

Analogien

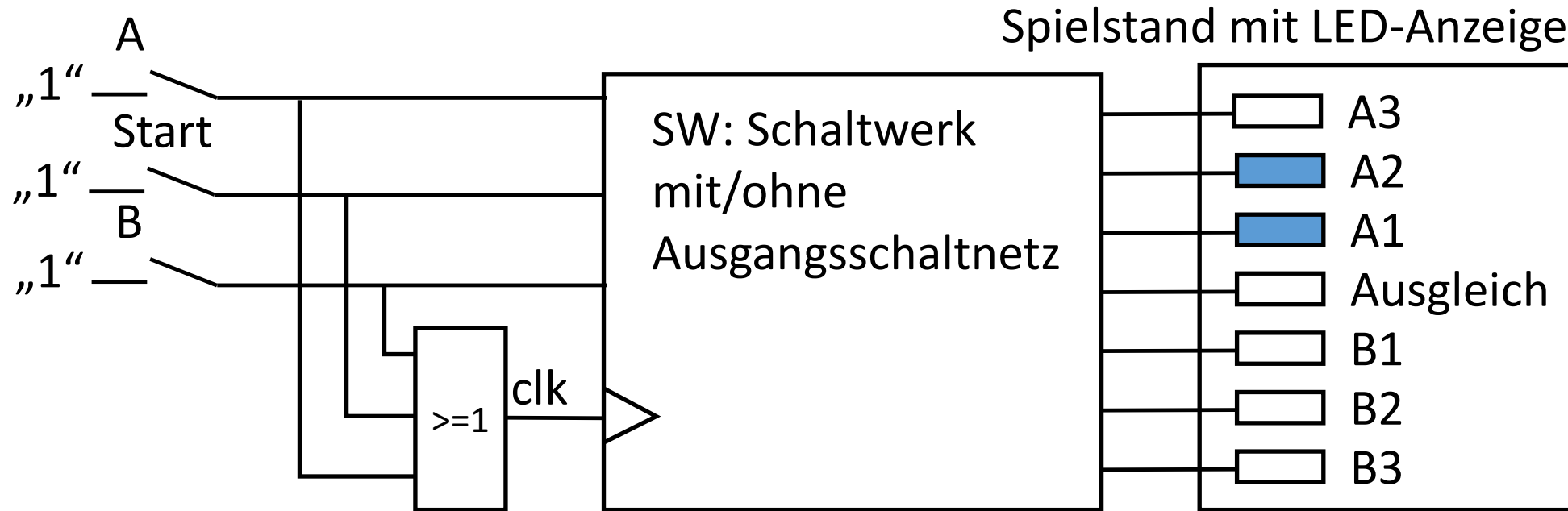
Schaltwerke und Schaltnetze



Ich bin Mik, Dein Mikrocontroller



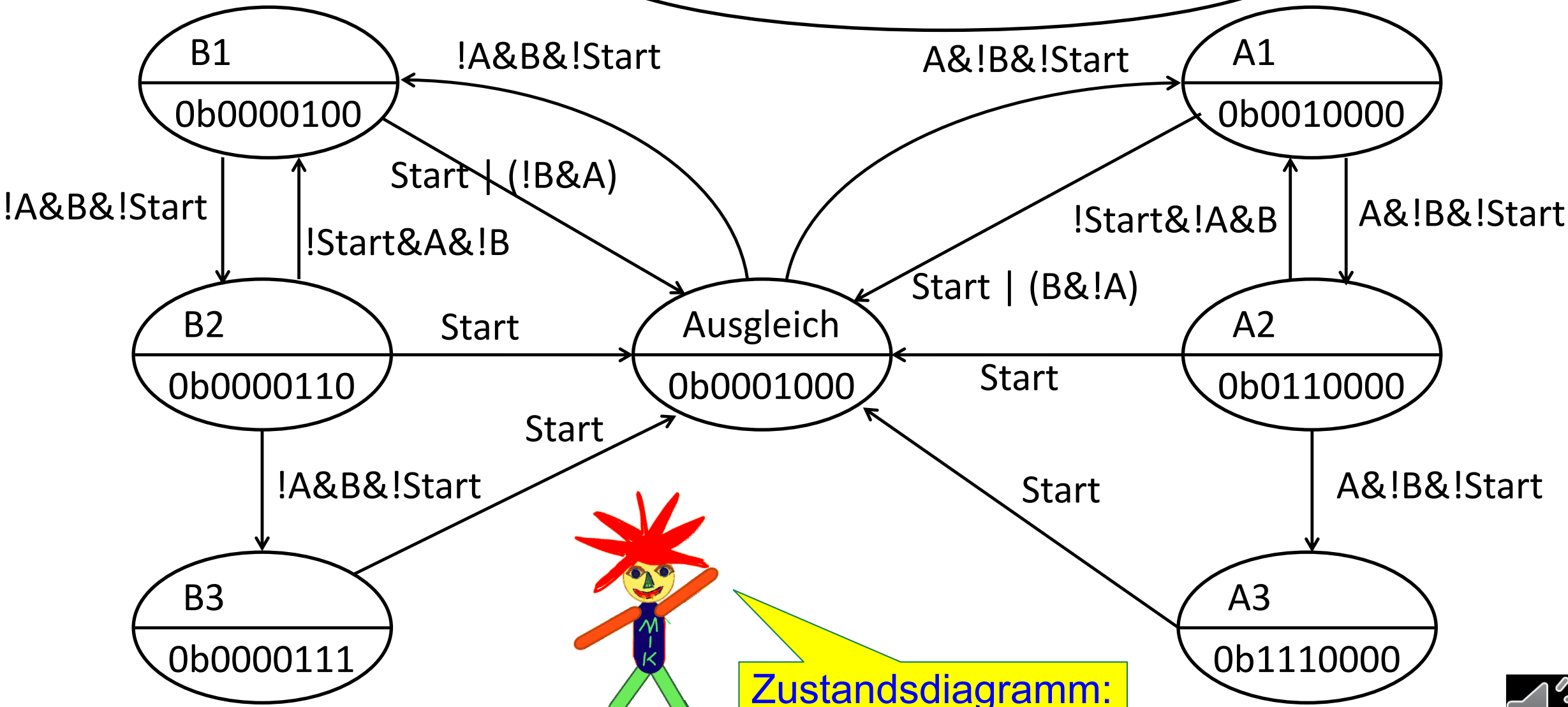
Analogien im Zustandsdiagramm



Mit dem Signal Start leuchtet die mittlere LED Ausgleich. Erzielt Spieler A einen Punkt, wechselt die Anzeige auf LED A1. Bei einem erneuten Punkt für A werden A1 und A2 eingeschaltet (Balkenanzeige). Bei Punkt für B wechselt die Anzeige von A1+A2 wieder auf A1. Gewonnen hat, wer zuerst 3 Punkte in Serie erzielt: Anzeige A1+A2+A3 bzw. B1+B2+B3. Mit Start kann jederzeit in den Zustand Ausgleich gewechselt werden.



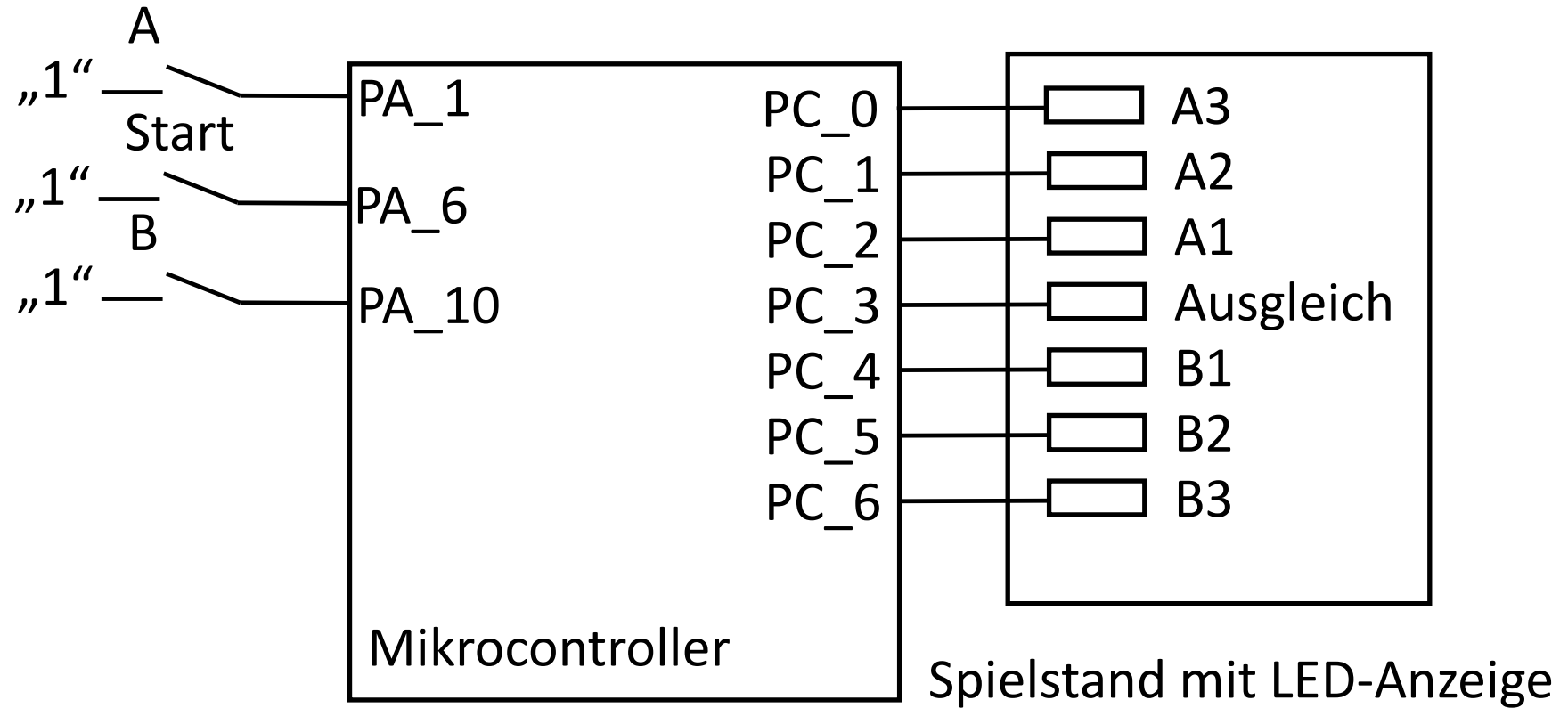
Analogien im Zustandsdiagramm



Zustandsdiagramm:



Analogien im Zustandsdiagramm



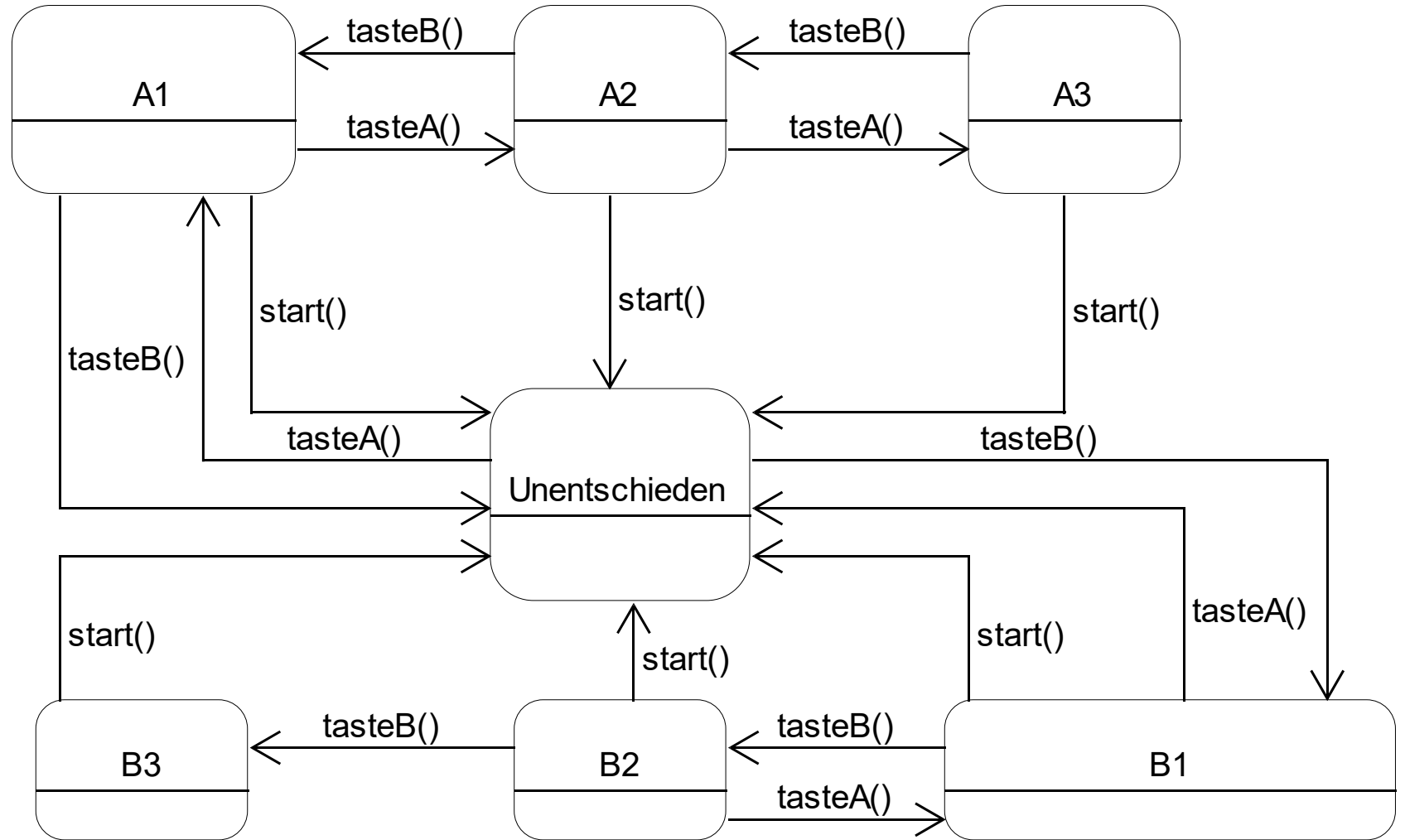
Lösung mit dem Mikrocontroller



Analogien im Zustandsdiagramm

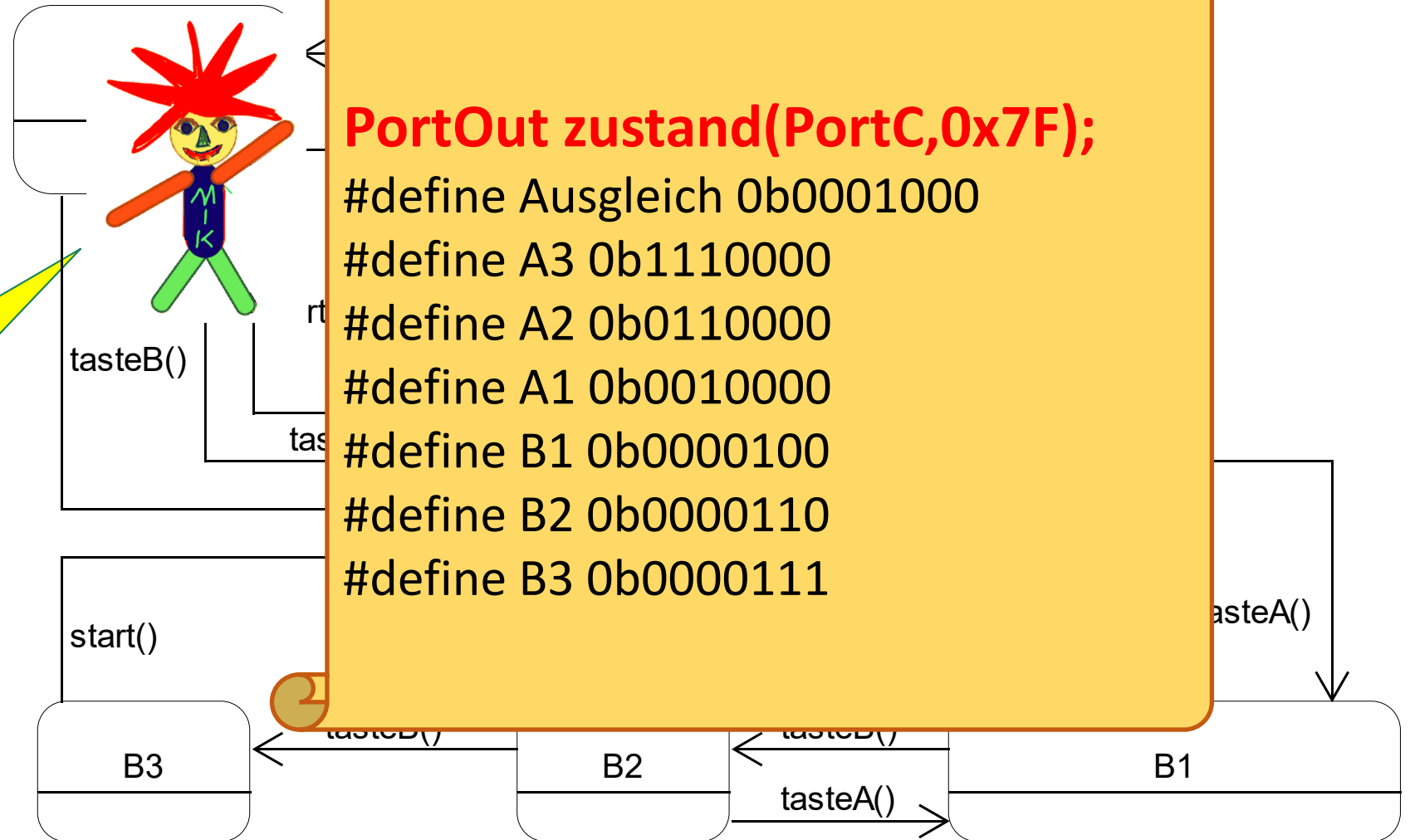


Das UML-
Zustandsdiagramm für
Lösung ohne
„Ausgangsschaltnetz“



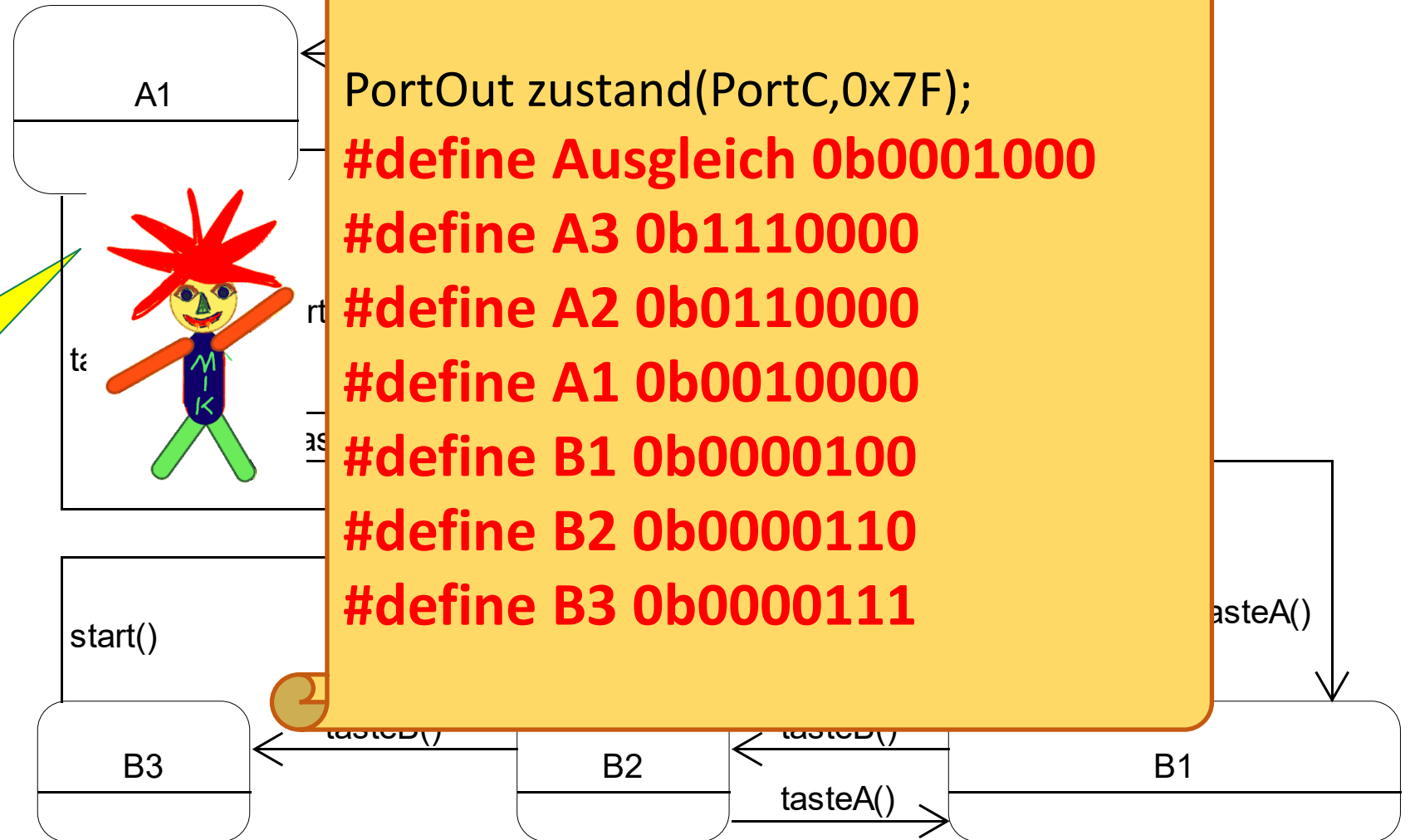
Analogien im Zustandsdiagramm

Der Zustand wird
direkt durch PortC
repräsentiert



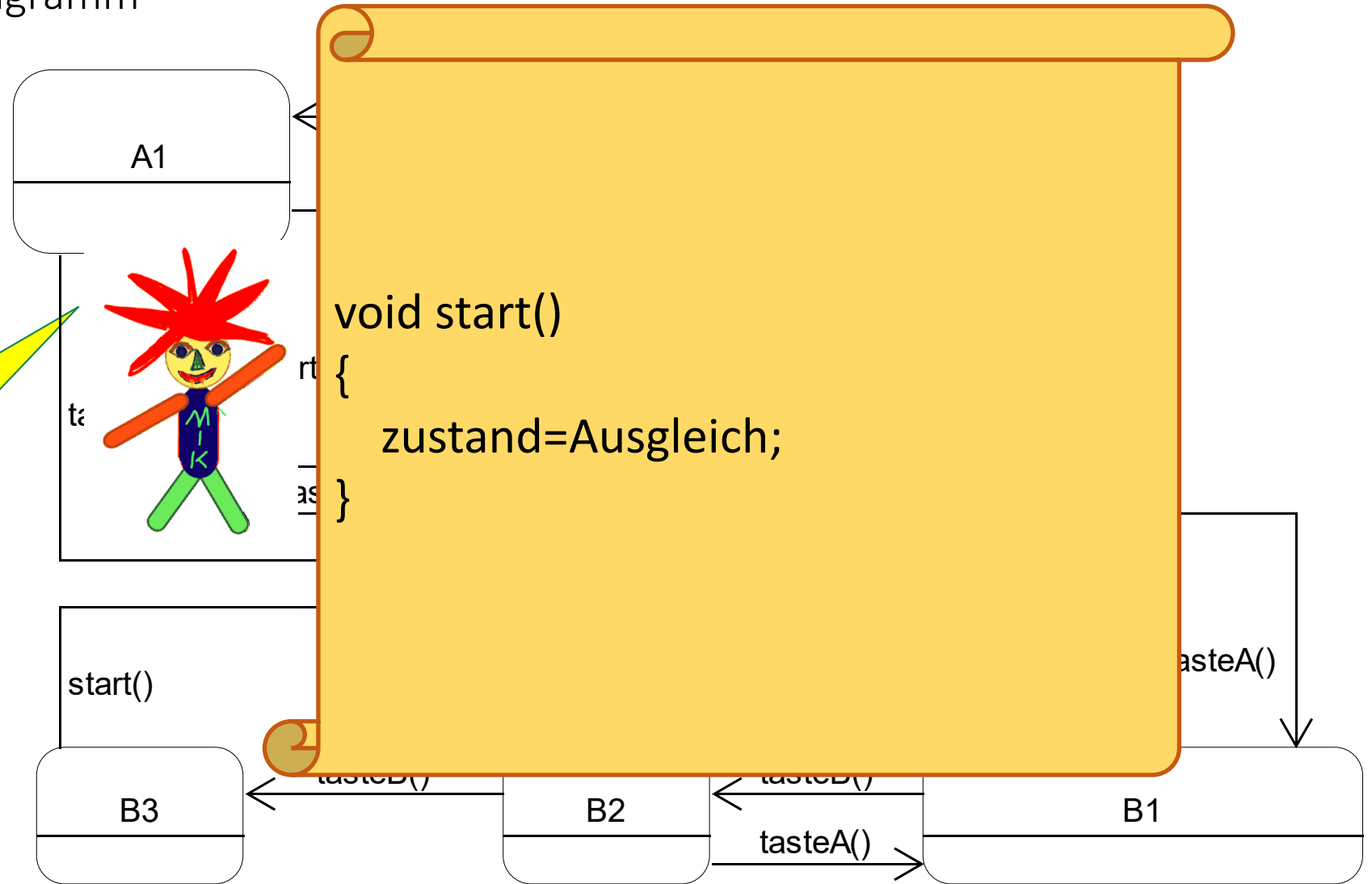
Analogien im Zustandsdiagramm

Wir definieren die Zustände so, dass in jedem Zustand die richtige Ausgabe unmittelbar erfolgt



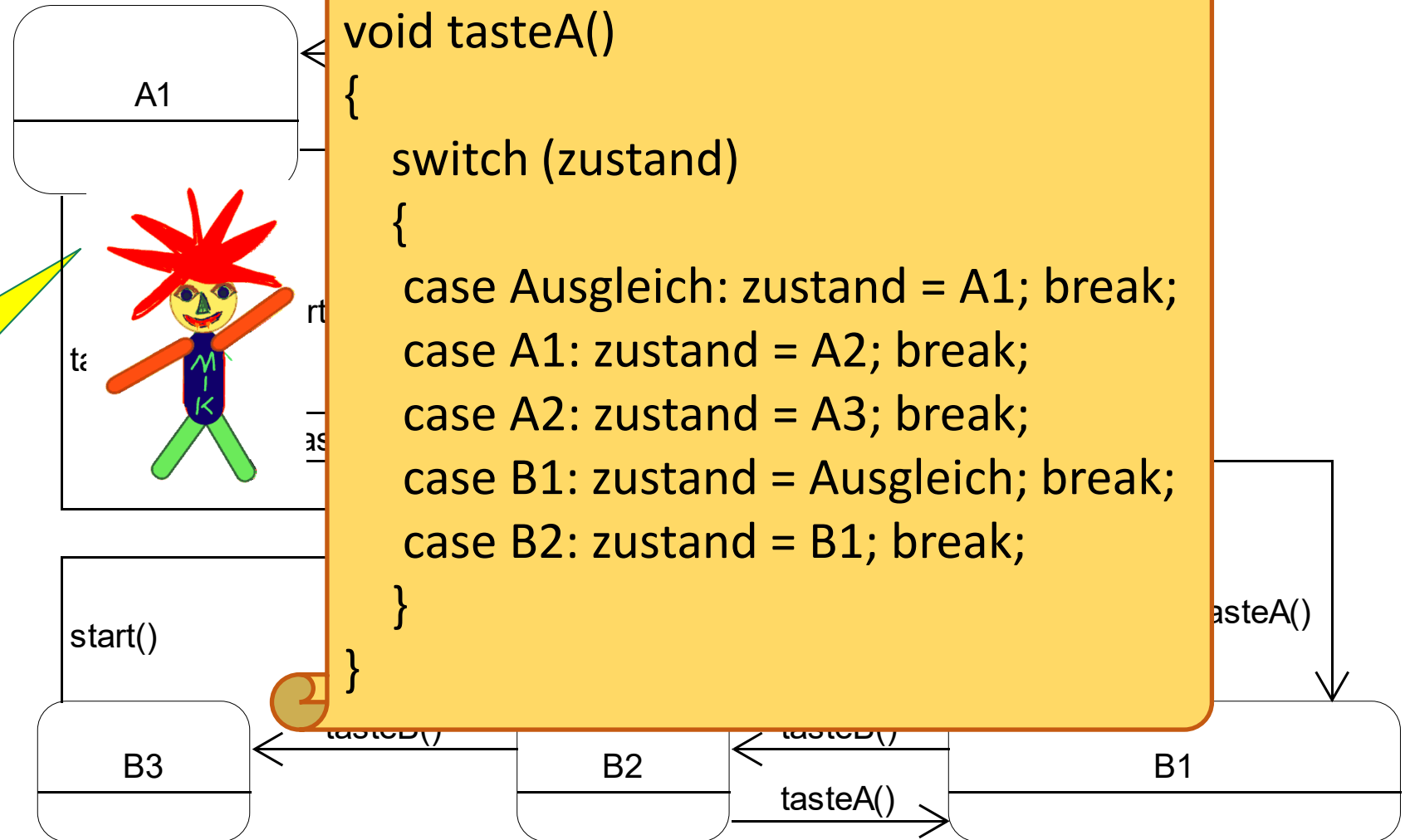
Analogien im Zustandsdiagramm

Bei Start erfolgt immer
ein Übergang in
Zustand Ausgleich



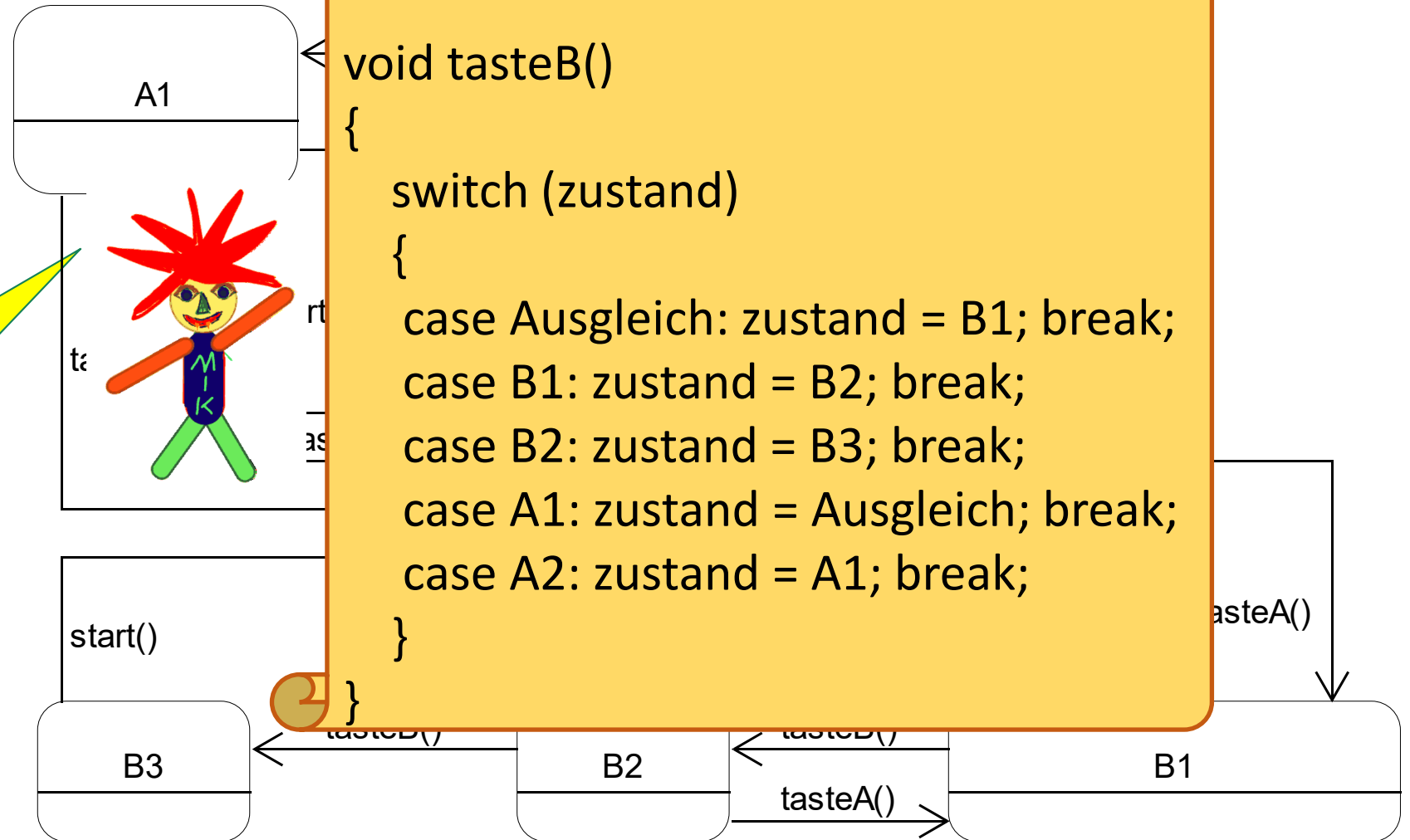
Analogien im Zustandsdiagramm

Bei Taste A erfolgt ein Zustandsübergang in Abhängigkeit vom Ausgangszustand



Analogien im Zustandsdiagramm

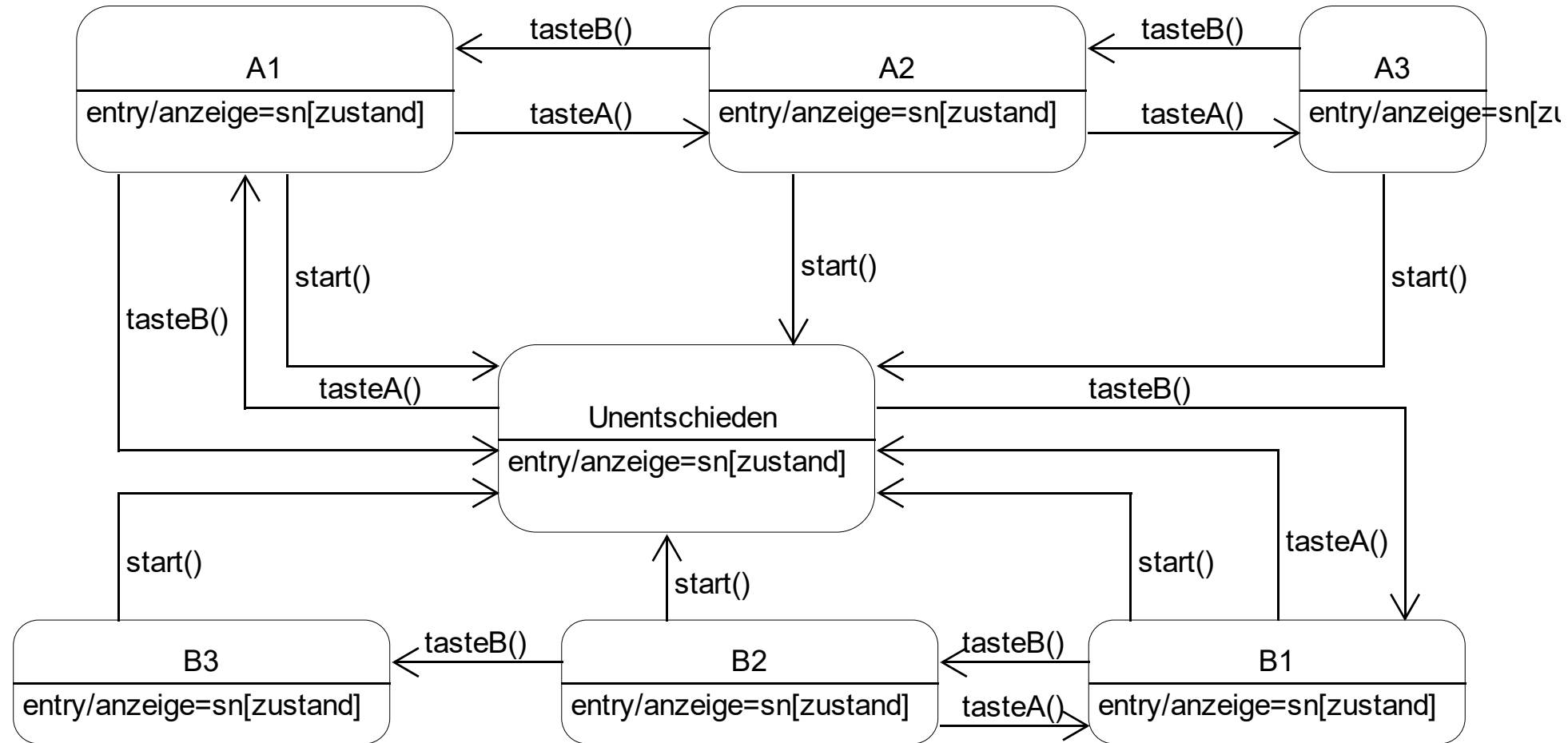
Bei Taste B erfolgt ein Zustandsübergang in Abhängigkeit vom Ausgangszustand



Analogien im Zustandsdiagramm



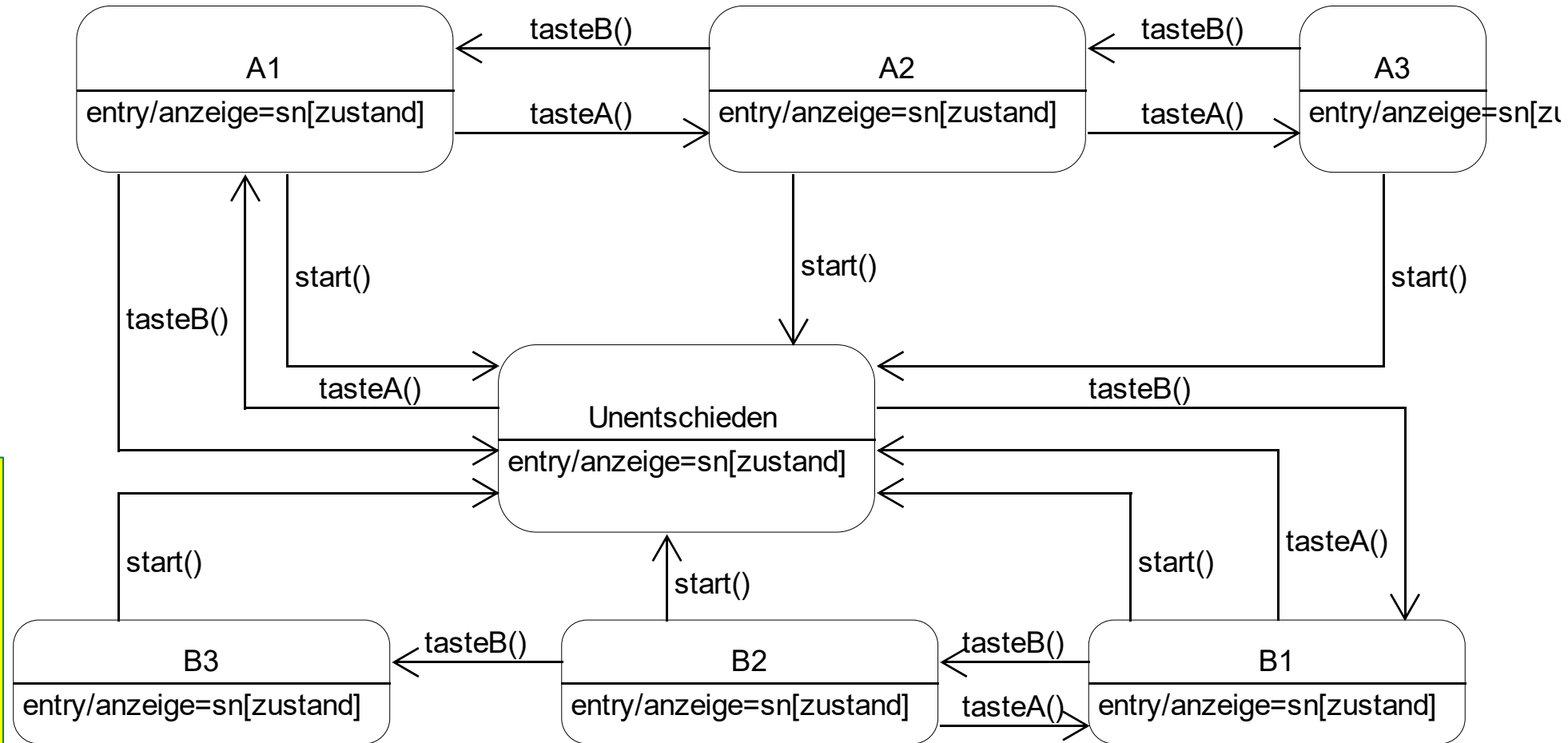
Das UML-Zustandsdiagramm für Lösung mit „Ausgangsschaltnetz“



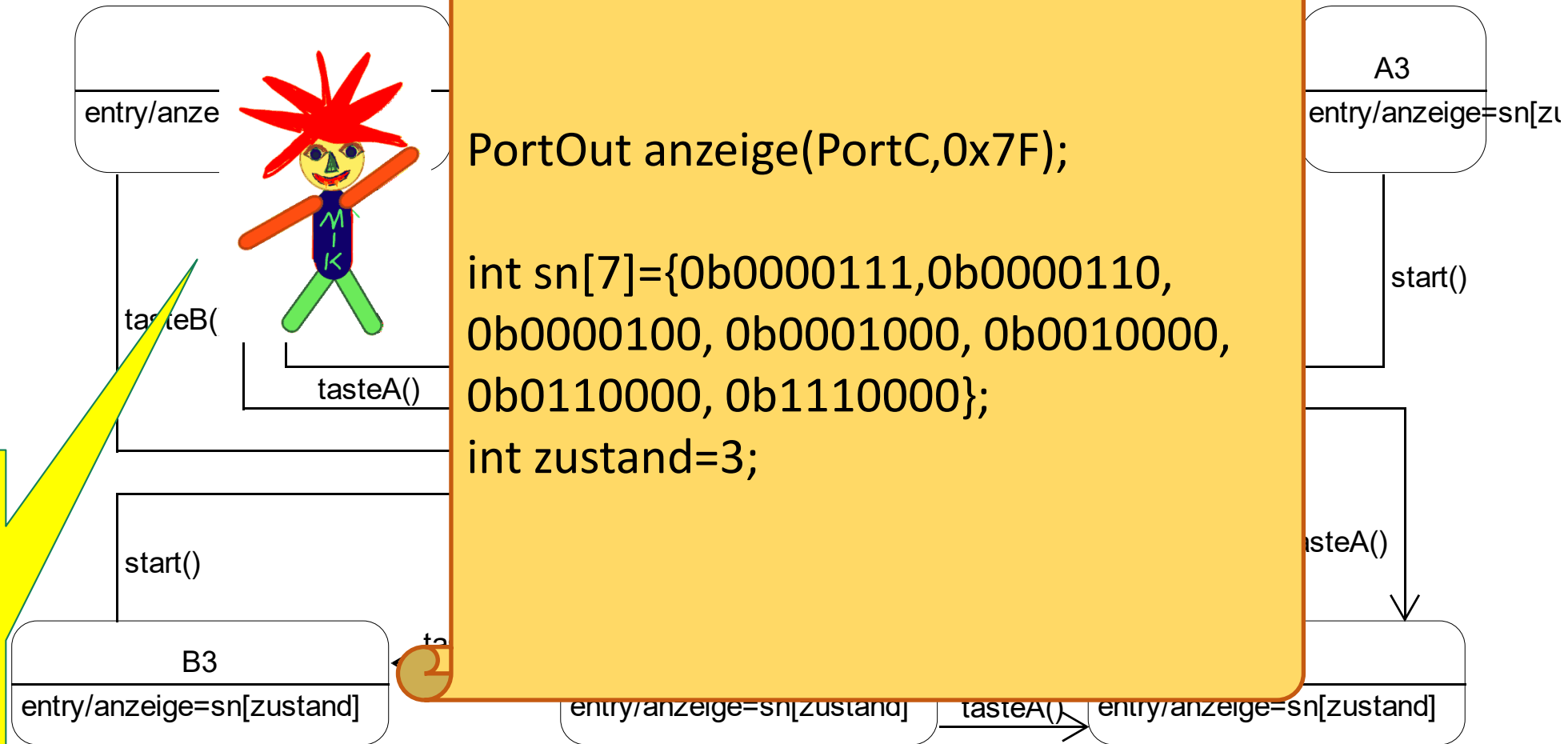
Analogien im Zustandsdiagramm



Die
Ausgangswerte
werden in einem
array `sn[]`
gespeichert und
zur Anzeige
gebracht



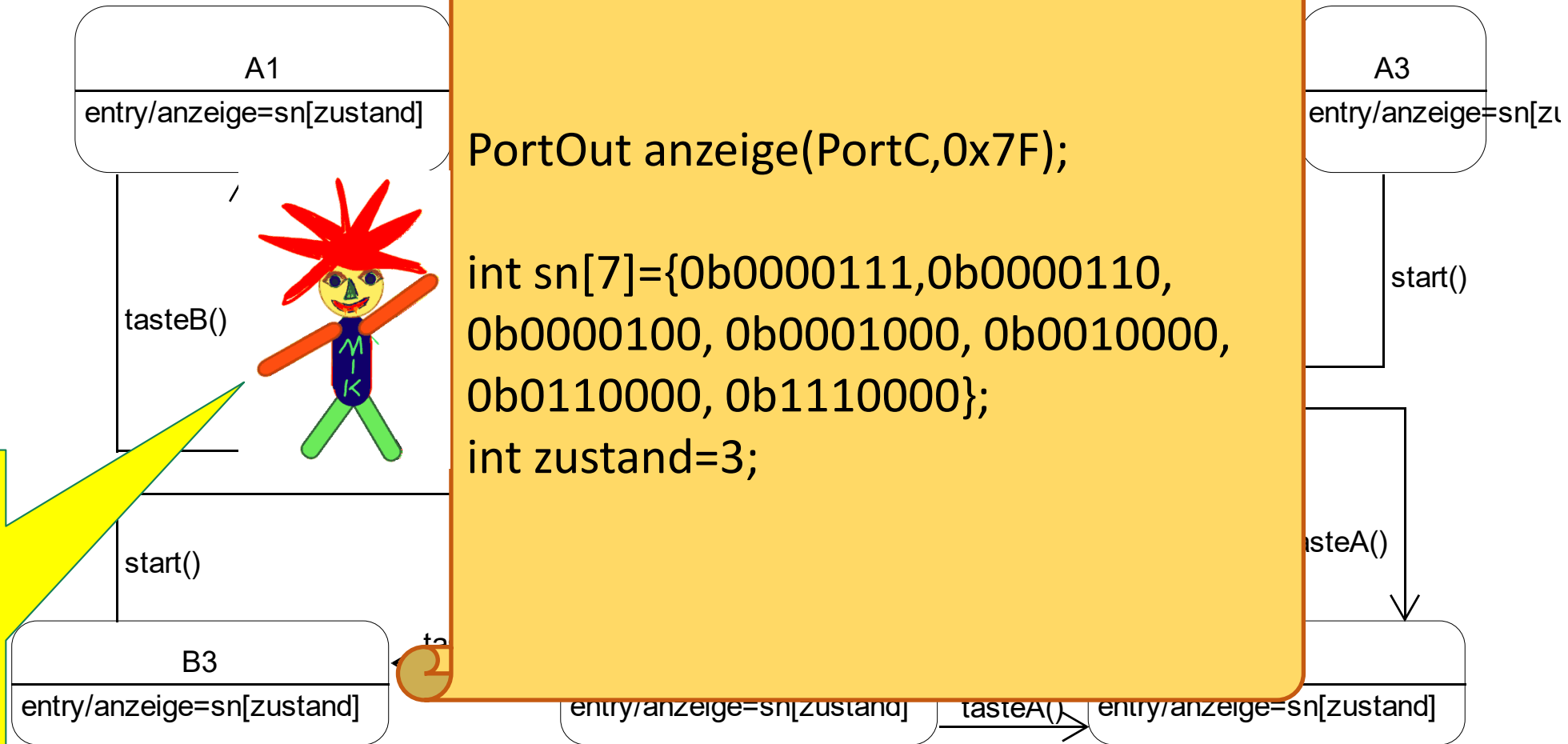
Analogien im Zustandsdiagramm



Die Anzeige erfolgt auf PortC



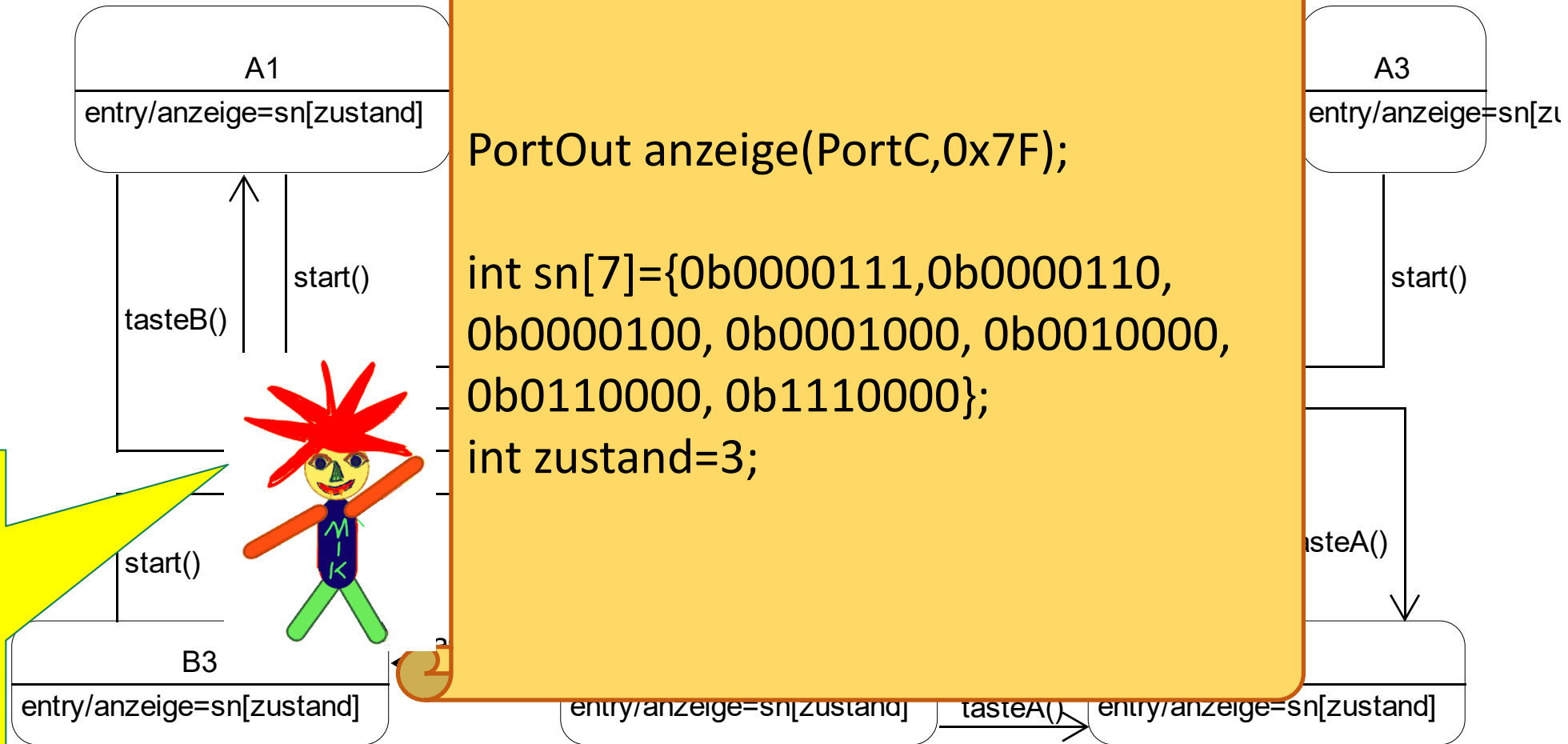
Analogien im Zustandsdiagramm



Das array `sn[]`
(Schaltnetz)
speichert die
Ausgangswerte



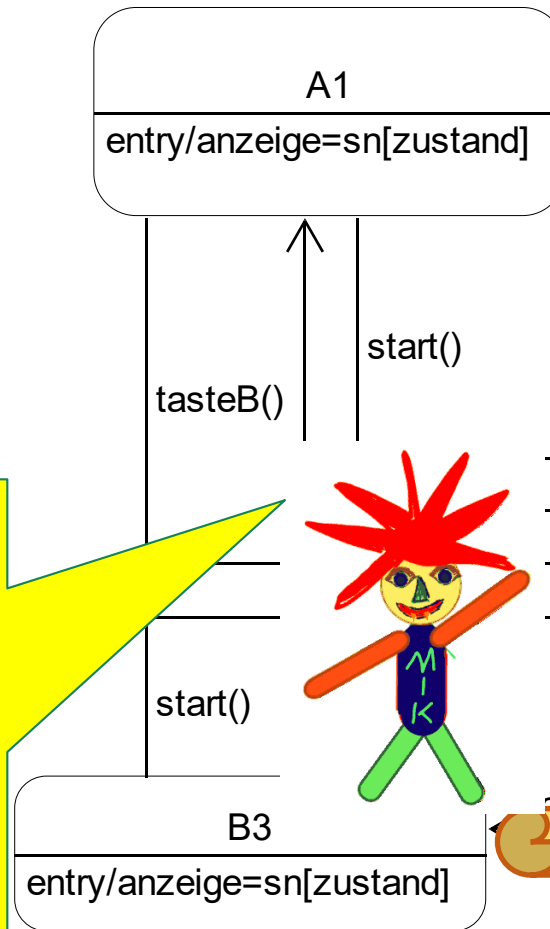
Analogien im Zustandsdiagramm



Der jeweilige
Zustand wird in
der
Zustandsvariablen
`int zustand;`
gespeichert



Analogien im Zustandsdiagramm



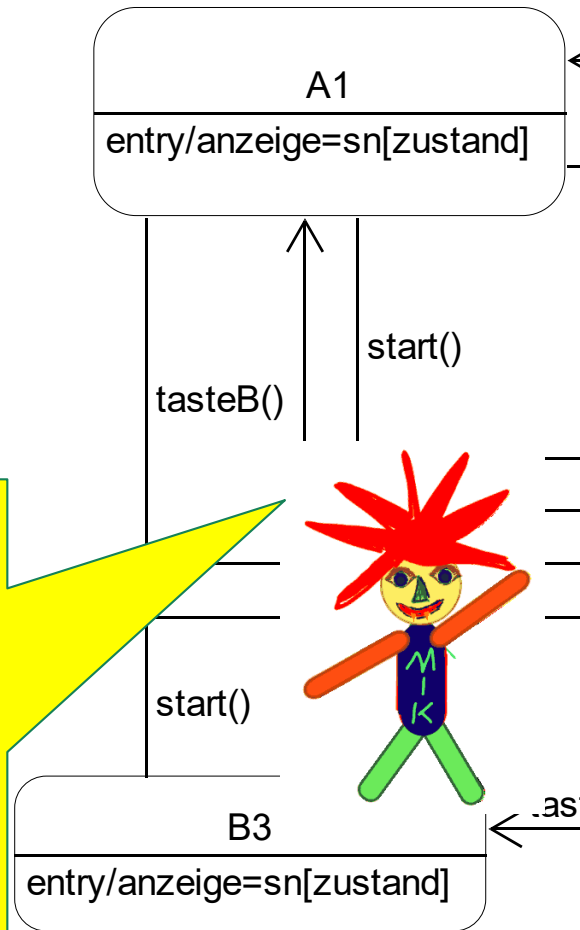
```
PortOut anzeige(PortC,0x7F);
```

```
int sn[7]={0b0000111,0b0000110,  
0b0000100, 0b0001000, 0b0010000,  
0b0110000, 0b1110000};  
int zustand=3;
```

0: B3
1: B2
2: B1
3: Unentschieden
4: A1
5: A2
6: A3



Analogien im Zustandsdiagramm



```
void start()
{
    zustand=3;
    anzeige=sn[zustand];
}

void aufA()
{
    if (zustand>0 && zustand < 6)
        zustand=zustand+1;
    anzeige=sn[zustand];
}

void aufB()
{
    if (zustand<6) zustand--;
    if (zustand== -1) zustand=0;
    anzeige=sn[zustand];
}
```

