<table>
<tr><td colspan="7">Digital Circuits Theory - Laboratory</td></tr>
<tr><td>Academic year</td><td>Laboratory exercises on</td><td>Mode of studies</td><td>Field of studies</td><td>Supervisor</td><td>Group</td><td>Section</td></tr>
<tr><td rowspan="2">2024/2025</td><td>Wednesday</td><td rowspan="2">SSI</td><td rowspan="2">Informatics</td><td rowspan="2">DP</td><td rowspan="2">1</td><td rowspan="2">1</td></tr>
<tr><td>11:45 – 13:15</td></tr>
</table>

Report from Exercise No 12

Performed on: 18.12.2024

Exercise Topic: Microprogrammable circuits

Performed by:

Piotr Copek
Jeremi Szczotka
Zuzanna Micorek

Introduction

introduction here

Task 1

Obtain Mod 15 adding counter.

Solution

We started by creating table of memory content.

Q_3^n	Q_2^n	Q_1^n	Q_0^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
A_3	A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

Figure 1 - Table of memory content.

Next we connected basic circuit and manually programmed memory.

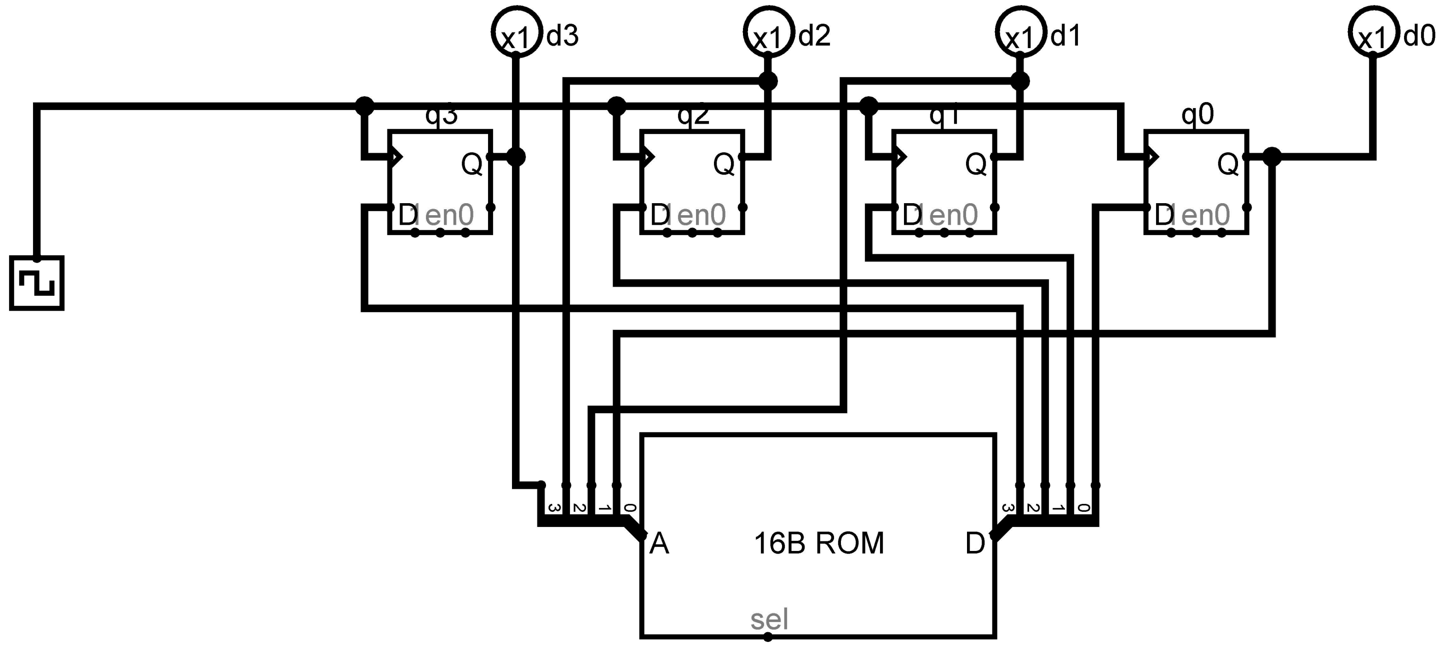
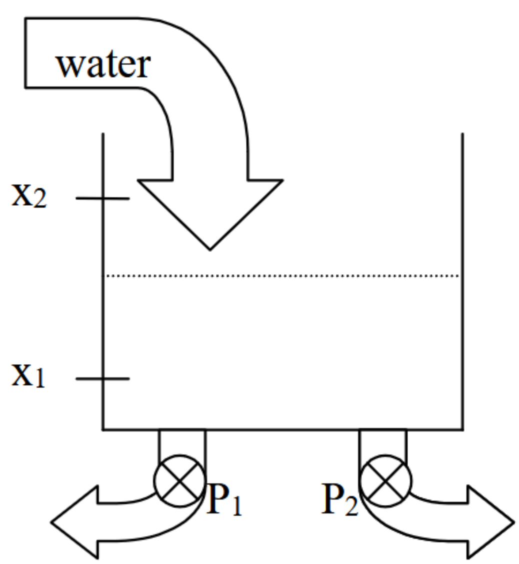


Figure 2 - Implementation of Mod 15 counter.

Task 2

Implement as a microprogrammable circuit a system controlling the task of emptying the water container with two pumps. The pumps P_1 and P_2 should be switched on alternately (only one pump can work at a time) when water exceeds the level of the sensor x_2 (i.e. when $x_2 = 1$). Working pump should be switched off when the water lever is below the sensor x_1 (i.e. when $x1 = 0$). Assume that water level grows when pumps are off, and that it decreases when any pump is working.



For the microprogrammable circuit obtain:

a) Universal Structure

b) Conditional Multiplexer-based Structure

Solution

A primitive flow map was constructed to identify system states and transitions based on sensor inputs.

Present State	$x_1 x_2$				P_1	P_2
	00	01	11	10		
1			(S_0)	S_1	1	0
2	S_2			(S_1)	1	0
3	(S_2)			S_3	0	0
4			S_4	(S_3)	0	0
5			(S_4)	S_5	0	1
6	S_6			(S_5)	0	1
7	(S_6)			S_7	0	0
8			S_0	(S_7)	0	0

Figure 3 - States flow map

We reduced states as following:

$$\begin{gathered} S_0 \to S_1 \\ S_2 \to S_3 \\ S_4 \to S_5 \\ S_6 \to S_7 \end{gathered}$$

Present State \ x_1x_2	00	01	11	10	P_1	P_2
S_0	S_2		S_0	S_0	1	0
S_2	S_2		S_4	S_2	0	0
S_4	S_6		S_4	S_4	0	1
S_6	S_6		S_0	S_6	0	0

Next State

Figure 4 - Karnaugh map used to obtain memory content.

We encoded states as following:

$$S_0 = 00$$
$$S_1 = 01$$
$$S_4 = 11$$
$$S_6 = 10$$

$Q_1^tQ_0^t$ \ x_1x_2	00	01	11	10
S_0	01		00	00
S_2	01		11	01
S_4	10		11	11
S_6	10		00	10

$Q_1^{t+1}Q_0^{t+1}$

Figure 5 - Encoded Karnaugh map used to obtain memory content.

Q_1^n	Q_0^n	x_1	x_2	Q_1^{n+1}	Q_0^{n+1}	P_1	P_0
A_3	A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1	1	0
0	0	0	1	-	-	-	-
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	0
0	1	0	0	0	1	0	0
0	1	0	1	-	-	-	-
0	1	1	0	0	1	0	0
0	1	1	1	1	1	0	0
1	0	0	0	1	0	0	0
1	0	0	1	-	-	-	-
1	0	1	0	1	0	0	0
1	0	1	1	0	0	0	0
1	1	0	0	1	0	0	1
1	1	0	1	-	-	-	-
1	1	1	0	1	1	0	1
1	1	1	1	1	1	0	1

Figure 6 - Memory content of microprogrammable circuit.

Finally we connected the circuit and manually programmed the memory.

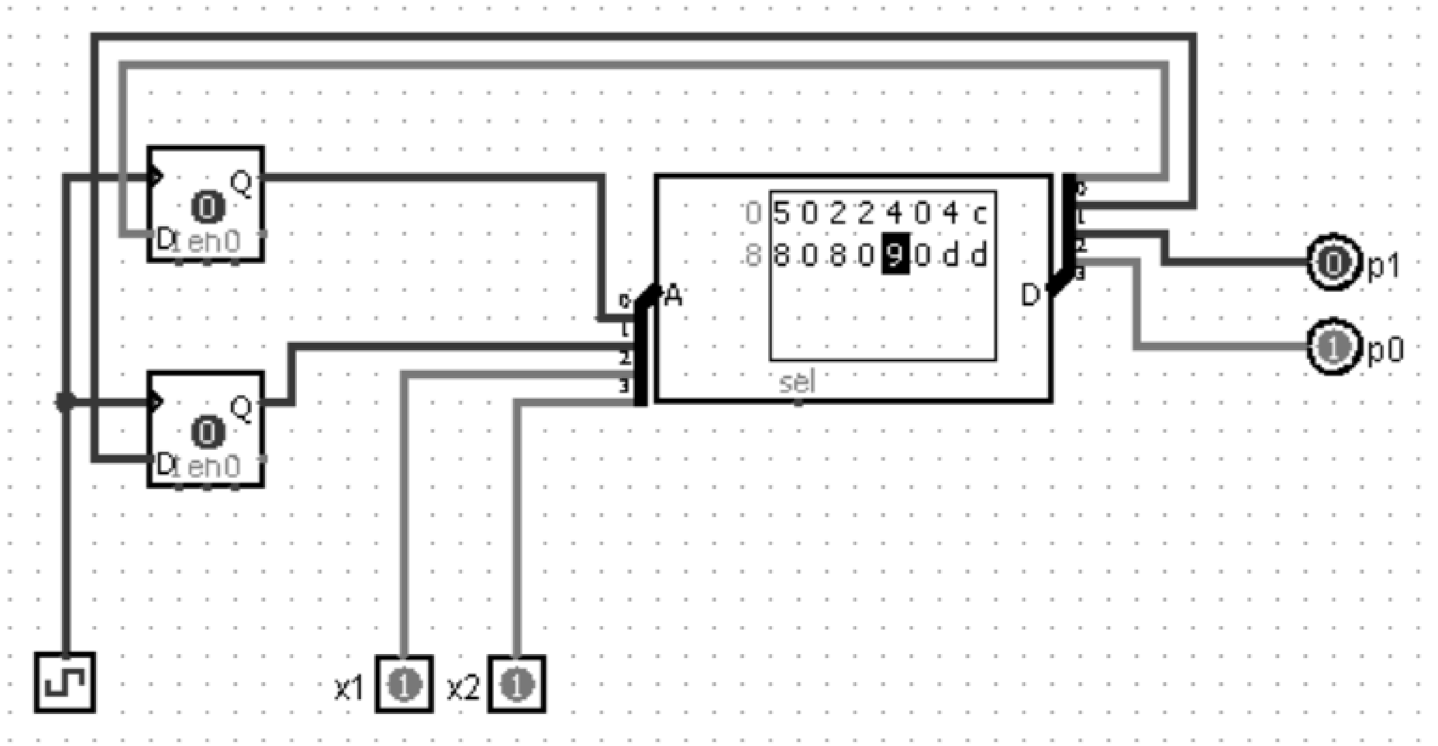


Figure 7 - Circuit with implementation of the task.

In the Logisim Hex Editor the memory content was written as:

$$6022\ 404c\ 8080\ 90dd$$

Task 3

For the microprogrammable circuit obtain:

a) Universal Structure

b) Conditional Multiplexer-based Structure.

Solution

We created state diagram using table of memory content from previous task.

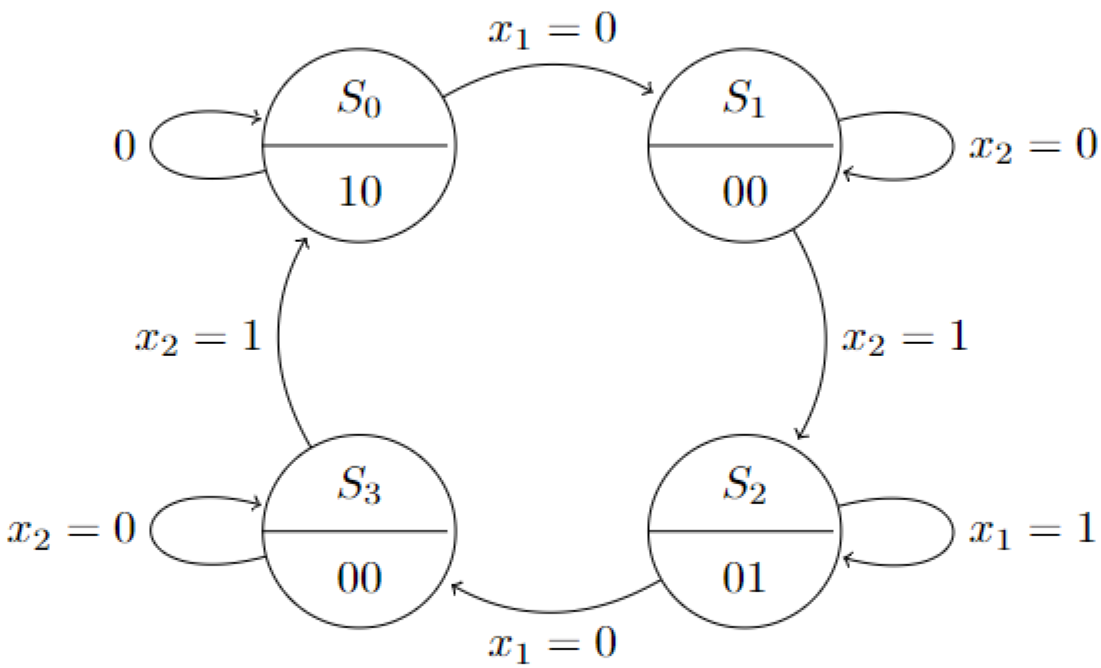


Figure 9 - State diagram used to optimize memory size used.

We noticed that each state change depends only on one variable. Using that information we created new table for the memory content.

Q_1^n	Q_0^n	w^t	Q_1^{n+1}	Q_0^{n+1}	P_1	P_0
A_2	A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	1	1	0
0	0	1	0	0	1	0
0	1	0	0	1	0	0
0	1	1	1	1	0	0
1	0	0	1	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	1
1	1	1	1	1	0	1

Figure 10 - Table of memory content of microprogrammable circuit.

Finally we connected the circuit and manually programmed the memory.

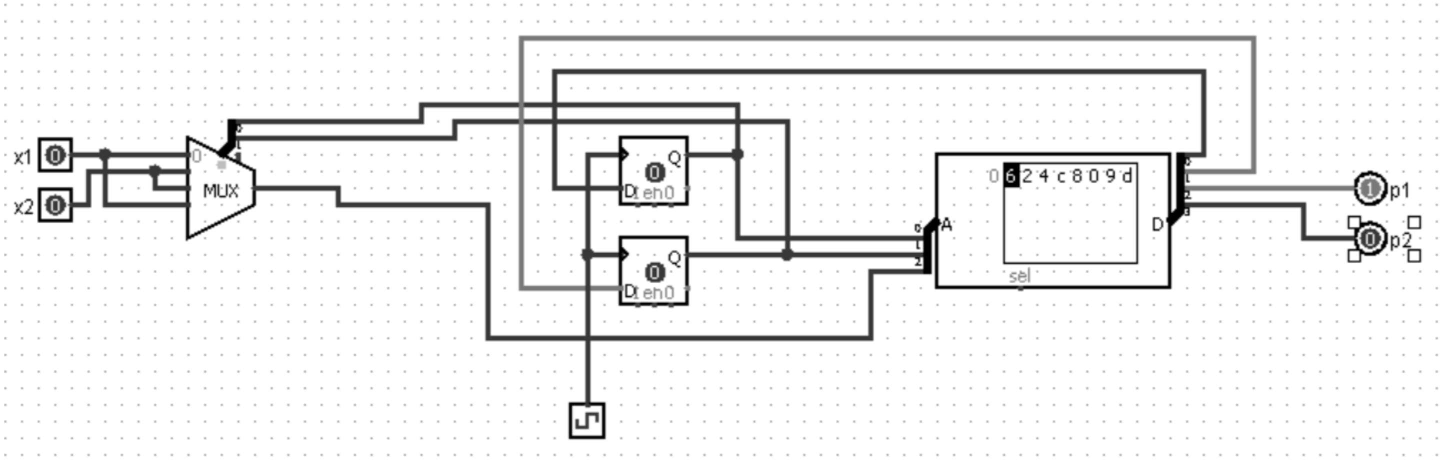


Figure 11 - Circuit with implementation of the task.

In the Logisim Hex Editor the memory content was written as:

$$624c\ 809d$$

Summary

The exercises successfully demonstrated the implementation of microprogrammable circuits. The following insights were gained:

- Microprogrammable circuits offer significant flexibility and reduced complexity in design.
- Optimization techniques, such as memory reduction, enhance efficiency without sacrificing functionality.