

Week 2 – Logic

Student number: 588406

Assignment 2.1: Parking lot

Which gates do you need?

You need a 3-input AND gate.

Because the parking is full only if:

$$\text{FULL} = A \wedge B \wedge C$$

Complete this table

Parking lot 1	Parking lot 2	Parking lot 3	Result (full) 1
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Assignment 2.2: Android or iPhone

Which gates do you need?

You need an XOR gate.

Reason:

- If the employee chooses Android = 1 and iPhone = 0, the output = 1
- If the employee chooses Android = 0 and iPhone = 1, the output = 1
- If the employee chooses both or none, the output = 0

This matches the XOR behavior.

Complete this table

- ❑ A = Android chosen
- ❑ B = iPhone chosen
- ❑ Y = Valid choice (output that is 1 only when exactly one is chosen)

Android phone	iPhone	Result (Phone in possession)
0	0	0
0	1	1
1	0	1

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

Assignment 2.3: Four NAND gates

Complete this table

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

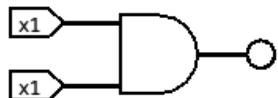
How can the design be simplified?

This circuit works like an XOR gate. So instead of using 4 NAND gates, you can use 1 XOR gate.

Assignment 2.4: Getting to know Logisim evolution

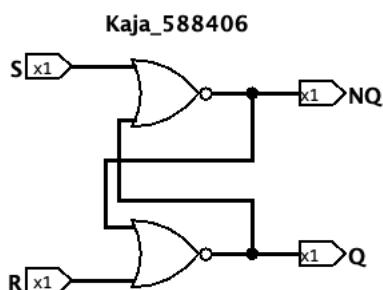
Screenshot of the design with your name and student number in it:

Kaja_588406



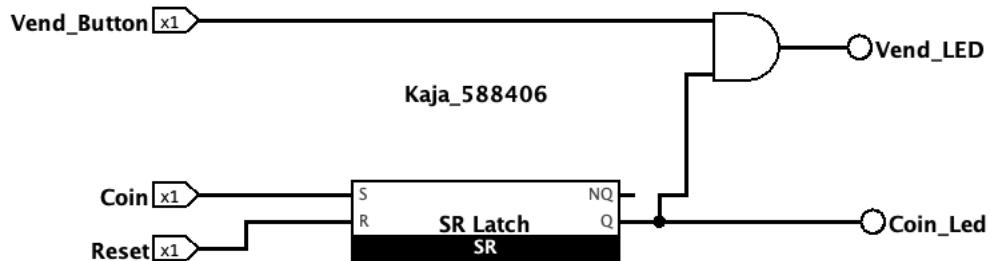
Assignment 2.5: SR Latch

Screenshot SR Latch in Logisim with your name and student number:



Assignment 2.6: Vending Machine

Screenshot Vending Machine in Logisim with your name and student number:



Assignment 2.7: Bitwise operators

Complete the java source code for bitwise operators. Put the source code here.

1#

```

4
5 >   public class Application implements Runnable {
6
...
7 >     public static void main(String[] args) { SaxonApp.start(new Application(), width: 800,
10
11    public void run() {
12        System.out.println("Kaja-588406");
13        int number = 5;
14        if(((number & 1) != 0)) System.out.println("number is odd");
15        else System.out.println("number is even");
16    }
17
18 }
19
Run Application
Run Application
/Users/KajaCingerla/Downloads/Sandbox/src/Application
/Downloads/Sandbox/resources/SaxonApp.jar Application
= SaxonApp version: 1.0.1 =
Kaja-588406
number is odd

```

2#

Sandbox

Current File

Sandbox6.../Application.java

Sandbox7.../Application.java

Sandbox8.../Application.java

```
5 public class Application implements Runnable {  
6     public static void main(String[] args) { SaxionApp.start(new Application(), width: 800,  
7         System.out.println("Kaja-588406");  
8         int number = 4;  
9         if (number > 0 && (number & (number - 1)) == 0)  
10            System.out.println("number is a power of 2");  
11        else  
12            System.out.println("number isn't a power of 2");  
13    }  
14 }  
15  
16  
17  
18  
19  
20  
21
```

Run Application

/Downloads/Sandbox/resources/SaxionApp.jar Application
= SaxionApp version: 1.0.1 =
Kaja-588406
number is a power of 2

19:1 LF UTF-8 4 spaces

3#

Sandbox

Current File

Sandbox6.../Application.java

Sandbox7.../Application.java

Sandbox8.../Application.java

```
5 public class Application implements Runnable {  
6     public void run() {  
7         System.out.println("Kaja-588406");  
8         final int READ = 4;  
9         final int WRITE = 2;  
10        final int EXECUTE = 1;  
11        int userPermissions = 7;  
12        if ((userPermissions & READ) != 0)  
13            System.out.println("User has read permissions");  
14        else  
15            System.out.println("User can't read. No permissions.");  
16    }  
17 }  
18  
19  
20  
21  
22
```

Run Application

/Downloads/Sandbox/resources/SaxionApp.jar Application
= SaxionApp version: 1.0.1 =
Kaja-588406
User has read permissions

Process finished with exit code 0

16:33 LF UTF-8 4 spaces

4#

The screenshot shows a Java application running in an IDE. The code in Application.java is as follows:

```
public class Application implements Runnable {
    public void run() {
        System.out.println("Kaja-588406");
        final int READ = 4;
        final int WRITE = 2;
        final int EXECUTE = 1;
        int userPermissions = READ | EXECUTE;
        System.out.println("User permissions: " + userPermissions);
    }
}
```

The output window shows the application's output:

```
/USERS/kajacingera /Downloads/Sandbox/out/production/Sandbox:/users/kajacingera
/Downloads/Sandbox/resources/SaxionApp.jar Application
= SaxionApp version: 1.0.1 =
Kaja-588406
User permissions: 5
```

The status bar at the bottom indicates the path: `/ > Users > kajacingera > Downloads > Sandbox > Sandbox8 > src > Application`, and the file was saved at 21:1.

5#

The screenshot shows a Java application running in an IDE. The code in Application.java is as follows:

```
public class Application implements Runnable {
    public void run() {
        System.out.println("Kaja-588406");
        final int READ = 4;
        final int WRITE = 2;
        final int EXECUTE = 1;
        int userPermissions = 6;
        userPermissions = userPermissions ^ WRITE;
        System.out.println("User permissions: " + userPermissions);
    }
}
```

The output window shows the application's output:

```
-- SAXIONAPP VERSION: 1.0.1 --
Kaja-588406
User permissions: 4

Process finished with exit code 0
```

The status bar at the bottom indicates the path: `/ > Users > kajacingera > Downloads > Sandbox > Sandbox8 > src > Application`, and the file was saved at 21:2.

6#

The screenshot shows a Java application running in an IDE. The code in Application.java is as follows:

```
public class Application implements Runnable {
    public void run() {
        System.out.println("Kaja-588406");
        int number = 5;
        number = ~number + 1;
        System.out.println("Number (negative): " + number);
        number = ~number + 1;
        System.out.println("Number (positive again): " + number);
    }
}
```

The output window shows the application's output:

```
/Downloads/Sandbox/resources/SaxionApp.jar Application
= SaxionApp version: 1.0.1 =
Kaja-588406
Number (negative): -5
Number (positive again): 5
```

The status bar at the bottom indicates the file path: /Users/kajacingera/Downloads/Sandbox/Sandbox8/src/Application, and the file was saved at 20:22, LF, UTF-8, with 4 spaces.

Assignment 2.8: Java Application Bit Calculations

Create a java program that accepts user input and presents a menu with options.

1. Is number odd?
2. Is number a power of 2?
3. Two's complement of number?

Implement the methods by using the bitwise operators you have just learned.

Organize your source code in a readable manner with the use of control flow and methods.

Keep this application because you need to expand it in week 6 for calculating network segments.

Paste source code here, with a screenshot of a working application.

A screenshot of a Java IDE interface. The top bar shows tabs for "Application" and "Sandbox8.../Application.java". The code editor displays the following Java code:

```
import java.awt.*;  
public class Application implements Runnable {  
    public static void main(String[] args) { SaxonApp.start(new Application(), width: 800);  
    public void run() {  
        System.out.println("Kaja-588406");  
        int number = 5;  
        int choice = 1;  
        System.out.println("Number: " + number);  
        System.out.println("1. Is number odd?");  
        System.out.println("2. Is number a power of 2?");  
        System.out.println("3. Two's complement of number?");  
        System.out.println("Chosen option: " + choice);  
    }  
}
```

The "Run" section shows the application's output:

```
= SaxonApp version: 1.0.1 =  
Kaja-588406  
Number: 5  
1. Is number odd?  
2. Is number a power of 2?  
3. Two's complement of number?  
Chosen option: 1  
Number is odd
```

The status bar at the bottom indicates the file path: "users > kajacingera > Downloads > Sandbox > Sandbox8 > src > Application > run", and the current time and date: "13:9 LF UTF-8 4 spaces".

A screenshot of a Java IDE interface. The top bar shows tabs for "Application" and "Sandbox8.../Application.java". The code editor displays the following Java code, which is a modified version of the previous one:

```
public class Application implements Runnable {  
    public void run() {  
        System.out.println("Two's complement of number?");  
        System.out.println("Chosen option: " + choice);  
  
        if (choice == 1) {  
            if ((number & 1) != 0) {  
                System.out.println("Number is odd");  
            } else {  
                System.out.println("Number is even");  
            }  
        }  
  
        if (choice == 2) {  
            if (number > 0 && (number & (number - 1)) == 0) {  
                System.out.println("Number is a power of 2");  
            } else {  
                System.out.println("Number is NOT a power of 2");  
            }  
        }  
  
        if (choice == 3) {  
            int twosComplement = ~number + 1;  
            System.out.println("Two's complement: " + twosComplement);  
        }  
    }  
}
```

The status bar at the bottom indicates the file path: "users > kajacingera > Downloads > Sandbox > Sandbox8 > src > Application > run", and the current time and date: "13:9 LF UTF-8 4 spaces".

Ready? Then save this file and export it as a pdf file with the name: [week2.pdf](#)

