

Návrh číslicových obvodů

Jan Kořenek

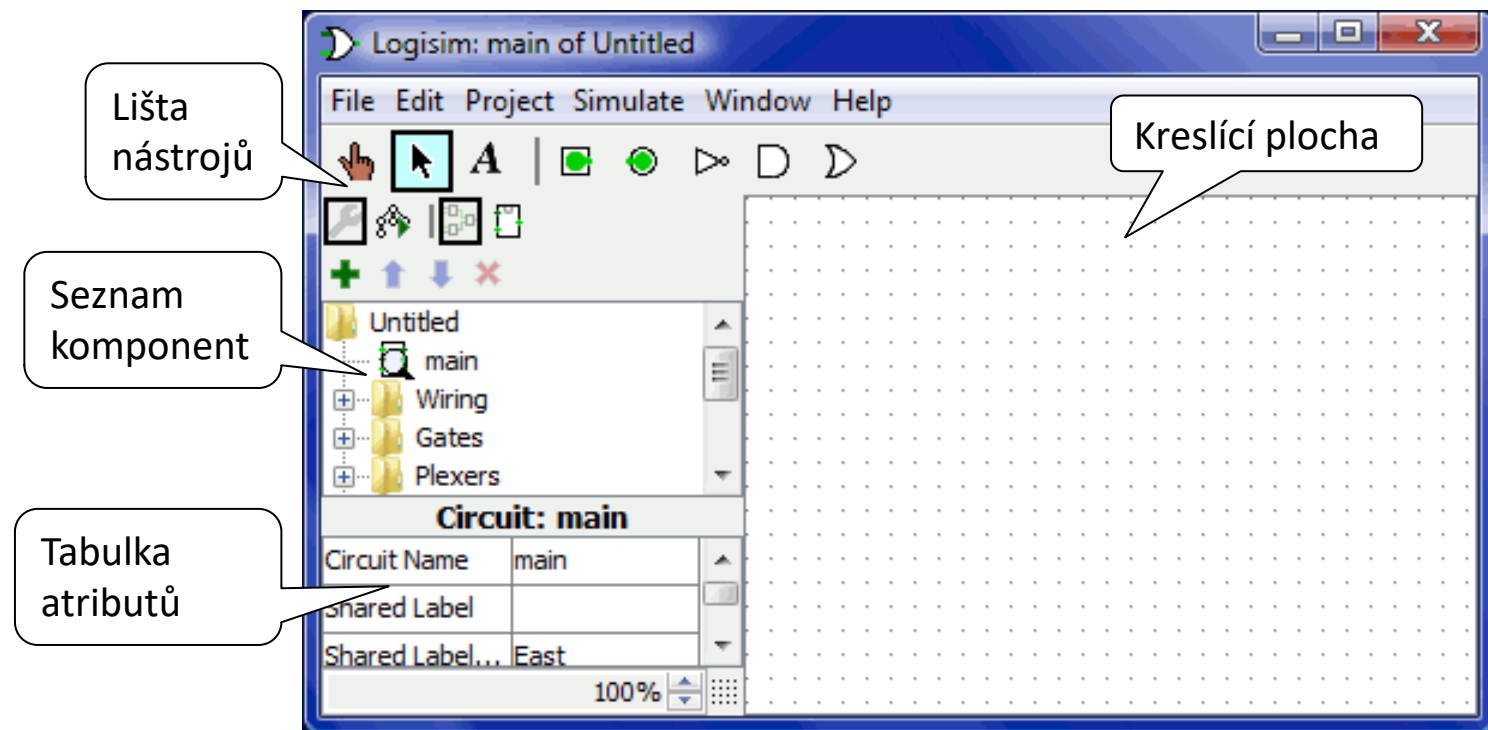
Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2, 612 66 Brno - Královo Pole
korenek@fit.vutbr.cz



- Z předchozích přednášek byste měli znát:
 - Základní logické prvky? AND, OR, NOT, NOR, NAND, XOR, ...
 - Kombinační moduly: kodér, dekodér, multiplexor, demultiplexor, sčítačka, ...
 - Sekvenční obvody: klopné obvody (RS, JK, T a D), registr, posuvný registr, čítač, ...
 - Modelování kombinačních obvodů pomocí pravdivostní tabulky, sekvenčních obvodů pomocí automatů (Mealyho nebo Moorův).
- Jak pomocí základních obvodových prvků navrhnout požadovaný obvod?

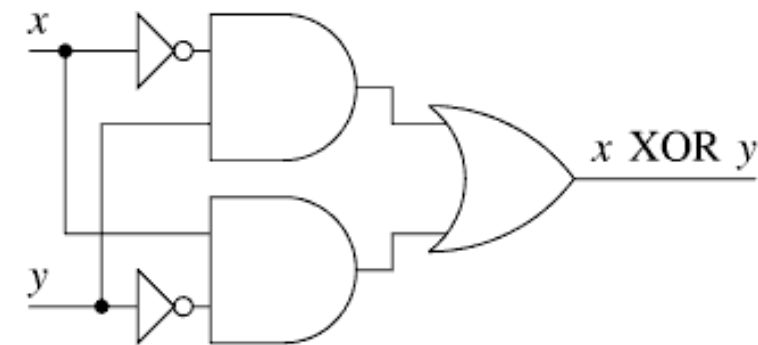
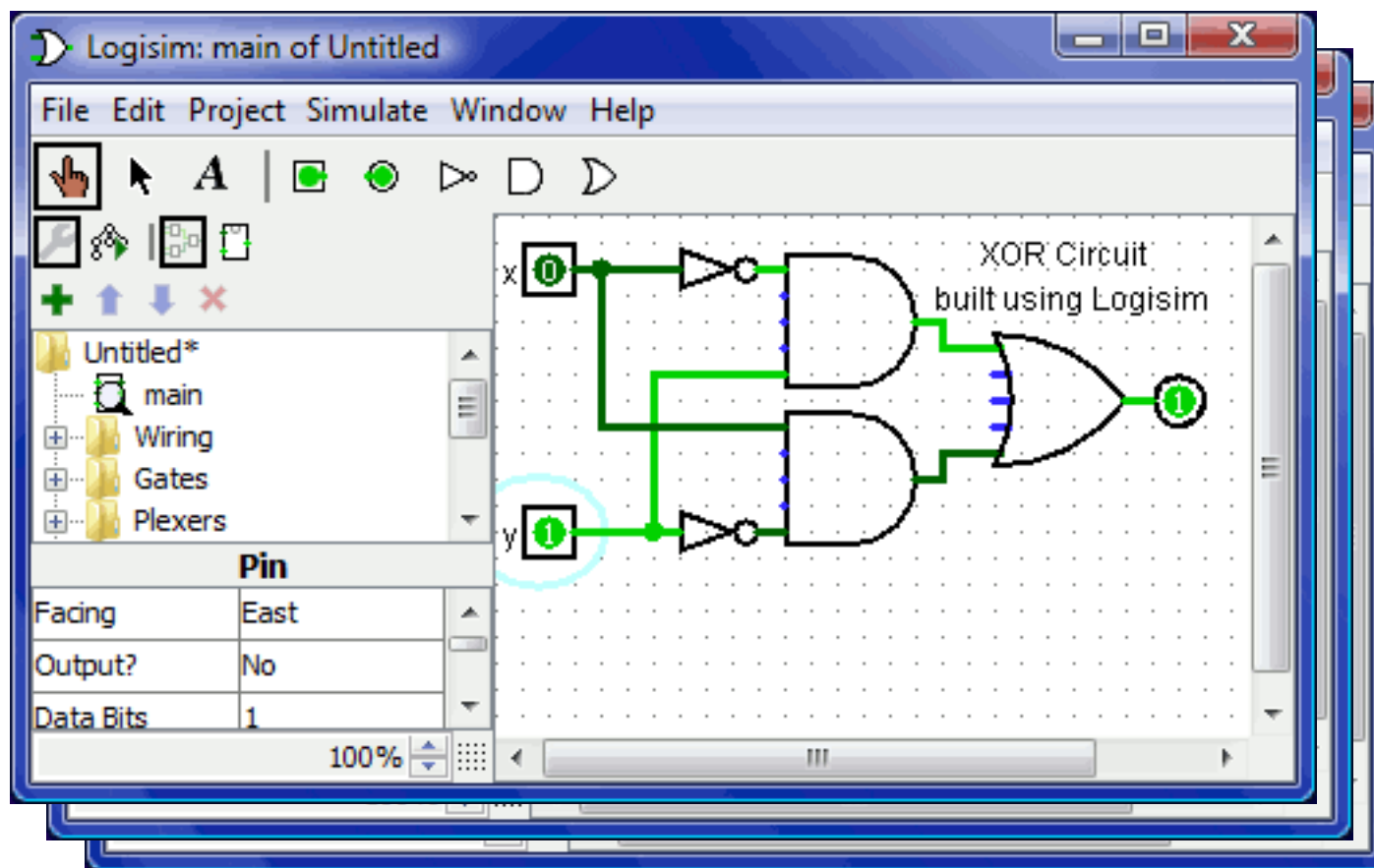
- Úvod do výukového programu Logisim
- Návrh číslicových obvodů
 - Jednoduchý synchronní obvod řízený automatem (LED indikace)
 - Základní zapojení, návrh automatu, automat jako dílčí obvod
 - **Princip synchronního návrhu obvodu**, vysvětlení, ukázka v Logisim, clock-by-clock behaviour
 - Způsoby řízení displeje (multiplexor, paměť, BCD dekodér)
 - Obvod s připojenými segmenty, obvod s multiplexorem, obvod s pamětí a obvod s BCD dekodérem
 - **BCD dekodér jako vytvořený modul (vstupy, výstupy, architektura, použití jako komponenta)**
 - Počítadlo nahoru/dolu ovládané tlačítkem (doplnění čítače, přidání řídicího automatu)
 - **Zobecnění datová a řídicí cesta**
 - Ukázka práce s aritmetikou (kalkulačka s operacemi plus, mínus, násobení, dělení – tlačítka, operace)
 - Vysílání dat po sériové lince pro ovládání periférií
 - Návrh odeslání jednoho slova po sériové lince
 - Práce s registrem a pamětí (aktuální hodnota, průměr, vstupních hodnot)
 - Generátor vstupu, zobrazení aktuální hodnoty, výpočet a zobrazení průměru, histogram teplot
- Shrnutí návrhových postupů a kontrolní otázky

- Nástroj pro návrh a simulaci jednoduchých číslicových obvodů
- Postup návrhu
 - Vložení vstupů, hradel a dalších komponent
 - Propojení komponent
 - Testování a simulace obvodu



- Postup návrhu

1. Vložení vstupů, hradel a dalších komponent
2. Propojení komponent
3. Testování a simulace obvodu

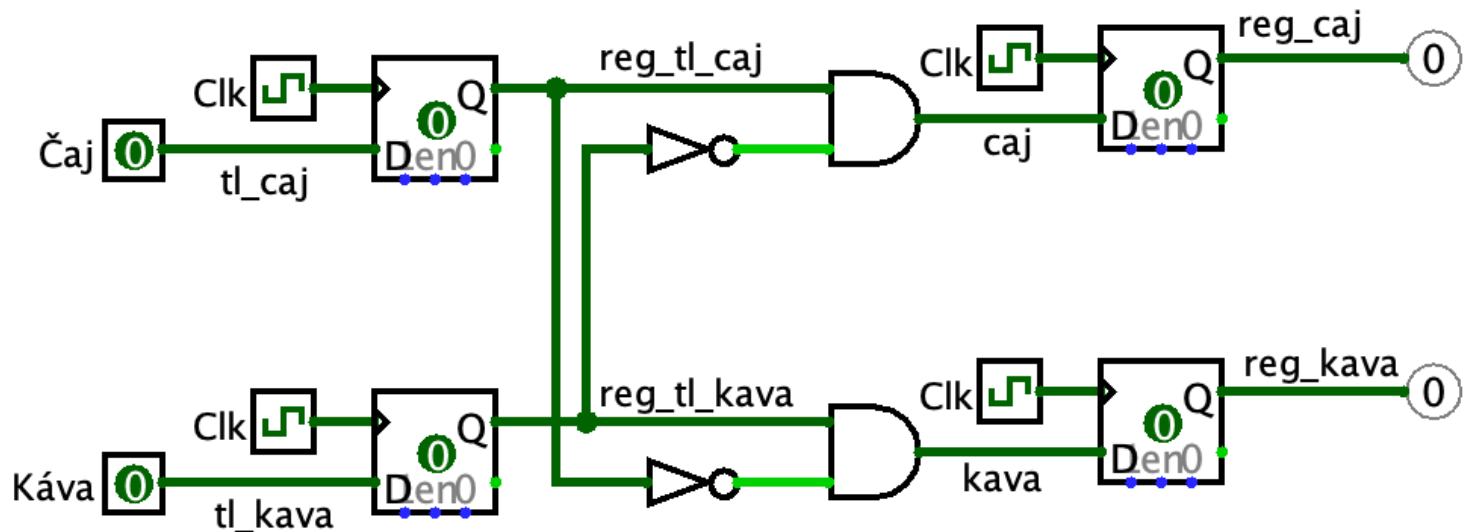


Modelovaný obvod

01-xor.circ

Více informací k programu **Logisim** najdete na webových stránkách kurzu

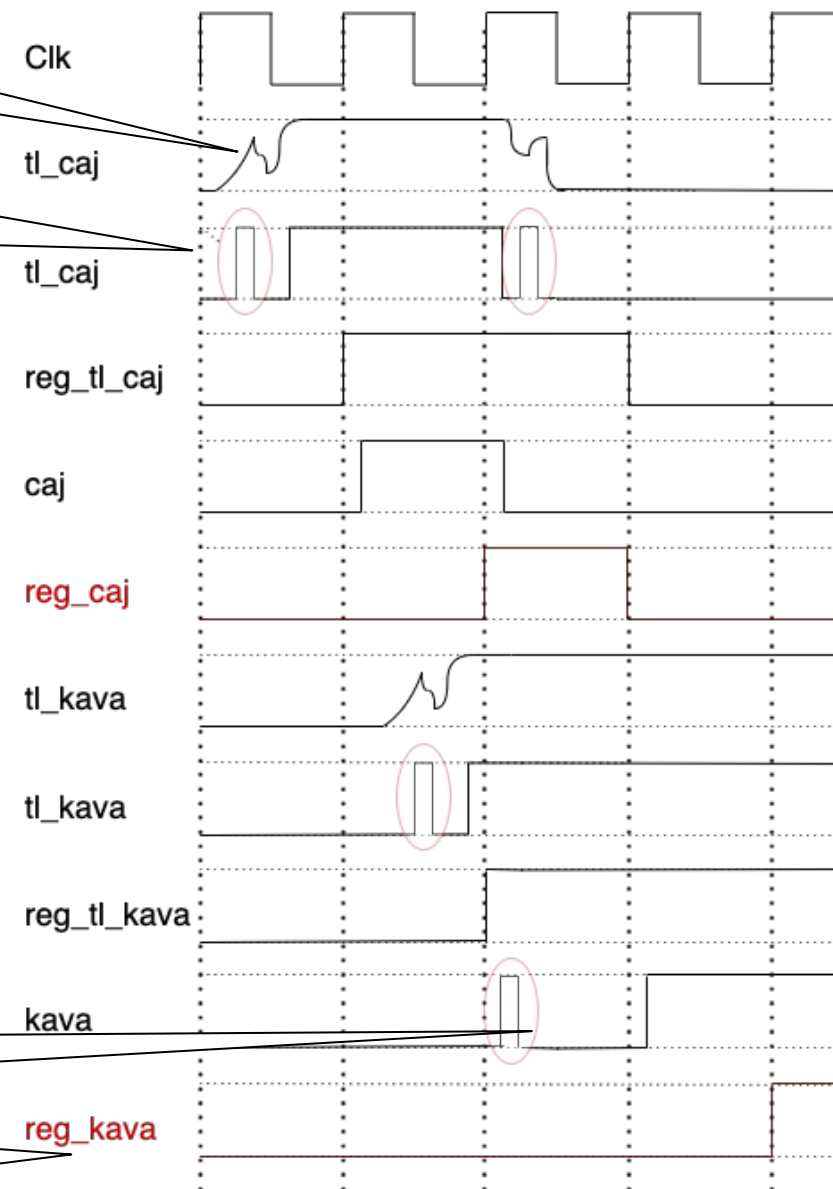
- Všechny výpočty v obvodu jsou řízeny hodinovým signálem
- Synchronizace na náběžnou nebo sestupnou hranu hodinového signálu



02-synchronni-obvod.circ

Postupné sepnutí
kontaktní tlačítka

Vzorkovaný
průběh (0/1)
(zámkity)



Zámkit způsobený
zpožděním invertoru

Odstranění zámkitů

- *Synchronní obvody se vyznačují tím, že jsou řízeny (synchronizovány) hodinovým signálem (Clk), většinou náběžnou nebo na sestupnou hranou.*
- Výhody synchronního návrhu
 - Do registrů se vzorkují data pouze v okamžiku synchronizace, náběžné nebo sestupné hrany hodinového signálu
 - Analyzují obvod pouze v ustáleném stavu, až všechny signály projdou kombinačními obvody ze výstupů jedněch registrů na vstupy druhých registrů
 - Není potřeba uvažovat zákmity, hazardy a podobně.
- Veškeré změny jsou koncentrovány na diskrétní okamžiky synchronizace, což vede na velké proudové odběry a vyšší příkon.

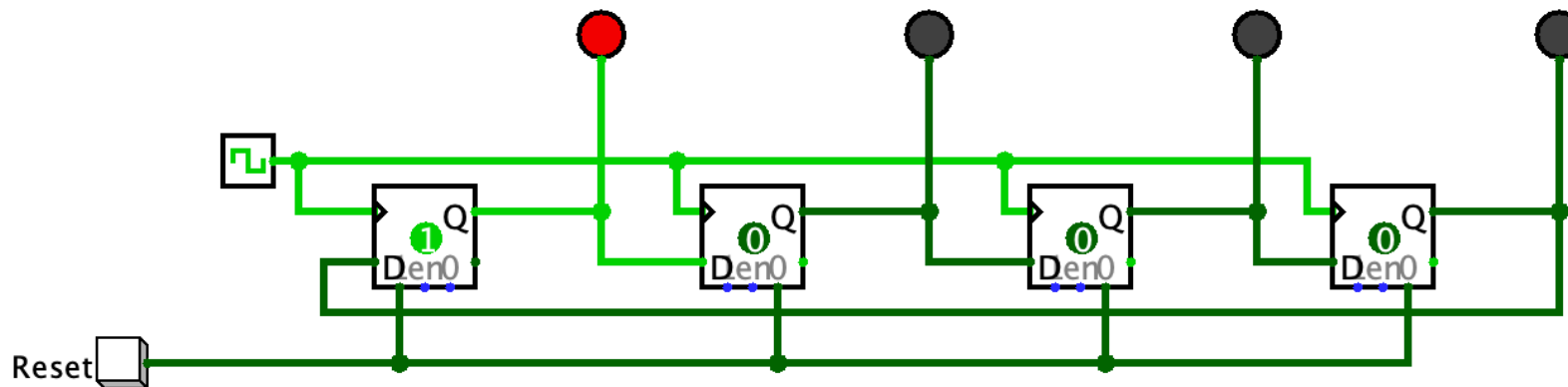
- Synchronní obvod pro cyklické blikání čtyř LED diod

- Řízení blikání synchronním automatem

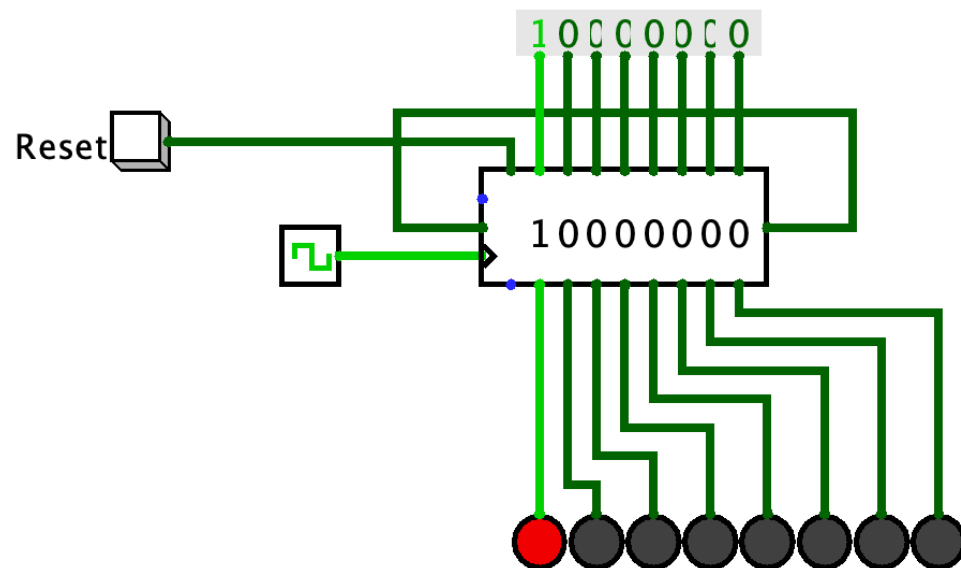


- Kdy přechází automat do dalšího stavu?
 - Přechody spouští náběžná hrana hodin.
- Měla by být náběžná hrana značena u přechodů automatu?
 - Ne, jedná se o synchronizaci.

- Obvodová realizace, kdy každý stav je reprezentován jedním bitem



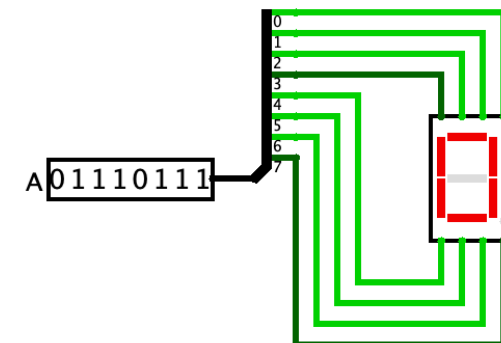
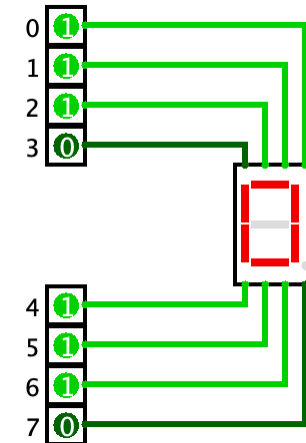
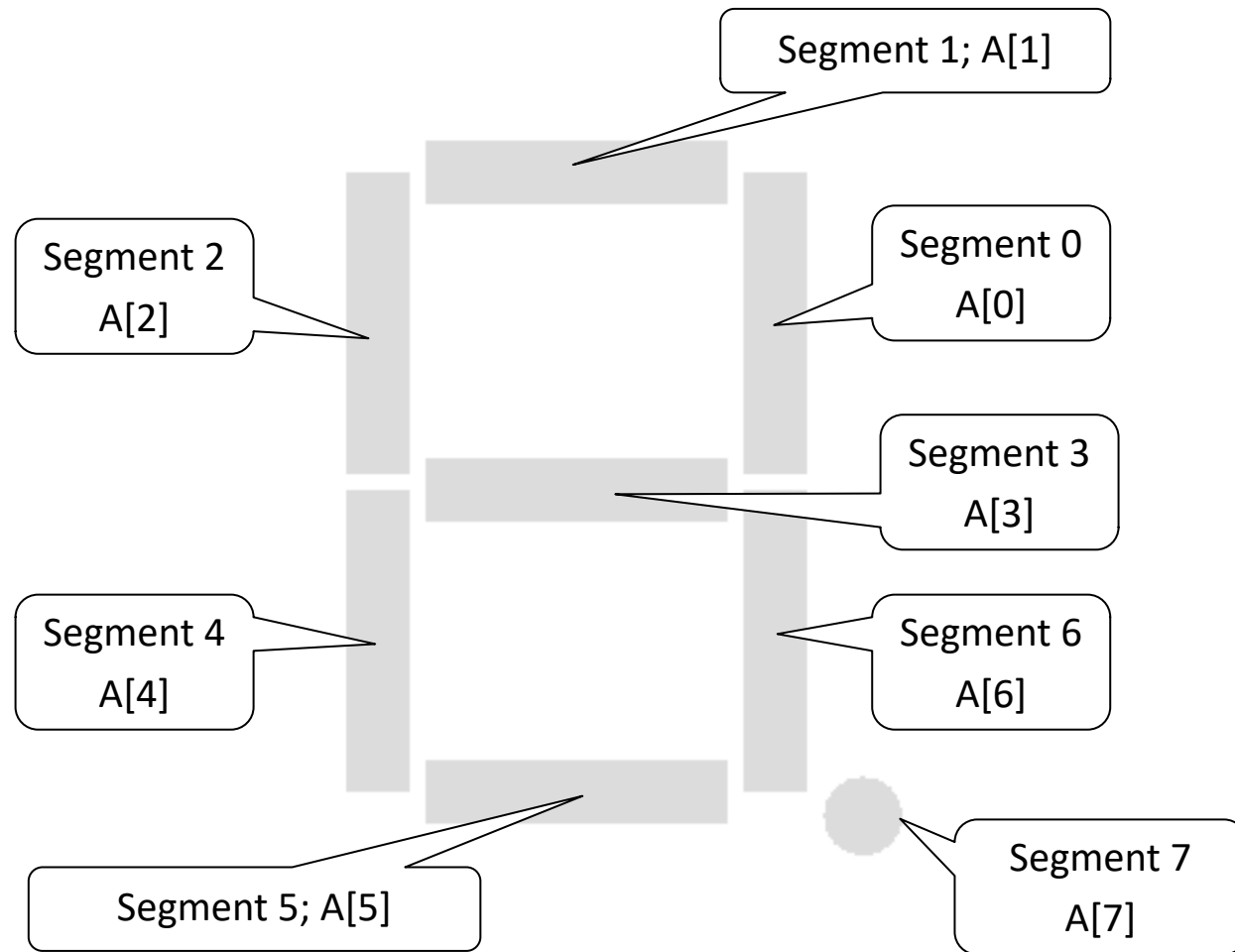
- Řešení pomocí synchronního posuvného registru



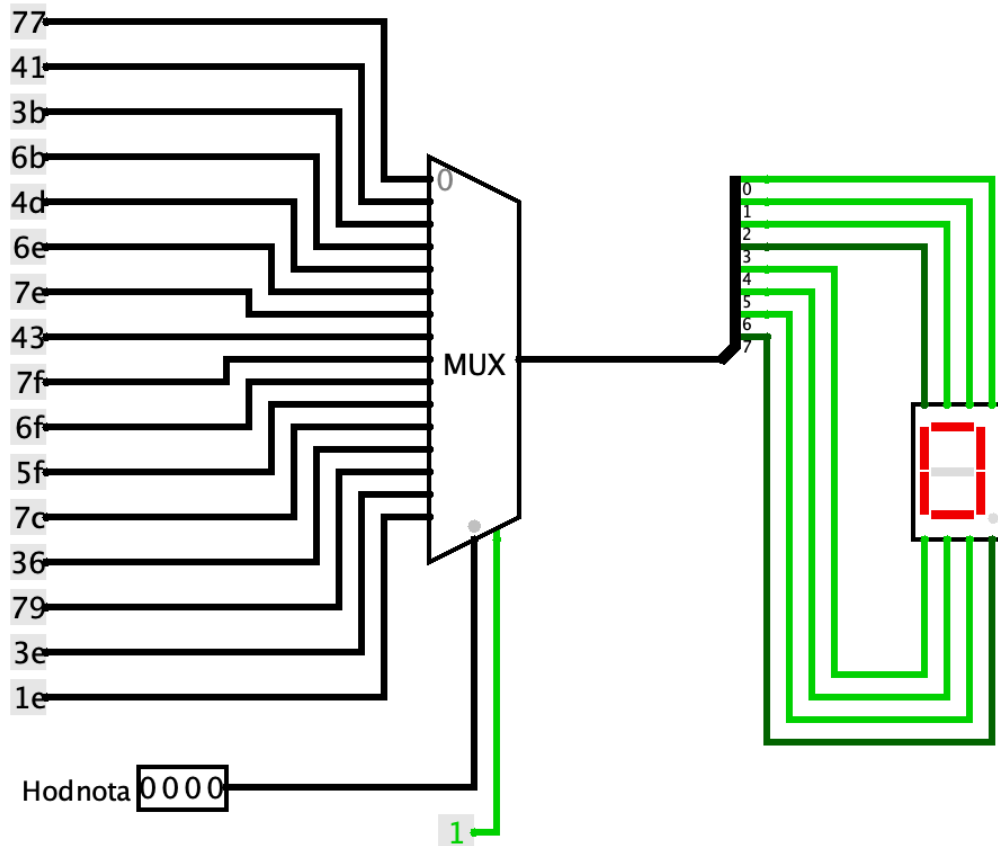
- Tlačítko Reset přednastaví hodnotu posuvného registru (svítící LED)
- S každou náběžnou hranou hodin se provede posun v registru a rotace nejnižšího bitu na pozici nejvyššího (cyklická rotace bitů)
- Jednotlivé bity posuvného registru jsou přivedeny k LED diodám
- Různá přednastavení hodnot v registru vedou na různé typy blikání LED

Stáhněte si připravený obvod v programu Logisim a sestavte si vlastní systém blikání na kolo

- Pomocí log.1 se rozsvítí nebo zhasne jeden ze 7 segmentů
- Zapnutím vybraných segmentů se mohou zobrazit čísla nebo písmena

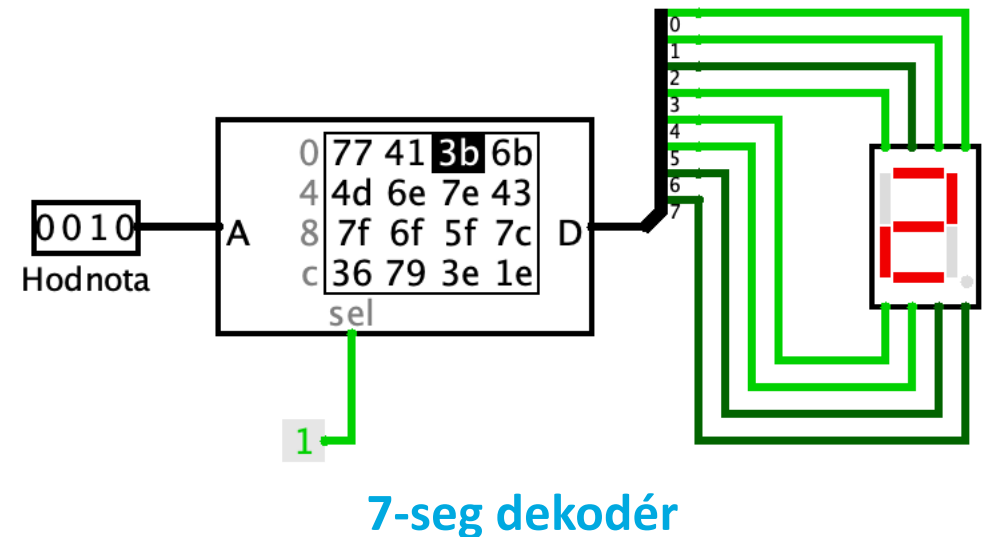


- Multiplexor přednastavených hodnot



06a-7seg-mux16.circ

- Efektivní implementace pomocí paměti ROM nebo RAM
 - Zobrazená hodnota je adresa do paměti, kde jsou uloženy řídicí signály 7-segmentového displeje
 - Pro změnu zobrazení stačí změnit obsah paměti ROM nebo RAM

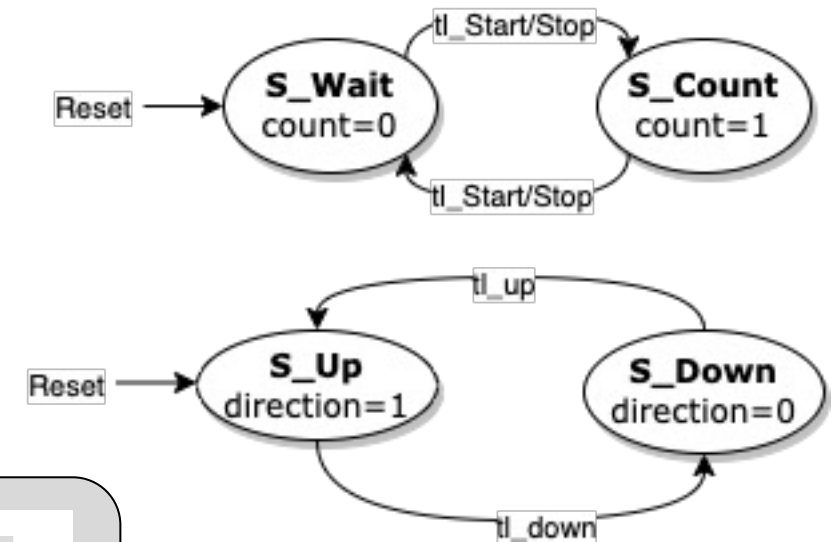
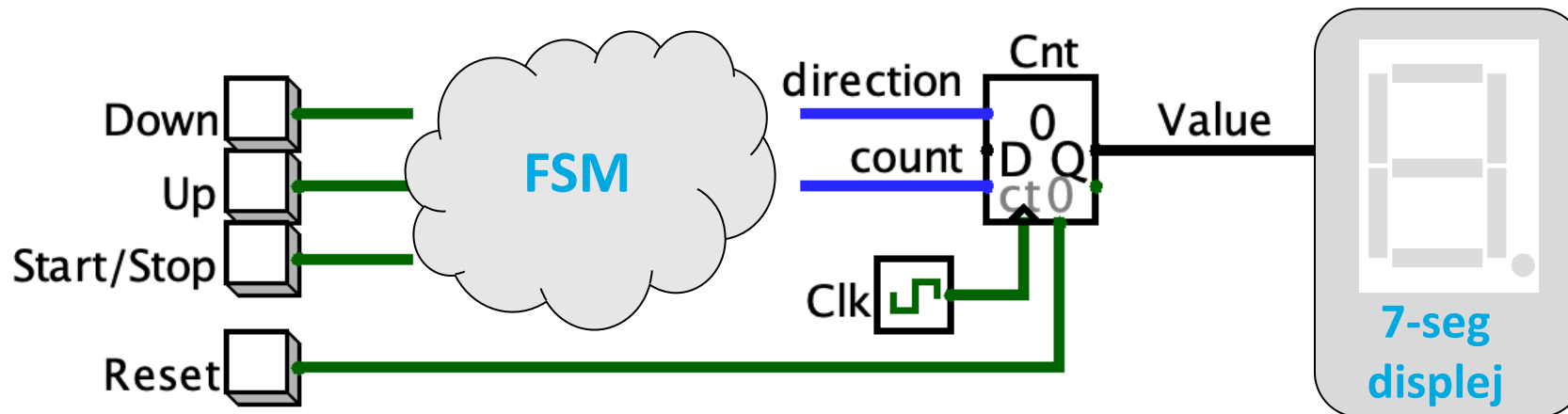


7-seg dekodér

06b-7seg-mem16.circ

- Ovládání odpočítávání nahoru/dolu s tlačítkem Start/Stop

- Počítání zajišťuje čítač „Cnt“
- Tlačítko „Reset“ uvádí zařízení do počátečního stavu
- Tlačítka „Up“, „Down“ přepínají čítání nahoru/dolů
- Tlačítko „Start/Stop“ pozastavují a znovu spouští čítání čítače



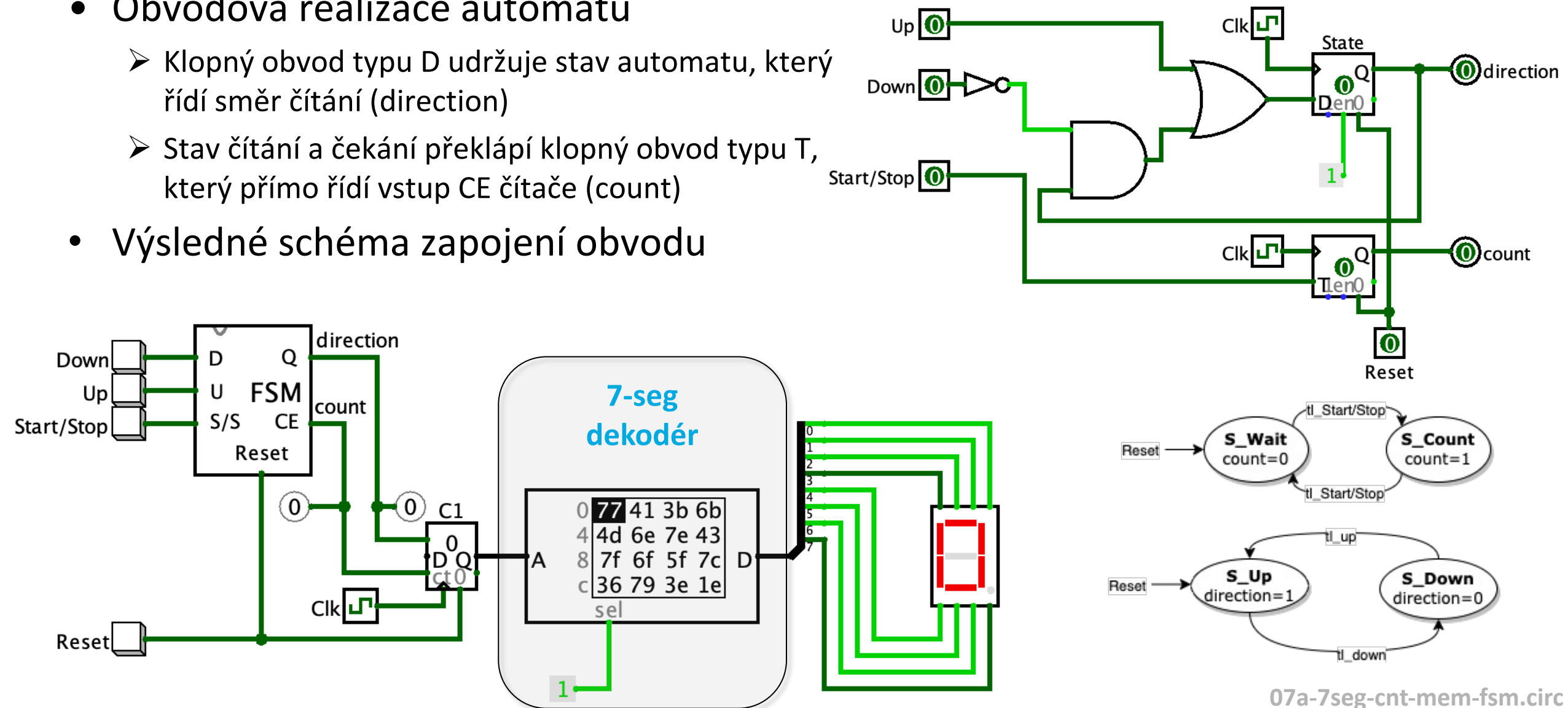
Řídící automaty

každý signál řídí samostatný automat

• Obvodová realizace automatu

- Klopný obvod typu D udržuje stav automatu, který řídí směr čítání (direction)
- Stav čítání a čekání překlápí klopný obvod typu T, který přímo řídí vstup CE čítače (count)

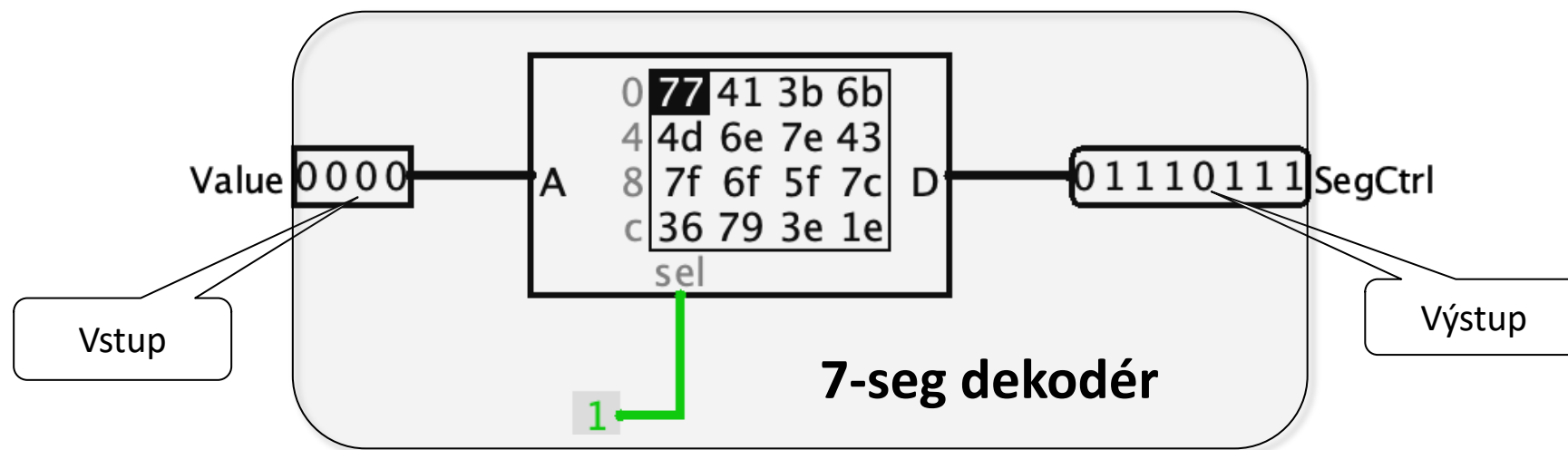
• Výsledné schéma zapojení obvodu



07a-7seg-cnt-mem-fsm.circ

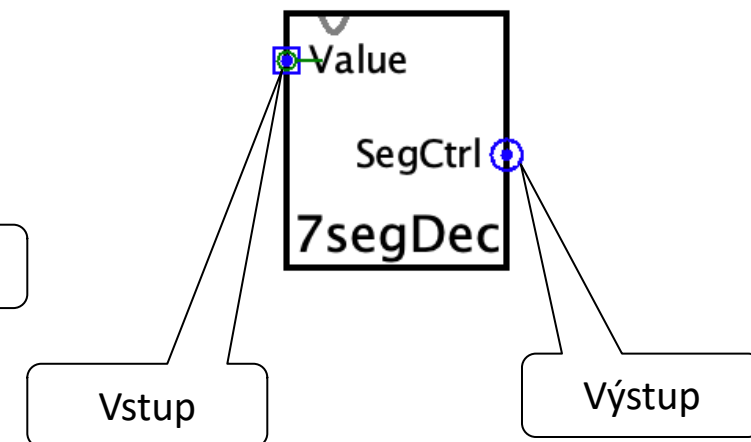
- Jak využít už existujících návrhů obvodu při návrhu složitějšího zapojení?
- Vytváření komponent (modulů)
 - Definice vstupních a výstupních pinů (portů)
 - Schéma zapojení pracující se vstupními piny a generující výstup na výstupní piny.
 - Vytvoření značka komponenty tak, aby se dala komponenta použít ve schématu

Definice vstupů a schéma zapojení



07b-7seg-cnt-mem-fsm-modul.circ

Schématická značka



• Vytvoření komponenty (modulu)

- Nejprve přidání komponenty tlačítkem + v módu editace, pak definice vstupu, výstupu a schéma
- Po definici vstupů a výstupů je možné přejít na kreslení značky: v kontextovém menu zobrazeném po stisku pravého tlačítka volba „Edit Circuit Appearance“

Přidání komponenty (modulu)

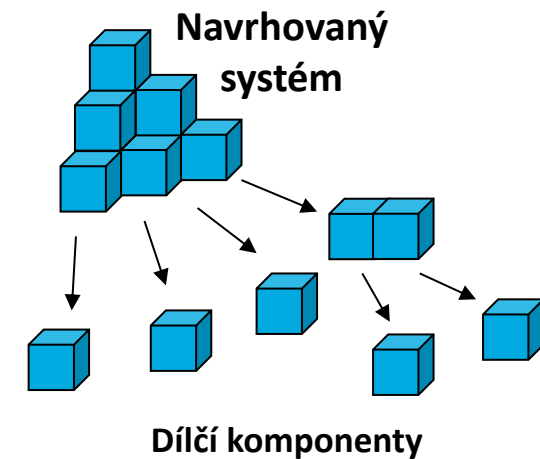
Všechny komponenty (moduly)

Definice schématické značky komponenty

The screenshots show the Logisim interface with a circuit titled "7segModul of 7seg-modul-a-counter-up-down". The left screenshot shows the component being added to the circuit. The right screenshot shows the context menu with "Edit Circuit Appearance" selected.

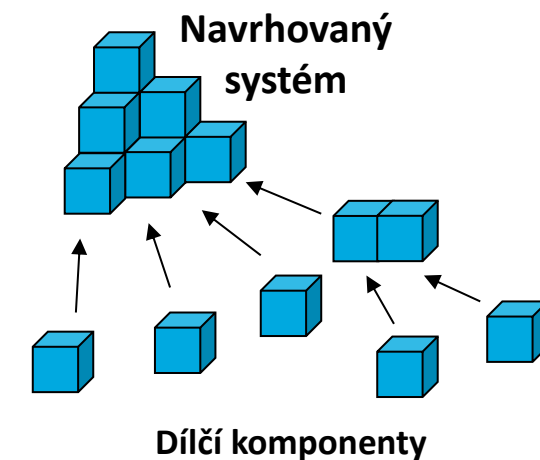
- **Metoda z hora dolů (top-down)**

- Nejprve je zběžně specifikovaná funkce celého systému.
- Každá část systému (komponenta) je následně zpřesněna do větší míry detailu. Současně jsou identifikovány nové dílčí komponenty systému.
- Opakuje se zpřesňování funkce pro každou nově identifikovanou komponentu až pro dostatečně přesnou specifikaci celé funkce systému.



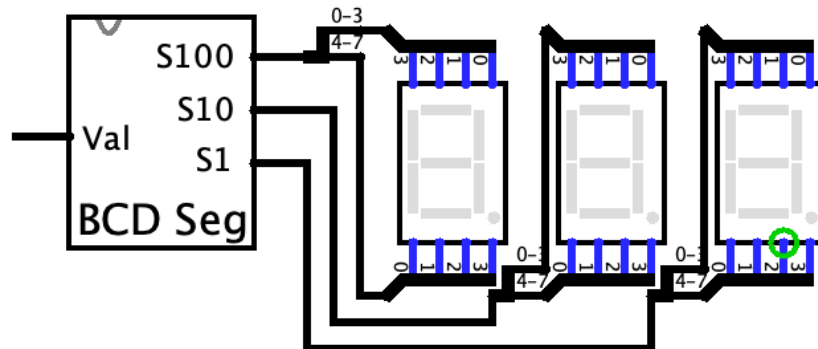
- **Metoda zdola nahoru (bottom-up)**

- Jednotlivé části systému jsou specifikovány do detailu jako samostatné části (komponenty).
- Komponenty jsou pak spojovány dohromady tak, aby tvořily složitější komponenty.
- Vytváření složitějších komponent z jednodušších pokračuje až do okamžiku vytvoření celého systému.

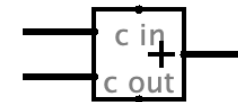


- Navrhněte jednoduchou kalkulačku pro sčítání, odčítání, násobení a dělení čísel
- Využijte již vytvořené komponenty (moduly)
 - Dekodér pro zobrazování čísel dekadicky na 7-segmentovém displeji
 - Komponenty pro jednotlivé operace: sčítání, odčítání, násobení dělení

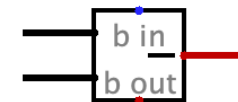
Dekodér pro zobrazení
čísel dekadicky



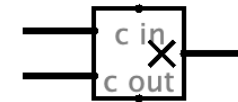
Sčítačka



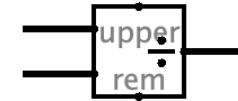
Odčítačka



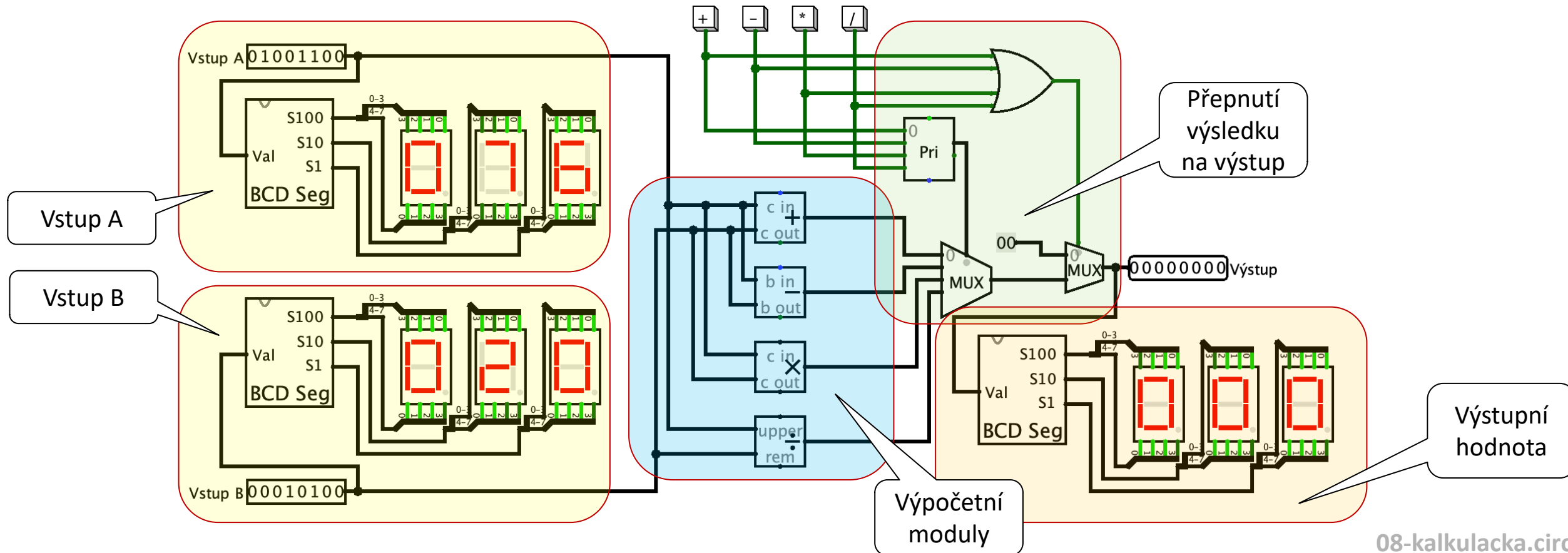
Násobička



Dělička



- Průběžně se počítají všechny operace (součet, rozdíl, násobení i dělení)
- Multiplexory přepínají tlačítka zvolenou operaci na výstup



08-kalkulacka.circ

- Sériová komunikace umožňuje přenášet i široká datová slova dat po malém počtu propojovacích vodičů (méně vodičů redukuje náklady na přenos)

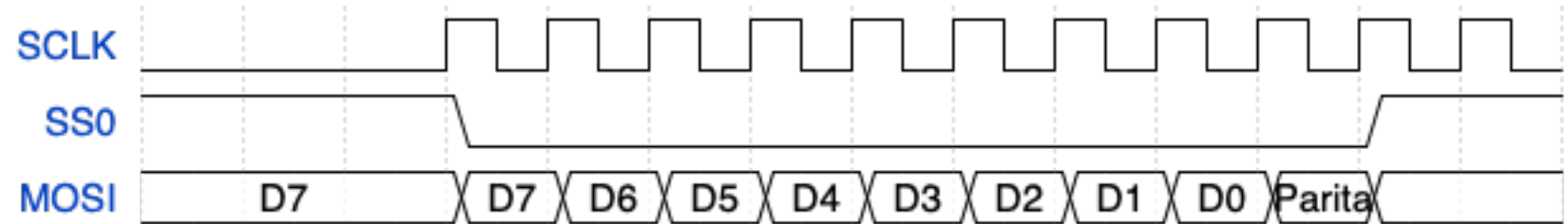
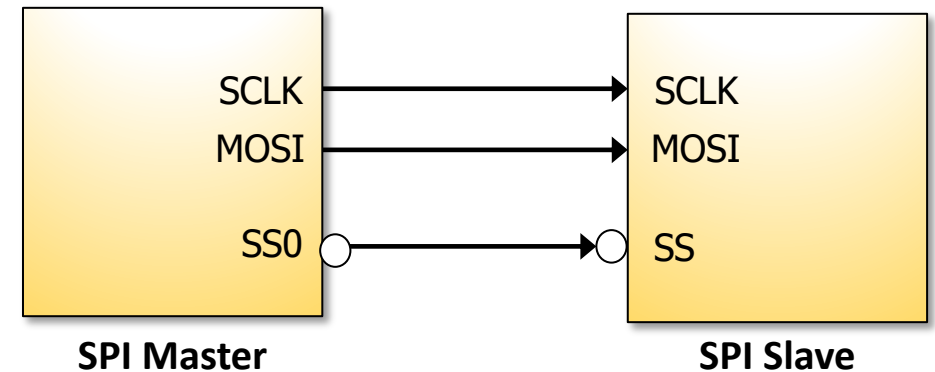


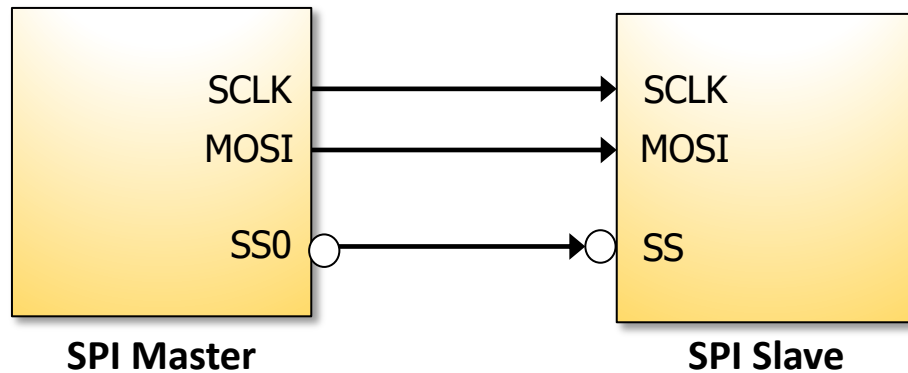
- Časový multiplex využití sběrnice – pro přenos každého bitu je vyhrazen konkrétní časový úsek na sériové sběrnici.
- Postupně se přikládají jednotlivé bity paralelního datového slova
 - Indikace začátku komunikace (start bit nebo dedikovaný signál indikuje přenos dat)
 - Přenos dat společně se zabezpečením přenosového kanálu (např. parita)
 - Indikace konce komunikace (stop bit nebo dedikovaný signál indikuje konec přenosu)

- **Zadání:** Navrhněte obvod pro vysílání dat po sériovém rozhraní SPI. Pro jednoduchost uvažujte pouze jednosměrný přenos s vysíláním 8 bitů dat se zabezpečením pomocí parity (9. bit).

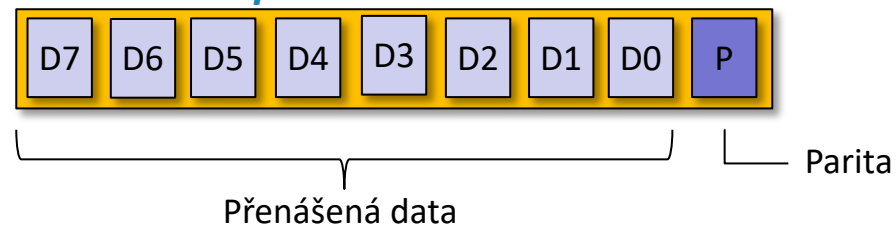
• SPI rozhraní

- **SCLK** - hodinový signál
- **MOSI** (Master Out Slave In) sériový datový přenos od zařízení Master k zařízení Slave.
- **MISO** (Master In Slave Out) sériový datový přenos od zařízení Slave k zařízení Master.
- **SS0** (Select Slave) vybírá zařízení, se kterým bude chtít komunikovat. Master může komunikovat s více zařízeními SSx





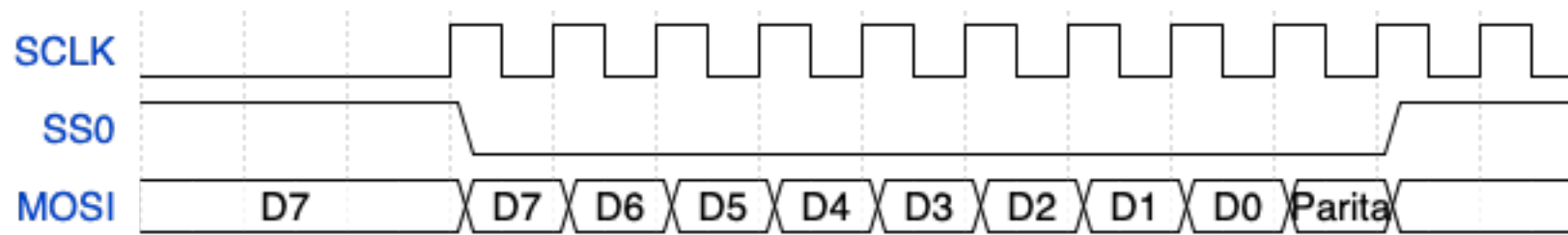
Přenášená zpráva



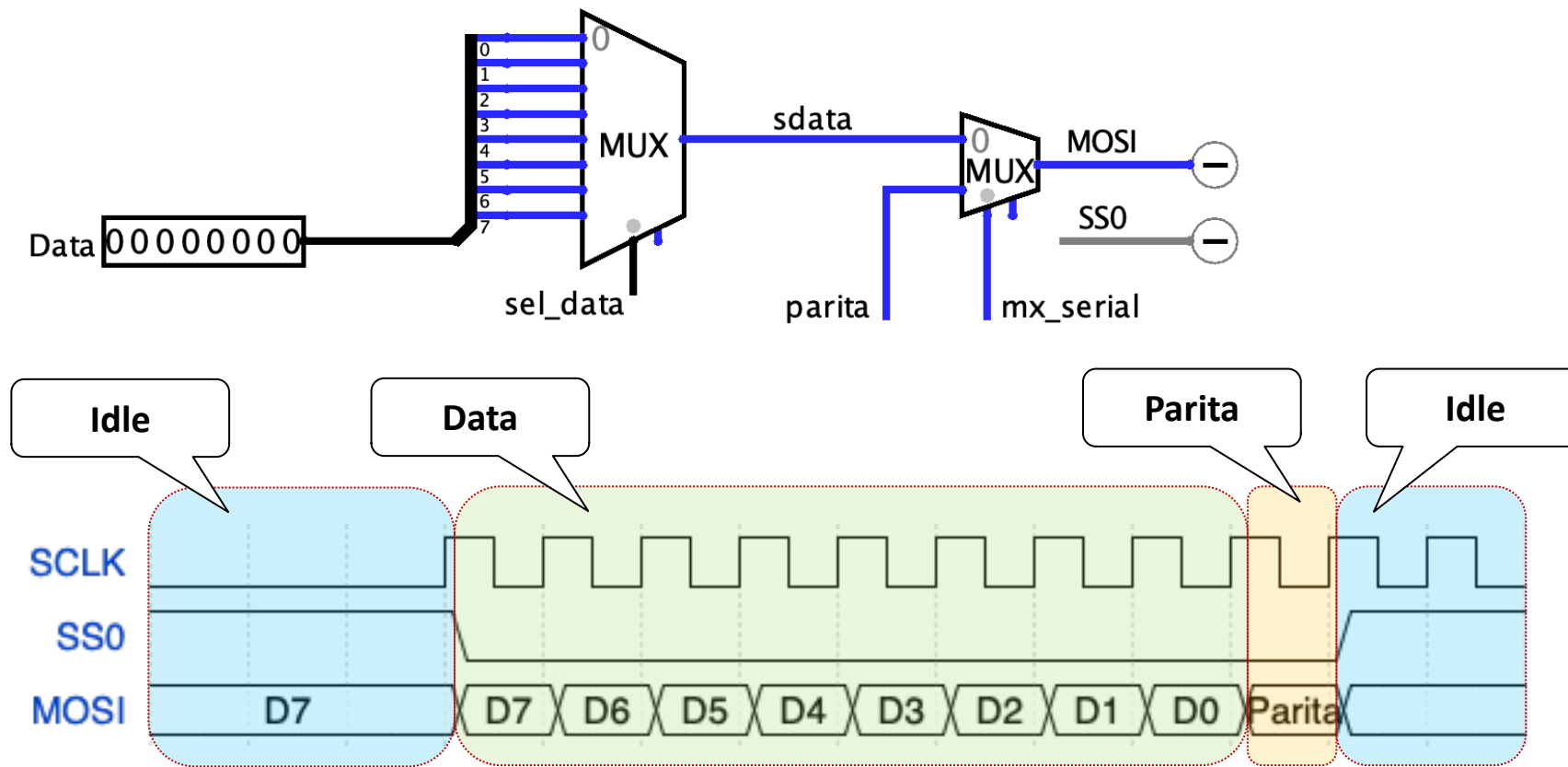
• Struktura přenášené zprávy

- **Zahájení přenosu** je indikováno signálem $SS0=0$
- **Nejprve je přeneseno 8 bitů dat:**
 $MOSI = Data[i]$, pro $i=7,6,5 \dots 0$
- **Pak je přenesen paritní bit:** Pouze zabezpečení komunikace $MOSI = Parita(D7 \text{ až } D0)$
- **Ukončení přenosu** je indikováno signálem $SS0=1$

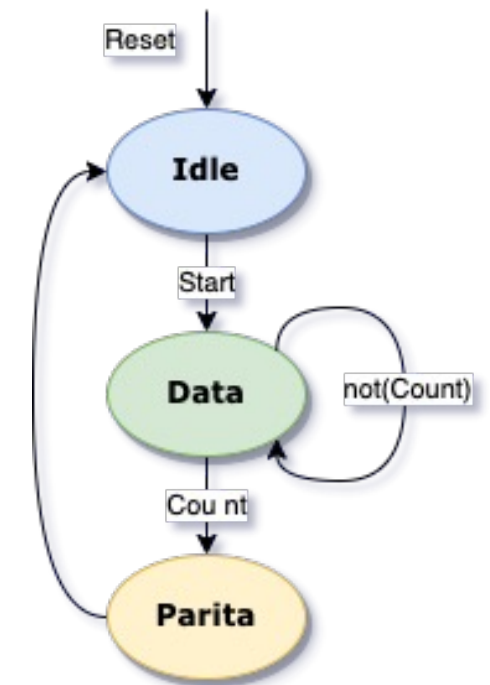
Časový průběh vysílání dat



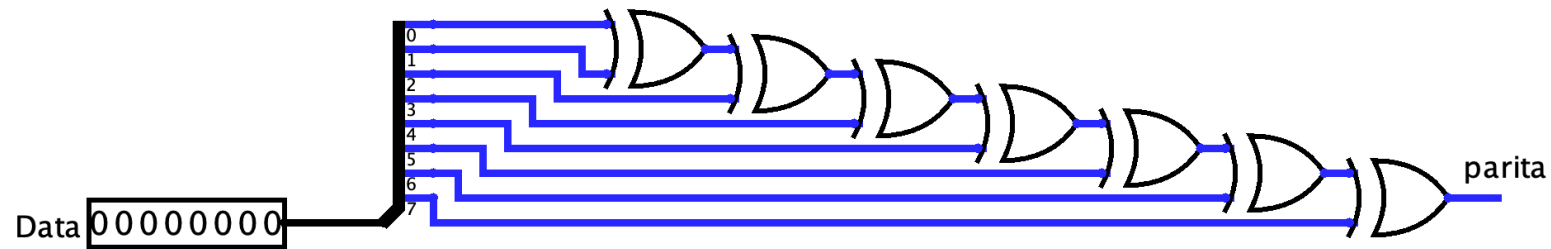
- Schéma zapojení tvoří multiplexory přepínající jednotlivé bity dat a paritu na sériový kanál MOSI sběrnice SPI
- Základní stavy automatu popisují fáze vysílání dat na sběrnici



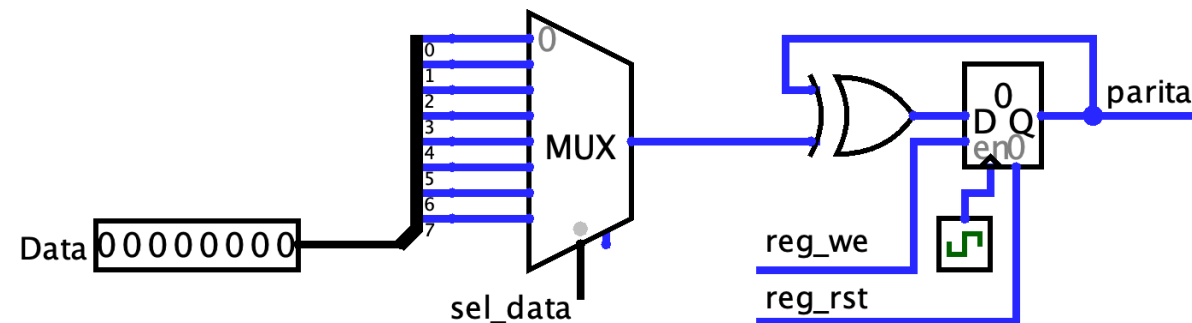
Řídící automat



- Výpočet parity z datového slova:



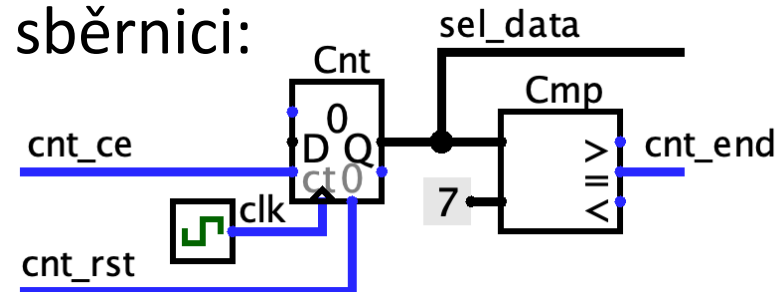
Generování parity nad vstupním datovým slovem
(Celkem 7 hradel XOR)



Sekvenční generování parity za multiplexorem
(Celkem 1 hradlo XOR a 1 registr)

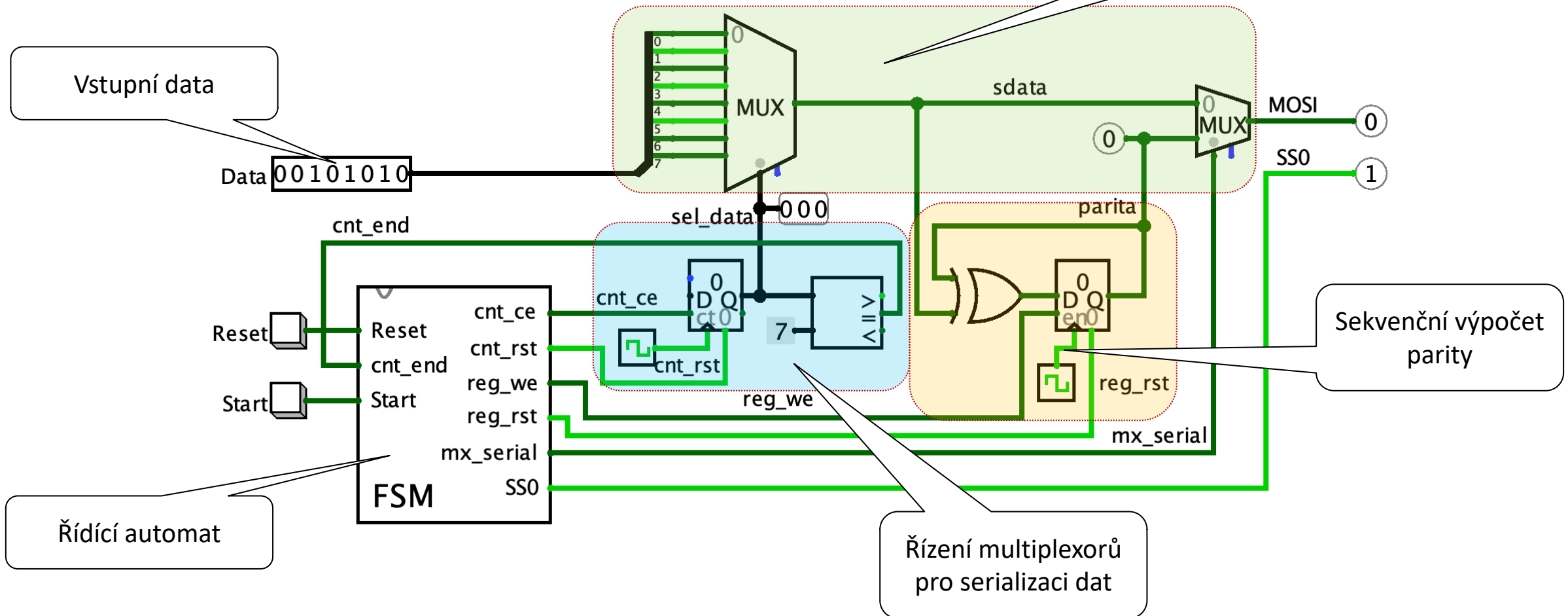
- Řízení multiplexoru pro serializaci dat na sběrnici:

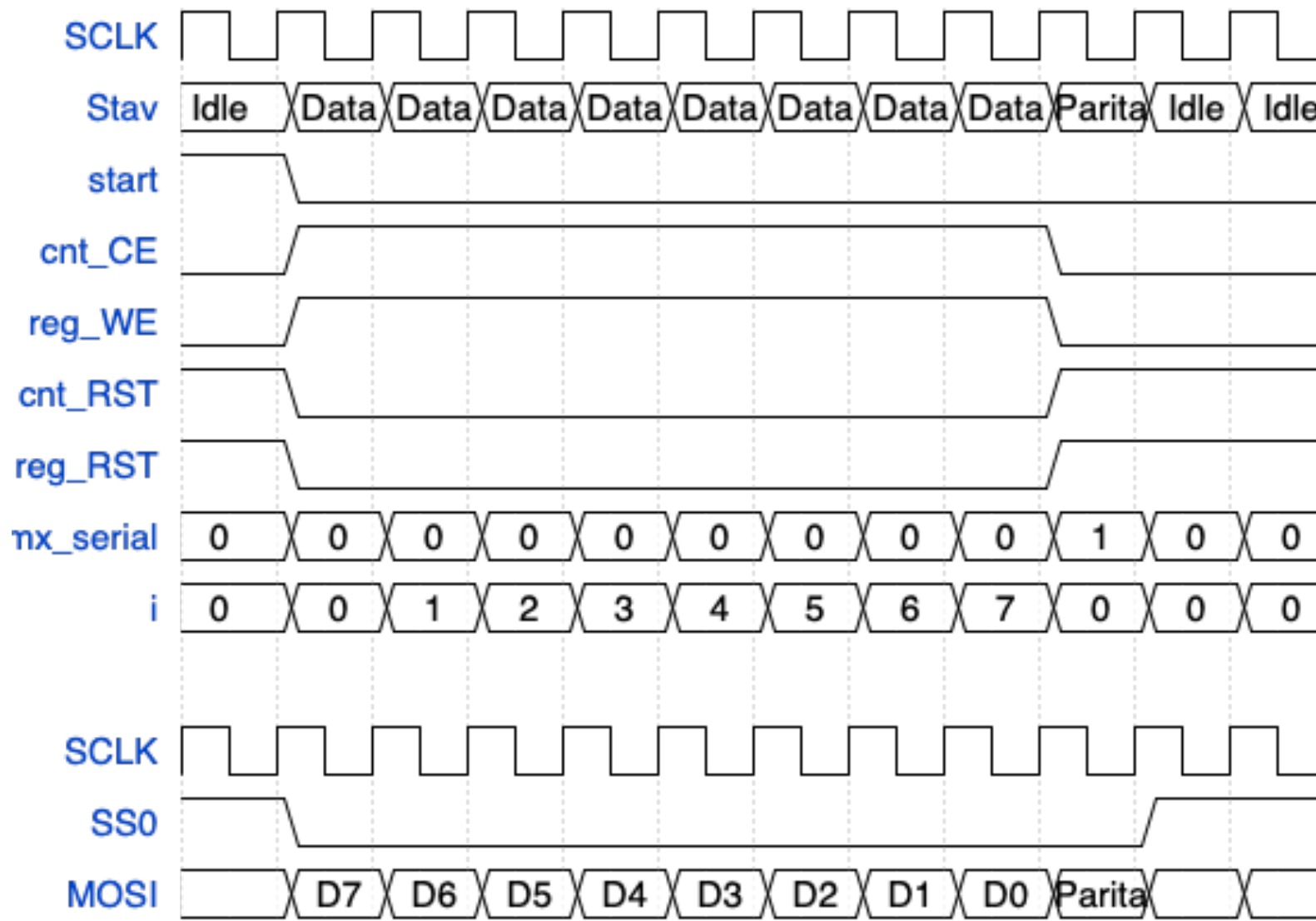
- Čítač pro indexaci přenášeného bitu
- Komparátor pro detekci posledního přenášeného bitu na sběrnici (poslední index dat)



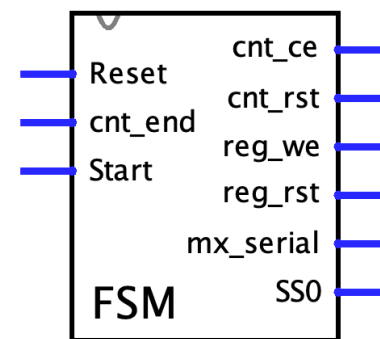
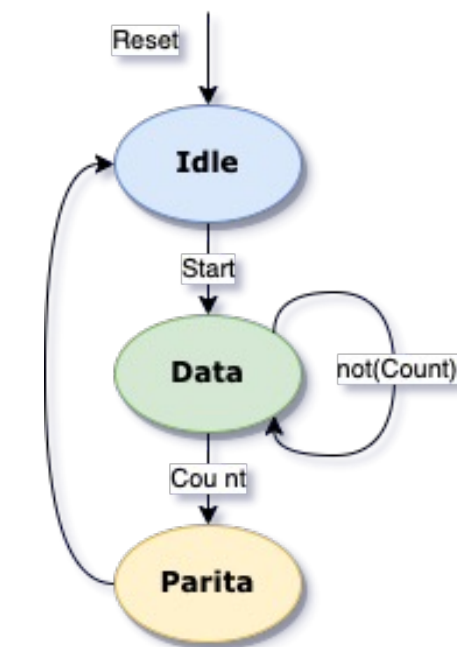
09b-SPI.circ

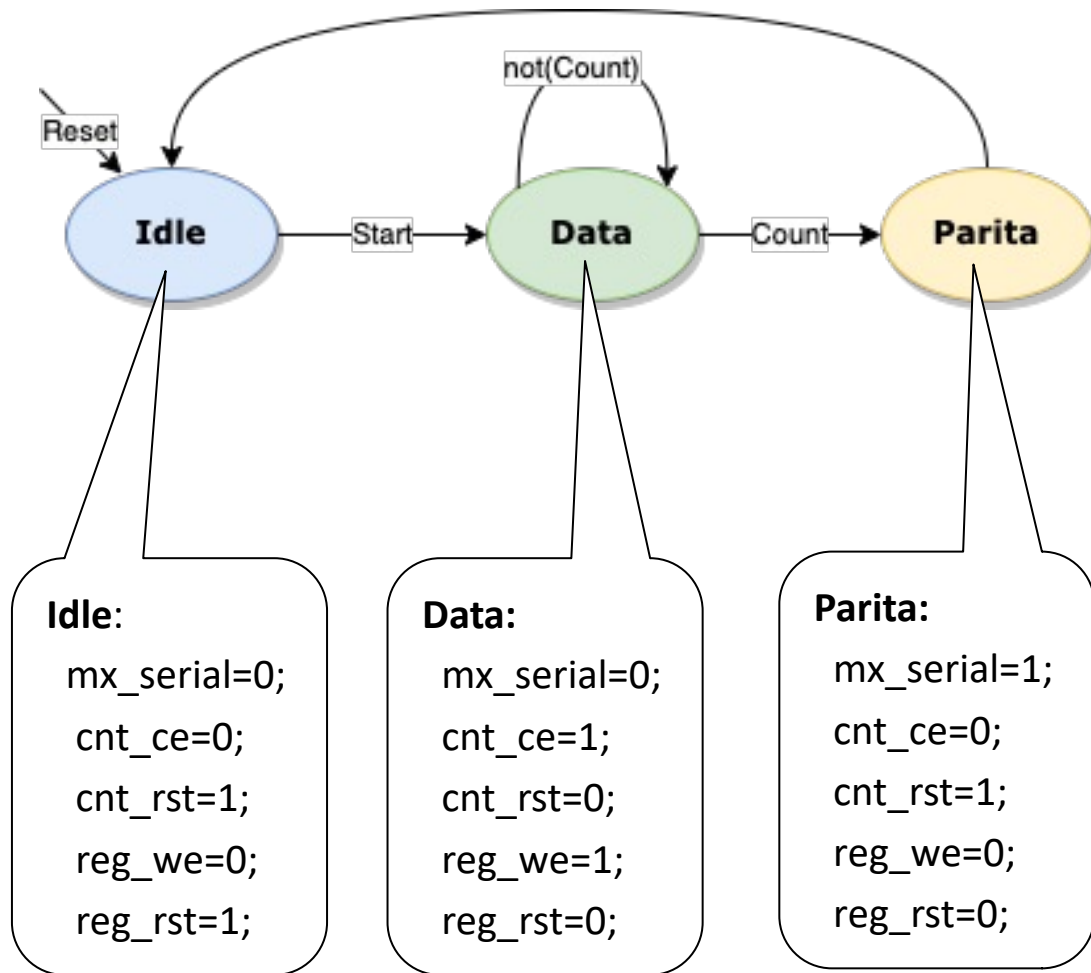
- Výsledné schéma zapojení pro řízení vysílání dat po sběrnici SPI



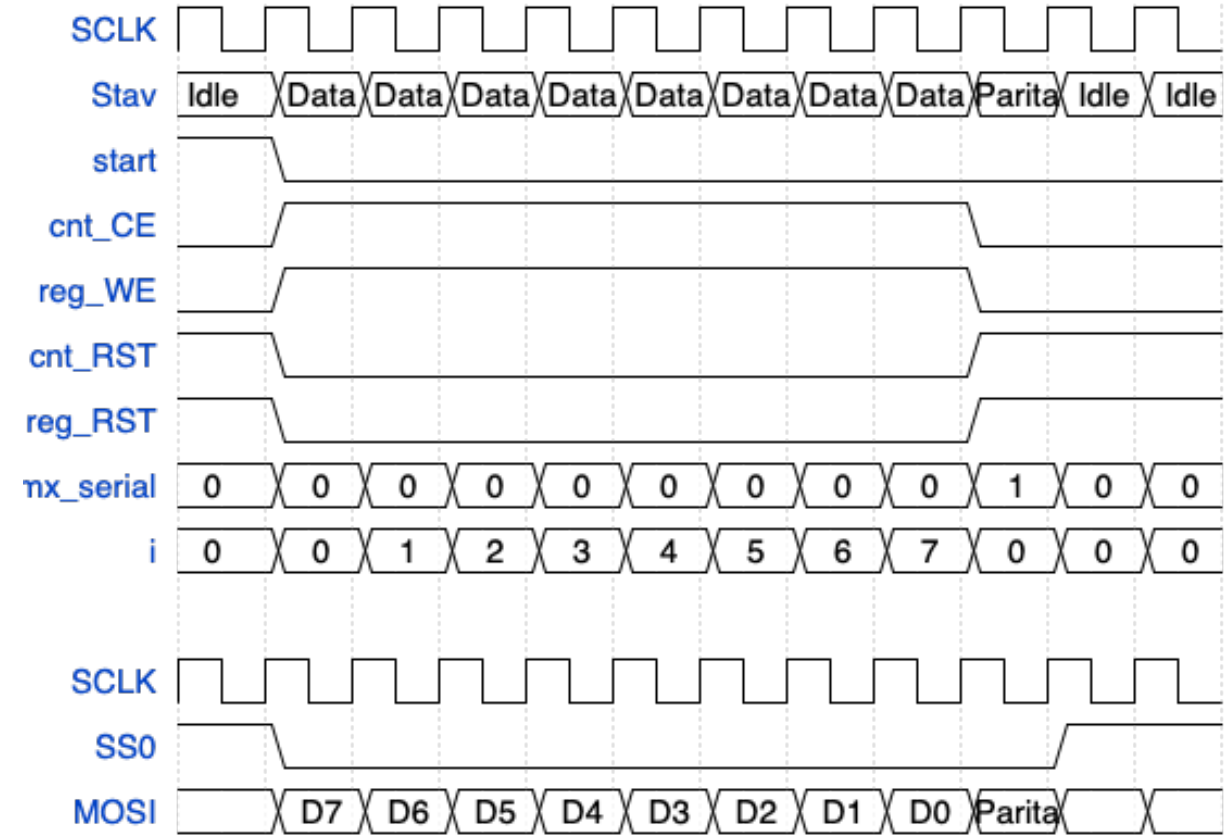


Návrh řídicího automatu

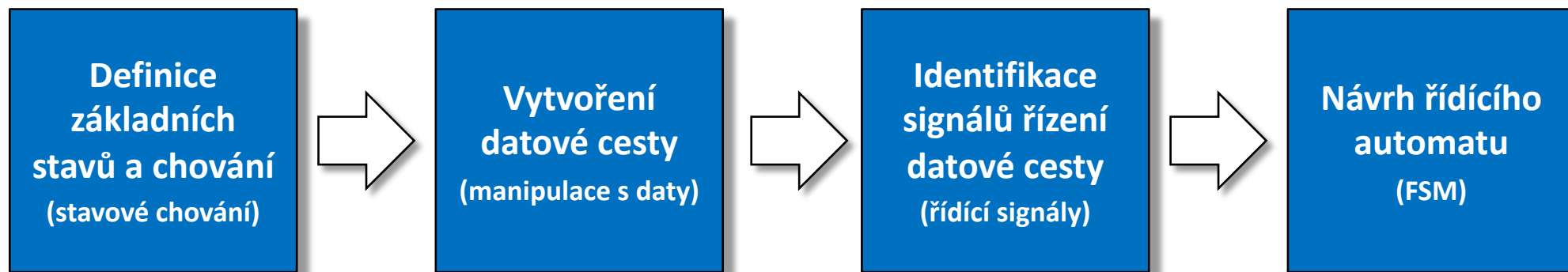




Časový diagram vysílání dat



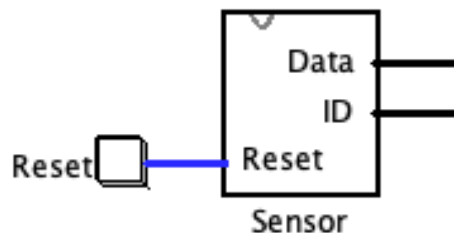
- Návrh je vhodné rozdělit do několika kroků
 - **Popis obvodu na úrovni základního stavového automatu** - automat popisuje základní stavy obvodu, ale neřeší konkrétní nastavení řídicích, vstupních nebo výstupních signálů.
 - **Vytvoření datové cesty (datapath)** - datová cesta provádí manipulaci s daty, a to na základě řízení z obecného automatu.
 - **Identifikace signálů pro řízení datové cesty** - definice signálů, které jsou potřeba pro napojení datové cesty na řídicí blok.
 - **Návrh řídicího automatu** - převedení obecného automatu na Mealyho nebo Moorův automat, který řídí datovou cestu.



- Registry umožňují na čipu efektivně ukládat data a poskytnout je k zobrazení nebo dalšímu výpočtu
- **Příklad:** Po sériové lince přicházejí periodicky data z celkem 4 senzorů, které jsou identifikovány pomocí dvou-bitového ID. Hodnota senzoru má 8 bitů. Udělejte zobrazovací zařízení, které bude ukazovat aktuální hodnoty všech 4 senzorů.

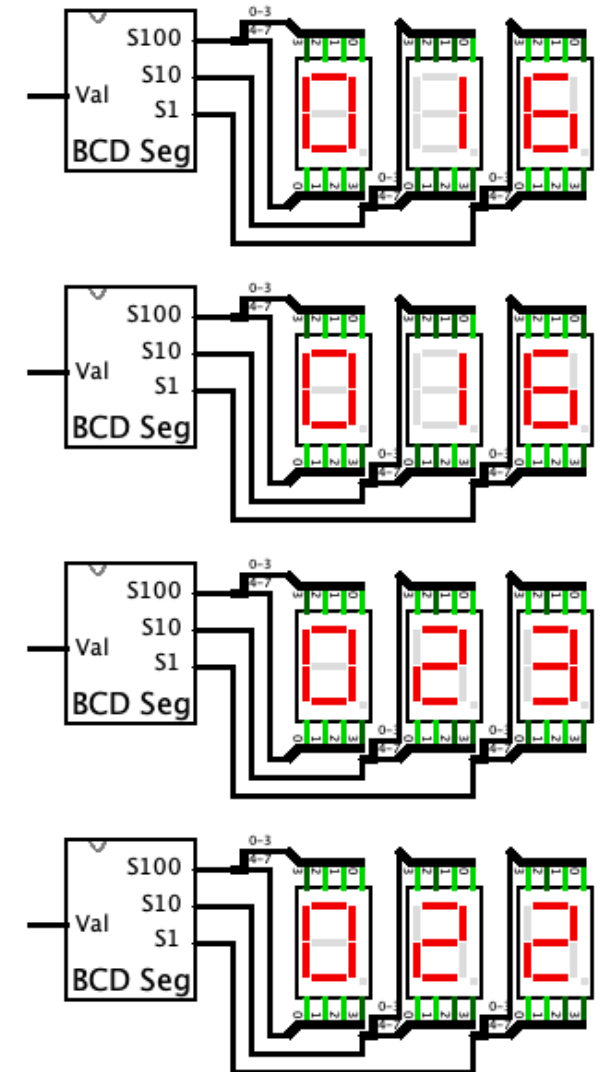
Periodické čtení dat ze senzorů

Většinou řešeno přes SPI nebo I2C sběrnici

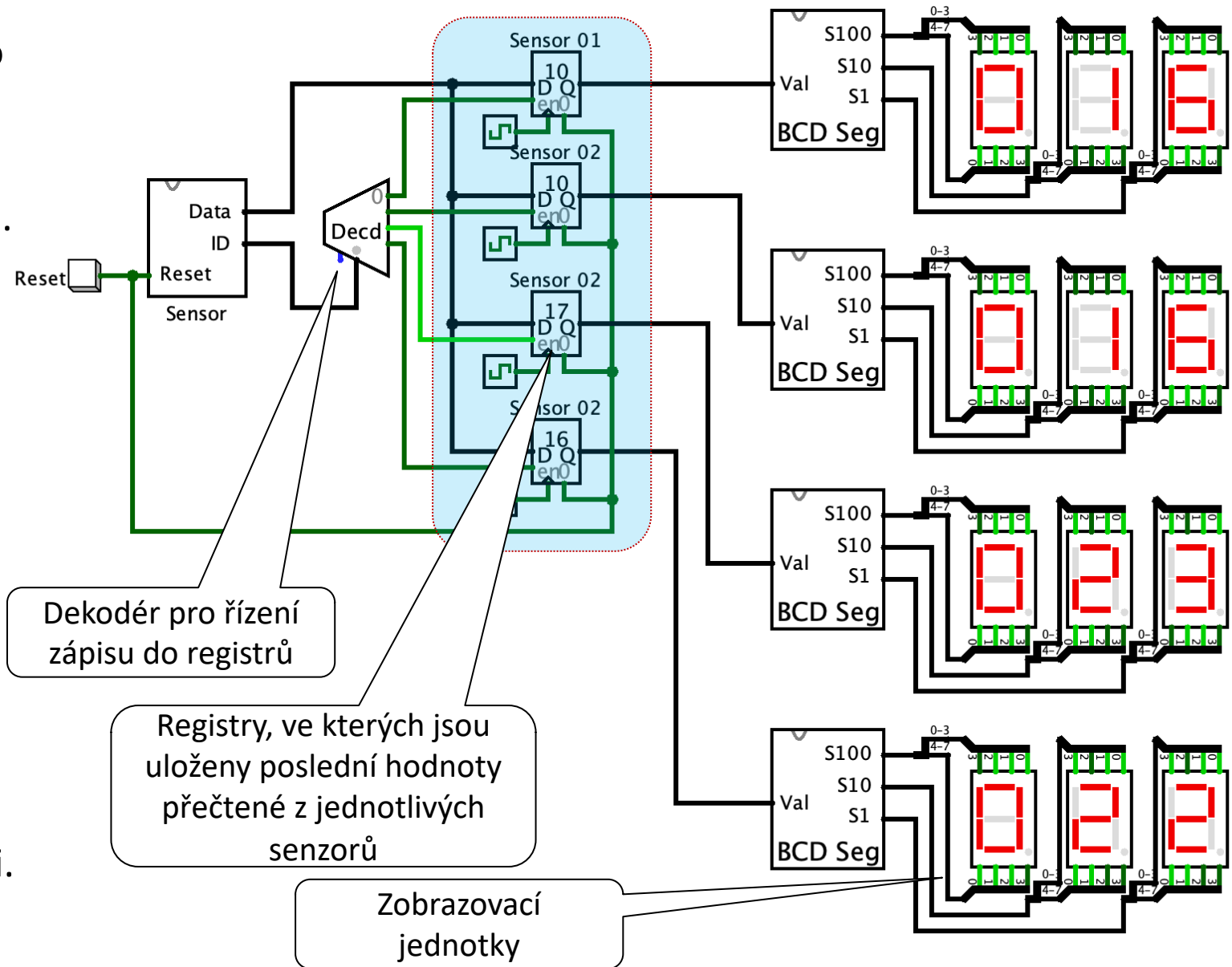


Přechodné
ukládání dat ze
senzorů

Zobrazení naměřených dat



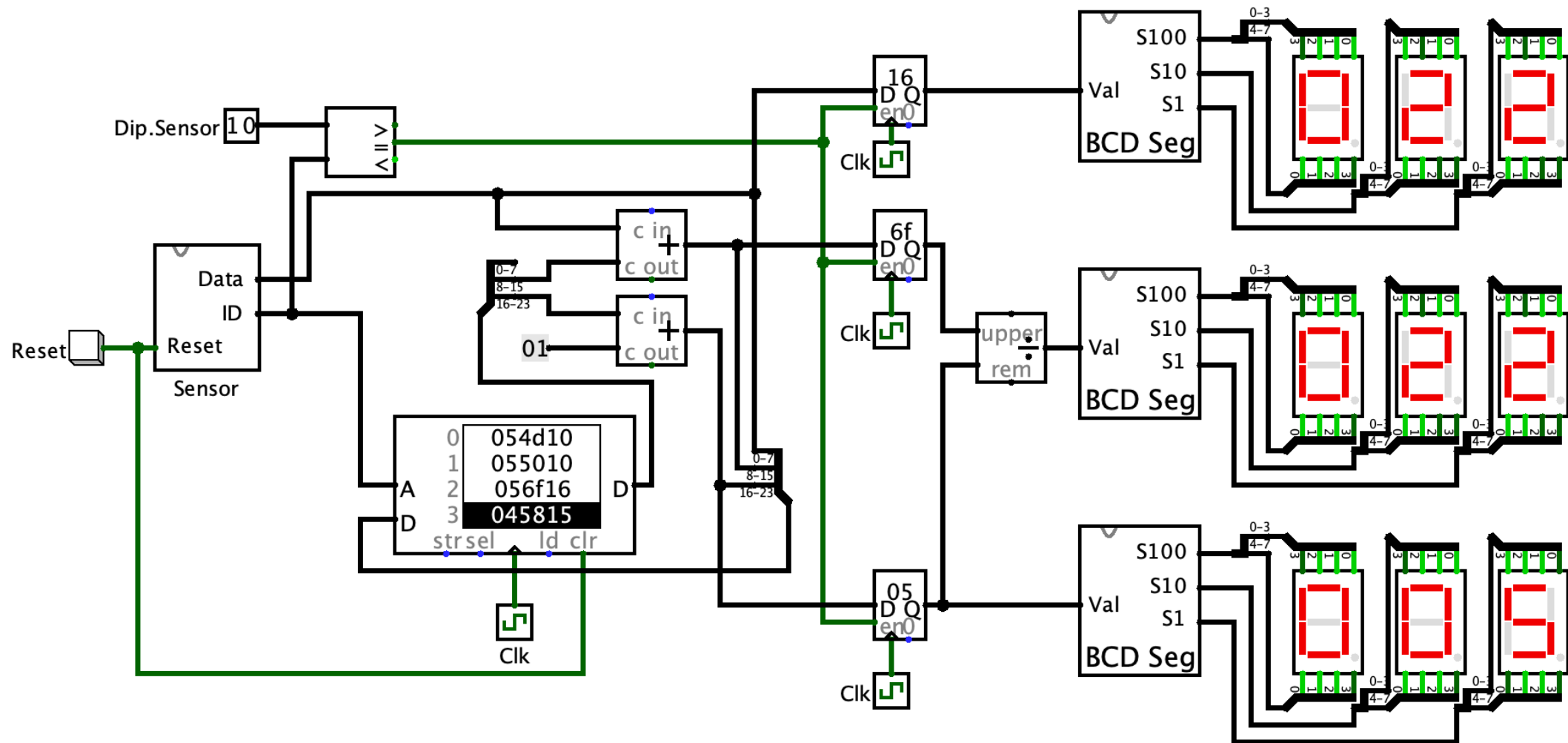
- Tlačítkem Reset je inicializováno zařízení do počátečního stavu
- Ze senzoru označeném pomocí ID přichází data na sběrnici Data.
- Dekodér identifikuje konkrétní senzor a řídí zápis do registru signálem WE (write enable). Podle dekódovaného ID je nastaven vždy jede signál WE.
- Obsah všech registrů je vyveden na zobrazovací jednotky (7-segmentové displeje).
- Jak se průběžně aktualizují hodnoty v registrech, jsou průběžně aktualizované i zobrazované hodnoty na displeji.



- Organizace paměti



- Výsledná architektura se zobrazením statistik zvoleného senzoru ID=„10“



- Dokážete vytvořit a analyzovat obvod v programu Logisim?
- Jak postupují při návrhu obvodu?
- Jak se doplňuje datová a řídicí cesta obvodu?
- Čím se vyznačuje synchronní obvod?
- Jak analyzujeme chování synchronního obvodu?
- Zapisuje se do řídicího automatu hodinový signál? Proč?
- Jak se definují moduly a jak se dají použít?
- Jak se postupuje při návrhu systému zhora dolů a zdola nahoru?
- Jaké máme k dispozici prvky pro uchování dat a jak rychle s nimi dokáží pracovat?
- Která komponenta umožňuje serializaci dat?

Děkuji za pozornost!