

# **Technologie pro realizaci logických hradel a obvodů, struktura a vývoj FPGA**

RTL, DTL, TTL, CMOS, napěťové úrovně, parametry, FPGA

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

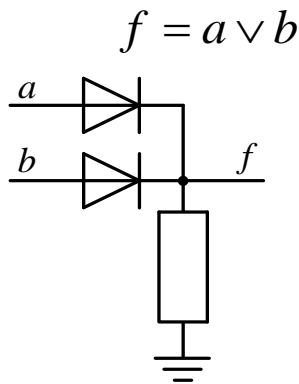
## Technologie pro HW realizaci – RTL, DTL, TTL, CMOS

### ▪ Technologie pro realizaci logických hradel

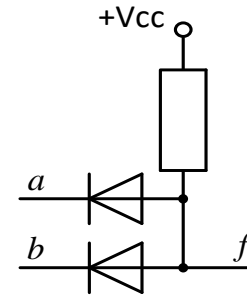
- různé technologie – historický vývoj, parametry a vlastnosti dané použitými prvky

#### 1. DL – Diodová logika

- nejjednodušší polovodičová logika – dioda a rezistory – logická hradla AND a OR
- OR – pokud alespoň jedna dioda je otevřena, na výstupu  $f$  se objeví napětí  $V_{cc}$
- AND – obě diody musí být uzavřeny, aby se na výstupu  $f$  objevilo napětí  $V_{cc}$
- problém – v diodové logice nelze realizovat negaci (NOT)
- logický součet OR



logický součin AND  
 $f = ab$



- okrajové použití, jen pro velmi jednoduché a nenáročné aplikace, výstupní napěťové úrovně závisí na vstupních, využívá se někdy jako předstupeň pro jiné logiky

#### 2. DTL – Diodově-tranzistorová logika

- hradlo má vždy 3 části:

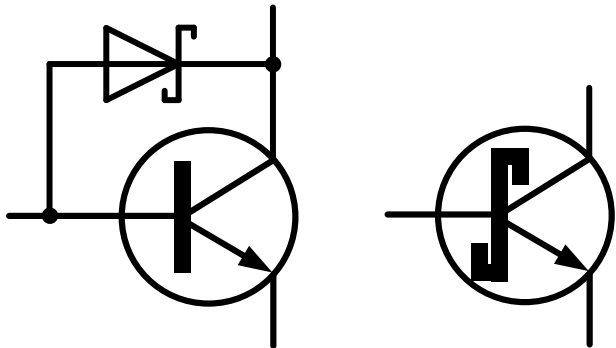
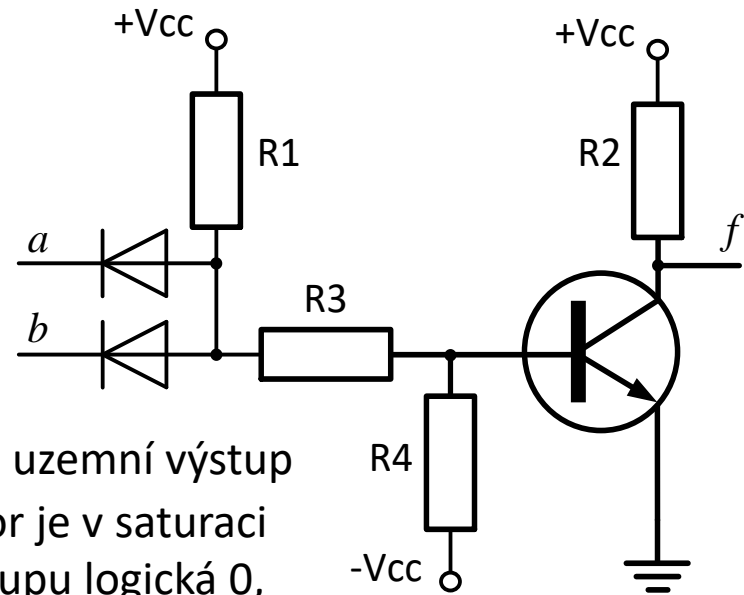
**diodová část** – realizuje logický součet OR nebo součin AND

**bipolární tranzistor** – realizuje negaci NOT, slouží jako zesilovač, stabilizátor úrovní

**rezistory** – síť rezistorů pro nastavení pracovního bodu tranzistoru a diod

## 2. DTL – Diodově-tranzistorová logika

- příklad **2vstupé hradlo NAND pomocí DTL**
- první část tvořená 2 diodami a rezistorem R1 je funkce AND – pokud oba vstupy  $a$ ,  $b$  jsou ve stavu logická 1 (+Vcc), diody jsou uzavřeny a na výstupu je logická 1 (+Vcc), pokud na alespoň jednom vstupu je logická 0 (zem), otevřená dioda uzemní výstup
- pokud je na vstupu tranzistoru logická 1, tranzistor je v saturaci – výstup  $f$  je logická 0 (uzemněn), pokud je na vstupu logická 0, tranzistor je uzavřen a na jeho výstupu  $f$  je logická 1 (+Vcc)
- různé modifikace – výměna za PNP tranzistor a R3, R4 pro odstranění -Vcc, přidání kondenzátoru přes R2 (zkrácení doby zpoždění), atd.
- problém DTL – **vysoká doba zpoždění** (až 100 ns)
- základní zapojení – tranzistor v hluboké saturaci -> dlouhé vybíjení přes diody
- **Schottkyho tranzistor – Schottkyho dioda + bipolární tranzistor**

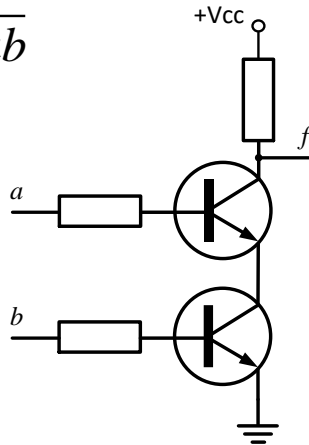


- zapojíme Schottkyho diodu mezi bázi a kolektor tranzistoru (tzv. Baker clamp) – tranzistor se nedostane do hluboké saturace -> **rychlejší zotavení tranzistoru ze saturace**
- **zkrácení doby zpoždění** – Schottkyho tranzistor je využit i v dalších technologiích, např. TTL -> LSTTL

## 3. RTL – Rezistorově-tranzistorová logika

- vlastní logická funkce realizována pomocí tranzistoru (zesílení, stabilizace úrovně), rezistory slouží jen pro správné nastavení úrovně pro tranzistory
- první technologie použita pro výrobu integrovaných obvodů (1961)
- logické hradlo NAND

$$f = \overline{ab}$$

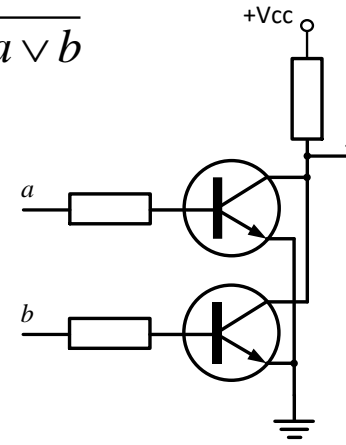


- pokud je na obou vstupech  $a, b$  logická 1 (+Vcc), jsou oba tranzistory v saturaci a výstup  $f$  je připojen na logickou 0 (zem), jinak je alespoň jeden tranzistor uzavřený a na  $f$  je napětí +Vcc – logická 1

- vysoký proudový odběr** (proudy  $I_B$  a  $I_E$ ), **velké zahřívání**
- omezené vstupní větvení a logický zisk** (typicky jen 3)
- nízká šumová imunita**
- vysoká doba zpoždění** (může být snížena použitím Schottkyho tranzistoru)

logické hradlo NOR

$$f = \overline{a \vee b}$$

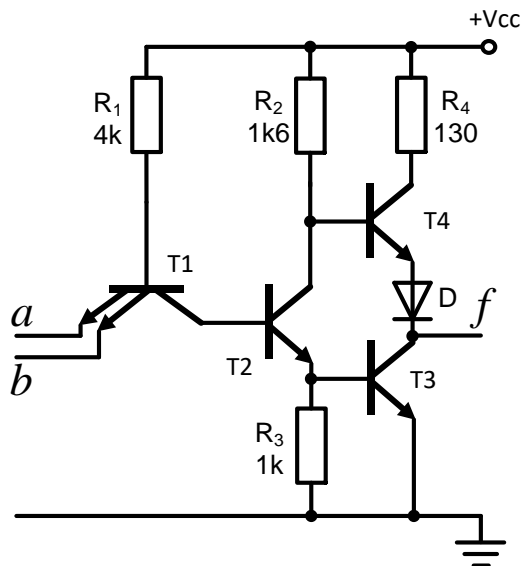


- pokud je na obou vstupech  $a, b$  logická 0, oba tranzistory jsou uzavřeny a výstup  $f$  je připojen na napětí +Vcc, logická 1, jinak pokud je alespoň jeden tranzistor v saturaci, je  $f$  připojen na zem, logická 0

## 4. TTL – Tranzistorově-tranzistorová logika

- **bipolární tranzistory jsou použity ve všech stupních** – realizace logických operací (AND, OR, NOT), zesilování, stabilizace úrovní, rezistory pouze pro nastavení proudů
- vynalezena 1961, 1963 první integrovaný obvod TTL, široké uplatnění a následný vývoj, použita i pro první počítače (realizace mikroprocesoru, grafického procesoru)

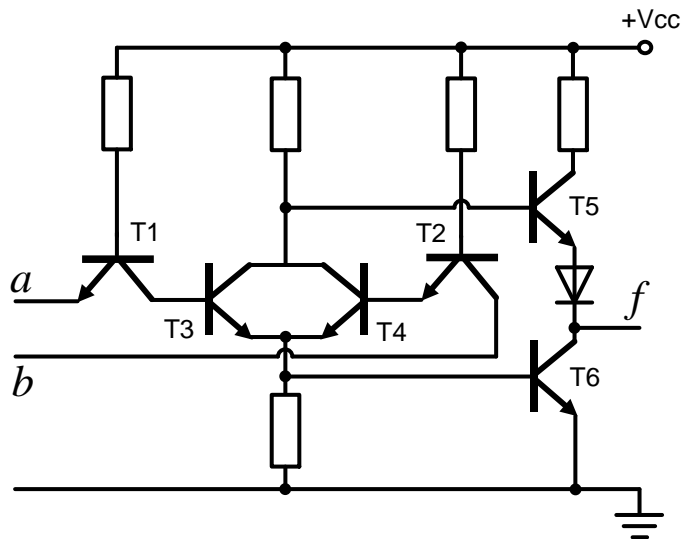
- **hradlo NAND v TTL**



- vstup – **víceemitorový tranzistor T1**
- pokud na alespoň jednom vstupu  $a$ ,  $b$  je logická 0, T1 je v saturaci a do báze T2 neteče žádný proud -> T2 je uzavřen
- tím je tranzistor T4 v saturaci a na výstupu  $f$  je logická 1 (+Vcc), opačně je T2 v saturaci, T4 uzavřen a tím je T3 v saturaci a na výstupu  $f$  je logická 0 (zem)
- tranzistory T4 + T3, dioda D a rezistor R4 představují koncový stupeň, tzv. „**totem-pole**” – jeden z tranzistorů je vždy v saturaci a druhý uzavřen -> T4 v saturaci = logická 1 na výstupu, T3 v saturaci = logická 0 na výstupu, dioda představuje dynamický odpor (propustná, závěrná)

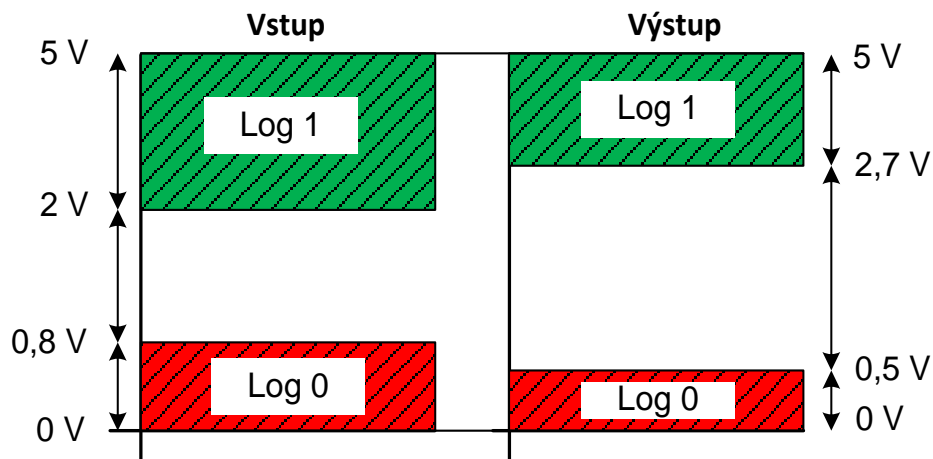
## 4. TTL – Tranzistorově-tranzistorová logika

### • hradlo NOR v TTL



- T3 – T4 tranzistory v paralelním zapojení
- pokud  $a = \text{logická } 1$   $\rightarrow$  T3 je v saturaci, pokud  $b = \text{logická } 1$   $\rightarrow$  T4 je v saturaci
- T5 + T6 + dioda = opět konfigurace „**totem-pole**”
- pokud alespoň jeden z T3, T4 tranzistorů je v saturaci – T6 je v saturaci (T5 je uzavřen) a výstup  $f = \text{logická } 0$ , v opačném případě je T5 v saturaci a T6 je uzavřen a pak  $f = \text{logická } 1$

### • základní napěťové úrovně TTL 5V – pozitivní logika



- pozitivní logika:  $V_{\log 1} > V_{\log 0}$
- základní napájecí napětí TTL =  $5 \text{ V} \pm 5\% = 4,75\text{-}5,25 \text{ V}$ , pro military  $\pm 10\%$  (4,5-5,5 V)
- Low Voltage TTL – LVTTTL (3,3; 2,5; 1,8; 1,2 V)
- vstupní úrovně vs. výstupní úrovně – tolerance **šumová imunita**  
pro logickou 0:  $0,8 - 0,5 = 0,3 \text{ V}$   
pro logickou 1:  $2,7 - 2 = 0,7 \text{ V}$
- mezi tolerančními pásmy na vstupu i výstupu – **pásmo neurčitého (zakázaného) stavu**

## Technologie pro HW realizaci – RTL, DTL, TTL, CMOS

### 4. TTL – Tranzistorově-tranzistorová logika

- **vývojové řady a integrované obvody TTL:**
- integrace TTL hradel do IO – tzv. série 7400 (komerční) a 5400 (military)
- **74XX, 74NXX** – standardní TTL hradla, spotřeba 10 mW/1 hradlo, 10 ns zpoždění
- **74LXX** – Low power TTL hradla, snížení spotřeby (1 mW), ale nárůst zpoždění (30 ns)
- **74HXX** – High speed TTL hradla, opak, snížení zpoždění (6 ns), ale nárůst spotřeby (22 mW)
- **74SXX** – Schottky TTL hradla, použití Schottkyho tranzistoru, spotřeba 20 mW, zpoždění 3 ns
- **74LSXX** – Low power Schottky TTL hradla, snížení spotřeby na 2 mW, zvýšení zpoždění 8-10 ns, dnes jedna z nejpoužívanějších
- **74ASXX** – Advanced Schottky TTL hradla, interní úprava 74LS, snížení zpoždění 1,7 ns, zvýšení spotřeby na 8 mW, napájecí napětí 4,5-5,5 V
- **74ALSXX** – Advanced low power Schottky TTL hradla, zpoždění 4 ns, spotřeba 1,2 mW
  
- 7400 – 4x 2vstupé hradlo NAND, 7403 – 4x 2vstupé hradlo NOR, 7410 – 3x 3vstupé hradlo NAND, 7411 – 3x 3vstupé hradlo AND, atd...
  
- řada výrobců: Siemens, Samsung, IBM, Texas Instruments, Motorola, AMD, Fairchild, ON Semiconductor, Tesla (značení MH74XX) aj.

### 4. TTL – Tranzistorově-tranzistorová logika

- každá řada TTL jiné parametry, zaměříme se na 74LS, dnes jedna z nejpoužívanějších:
- **vstupní větvení** – počet vstupů TTL hradel 1, 2, 3, 4, 8
- **logický zisk** – dané napěťovými úrovněmi TTL, vstupní/výstupní impedancí hradel, vstupními proudy a spotřebou – 74LS typicky ~**10-16**
- **šumová imunita** – maximální úroveň šumu, která na vstupu hradla nezpůsobí změnu výstupní hodnoty  
pro logickou 0:  $0,8 - 0,5 = 0,3 \text{ V}$   
pro logickou 1:  $2,7 - 2 = 0,7 \text{ V}$
- **zpoždění hradla** – pro 74LS = cca **8-10 ns**, jiné řady viz předchozí slide
- **spotřeba hradla** – pro 74LS = **2 mW**, jiné řady viz předchozí slide  
spotřeba při výstupu logická 0 je mnohem menší než při výstupu logická 1, přepínání výstupu hradla generuje proudové špičky -> generování rušení v obvodu, nároky na napájecí zdroj  
**se zvyšující se přepínací frekvencí se spotřeba TTL hradel zvyšuje jen omezeně**
- **TTL hradla s otevřeným kolektorem**
- z výstupní konfigurace „totem-pole“ je ponechán pouze spodní tranzistor (horní + dioda je vyjmuta) a na výstupu je tak jen jeden tranzistor s tzv. **pull-up rezistorem**
- ten umožňuje odebírat z kolektoru tohoto tranzistoru **větší (nastavitelné) proudy** -> vhodné pro spínání větší zátěže (výkonová hradla)
- **montážní součin** – můžeme spojovat výstupy dvou (a více) hradel
- nevýhoda – **delší zpoždění hradel** s ot. kolektorem (pomalejší vybíjení kapacit)



## Technologie pro HW realizaci – RTL, DTL, TTL, CMOS

### 4. TTL – Tranzistorově-tranzistorová logika

- TTL hradla s tzv. **třístavovým výstupem**
- kromě výstupní hodnoty logická 1 a logická 0 i tzv. stav **vysoké impedance Z**
- zapojení hradla upraveno, přidán tzv. **blokovací (enable) vstup** -> pomocí něho jsou uzavřeny oba tranzistory v „totem-pole“ zapojení a výstup hradla je tak zcela odpojen
- díky tomu lze spojovat výstupy více hradel dohromady – montážní součiny

### 5. ECL – Emitorově vázaná logika

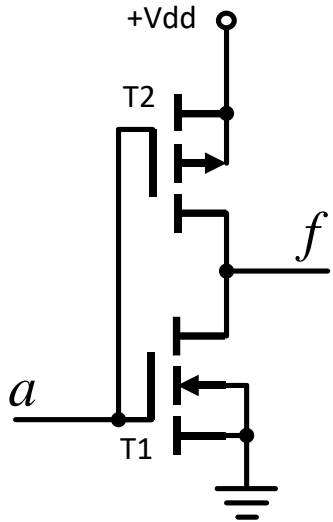
- bipolární tranzistory propojeny pomocí svých emitorů – **pracují jako zesilovače**
- tranzistory zapojené jako tzv. proudové spínače – přepínání definovaného proudu velmi malými změnami řídicího napětí (desetiny voltu)
- na rozdíl od TTL – tranzistory **nejsou nikdy v saturaci**, pouze v aktivním (zesilovacím) režimu a závěrném stavu -> výrazné zkrácení doby zpoždění
- **nejrychlejší logika** – doba zpoždění dnes ~0,7-1 ns
- **nevýhody** – vysoká spotřeba (generované teplo), velmi drahé, nízká šumová imunita, pro dosažení vysoké rychlosti nutnost optimalizace celého obvodu (plošného spoje), negativní logika (potřeba záporného i kladného zdroje napájení)
- další vývoj a řady: PECL (Positive ECL), LVPECL (Low-voltage PECL)
- dnes hlavně **speciální aplikace** (speciální paměti a mikroprocesory)

### 6. CMOS logika – Complementary metal-oxide-semiconductor logika

- dnes nejvíce rozšířená a propracovaná technologie – výroba integrovaných obvodů, mikroprocesorů, hradlových polí, pamětí, opto-elektrických prvků apod.
- základem je unipolární tranzistor **MOSFET**
- **complementary** = použití **párů tranzistorů MOS** – **jeden pár = 1 typ P**, pMOSFET + **1 typ N**, nMOSFET
- **MOSFET v odporovém režimu** – nejsou potřeba samostatné rezistory -> řízené odpory
- **obecná konstrukce hradel CMOS:**
- použití MOSFETů s **indukovaným kanálem** (v obohaceném módu)
- všechny pMOSFETy jsou vždy připojeny k napájecímu zdroji (+Vdd) nebo jinému pMOSFETu, zatímco všechny nMOSFETy musí být naopak připojeny k uzemnění nebo jinému nMOSFETu
- **pMOSFET** – kanál mezi source-drain je vodivý pokud je na gate přiloženo nižší napětí než na source, v opačném případě je kanál uzavřen nebo v odporovém režimu
- **nMOSFET** – naopak, pokud je na gate vyšší napětí než na source, kanál mezi source-drain je vodivý, jinak je kanál uzavřený nebo v odporovém režimu
- **CMOS tvoří vždy páry pMOSFET + nMOSFET** – jejich drainy a sourcy jsou navzájem propojeny – jeden tranzistor je vždy v saturaci a druhý uzavřený = komplementární
- díky tomu lze snadno vytvářet jednotlivé logické funkce, snížení spotřeby (malé proudy) a generovaného tepla

## 6. CMOS logika

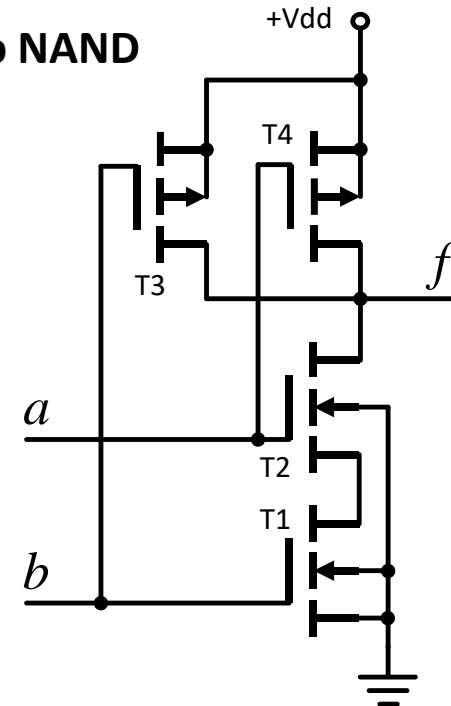
### • CMOS invertor (NOT)



- horní tranzistor **T2 je pMOSFET**, zatímco spodní tranzistor **T1 je nMOSFET**
- pokud je na vstupu  $a$  logická 0, T2 pMOSFET má otevřený kanál source-drain, zatímco T1 nMOSFET je uzavřený, takže na výstupu  $f$  je **logická 1** díky napětí zdroje +Vdd
- naopak, pokud vstup  $a$  = logická 1, T2 je uzavřený a T1 tranzistor je v saturaci, tedy na výstupu  $f$  je **logická 0** (je uzemněn)

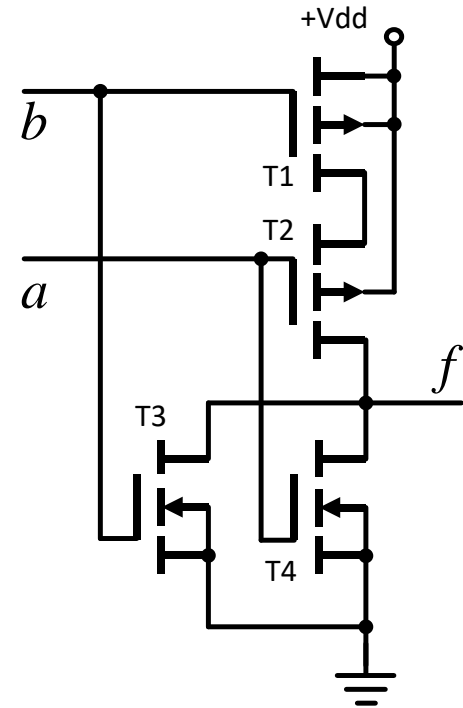
- **T1 a T2** tranzistory jsou **nMOSFETy** zapojené sériově, zatímco **T3 a T4** tranzistory jsou **pMOSFETy** zapojené paralelně – oba vstupy  $a, b$  jsou vždy připojeny na **jeden p a jeden n**
- pokud oba vstupy  $a, b$  = logická 1, oba T1 a T2 tranzistory jsou v saturaci, a oba T3 a T4 tranzistory jsou uzavřeny, na výstupu  $f$  = **logická 0** (výstup uzemněn)
- pokud alespoň jeden ze vstupů  $a, b$  = logická 0, alespoň jeden z obou pMOSFETů T3, T4 je v saturaci, zatímco alespoň jeden z nMOSFETů T1, T2 je uzavřen – výstup  $f$  je tak připojen ke zdroji +Vdd = **logická 1**

### • CMOS hradlo NAND



## 6. CMOS logika

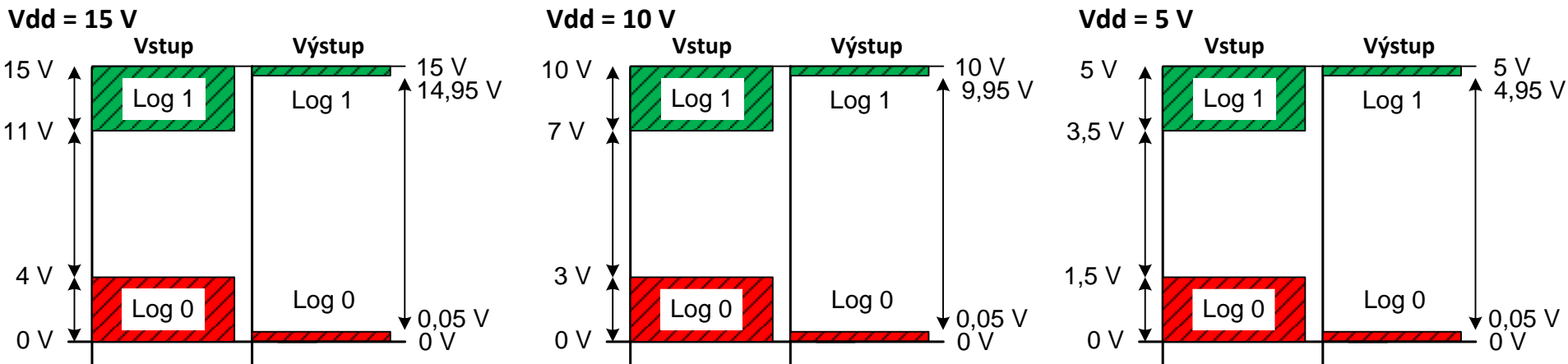
### • CMOS hradlo NOR



- **komplementární zapojení k hradlu NAND** – nMOSFETy T3, T4 jsou zapojeny paralelně, zatímco pMOSFETy T1, T2 sériově, opět každý z obou vstupů *a*, *b* je připojen vždy **k jednomu p a jednomu n**
- díky tomu, pokud jsou oba vstupy *a*, *b* = logická 0, oba tranzistory T1 a T2 jsou v saturaci, zatímco oba tranzistory T3 a T4 jsou uzavřeny – výstup *f* je připojen ke zdroji +Vdd = **logická 1**
- jinak, pokud alespoň jeden ze vstupů *a*, *b* = logická 1, alespoň jeden z nMOSFETů T3, T4 je v saturaci a alespoň jeden z pMOSFETů T1, T2 je uzavřený – tím pádem je výstup *f* uzemněn = **logická 0**

## 6. CMOS logika

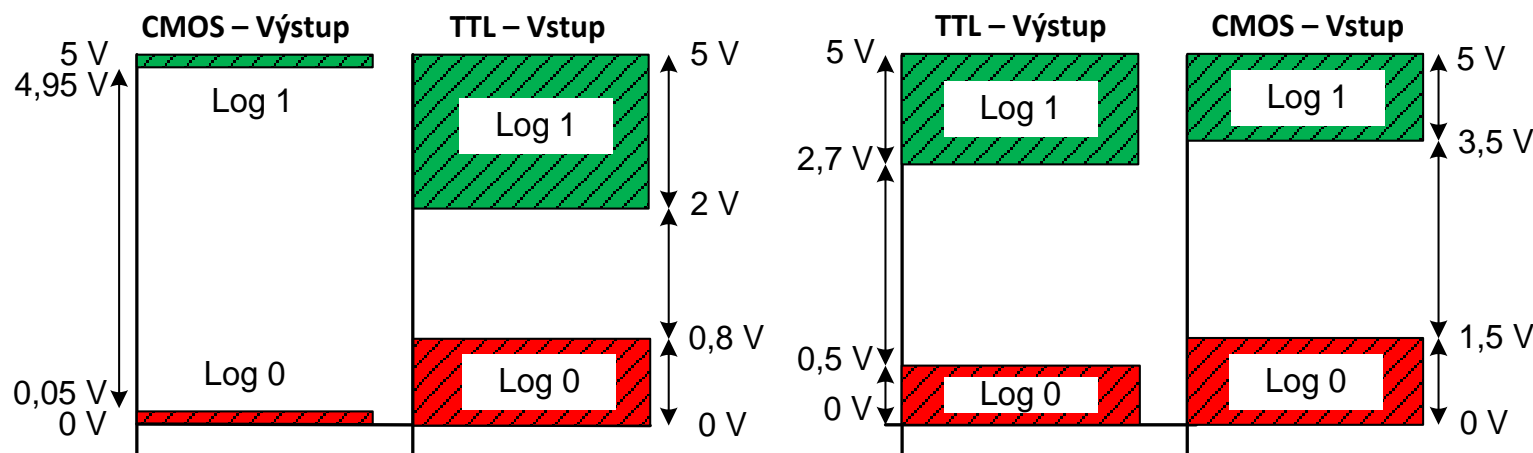
- **napěťové úrovně CMOS logiky – široký rozsah napájení V<sub>dd</sub> – základní řada 3-15 V**



- standardní řada CMOS má **mnohem širší toleranci napájecího napětí v porovnání s TTL – V<sub>dd</sub> = 3-15 V**, obvykle se používají V<sub>dd</sub> = **15 V**, = **10 V** a **nejčastěji pak = 5 V**
- vstupní úrovně pro logickou 1 a 0 jsou pak dle toho uzpůsobeny, toleranční rozpětí pro výstupní napěťové úrovně je pak vždy stejné a stejně úzké – **0,05 V**
- to je z toho důvodu, že kanál source-drain MOSFETu v saturaci má velkou vodivost a malý odpor (velmi malý úbytek napětí jen max. 0,05 V v propustném směru), a naopak velmi malou vodivost a velký odpor v uzavřeném stavu (protéká jen velmi malý závěrný proud, vytvářející napěťovou úroveň do max. 0,05 V)
- výhoda – **vysoká šumová imunita CMOS** (např. pro V<sub>dd</sub> = 5 V):  
pro logickou 0: 1,5-0,05 = **1,45 V**  
pro logickou 1: 4,95-3,5 = **1,45 V**
- existují i Low voltage CMOS řady – **LVC MOS**: 3,3 V; 2,5 V; 1,8 V; 1,5 V a 1,2 V

## 6. CMOS logika

- kompatibilita napěťových úrovní CMOS vs. TTL – zapojení hradel CMOS a TTL



- připojení hradla TTL na výstup hradla CMOS:**
  - napěťové úrovně na výstupu CMOS splňují tolerance napěťových úrovní na vstupu TTL
  - ale kromě napěťových úrovní musí být hradlo schopné zajistit i minimální dodávaný proud (výkon) do vstupu druhého hradla -> TTL vyžaduje větší proudový odběr ve stavu logická 0, než je schopné standardní CMOS hradlo dodat na výstupu
  - nutný přídatný zesilovač (buffer): CMOS – buffer – TTL, např. CD4050, CD4010 apod.
- připojení hradla CMOS na výstup hradla TTL:**
  - napěťová úroveň pro logickou 1 na výstupu TTL nemusí splňovat vstupní úroveň CMOS
  - bud' lze použít TTL s otevřeným kolektorem a vhodným pull-up rezistorem nastavit úroveň, nebo opět použít přídatný obvod (buffer): TTL – buffer – CMOS, např. CD4504
- řešení – řada CMOS přímo kompatibilní s TTL logikou** – zaručena přímá kompatibilita s TTL úrovněmi – **tato řada CMOS hradel má v názvu „T”** (např. 74HCTxx apod.)

### 6. CMOS logika

- **výhoda CMOS** – maximálně zvládnutý proces výroby a integrace, neustálý vývoj  
SSI – small-scale integration (1964) – integrace jen desítek tranzistorů  
MSI – medium-scale integration (1968) – maximálně do 100 tranzistorů  
LSI – large-scale integration (1971) – maximálně 10.000 tranzistorů  
VLSI – very large-scale integration (1980) – maximálně 100.000 tranzistorů  
ULSI – ultra large-scale integration (1984) – stovky tisíc tranzistorů  
GSI – giga-scale integration (90. léta) – miliony tranzistorů  
WSI – wafer-scale integration, SoC – system-on-a-chip, 3D-IC – three-dimensional integrated circuits... – dnes integrace již **miliard až biliónů tranzistorů** (max.  $2,6 \cdot 10^{12}$ )
- **výrobní proces CMOS – vyjadřuje polovinu vzdálenosti mezi dvěma identickými strukturami v poli jedné buňky** (pouze vybrané):  
10 μm (1971), 1 μm (1985), 600 nm (1994), 130 nm (2001), 45 nm (2008), 14 nm (2014), 10 nm (2017), 7 nm (2018), 5 nm (2020), 2 nm (2024) – dnes již pokusy s 1 nm (pro velikosti pod 5 nm bude potřeba nově vyřešit řadu problémů)
- 1965 (75) G. Moore – tzv. **Moorův zákon** „počet tranzistorů, které mohou být umístěny na integrovaný obvod, se při zachování stejné ceny zhruba každých 18 měsíců zdvojnásobí“
- dnes období zdvojnásobení spíše 20-24 měsíců
- v budoucnu bude nutno překonat některé obtíže – velikost tranzistorů se blíží atomární velikosti, nové jevy, nové postupy, nové materiály

### 6. CMOS logika

- opět jako v případě TTL, **vývoj a vývojové řady CMOS logiky:**
- první řada CMOS hradel – obvody řady **40xx**  
velký rozsah napájecího napětí (3-15 V), vysoká šumová imunita, nízká spotřeba (25 nW-1μW), vysoká doba zpoždění (závisí na V<sub>dd</sub>) 60-100 ns
- další řady již začleněny v rámci série **74xx** (54xx military):
  - 74Cxx** – CMOS – stejná řada 40xx pouze přepracovány piny IO pro kompatibilitu s TTL
  - 74HCxx, 74HCTxx** – High-speed CMOS (TTL kompatibilní), rozsah napájecího napětí snížen na 2-6 V u HC a 4,5-5,5 V u HCT, nižší šumová imunita, nízká spotřeba (desítky μW), doba zpoždění srovnatelná s TTL – ~10 ns
  - 74ACxx, 74ACTxx** – Advanced CMOS (TTL kompatibilní), napájecí napětí stejné jako u TTL, další snížení zpoždění – ~9 ns, zvýšení spotřeby
  - 74AHCxx, 74AHCTxx** – Advanced High-speed CMOS (TTL kompatibilní), napájecí napětí stejné jako u TTL, velmi nízké zpoždění – ~4-5 ns, zvýšení spotřeby – stovky μW, snížení šumové imunity
  - 74AUCxx** – Advanced ultra-low voltage CMOS, **74AUPxx** – Advanced ultra-low power CMOS – aplikace s minimální spotřebou (napájení z baterií)
- ... a další



### 6. CMOS logika

- opět každá řada CMOS odlišné parametry, srovnání s TTL:
- **vstupní větvení** – zapojení hradel v CMOS – paralelní a sériové uspořádání, zvýšení počtu vstupů, přidání jedné dvojice p a n tranzistorů – zvyšování spotřeby, plochy hradla, zvyšování zpoždění – obvykle 1, 2, 3, 4 vstupy (pro spec. aplikace i 8)
- **logický zisk** – v porovnání s TTL mají CMOS řady nižší spotřebu, na druhou stranu zvyšování výstupního větvení vede k nárůstu zpoždění (nabíjení a vybíjení parazitních kapacit – obvykle **řada 40xx desítky-100, řady HC, AC a AHC desítky**
- **šumová imunita** – původní řada 40xx vysoká šumová imunita (**1,45 V**), modernější řady nižší šumová imunita (snižování Vdd), HC 0,9 V, AC 1,4 V, **AHC 0,55 V**
- **zpoždění** – původní řada 40xx vysoká doba – **desítky ns** (60-100 ns), moderní řady HC, AC, AHC snižování zpoždění – **srovnatelné s řadami TTL, nebo i nižší (AHC)**  
**závislost doby zpoždění na Vdd** – vyšší napětí -> nižší zpoždění, dále závislost na vstupním/výstupním větvení, přepínací frekvenci apod.
- **spotřeba** – **původní řada 40xx ~1μW, moderní řady AHC ~10-100 μW**  
silná závislost spotřeby na zátěži hradla (kapacitní spínání)  
závislost spotřeby na přepínací frekvenci – do určité frekvence lineární nárůst, později však **exponenciální** -> pro vysoké frekvence je spotřeba již srovnatelná s TTL  
opět přepínání hradla generuje proudové špičky -> generování rušení v obvodu, nároky na napájecí zdroj

### 6. CMOS logika

#### • problémy CMOS

- zvyšování přepínací frekvence, zmenšování velikosti integrovaných tranzistorů -> problémy s odvodem generovaného tepla
- dále problém tzv. svodových proudů (**leakage currents**), při zmenšování a integraci miliónů tranzistorů začínají hrát podstatnou roli -> zvyšování spotřeby, tepla
- **tranzistory jsou náchylné ke zničení**
  - statická elektřina (náboj) snadno zničí izolační vrstvu gatu
  - výstupy CMOS hradel nesmí být nikdy přímo navzájem spojovány
  - při manipulaci s logickými vstupy/výstupy hradel musíme vždy odpojit napájecí zdroj, před přivedením logických úrovní na vstupy hradel je nejprve potřeba připojit napájení hradel
- **nezapojené vývody** – musí být vždy správně ošetřeny, hrozí vznik statického náboje, zvyšují navíc spotřebu hradla (kapacitu)

### 7. BiCMOS – Bipolar CMOS logic

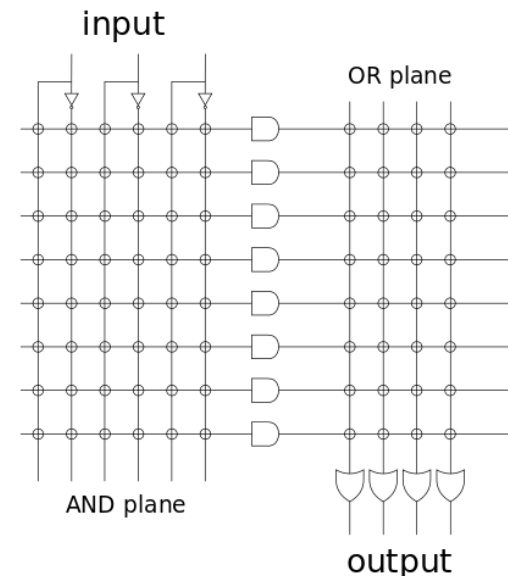
- kombinování bipolárních (výstupy) a MOSFET tranzistorů (vstupy) v jednom hradle
- vhodná kombinace výhod obou technologií – rychlost, spotřeba, výstupní výkon
- prozatím složitý a drahý proces výroby (využívá např. Intel)

### 8. IIL (I<sup>2</sup>L, I<sup>2</sup>L) – Integrated injection logic

- využití tranzistoru PNP jako injektoru minoritních nosičů náboje
- snížena spotřeba bipol. tranzistorů (a generované teplo) -> možnost vysoké integrace

## FPGA, CPLD – Programovatelná pole, programovatelné obvody, vývoj

- **VHDL** = VHSIC Hardware Description Language
  - VHSIC = Very High Speed Integrated Circuits – rychlé integrované obvody, FPGA, CPLD, ....
  - HDL – Hardware Description Language – nejedná se o programovací jazyk! **v jazyce VHDL neprogramujeme!**, navrhujeme, simulujeme, provádíme syntézu digitálních obvodů tím, že je popisujeme – description language = **jazyk pro popis!**
  - rodina HDL jazyků – dnes hlavně **VHDL** (akademická sféra, Evropa) a **Verilog** (průmysl, USA)
- **vývoj programovatelných obvodů a polí**
  - 50. a 60. léta – výroba prvních typů TTL a CMOS, vývoj prvních komunikačních protokolů, prvních integrovaných obvodů, prvních pamětí, prvních počítačů...
  - 1970 – PROM (Programmable Read-Only Memory), 1971 – EPROM (Erasable PROM)
  - 1975 – **PLA** (Programmable Logic Array) – první programovatelný logický obvod
    - matice programovatelných hradel AND a programovatelných hradel OR
    - realizace jednoduchých kombinačních obvodů v disjunktivním i konjunktivním tvaru
    - $n$  – vstupů,  $m$  – výstupů,  $2^n$  – hradel AND,  $m$  – hradel OR
    - na výstupu matice invertorů – NOT
    - v názvu uvedené „programmable“ je zavádějící – PLA může být naprogramováno jen jednou při výrobě (vypálením příslušných cest a propojů), je tedy jen read-only
    - některé ze základních myšlenek použitých v PLA slouží i dnes v moderních polích FPGA a CPLD



**FPGA, CPLD** – Programovatelná pole, programovatelné obvody, vývoj

- **vývoj programovatelných obvodů a polí**

- 1978 – **PAL** (Programmable Array Logic)
  - opět matice programovatelných hradel AND následována pevnou maticí neprogramovatelných hradel OR a maticí invertorů NOT
  - lze realizovat kombinační obvody jen pomocí disjunktivní formy
  - programovatelná matice hradel AND (PROM paměť):
    - opět lze naprogramovat jen jednou pomocí přepalování cest a propojů
    - po vynálezu EPROM (PROM paměť umožňující vymazání pomocí UV záření a reprogramování) využita pro přeprogramování obvodů PAL
    - později i moderní verze obvodů PAL s paměťmi typu flash a EEPROM - Electrically-Erasable PROM
- 1983 – EEPROM (Electrically-Erasable PROM) – umožnila velký pokrok i v oblasti programovatelných hradlových polí
- 1983 – **GAL** (Generic Array Logic)
  - vylepšení předchozích obvodů PAL od Lattice Semiconductor
  - použití EEPROM pro přeprogramování hradlového pole
  - konfigurovatelné výstupy označeny OLMC (Output Logic Macrocells) – možnost použít dodatečné kombinační obvody na výstupu, registry apod.
  - dnes se obvody GAL omezeně používají v některých velmi jednoduchých aplikacích
- 1984 – paměť Flash – nevolatilní paměť nabízející snadné přepisování, přístupu atd.
- obvody PAL i GAL – ekvivalentní počet hradel desítky až stovky

## **FPGA, CPLD** – Programovatelná pole, programovatelné obvody, vývoj

- **vývoj programovatelných obvodů a polí**

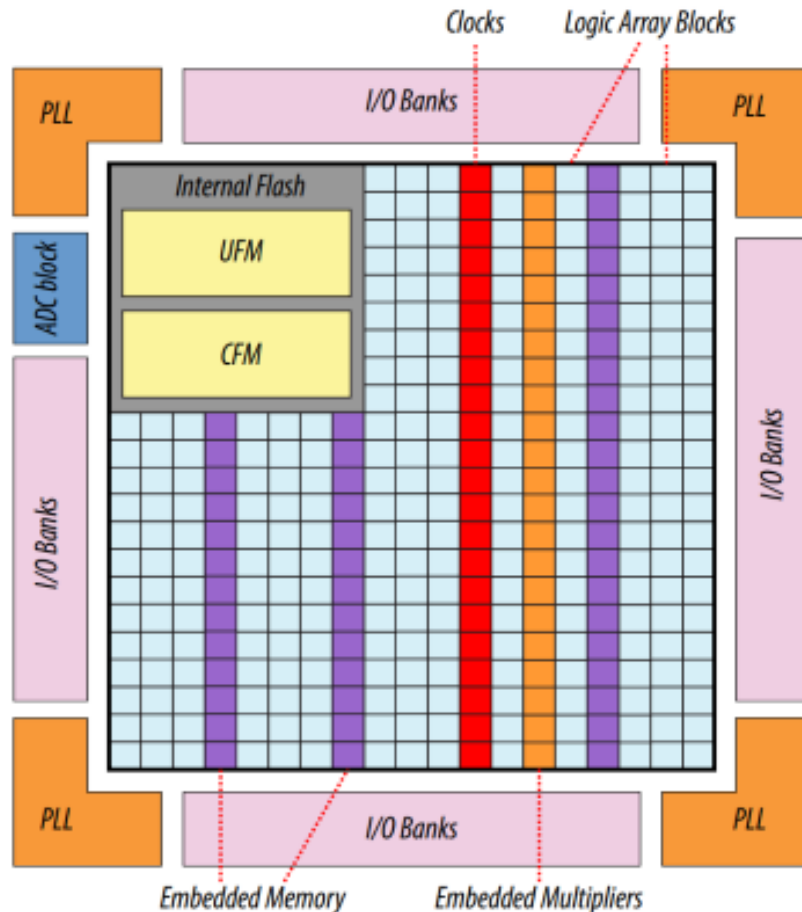
- 1985~88 – **CPLD** (Complex Programmable Logic Device)
  - rozsáhlá matice hradel (ekvivalentní počet 1.000-10.000 logických hradel)
  - vychází ze stejných principů jako předchozí PAL, GAL – makrobuňky (matice) programovatelných hradel AND a OR (disjunktivní tvar funkce) + invertory na vstupu i výstupu, makrobuňky pro speciální logické i aritmetické operace (násobičky...),...
  - vybavené pamětí Flash i EEPROM
- 1985 – **FPGA** (Field-Programmable Gate Array)
  - v současnosti nejpokročilejší typ hradlového pole, současná FPGA obsahují miliony ekvivalentních logických hradel
  - založena na odlišném principu – LUT – Lookup Table – v podstatě programovatelná pravdivostní tabulka (syntéza logických funkcí), která generuje výstupní hodnoty na základě vstupů a definované funkce (chování) obvodu
  - FPGA obsahuje navíc řadu specializovaných bloků – komparátory, klopné obvody, matice cest a propojů, specializované bloky (rychlé násobičky, registry, paměti, A/D převodníky...) – každý výrobce jiné řešení
  - dnes největší výrobci – **Xilinx**, **Altera** (vlastněna Intelem), menší výrobci – Lattice Semiconductor, Microsemi, Atmel, atd.
  - FPGA jsou dnes použita v řadě moderních aplikací – rychlé zpracování obrazových, audio i video signálů, síťové prvky, bezdrátová komunikace, mikroprocesory, spotřební elektronika, nejrůznější průmyslové a military aplikace...

## FPGA, CPLD – FPGA detailní pohled

- **blíží pohled na FPGA – co je uvnitř a jak funguje?**

- základy architektury a principy jsou stejné u různých výrobců, ale každý přidává vlastní modifikace a úpravy – i jednotlivé řady jednoho výrobce se liší (výkon, cena...)

### FPGA MAX10



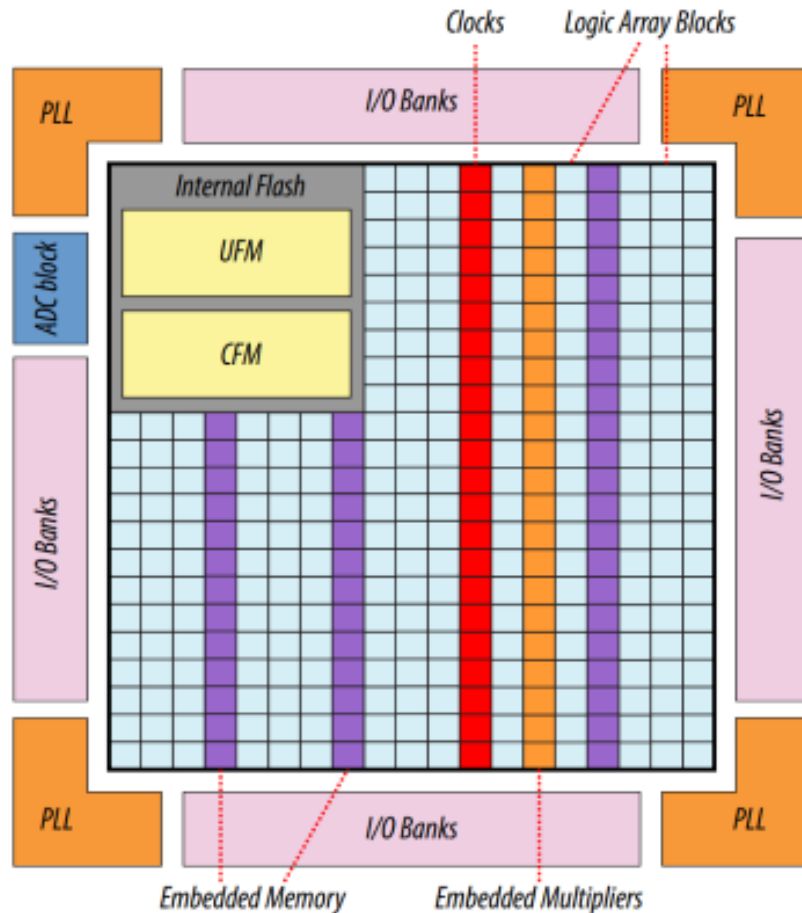
- FPGA jsou běžné integrované obvody, vyráběné běžnou technologií, v pouzdru
- **IO bloky** – Vstupní/Výstupní Bloky – pro připojení vstupů/výstupů pole a vnitřních částí FPGA s vnějšími periferiemi, obvykle obsahují registry, multiplexory, budiče, invertory, nastavitelné pull-up a úrovně...
- **LAB bloky** – Logic Array Blocks – základní část FPGA obsahující LUT tabulky a provádějící logické operace
- **PSM** – Programmable Switch Matrix – programovatelná matice cest a propojů
- **Paměťové bloky** – bloky paměti typu RAM (volatilní) použitelné pro různé účely
- **Násobičky** – bloky provádějící různé aritmetické operace, zejména násobení
- **Distribuce a práce s hodinovým signálem** – PLL fázové závěsy, cesty pro distribuci hodinového taktu

## FPGA, CPLD – FPGA detailní pohled

- **blíží pohled na FPGA – co je uvnitř a jak funguje?**

- základy architektury a principy jsou stejné u různých výrobců, ale každý přidává vlastní modifikace a úpravy – i jednotlivé řady jednoho výrobce se liší (výkon, cena...)

### FPGA MAX10

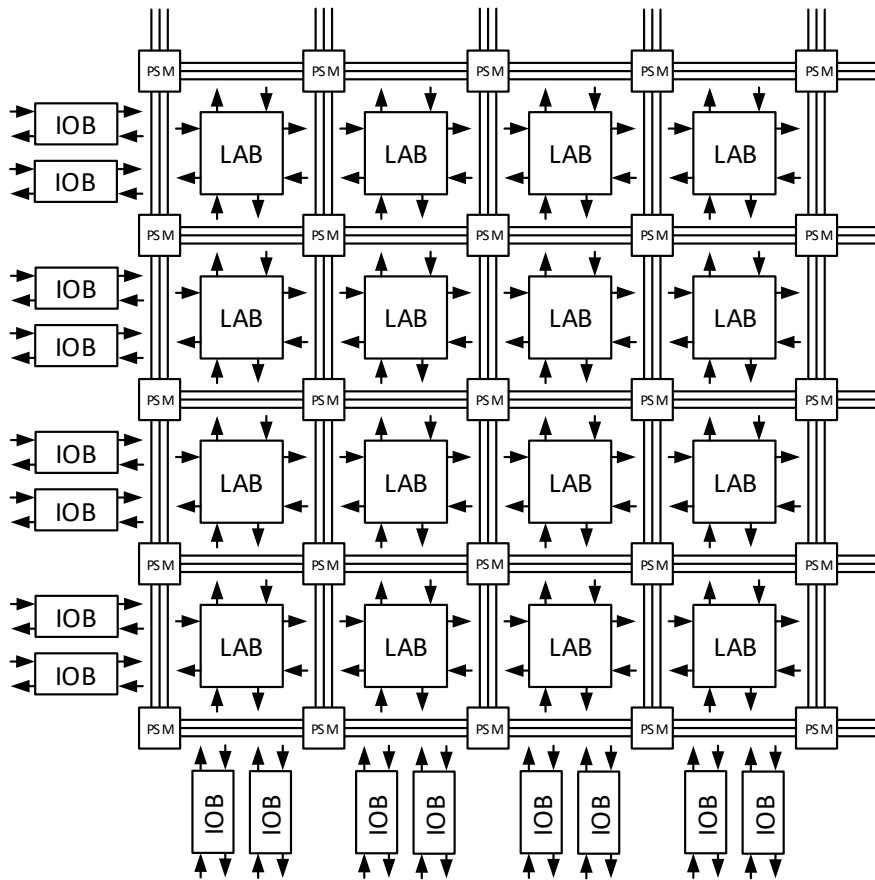


- **ADC** – Analog/Digital Converter – převodníky analogových-digitálních signálů, umožňují FPGA komunikovat a pracovat s analogovými signály (např. analogové senzory veličin, analogové periferie apod.)
- **Vnitřní FLASH paměť** – UFM (User Flash Memory) a CFM (Configuration Flash Memory), paměti pro uložení konfigurace a naprogramování FPGA pole, nevolatilní paměť, obvykle JTAG rozhraní pro přístup

## FPGA, CPLD – FPGA detailní pohled

- **blíže pohled na FPGA – co je uvnitř a jak funguje?**

- toto je jen základ, FPGA různých výrobců a produktových řad často navíc obsahují – CPU (moderní procesory typu ARM), DSP (Digital Signal Processing units – signálové procesory), různé řadiče a kontroléry (USB, Ethernet, DRAM...) a spoustu jiných
- podrobnější popis LAB a PSM

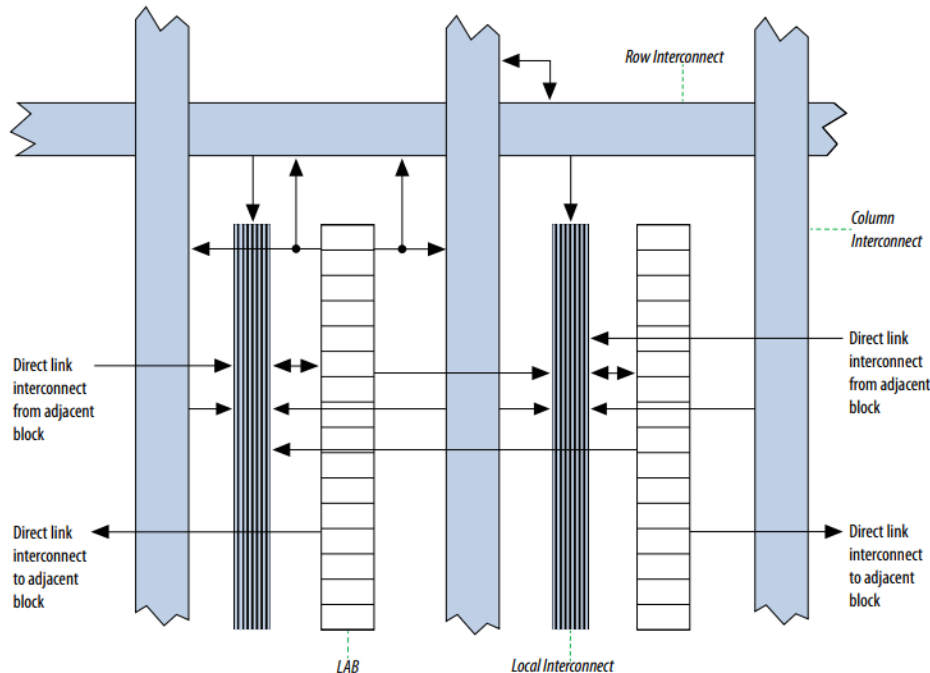


- **LAB** jsou obklopeny a navzájem odděleny horizontálními i vertikálními signálovými cestami – metalickými propoji
- soubor všech těchto cest a propojů a jejich ovládání tvoří matici **PSM** – Programmable Switch Matrix
- jedna z předností FPGA – snadné a propracované propojování bloků FPGA
- dnes v průměru až 75% plochy FPGA čipu tvoří matice cest a propojů (PSM) a pouze 25% plochy zbývá na samotné bloky pro syntézu logických funkcí LAB!



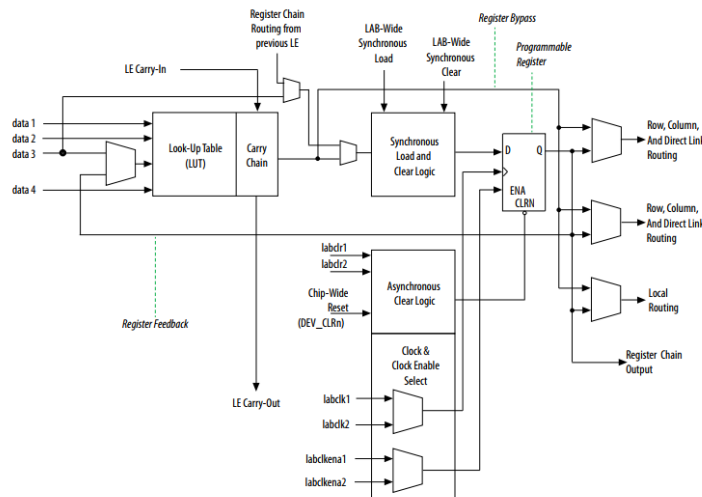
## FPGA, CPLD – FPGA detailní pohled

- LAB bloky jsou primárně určeny pro syntézu kombinačních a sekvenčních log. obvodů
- různí výrobci různá struktura bloků a názvů, i jeden výrobce v různých produktových řadách odlišná struktura a parametry – zaměříme se na MAX10 (použit na cvičení)
- **MAX10 LAB:**
  - **16 LE** (Logic Element) – základní logická buňka (viz dále)
  - **LE přenos** (LE carry chain) – sériový přenos přes všechny LE v daném LAB
  - **LE sousední přenos** – přenos mezi dvěma sousedními LE v LAB
  - **LAB řídicí signály** – generování a přenos řídicích signálů v jednom LAB mezi jednotlivými LE
  - **Lokální propoje a cesty** – vnitřní signálové cesty v LAB mezi jednotlivými LE



## FPGA, CPLD – FPGA detailní pohled

- **LE (Logic Element)**
- Základem je 4vstupá LUT tabulka – **Look-up Table**
- LUT je v podstatě „programovatelná“ pravdivostní tabulka, paměťový (RAM) **generátor logických funkcí na základě definované pravdivostní tabulky**
- každá **LUT má 4 vstupy a 1 výstup** – jedna LUT tak může implementovat libovolnou funkci 4 proměnných (dle pravidel Booleovy algebry), pokud potřebujeme vytvořit funkci více proměnných → kaskádně spojíme více LUT tabulek (pomocí přenosů mezi LE)
- LE dále obsahuje registry (multiplexory), programovatelný klopný obvod (lze využít jako JK, D, T, RS), různé cesty a propoje, atd.
- LE může pracovat ve 2 režimech: **normální** a **aritmetický**
- **Normální** – využita 4vstupá LUT pro realizaci log. funkce
- **Aritmetický** – LUT využita jako 2bitová sčítačka s přenosem pro aritmetické operace (vhodné pro realizace sčítaček, čítačů, komparátorů apod.)



## FPGA, CPLD – FPGA detailní pohled

- **Hodinové obvody**

- globální distribuce taktu **GCLK** (Global Clock Network) a fázové závěsy **PLL** (Phase-locked Loop)
- Interní oscilátor – 232 MHz s možností děliček na nižší takty
- **GCLK** – interní cesty a síť pro distribuci hodinového signálu z externího zdroje
- **PLL** – dvojice či čtveřice fázových závěsů umístěných v kvadrantech FPGA pole pro práci s frekvencí a fází vstupního taktovacího signálu (děličky, fázové posuvy, atd.)

- **Vestavěná paměť**

- paměť přímo integrovaná v FPGA poli – celkem 9,2 kbitů
- využitelná jako ROM, RAM, posuvné registry, fronty apod.

- **Násobička**

- blok integrované násobičky v FPGA poli, lze využít ve 2 režimech
  - **jedna 18 x 18 bitová násobička** – jedna násobička pro 10-18 bitů
  - **dvojice 9 x 9 bitových násobiček** – dvě samostatné násobičky pro max. 9 bitů na vstupech

- **IO Bloky**

- vstupy/výstupy FPGA pole
- komunikace s externími periferiemi, zdroji, paměťmi, atd.
- rozděleny na 3 druhy (dle rychlosti a použitých protokolů komunikace)

- **AD převodník**

- jeden či dva 12bitové AD převodníky