

2 DIL logische poorten tester

2.1 Einddoel

Je gaat een logische poorten tester maken die aangeeft welke logische poort je hebt aangesloten.

Als je klaar bent met de opdracht lever je in It's learning in:

- De definitieve python code.
- Een mp4-filmpje met een demo van je programma.

Als dit gedaan is laat je de opdracht zien aan de docent zodat deze de opdracht kan goedkeuren.

2.2 Kennis

Voor deze opdracht heb je kennis nodig van microPython.

2.3 Benodigheden

Voor deze opdracht heb je de onderstaande materialen nodig. Controleer aan het begin of al deze spullen aanwezig zijn. Bij het opruimen dien je weer te controleren of alles aanwezig is. Indien er iets defect is geraakt moet je de docent op de hoogte brengen.

- Raspberry Pi Pico – H
- UBS-usb mini kabel
-



2.4 Opdracht: Logische poort tester

2.4.1 Wat is het probleem

We hebben een hele bak met zowel de 74HC00 (AND) als de 74HC32(OR) DIL-componenten. Er is alleen een groot probleem. Alle nummers zijn ervan af, en we willen deze componenten wel gebruiken. We gaan daar een tester voor maken met de Raspberry pi Pico. We weten ook dat sommige componenten kapot zijn, dus we moeten alle poorten testen. Als de rode LED gaat branden is de DIL defect. Als de Groene LED gaat branden is de DIL een 74HC00 en als de gele LED gaat branden is het een 74HC32. De elektronische opstelling is hieronder gegeven. Ontwikkel de software voor deze tester. Om de defecte te testen kan je een 74HC86 nemen.

2.4.2 Logische poorten

We gebruiken een *if*-vergelijking als we iets willen uitvoeren als een conditie geldig is. Bijvoorbeeld indien een waarde groter is dan 6 dan willen we iets printen.

Het kan ook zo zijn dat we meerdere condities tegelijk willen bekijken. Oftewel we willen kijken of de waarde groter is dan 6, maar het moet wel kleiner zijn dan 20. Dit kunnen we in een regel schrijven met de *AND*-operator.

```
1  # C
2  if ( x > 6 && x < 20)
3  {
4      Serial.println("Ja");
5  }
6  # python
7
8  if x > 6 and x < 20:
9      print("Ja")
10
```

Of we willen dat indien de waarde kleiner is dan 6 of groter dan 20 de output neerzetten, dat doen we dan met de *OR*-operator.

```

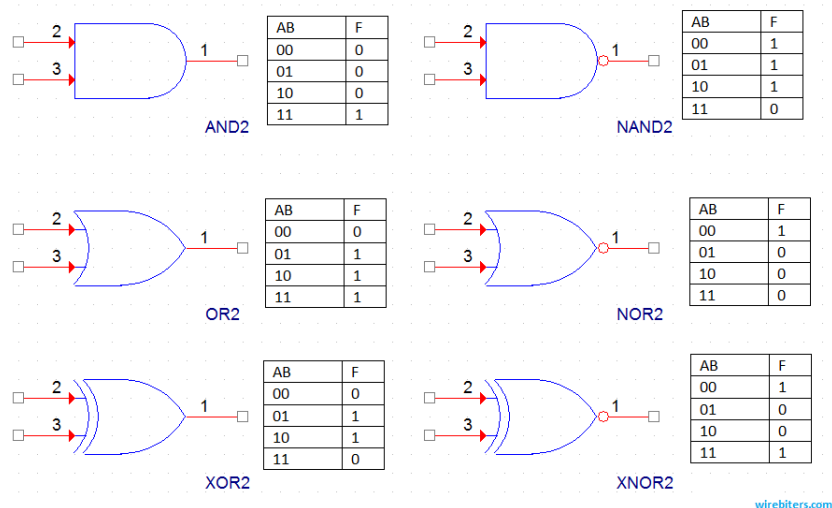
1  # C
2  if ( x < 6 || x > 20)
3  {
4      Serial.println("Ja");
5  }
6  # python
7
8  if x < 6 or x > 20:
9      print("Ja")
10

```

Deze logica kunnen we ook in elektronica nabouwen en hiervoor zijn er logische poorten. Dit zijn DIL-componenten met daarin meerdere van dezelfde vergelijkers. Alle onderstaande vergelijkers hebben 2 ingangen (nummer 2 en 3) en 1 uitgang (nummer 1).

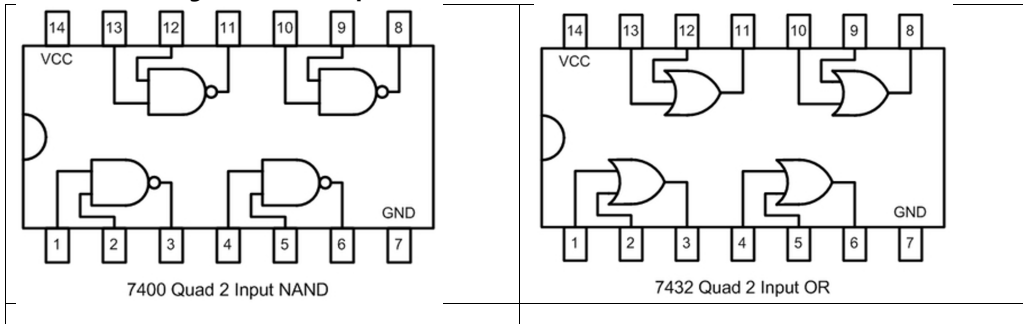
Naast het symbool zie je ook de waarheidstabel. Ik bespreek de waarheidstabel van de AND. Als de ingangen A en B beide 1 zijn is de uitgang F ook 1, in alle andere gevallen is de F de waarde 0.

Naast de AND staat de NAND, dis is een AND maar dan alles net andersom.



Je moet wel opletten dat op de ingangen een pull-up of pull-down weerstand gebruikt wordt indien je met schakelaars werkt.

2.4.3 Aansluitingen van de componenten



We zien op bovenstaande aansluitschema dat pin 14 de VCC heeft, en pin 7 de GND.

Op welke spanning kunnen deze poorten werken?

Verder zijn alle ingangen en uitgangen op dezelfde plek.

Als we een component er omgekeerd inzetten zal ook de rode LED gaan branden.

Je moet alle logische poorten testen. We willen geen goedgekeurde componenten waarbij maar 1 poort