# Datateknik
# Lab 1: Digital Logic Circuits

**Alfred Elfving**
**Fahad Rejabo**

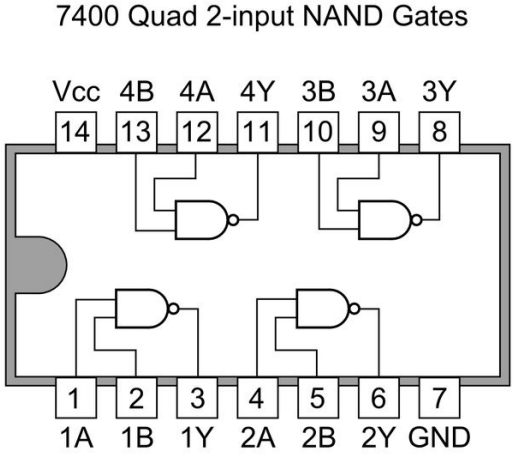
## Materials

• A solderless breadboard.
• An Adafruit Trinket microcontroller board.
• Wiring kit.
• Micro USB cable.
• LED set.
• 3x 7400-series (SN74ACT00) quad NAND gate arrays.
• 2x 7402-series (SN74HCT02) quad NOR gate arrays.

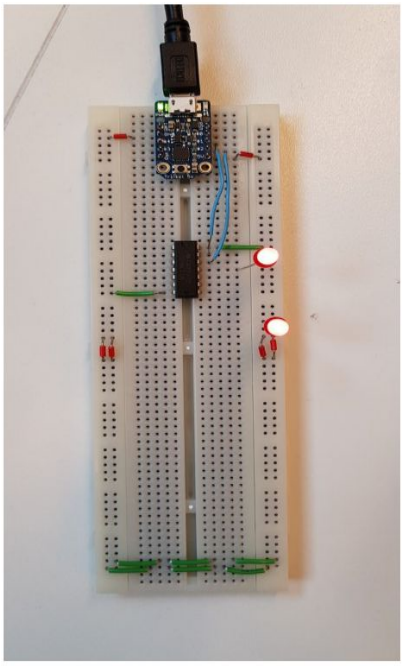
## Task 1: Breadboard Setup and a NAND gate

We installed Arduino IDE and downloaded the library Adafruit Trinket board. We started with setting up our breadboard, which took us a lot of time, because we noticed that there were something wrong with our microcontroller.

After that, it was time to start with the first part of the task 1, assignment a). We followed the instruction in the lab, which was about to test if the led lamp blinks, which it did. Now now it's time for assignment b). We connected a NAND gate to our bredbord, and connected cables from #4 and #3 (the source in the microcontroller) to the NAND gate inputs 1A and 1B (se the figure), and the output of the NAND gate (which is 1Y) was connected to the led lamp.



Our breadboard setup



Our code implementation of task1.

lab1-Datateck §

```
int pinA=3;
int pinB=4;

void setup(){
  pinMode(pinA, OUTPUT);
  pinMode(pinB, OUTPUT);
  }
  void loop(){
//    digitalWrite(pinA, HIGH);
//    digitalWrite(pinB, HIGH);
//    delay(1000);
//    digitalWrite(pinA, LOW);
//    digitalWrite(pinB, LOW);
//    delay(1000);
    digitalWrite(pinA, HIGH);
    digitalWrite(pinB, HIGH);
    delay(1000);
    digitalWrite(pinA, LOW);
    //digitalWrite(pinB, HIGH);
    delay(1000);
    }
```

The NAND truth table.

| A | B | A **NAND** B |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

### Result

So when we upload the code will both A and B turn on (high) and then after 1s turn off, after that A will turn on (low) for 1s, and then A and B will turn on (high) and so on. i.e. the NAND gate works.

## Task 2: Half-adder

Task 2 is about to construct a half-adder circuit which accepts two input bits A0 and B0, and outputs the sum as a bit C0 and a CARRY bit (se the figure 1).

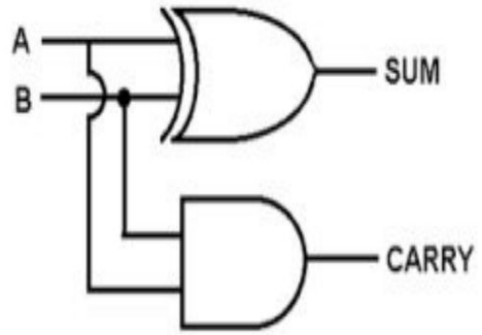
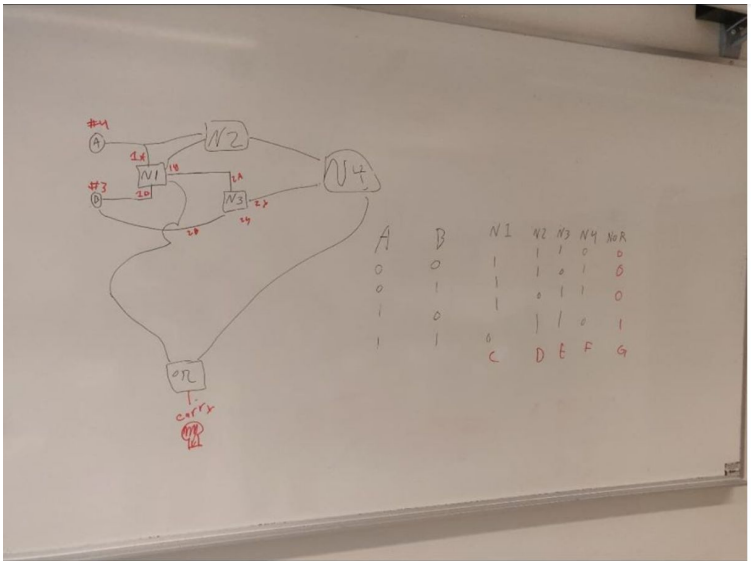
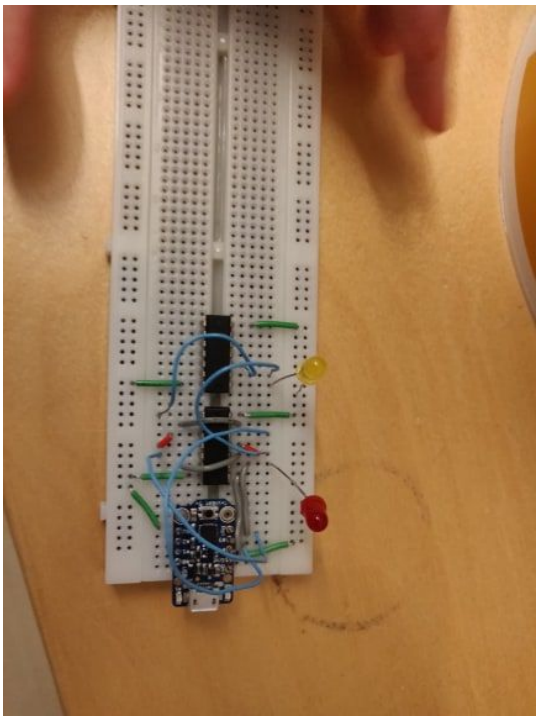


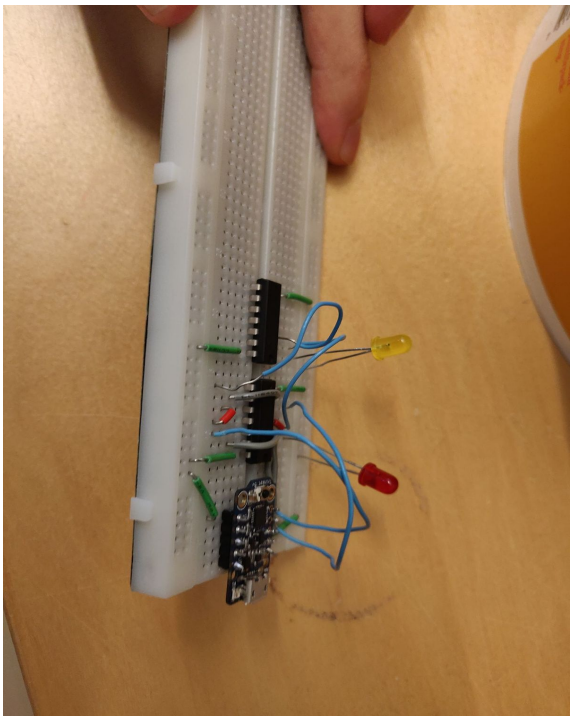

Figure 1 Half adder circuit

Our illustrations for the 4 NAND and NOR gates



*Our breadboard setup*





Due to some USB error we could not connect the output from A0 (pin 3) and B0 (pin4) which would connect to the red LED. The red LED is supposed to show the sum, but we had to put in into the breadboard after we connected the USB and transferred the Arduino program.

## Inputs to every part in the circuit

| NAND 1 | NAND 2 | NAND 3 | NAND 4 | NOR |
|---|---|---|---|---|
| A0 | A0 | B0 | NAND 2 | NAND 1 |
| B0 | NAND 1 | NAND 1 | NAND 3 | NAND 4 |

## Outputs from every part in the circuit (truth table)

| A0 | B0 | NAND 1 | NAND 2 | NAND 3 | NAND 4 | NOR |
|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 |

## Result

As we can see in the truth table on the last column, the *NOR* gate will act as a carry transmitter. We simply had to put a LED on the output of it to show if we have a carry load from the A0 and B0 signal.
This is practically the hard part, the carry LED worked fully as intended after setting up the breadboard with the parts according to our blueprint. The carry (yellow) LED would turn on if both A0 and B0 was sending a signal.

The sum (red) LED would only be turned on if either A0 or B0 was sending a signal, this works when we connect the lamp to the output of NAND 4. As we can see in the previous table (the output table), the NAND 4 outputs only if A0 or B0 is on, not when both or none of them are.
It is acting as an XOR output for A0 and B0.