excitation functions for D and JK flip-flops

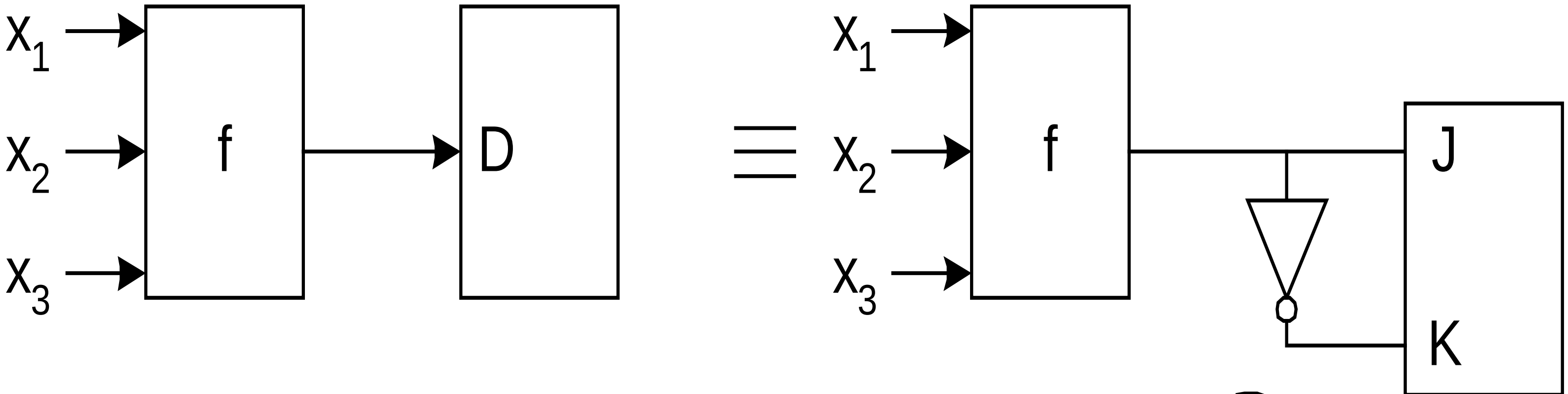


Fig 5.36. Example of excitation function for D and JK flip-flops

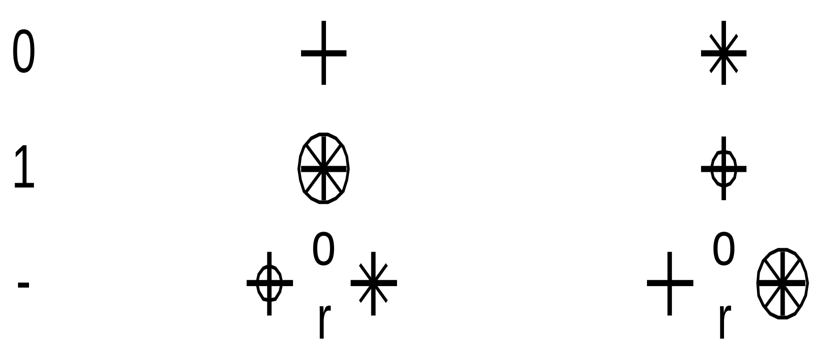
Fig.5.36

QUESTION: How to do state assignment for JK flip-flops?

Let us first recall excitation tables for JK Flip-flops

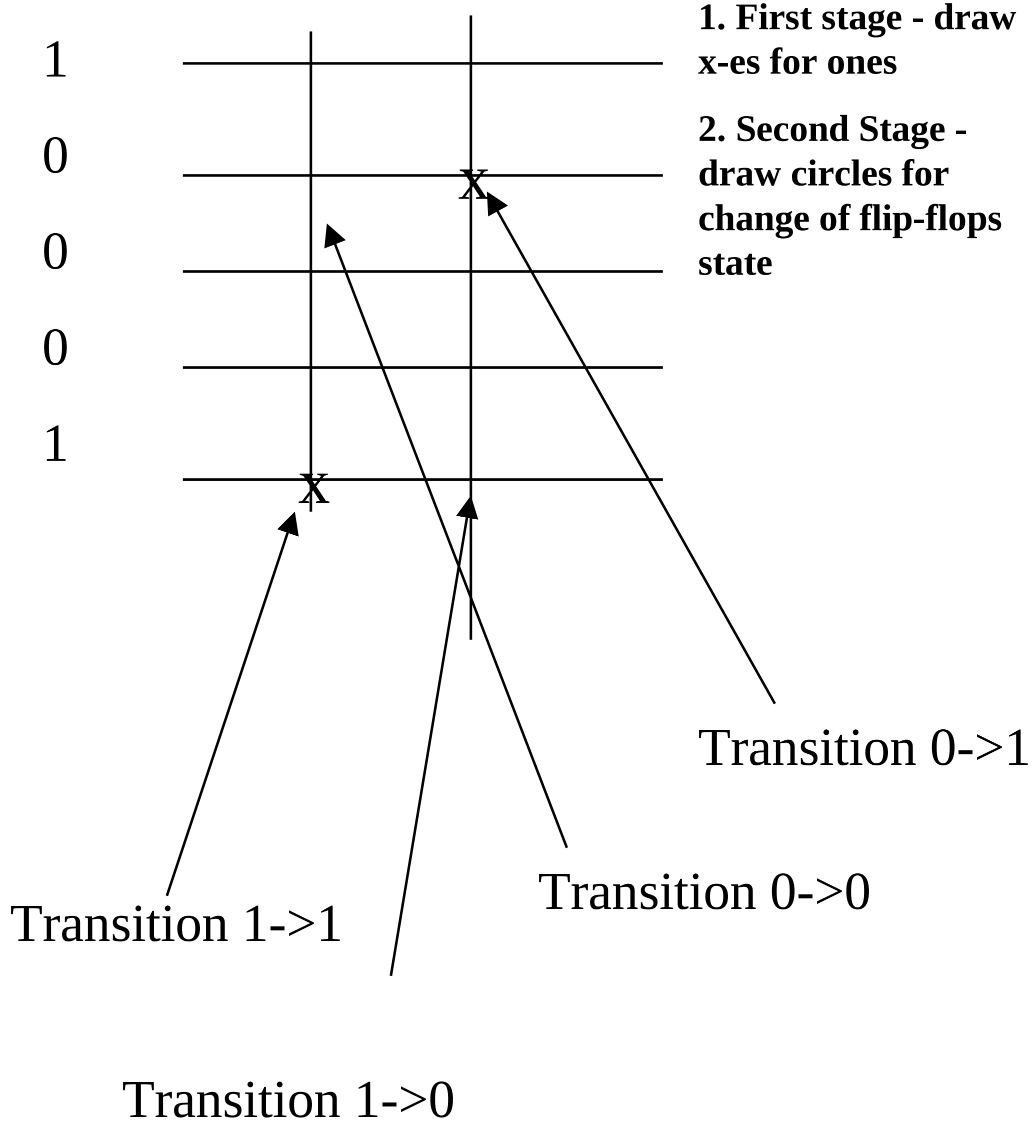
Q	Q^+	J	K
0	0	0	-
0	1	1	-
1	0	-	1
1	1	-	0

for input J   for input K



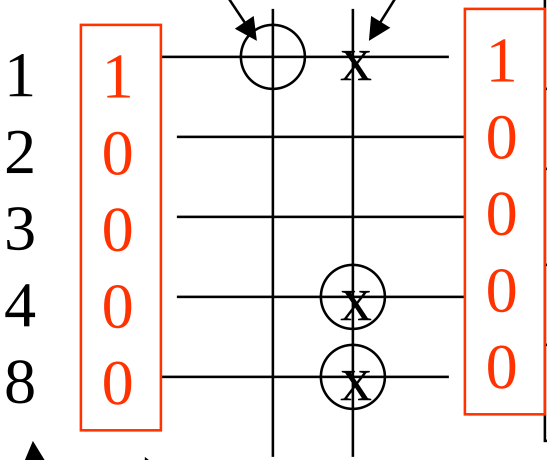
1. First stage - draw x-es for ones

2. Second Stage - draw circles for change of flip-flops state



Draw circles

Draw exes



state

encoding

transitions

0	1
0	0
0	0
0	1
0	1

Obtained from transitions

-1	-0
0-	0-
0-	0-
0-	1-
0-	1-

J

-	-
0	0
0	0
0	1
0	1

K

1	0
-	-
-	-
-	-
-	-

Now, thanks to don’t cares from J we can write :

$(123,48) \text{-->} T_1$

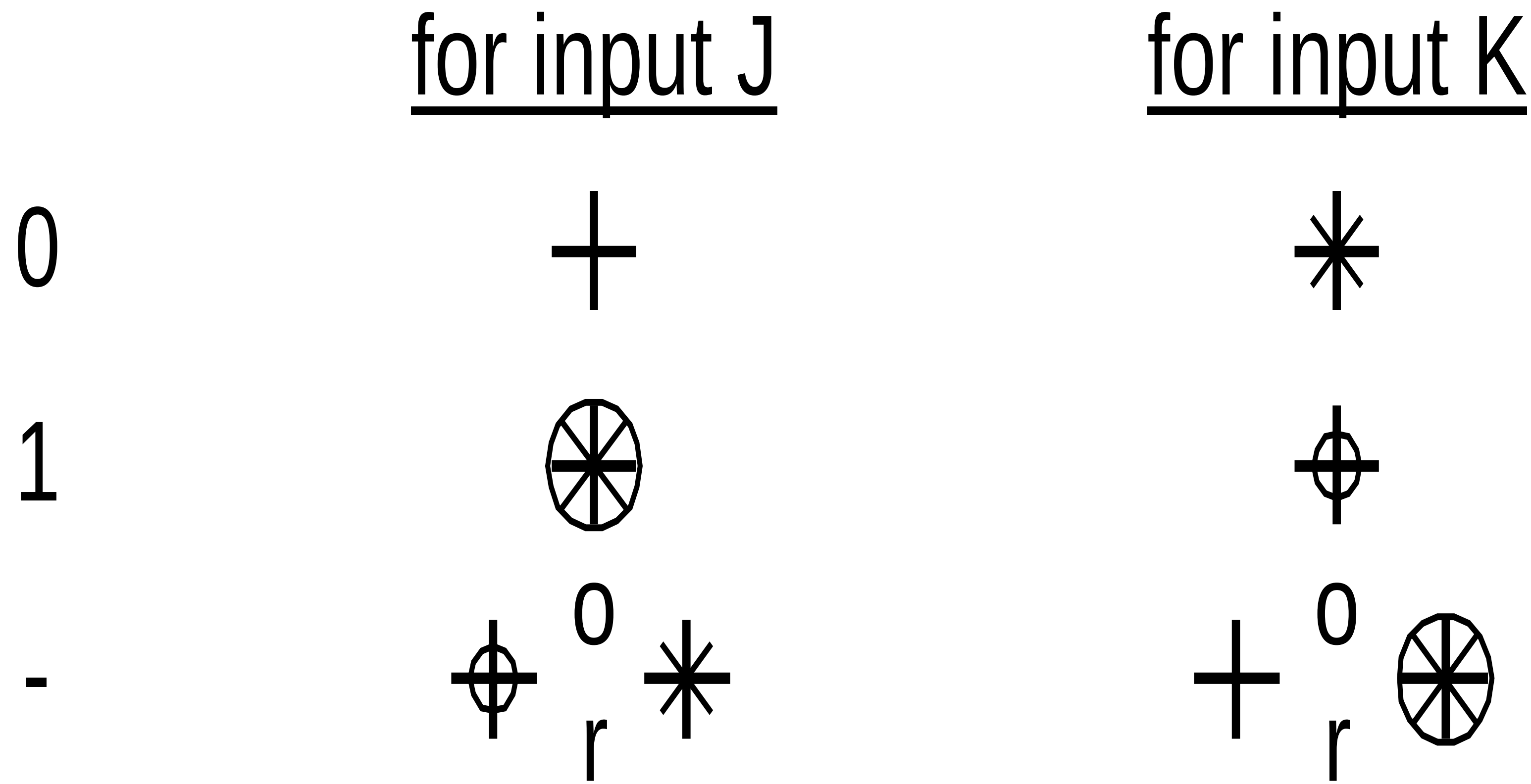
$(23,148) \text{-->} T_1$

From K we can write :

$\mathbf{1} \text{ -->} T_1$

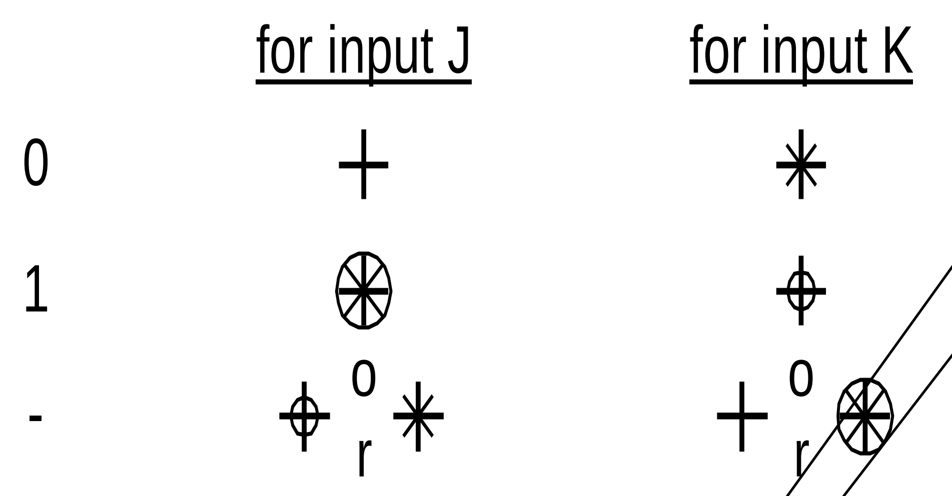
For this task we will adapt the Multi-line method

Rules for State Assignment of JK Flip-Flops



These are the mechanical rules for you to follow, but where they come from?

Calculation of partition pairs assuming JK flip-flops for machine M3



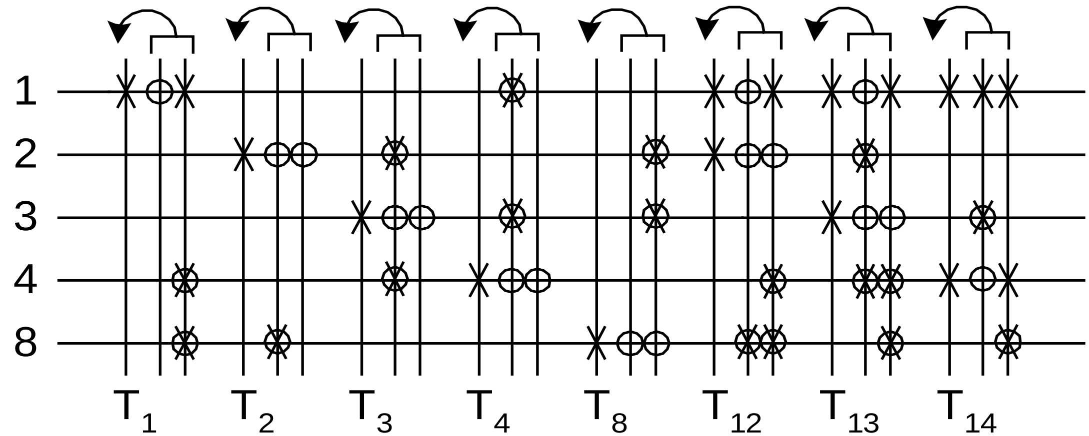
Transition 0->0 has excitations 0- for JK.

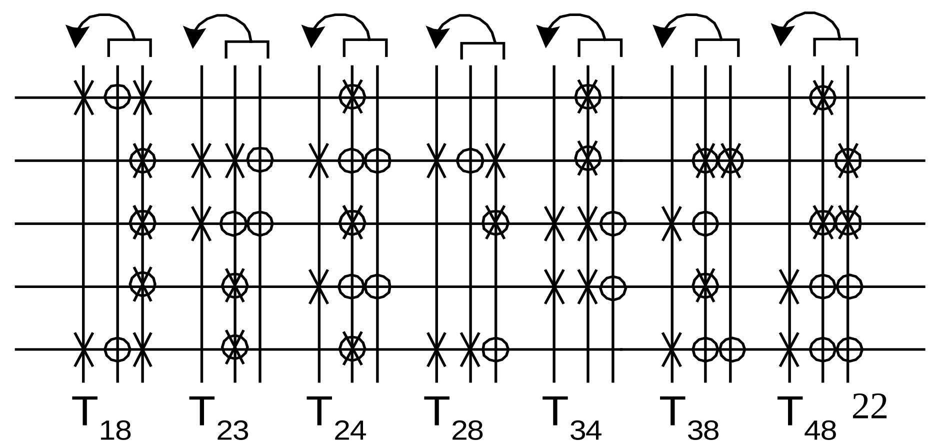
Transition 1->1 has excitations -0 for JK.

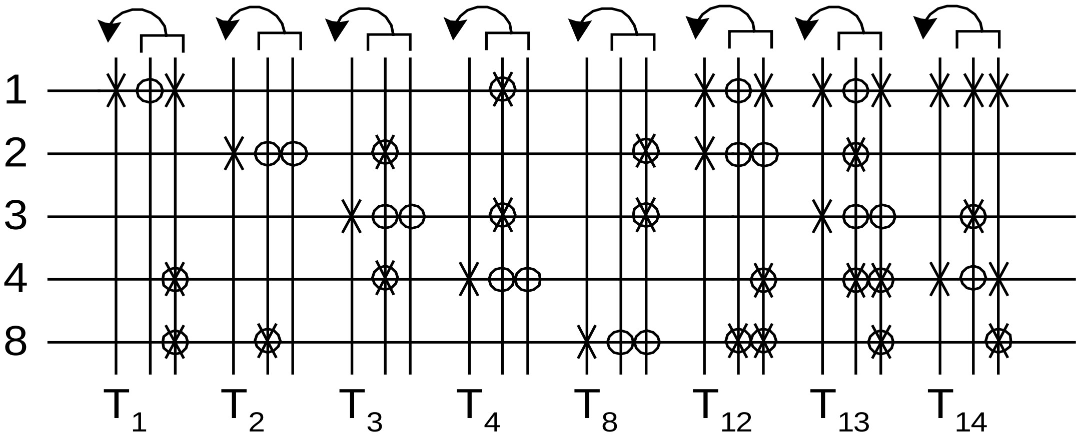
Current encoding

	X_1	X_2			
1	**4**	**1**	1	0	1
2	**3**	**8**	0	0	0
3	**4**	**8**	0	0	0
4	**3**	**1**	0	0	1
8	**2**	**1**	0	0	1

Machine M3





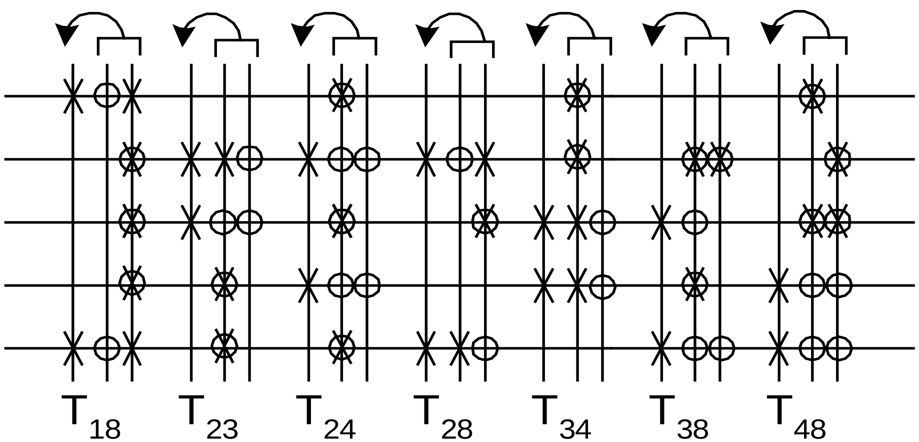


The subsequent stages are the following.

1. From multiline draw the graph of transitions for both J and K inputs.

2. Mark partitions good for output

3. Find partition pairs that simplify the total cost, exactly the same as before.



There fore the multi-line method can be extended for any type of flip-flops and for incompletely specified machines.

Fig.5.43. Schematic of FSM from Example 5.7 realized with JK Flip-flops