

# **Klopné obvody, čítače, stavové automaty, technologie pro realizaci log. obvodů a hradel**

Klopné obvody, čítače, registry, stavové automaty, polovodičové  
technologie pro realizaci log. obvodů, TTL, CMOS

Ing. Pavel Lafata, Ph.D.  
lafatpav@fel.cvut.cz

## Klopné obvody, sekvenční obvody – RS, JK, D a T klopné obvody

### ▪ Klopné obvody – shrnutí

- **synchronní vs. asynchronní**
- synchronní – **hladinové (latch)** vs. **hranové (flip-flop)**
- **master-slave** konfigurace – každý **hladinový** klopný obvod můžeme takto upravit na **hranový** klopný obvod – dvojice klopných obvodů, master část připojena přímo na hodinový vstup Clock, slave část přes invertor na negaci Clock
- ke každému klopnému obvodu můžeme přidat **asynchronní vstupy S a R**:
  - asynchronní Set obvykle nazýván **nastavení (preset)**
  - asynchronní Reset obvykle nazýván **nulování (clear)**
  - asynchronní vstupy jsou vždy **aktivní v logické 0**
  - **asynchronní vstupy jsou aktivovány okamžitě** (bez čekání na Clock) a jsou **dominantní** – převáží funkci synchronních vstupů
  - protože pro RS klopný obvod je  $R = S = 1$  zakázaný stav, nesmí oba asynchronní vstupy žádného klopného obvodu **nabývat současně logické 0**
- **klopné obvody v jazyce VHDL** -> lze vytvořit synchronní i asynchronní klopné obvody, hladinové i hranové, obvykle používáme behaviorální popis (více později)

## Sekvenční logické obvody (SLO) – sekvenční logické obvody

### ▪ Sekvenční logické obvody

- paměťová část SLO tvořena některým klopným obvodem (obvykle D, JK), příklady SLO: **čítače, registry, konečné stavové automaty**, atd.

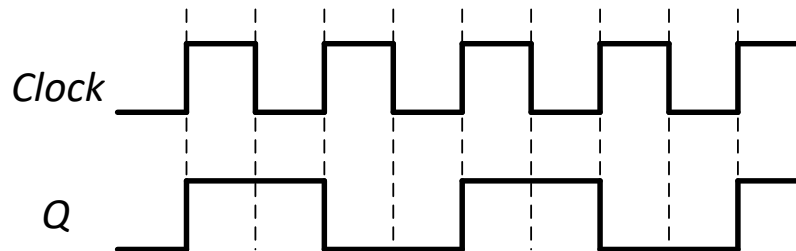
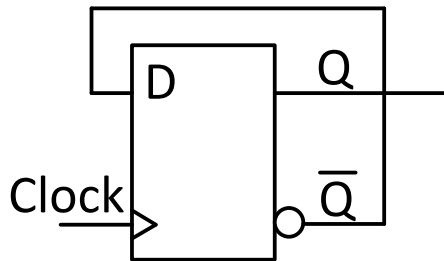
### • Čítače

- obvody **čítající požadované výskyty** v průběhu dané sekvence – obvykle **vzestupné či sestupné hrany** vstupního signálu -> výstupem je **kódové vyjádření načítaného počtu**
- výstup obvykle v **binárním kódu**, ale např. i Grayův kód apod.
- **čítač modulo N** = čítající v intervalu hodnot **0 až N-1**
- **každý 1 bitový řád čítače = 1 klopný obvod** -> čítač modulo N =  $\log_2 N$  klopných obvodů (kde N je nejbližší vyšší mocnina 2)
- **synchronní čítač vs. asynchronní čítač**
- dle způsobu distribuce buzení (hodinového taktu)
- **vzestupný čítač vs. sestupný čítač vs. obousměrný čítač**
- dle směru čítání (přičítání, odečítání, volitelný směr)
- dodatečné vstupy – blokování (enable), nulování, nastavení
- tyto vstupy mohou být synchronní nebo asynchronní
- konstrukce čítačů – klopné obvody D nebo JK (různé výhody, nevýhody)

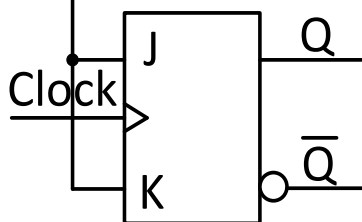
## Sekvenční logické obvody (SLO) – čítače, sekvenční logické obvody

### ▪ Asynchronní čítač

- vnější hodinový signál Clock je připojen pouze na vstup prvního klopného obvodu, hodinové vstupy všech dalších jsou připojeny na výstup předchozího klopného obvodu
- buzení se tak šíří postupně -> nejprve překlopí první KO, po jeho překlopení druhý, po jeho překlopení třetí... nevýhoda – tzv. **ripple-effect**
- $T$  – doba zpoždění (překlopení) jednoho KO ->  $n$ -bitový čítač =  **$n \times T$  celkové zpoždění**
- pokud je perioda vstupního signálu **Clock**  $< n \times T$  -> výstup čítače se nestihne ustálit před příchodem další hrany Clock -> **nestabilní stav čítače**, nelze jeho hodnotu odečíst
- **výhoda** – pro konstrukci čítače si vystačíme jen s klopnými obvody (D, JK), **nejsou potřeba žádná jiná hradla** -> **jednoduchá konstrukce**, menší plocha, menší spotřeba
- základem je D klopný obvod se zpětnou vazbou, nebo JK se spojenými vstupy – **dělič 2**



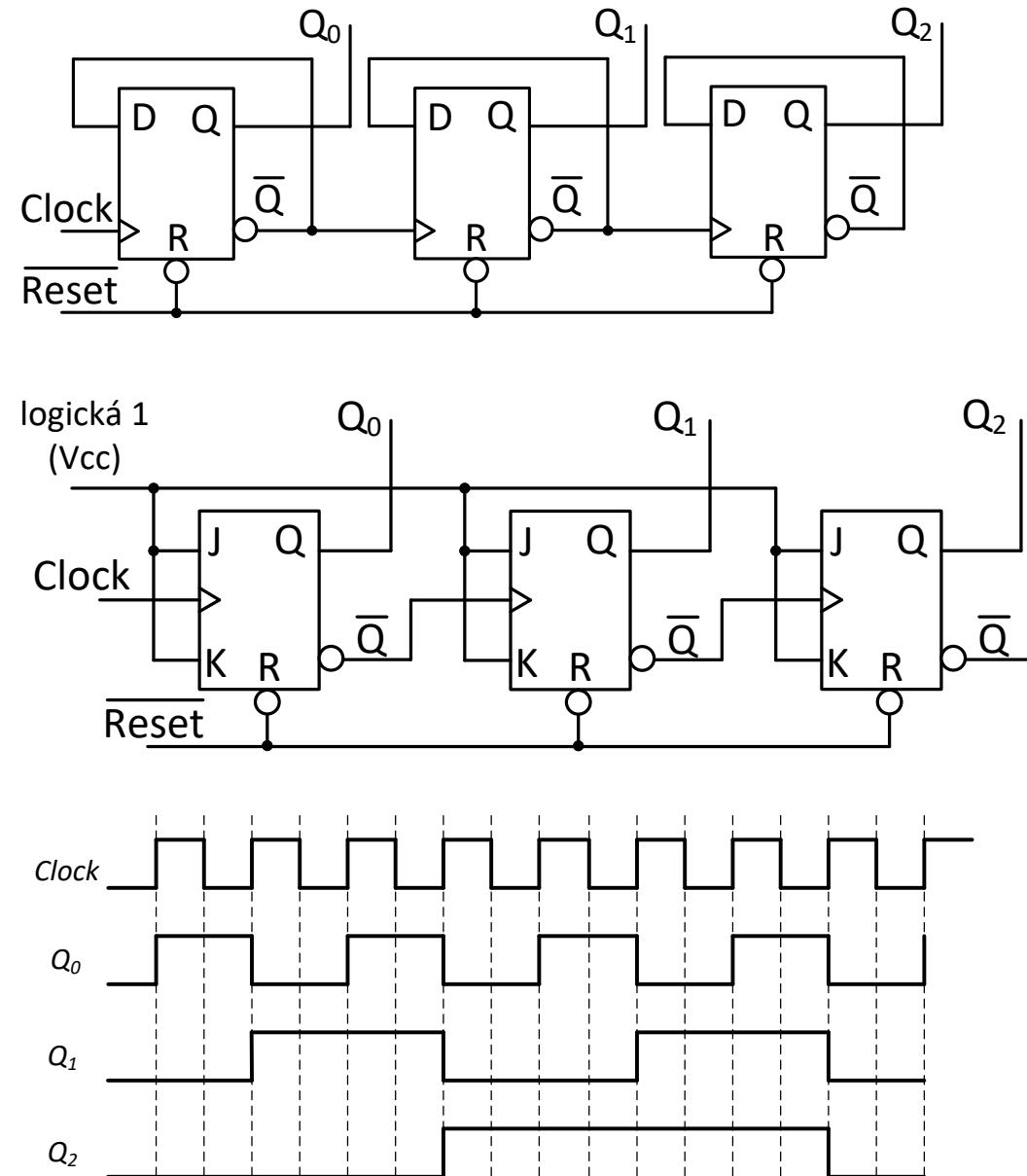
logická 1  
(Vcc)



- výstup Q představuje signál s poloviční frekvencí (dvojnásobnou periodou) v porovnání se vstupem Clock
- **kaskádním řazením  $n$ -stupňů (D či JK) obdržíme  $n$ -bitový asynchronní čítač**

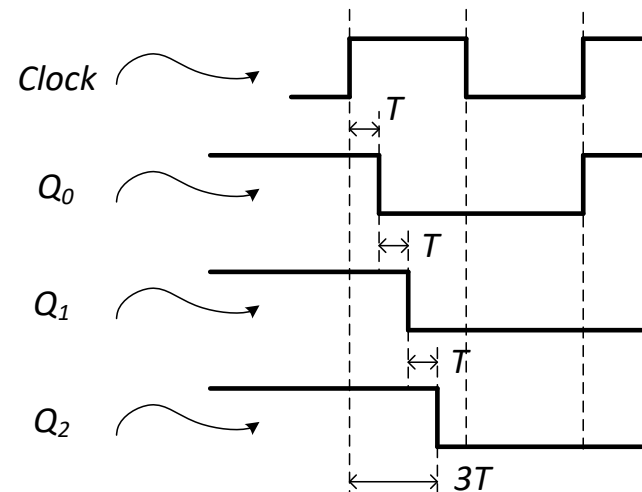
## Sekvenční logické obvody (SLO) – čítače, sekvenční logické obvody

### ▪ Asynchronní čítač – příklad 3bitového (modulo 8), realizace z D i JK obvodů



- každý stupeň představuje děličku 2 předchozího výstupu
- **ripple-effect** – kaskádní narůstání zpoždění při průchodu jednotlivými stupni =  $n \times T$  pro n-bitový čítač
- pokud  $T_{Clock} < n \times T$  – čítač nepoužitelný
- řešení – zvýšit periodu  $T_{Clock}$  (nelze vždy), snížit počet bitů čítače  $n$  (nelze vždy), použít synchronní čítač

#### detail přechodu 111 -> 000



- **Synchronní čítač – příklad 3bitového (modulo 8), realizace z JK obvodů**

## Sekvenční logické obvody (SLO) – čítače, sekvenční logické obvody

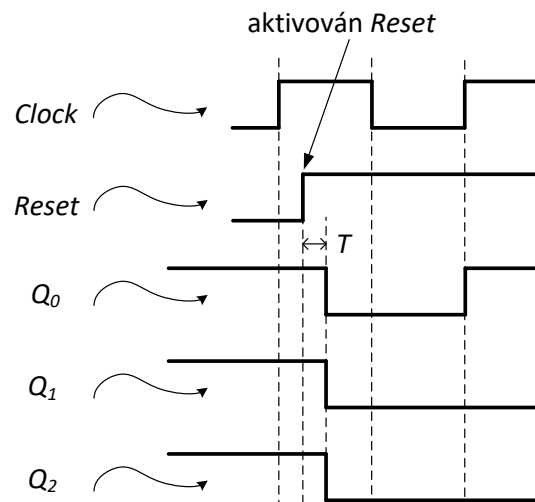
### ▪ Čítače – synchronní vs. asynchronní Reset (nulování), blokování (Enable)

- využití asynchronních vstupů Set a Reset klopných obvodů
- můžeme je dodatečně připojit (synchronizovat) s hodinovým vstupem Clock

### • čítač s asynchronním resetováním vs. čítač se synchronním resetováním

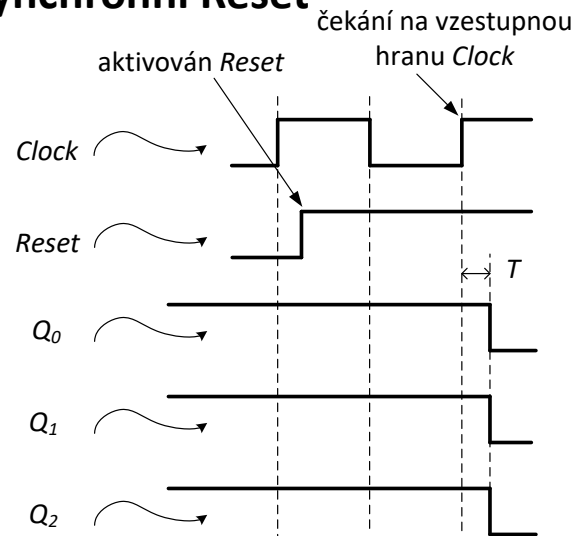
- asynchronní vstup Reset nezávislý na stavu hodinového vstupu Clock -> proveden okamžitě
- synchronní vstup Reset závislý na Clock -> proveden až při detekci vzestupné (sestupné) hrany hodinového vstupu

#### asynchronní Reset



- realizace v jazyce VHDL? snadná – ukážeme si později
- obdobně jako Reset – blokovací vstup (Enable)

#### synchronní Reset



### • asynchronní Reset:

- výhoda – okamžité provedení (zpoždění  $T$ )
- nevýhoda – výstup po Resetu je asynchronní (pokud je následující připojený obvod synchronní – problém)

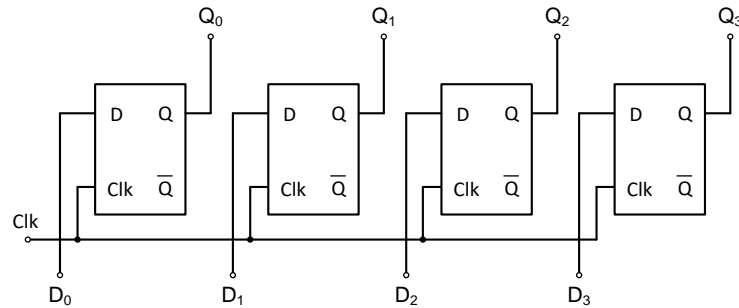
### • synchronní Reset:

- výhoda – výstup čítače zůstává plně synchronní
- nevýhoda – zpoždění dané čekáním na hranu Clock
- co se stane, pokud před příchodem hrany Clock Reset vrátíme do logické 0?

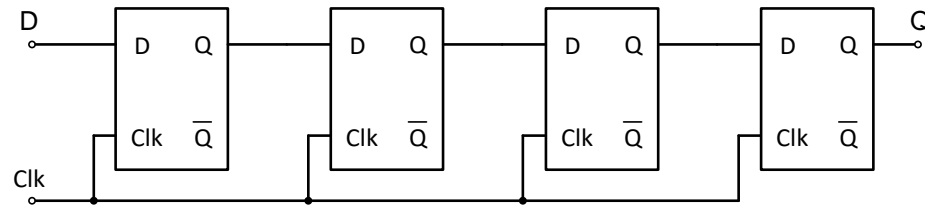
## Sekvenční logické obvody (SLO) – registry, sekvenční logické obvody

- **Registry – statické (paměti) a posuvné**

- 4 základní varianty dle způsobu čtení a zápisu – **PIPO**, **SISO**, **SIPO**, **PISO**
- základem jsou klopné obvody D nebo JK
- **parallel in-parallel out, PIPO** – paměť s paralelním vstupem a paralelním výstupem



- **serial in-serial out, SISO** – sériová paměť (zpožďovací článek – výstup  $n$  x zpožděn)



- **serial in-parallel out, SIPO** – sériová paměť s paralelním čtením při zaplnění
- **parallel in-serial out, PISO** – serializér, převod paralelních toků do sériového výstupu
- registry můžeme doplnit např. o možnost nulování (mazání), posuvu o  $n$ -bitových pozic vlevo či vpravo, provádění jednoduchých operací (XOR) apod.



## Sekvenční logické obvody (SLO) – konečné stavové automaty

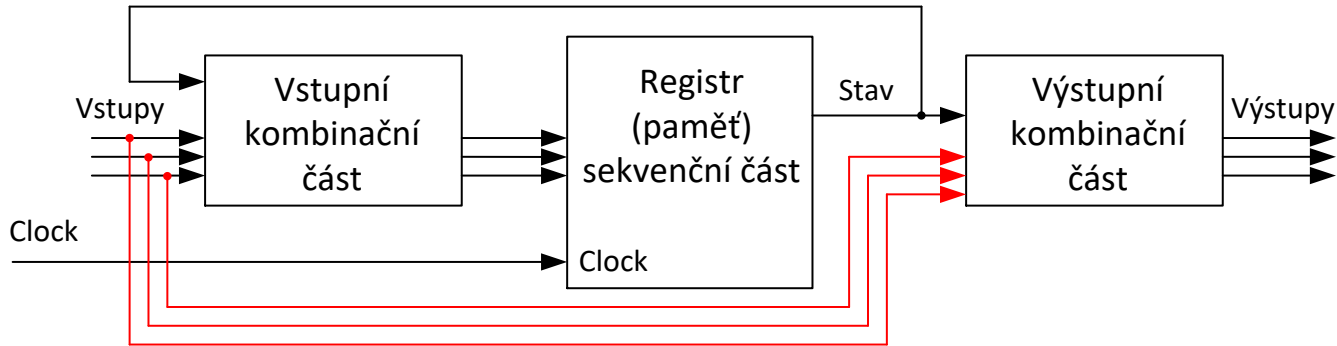
### • Konečné stavové automaty (FSM – Finite state machines)

- obecně, konečný stavový automat = jakýkoliv sekvenční logický obvod, který může nabývat konečného počtu vnitřních stavů, vstupů, výstupů (např. čítač)
  - automat smí být vždy právě jen v jednom stavu v čase  **$t$**  – **současný stav**
  - na základě budících podmínek přechází automat z jednoho stavu do druhého – nazýváme **přechod** – automat se tak v čase  **$t+1$**  dostane do **následného stavu**
- 
- každý stavový automat tak můžeme popsat (uvažujeme dál jen synchronní obvody):
    1. konečnou množinou vstupních stavů  **$\mathbf{X}$** :  $\mathbf{X} = \{x_1, x_2, \dots, x_M\}$
    2. konečnou množinou výstupních stavů  **$\mathbf{Y}$** :  $\mathbf{Y} = \{y_1, y_2, \dots, y_N\}$
    3. konečnou množinou vnitřních stavů  **$\mathbf{Q}$** :  $\mathbf{Q} = \{q_1, q_2, \dots, q_P\}$
    4. konečnou množinou přechodových funkcí  **$\mathbf{G}$** :
$$\mathbf{Q}^{t+1} = \mathbf{G}(\mathbf{X}^t; \mathbf{Q}^t) = g(x_1^t, x_2^t, \dots, x_m^t; q_1^t, q_2^t, \dots, q_p^t)$$
    5. konečnou množinou výstupních funkcí  **$\mathbf{F}$** :
      - a. **obvod typu Mealy**  $\mathbf{Y}^{t+1} = \mathbf{F}(\mathbf{X}^t; \mathbf{Q}^t) = f(x_1^t, x_2^t, \dots, x_m^t; q_1^t, q_2^t, \dots, q_p^t)$
      - b. **obvod typu Moore**  $\mathbf{Y}^{t+1} = \mathbf{F}(\mathbf{Q}^t) = f(q_1^t, q_2^t, \dots, q_p^t)$

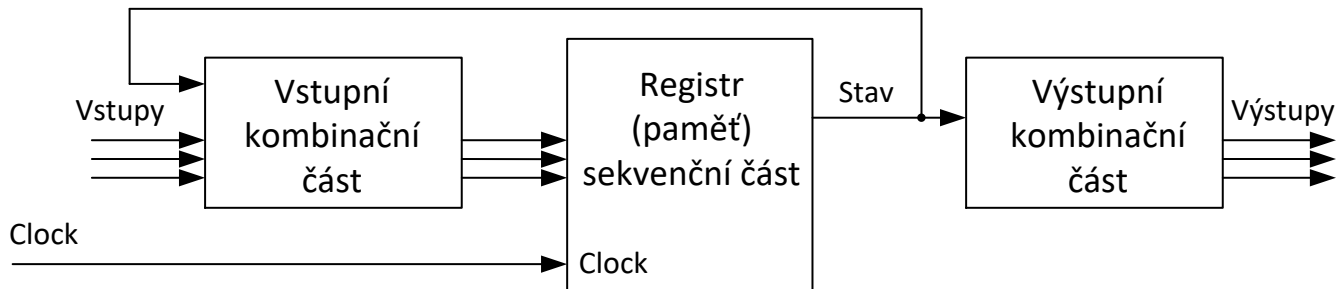
## Sekvenční logické obvody (SLO) – konečné stavové automaty

- **Obecné schéma automatu – typ Mealy a typ Moore**

- **typ Mealy**



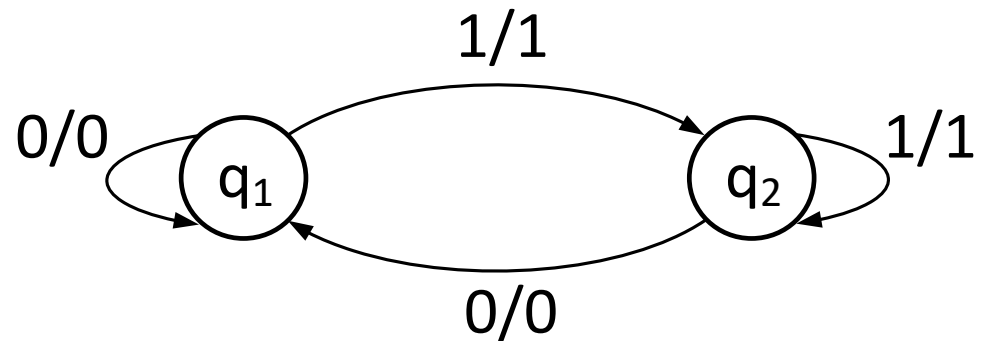
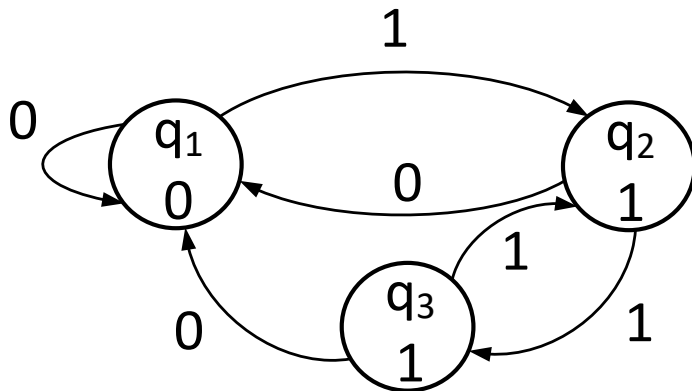
- **typ Moore**



- výstup obvodu typu Moore je závislý pouze na současném vnitřním stavu, zatímco typu Mealy je závislý kromě stavu i na současných vstupech obvodu
- uvedená bloková schémata jsou jen obecná a pro specifické obvody mohou být upravena

## Sekvenční logické obvody (SLO) – konečné stavové automaty

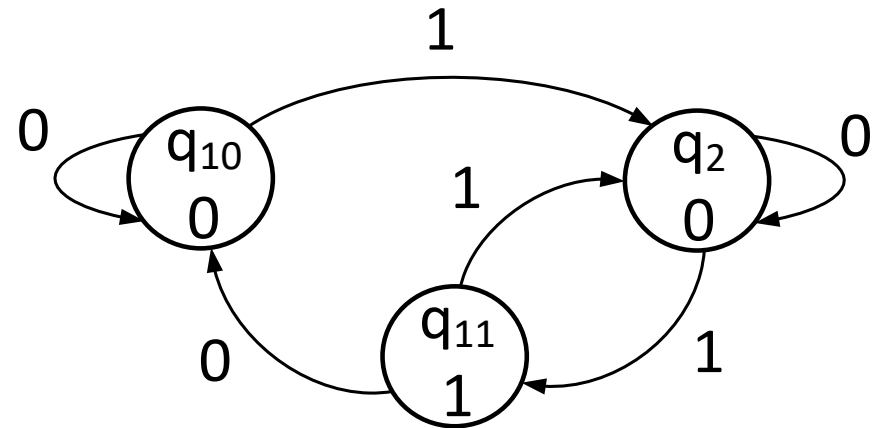
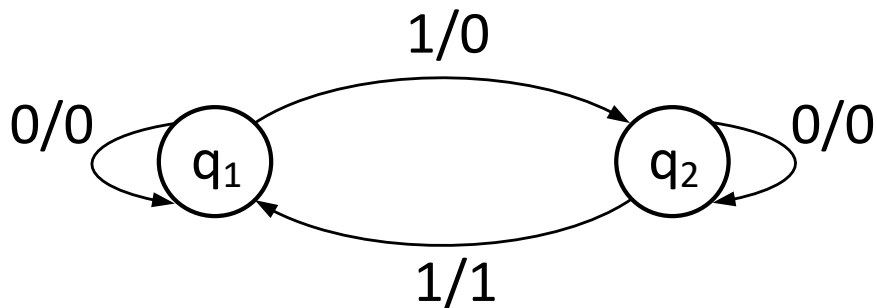
- **Konečné stavové automaty – převody typů Mealy a Moore**
- převod automatu typu **Moore** -> na typ **Mealy**
  - v prvním kroku **přepíšeme hodnoty výstupů z uzlů k jednotlivým hranám grafu**, které do těchto uzlů směřují při daném vstupu obvodu
  - v druhém kroku **můžeme sloučit ty hrany**, které vycházejí i končí ve stejných stavech a mají stejné výstupní hodnoty
  - ve třetím kroku **můžeme sloučit ty stavy**, pokud hrany, které z nich vycházejí, vstupují do stavů se stejnou výstupní hodnotou při stejné hodnotě vstupu
  - příklad – převedte zadaný automat Mooreova typu na typ Mealy



- můžeme sloučit stavy  $q_2$  a  $q_3$  z Mooreova typu do stavu  $q_2$  Mealyho typu, protože oba dávají stejnou výstupní hodnotu a přechody z obou do stavu  $q_1$  jsou shodné

## Sekvenční logické obvody (SLO) – konečné stavové automaty

- **Konečné stavové automaty – převody typů Mealy a Moore**
- převod automatu typu **Mealy** -> na typ **Moore**
  - pokud do jednoho stavu směřují dvě hrany s odlišnými výstupními hodnotami, musíme tento stav rozštěpit na dva samostatné – do každého nového stavu pak budou vcházet jen hrany se stejným výstupem – pak již jen přesuneme výstupy do uzlů grafu
  - příklad – převedte zadaný automat Mealyho typu na typ Moore



- musíme rozštěpit stav  $q_1$  v Mealyho automatu na stavy  $q_{10}$  – dává výstup logická 0 a  $q_{11}$  – dává výstup logická 1 při realizaci typem Moore
- porovnání automatů **Mealy** vs. **Moore**
- Mealy – obvykle potřeba méně stavů (méně klopných obvodů pro realizaci), ale výstup je nesynchronní (výstupní hodnota se objeví již při přechodu do následného stavu)
- Moore – obvykle potřeba více stavů (složitější realizace), výstup je ale vždy synchronní (výstupní hodnota se objeví až po přechodu a dosažení následného stavu)

# **Technologie pro realizaci logických hradel a obvodů**

Polovodičové prvky pro HW realizaci logických obvodů

## Technologie pro HW realizaci – základní pojmy a kritéria

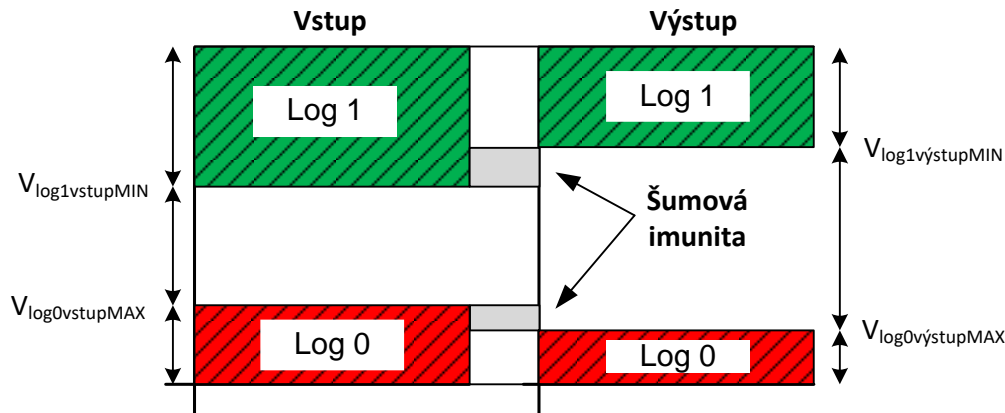
### ▪ Technologie pro realizaci logických hradel a obvodů

- zaměříme se jen na elektrické logické signály a polovodičové součástky
- pro porovnání jednotlivých technologií definujeme nejprve důležité pojmy:
  1. **vstupní větvení** – počet vstupů hradla
    - obvykle omezené danou technologií výroby a složitostí, např. na vstupu TTL hradla NAND je víceemitorový tranzistor, vyrábí se maximálně 8 vstupů
    - vyšší počet vstupů hradla -> obvykle větší parazitní kapacita na vstupu -> větší doba zpoždění hradla, v praxi obvykle hradla s 2, 3, 4 vstupy
  2. **logický zisk (výstupní větvení)** – počet vstupů následně připojených hradel stejné technologie, které lze zapojit na výstup jednoho předchozího hradla
    - logický zisk je dán výstupním výkonem a vstupní spotřebou – reálná hradla mají konečnou výstupní impedanci a nenulovou vstupní impedanci -> logický zisk tak vyjadřuje, kolik lze napájet hradel stejné technologie z výstupu jednoho hradla při zajištění dostatečných napěťových úrovní pro jednotlivé logické hodnoty
  3. **logické (napěťové) úrovně** – logické hodnoty (logická 1 a 0) mají přiřazeny definované napěťové úrovně -> různé hodnoty pro různé technologie logických hradel
    - pro binární (2stavovou logiku) – dvě pásma napěťových úrovní, jedno pro logickou 1 a jedno pro logickou 0, mezi nimi pásmo neurčitého stavu
    - **pozitivní logika** – napěťová úroveň definovaná pro logickou 1 je vyšší než napěťová úroveň logické 0 (obě mohou být klidně záporné, kladné, různé),  $V_{\log 1} > V_{\log 0}$
    - **negativní logika** – opak pozitivní,  $V_{\log 0} > V_{\log 1}$

## Technologie pro HW realizaci – základní pojmy a kritéria

### ▪ Technologie pro realizaci logických hradel a obvodů

- různé technologie pro realizaci logických hradel mají různé napěťové úrovně
- 4. **šumová imunita** – přesah napěťových úrovní logické 1 a logické 0 mezi vstupními a výstupními hodnotami =  $(V_{\log 1 \text{ výstup MIN}} - V_{\log 1 \text{ vstup MIN}})$ ,  $(V_{\log 0 \text{ vstup MAX}} - V_{\log 0 \text{ výstup MAX}})$



**šumová imunita** – představuje odolnost hradla proti rušení (proti aditivnímu šumu na vstupu)

- 5. **napájecí napětí, jeho zvlnění** – pro každou technologii dané tolerančním pásmem (přípustným rozsahem napájecího napětí  $V_{cc}$ )
- často v rámci jedné technologie existuje několik řad, např. tzv. LV – low voltage
- 6. **spotřeba hradel** – daná vstupními proudy a napájecím napětím
- pro různé technologie různá – i v rámci jedné technologie řady s různou spotřebou
- se zvyšující se přepínací frekvencí spotřeba hradel narůstá – pro různé technologie ale jinak (TTL vs. CMOS)
- spotřeba -> generování odpadního tepla

## Technologie pro HW realizaci – základní pojmy a kritéria

### ▪ Technologie pro realizaci logických hradel a obvodů

7. **doba zpoždění** – obecně časová prodleva mezi okamžikem, kdy signál na vstupu hradla překročí požadovanou rozhodovací úroveň a okamžikem, kdy se na výstupu hradla objeví správná logická hodnota odpovídající dané změně
- různé technologie, různé doby zpoždění, i v rámci jedné technologie různé řady s odlišnými dobami zpoždění, např. tzv. HCMOS – High speed CMOS
- obecně **dva hlavní důvody vzniku zpoždění**:
  - reálné digitální signály mají **konečné strmosti** vzestupných (sestupných) **hran**, krátký okamžik vždy trvá, než úroveň signálu dosáhne rozhodovací úrovně
  - logická hradla obsahují reálné integrované prvky s jejich dynamickými parametry – **nabíjení parazitních kapacit, doba zaplavování a vyprazdňování PN přechodů polovodičových součástek, parazitní indukčnosti** signálových cest atd.

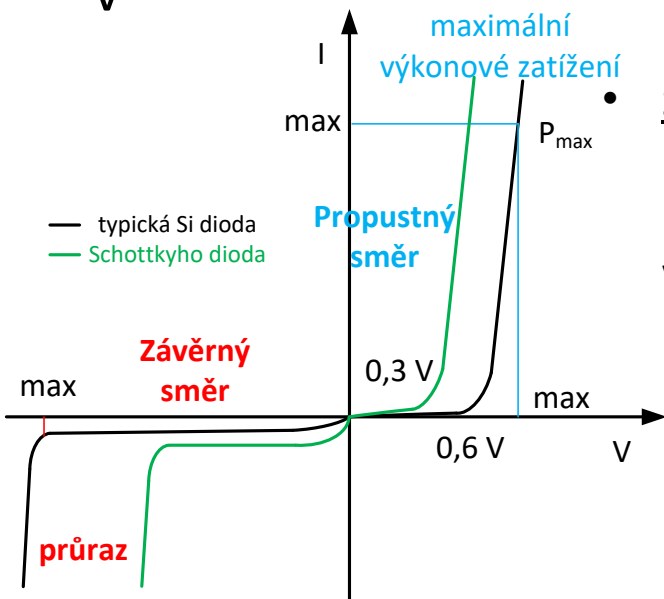
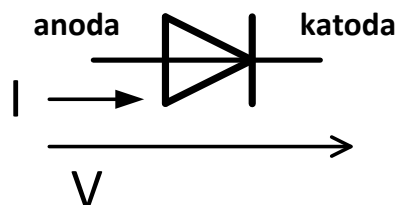


# Technologie pro HW realizaci – základní pojmy a kritéria

## ▪ Polovodičové součástky – opakování

- základní stavební prvky – polovodičové diody, **bipolární tranzistory**, **unipolární tranzistory**

### 1. Polovodičová dioda



Schottkyho dioda

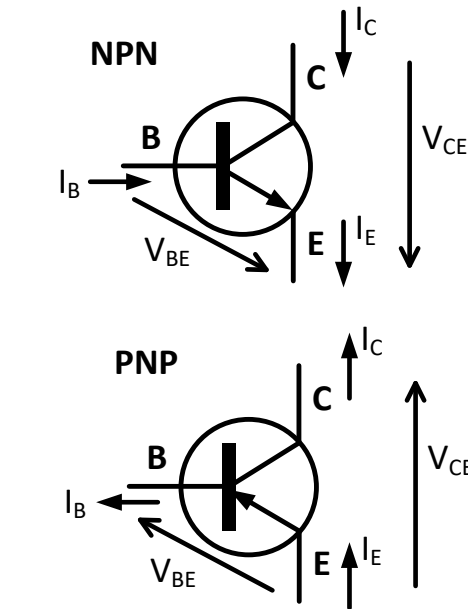


- 1 PN přechod – polovodič Si, Ge, GaAs,... – anoda, katoda
- propustný směr:  
**napětí anoda > katoda** – pro otevření křemíkové diody **0,6 V** protéká **maximální proud**, **výkonové omezení** (tepelný průraz)
- závěrný směr:  
**napětí anoda < katoda**  
**protéká jen minimální závěrný proud**, **průraz diody** (při příliš velkém napětí, záleží na konstrukci a typu diody)
  - dioda tak slouží jako spínač – ovládání napětím anoda-katoda můžeme diodu spínat, rozspínat
  - PN přechod se chová jako kapacitor – nutnost odvést (přivést) nahromaděný náboj z/do vyprázdněné oblasti – tzv. **zotavovací čas diody**
- **Schottkyho dioda** – místo PN přechodu (P + N) **přechod kov-polovodič** (kov + N) → obsahuje jen majoritní nosiče náboje
- **rychlejší zotavení** (kratší spínací časy) – vysokofrekv. aplikace
- **nižší úbytek napětí v propustném směru**, vyšší závěrný proud

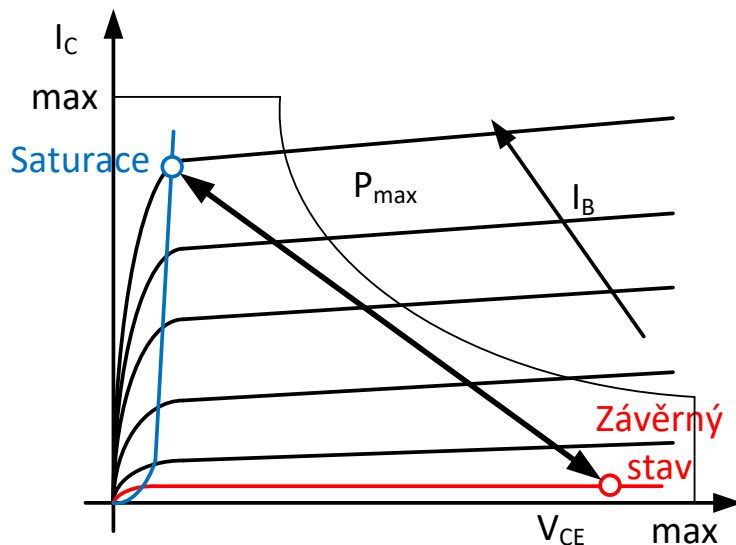
# Technologie pro HW realizaci – základní pojmy a kritéria

## ▪ Polovodičové součástky – opakování

### 2. Bipolární tranzistor

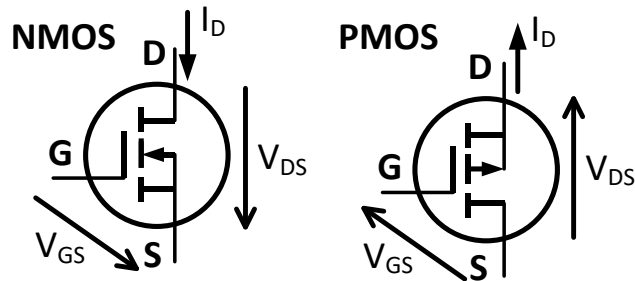


- 3 kontakty (3 oblasti), 2 polovodičové přechody: **Báze-Emitor** a **Báze-Kolektor**
- **NPN** nebo **PNP** – v případě NPN elektrony majoritní nosiče náboje, PNP majoritní nosiče díry, v obou typech se ale uplatňují i minoritní nosiče (bipolární)
- každá ze 3 částí jinak dotovaný polovodič: **emitor >> báze > kolektor**
- nastavením napětí  $V_{BE}$  a  $V_{CE}$  a proudu  $I_B$  můžeme nastavit přechody B-E a B-K do propustného či závěrného směru -> 4 pracovní režimy, v logických hradlech používáme 2 režimy:
  - **saturace** – přechody B-E i B-K plně otevřeny, tranzistorem protéká maximální kolektorový proud – tranzistor = sepnutý spínač
  - **závěrný režim** – oba přechody B-E i B-K jsou v závěrném režimu a tranzistorem protéká jen velmi malý závěrný proud – tranzistor = rozepnutý spínač
    - tranzistor v logických hradlech = **programovatelný spínač**

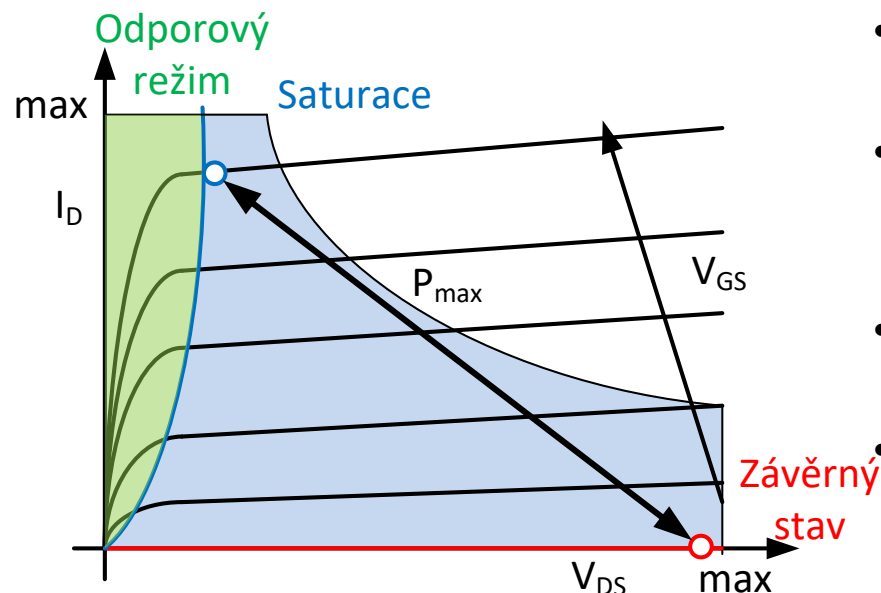


## ▪ Polovodičové součástky – opakování

### 3. MOSFET (Metal-Oxide Semiconductor Field Effect Transistor) – unipolární tranz.



- **tranzistor řízený elektrickým polem** – v logických hradlech používáme MOSFETy s **indukovaným kanálem** (v tzv. obohaceném módu) – kanál vytvořen přiložením napětí, využíváme oba typy: **typ-n (NMOS)** i **typ-p (PMOS)**
- 4 oblasti (části): **source**, **gate**, **drain**, **substrát** (nižší dotace)
- 3 kontakty: **S – source**, **G – gate**, **D – drain**, substrát je propojen se sourcem



- unipolární = jen majoritní nosiče: **elektrony – NMOS, díry – PMOS**
- napětím mezi gatem-source  $V_{GS}$  **řídíme šířku kanálu a jeho vodivost** – nulovým napětím  $V_{GS}$  můžeme kanál uzavřít
- proud tekoucí drainem  $I_D$  působí na šířku kanálu u drainu – postupné nasycení
- 3 režimy v logických hradlech – **saturace** (sepnutý spínač), **závěrný stav** (rozepnutý spínač), **odporový režim** (řízený odpor)

## Technologie pro HW realizaci – základní pojmy a kritéria

### ▪ Polovodičové součástky – opakování

- shrnutí bipolární tranzistor vs. MOSFET

### • porovnání vlastností z hlediska použití pro konstrukci logických hradel

- bipolární tranzistor – oba typy nosičů náboje
- režimy bipolárního tranzistoru dle jeho zapojení ovládáme pomocí proudu tekoucího do jeho báze (nebo emitoru) – obvykle řádově **~mA**
- oproti tomu MOSFET řídíme napětím G-S, gate je oddělený nevodivou vrstvou  $\text{SiO}_2$  -> velmi vysoký vstupní odpor -> typický proud tekoucí do gate **~nA**
- MOS – mnohem **menší proud pro řízení** -> **nižší spotřeba, menší generované teplo** -> vyšší možnosti integrace (integrace bipol. tranzistorů omezena odvodem tepla)

### • spínací doby tranzistorů

- bipolární tranzistor – použité oba typy nosičů náboje, při přepnutí ze saturace do závěrného stavu nutnost odsát volné nosiče z báze (rekombinace v bázi)
- MOSFET – jen majoritní nosiče náboje, vysoký odpor gate v kombinaci s parazitními kapacitami gate-drain a gate-source – zpoždění dané vybíjením kapacit
- bipolární tranzistory – citlivější z hlediska **teplotní stabilizace a generovaného tepla**
- MOSFETy – tenká **izolační vrstva  $\text{SiO}_2$** , může být snadno proražena (nevratně zničena) i statickou elektřinou při ruční manipulaci, **se zvyšující se frekvencí signálů narůstá strměji spotřeba** v porovnání s bipolárními tranzistory