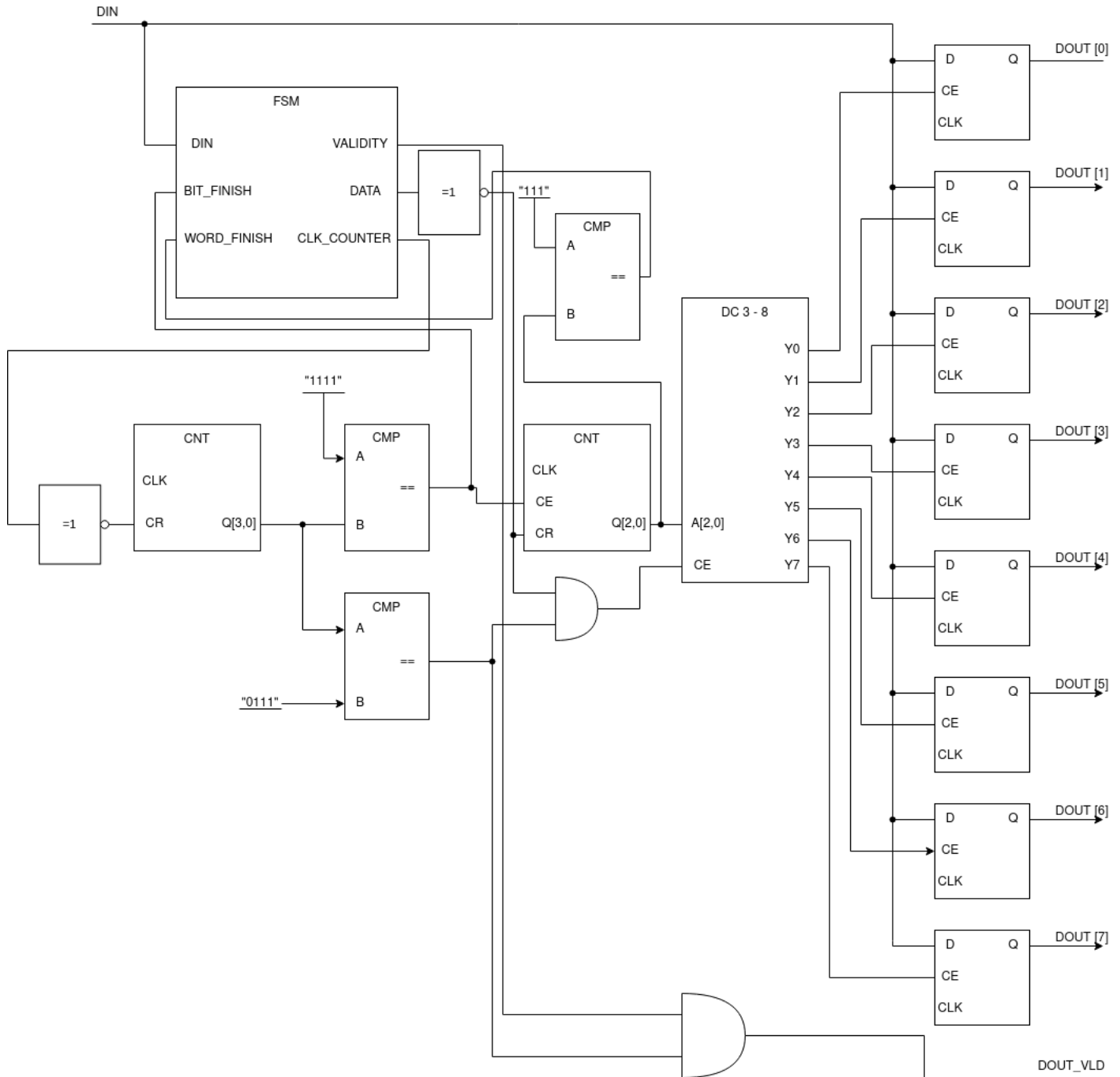


# INC 1 část úkolu

Jméno: Ondřej Novotný  
Login: xnovot2p

## Architektura navrhnutého obvodu (na úrovni RTL)

### Schéma obvodu



## Popis funkce

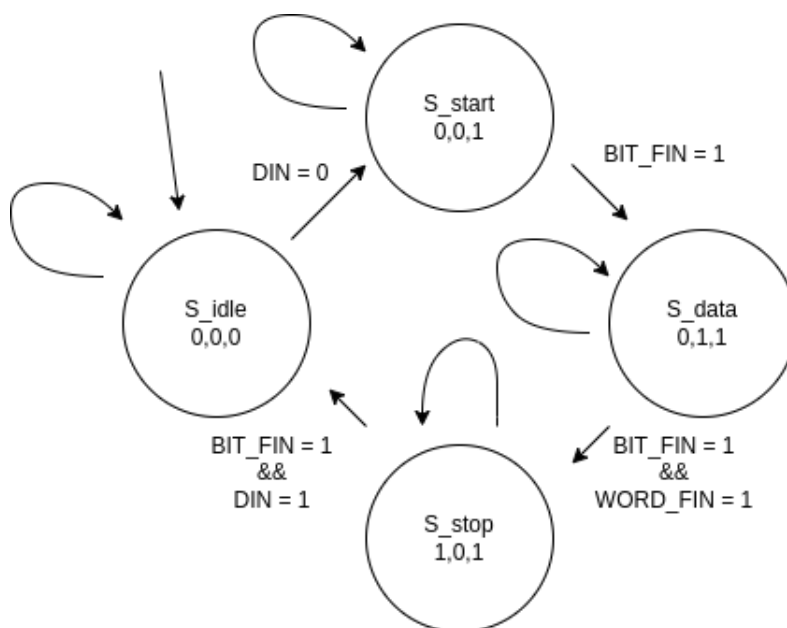
Jakmile FSM pošle na Clk\_counter '1', CNT na CR bude mít vstup 0, tudíž se přestane restartovat a začne počítat. Jedna hodnota vstupu trvá 16 ticků, proto vím, že jakmile mám na výstupu CNT "0111", jsem na mid\_bitu, a povolím zapsat DIN hodnotu do registru. Zápis do určitého registru mi určuje výstup druhého counteru, který se inkrementuje na základě prvního CNT. CNT2 se inkrementuje pouze pokud se výstup CNT1 rovná "1111". Vždy když se výstup CNT1 rovná "1111", odešlu do FSM na BIT\_FINISH jedničku. Pokud na CNT2 přečtu "111" vím, že jsem dočetl 8. bit, a na vstupu očekávám 1 jako Stop\_bit. Poté znovu odesílám na CLK\_COUNTER '0' a čekám opět na start\_bit.

## Návrh automatu (Finite State Machine)

### Schéma automatu

Legenda:

- Stavy: S\_idle, S\_start, S\_data, s\_stop
- Vstupy: DIN, BIT\_FINISH, WORD\_FINISH
- Moorovy výstupy: Validity, Data, Clk\_count



### Popis funkce

Automat čeká ve výchozím stavu S\_idle, kde jsou všechny výstupy rovny '0'. Při obdržení DIN '0' se automat přesune do stavu S\_start, kde se CLK\_COUNTER nastaví na 1. Při obdržení BIT\_FINISH = '1', se přesune do stavu S\_data, se DATA a CLK\_COUNTER nastaví na '1'. Při obdržení BIT\_FINISH = '1' a WORD\_FINISH = '1' se přesune do stavu S\_stop, kde se DATA nastaví na 0 a VALIDITY se nastaví na 1. Pokud dostanu BIT\_FINISH = '1' a DIN = '1', přesune se zpět do stavu S\_idle a čekám opět na DIN '0'.