



TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky
a mezioborových studií



Zásady návrhu elektronických systémů

Milan Kolář

Ústav mechatroniky a technické informatiky



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost

Projekt ESF CZ.1.07/2.2.00/28.0050
**Modernizace didaktických metod
a inovace výuky technických předmětů.**

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



Zápis v HDL jazycích

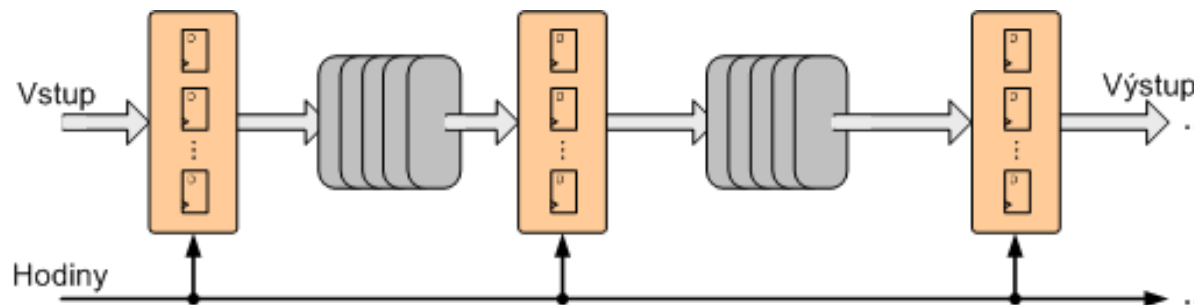
Nejčastěji používáme úroveň **RTL** (Register Transfer Level).

Datová část - v podstatě se jedná o střídání kombinační logiky (výpočetní jednotky) a registrů pro uchovávání dat.

Řídící část – stavové automaty - opět se střídá kombinační logika (přechodové a výstupní funkce) s registry (pro vnitřní stav).

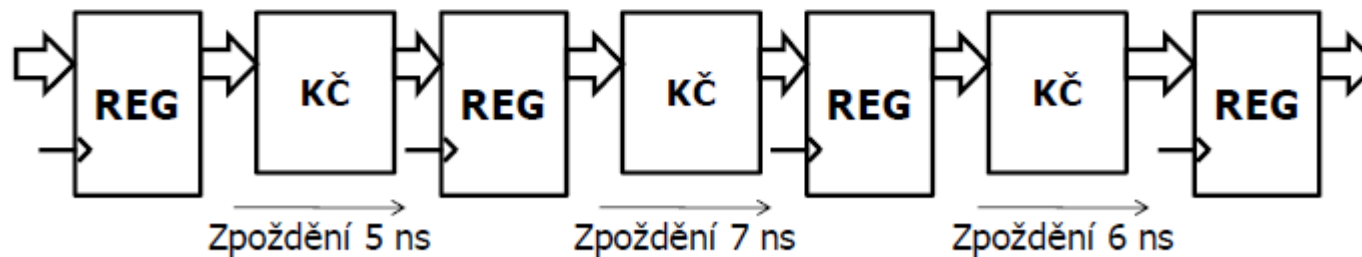
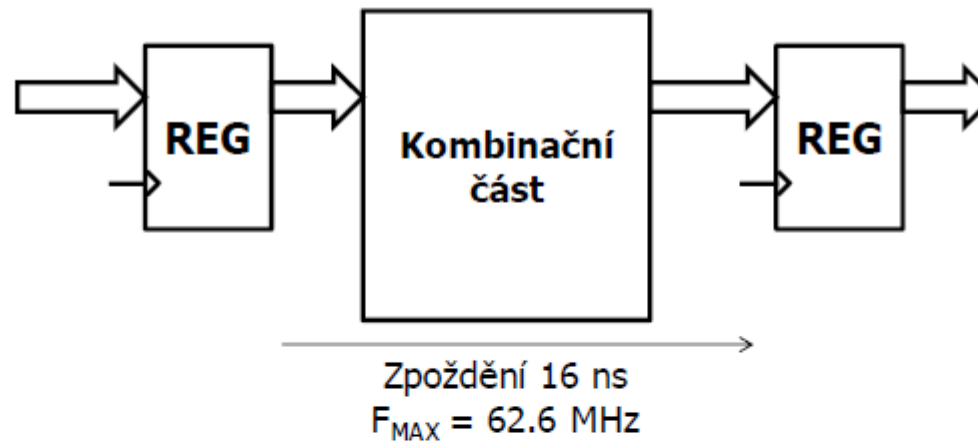
=> vzniká „sendvičová“ struktura registrů a kombinační logiky (přelévání dat mezi registry)

– podpora v hardwaru FPGA (LE = LUT + klopný obvod).



Technika zřetězení (pipelining)

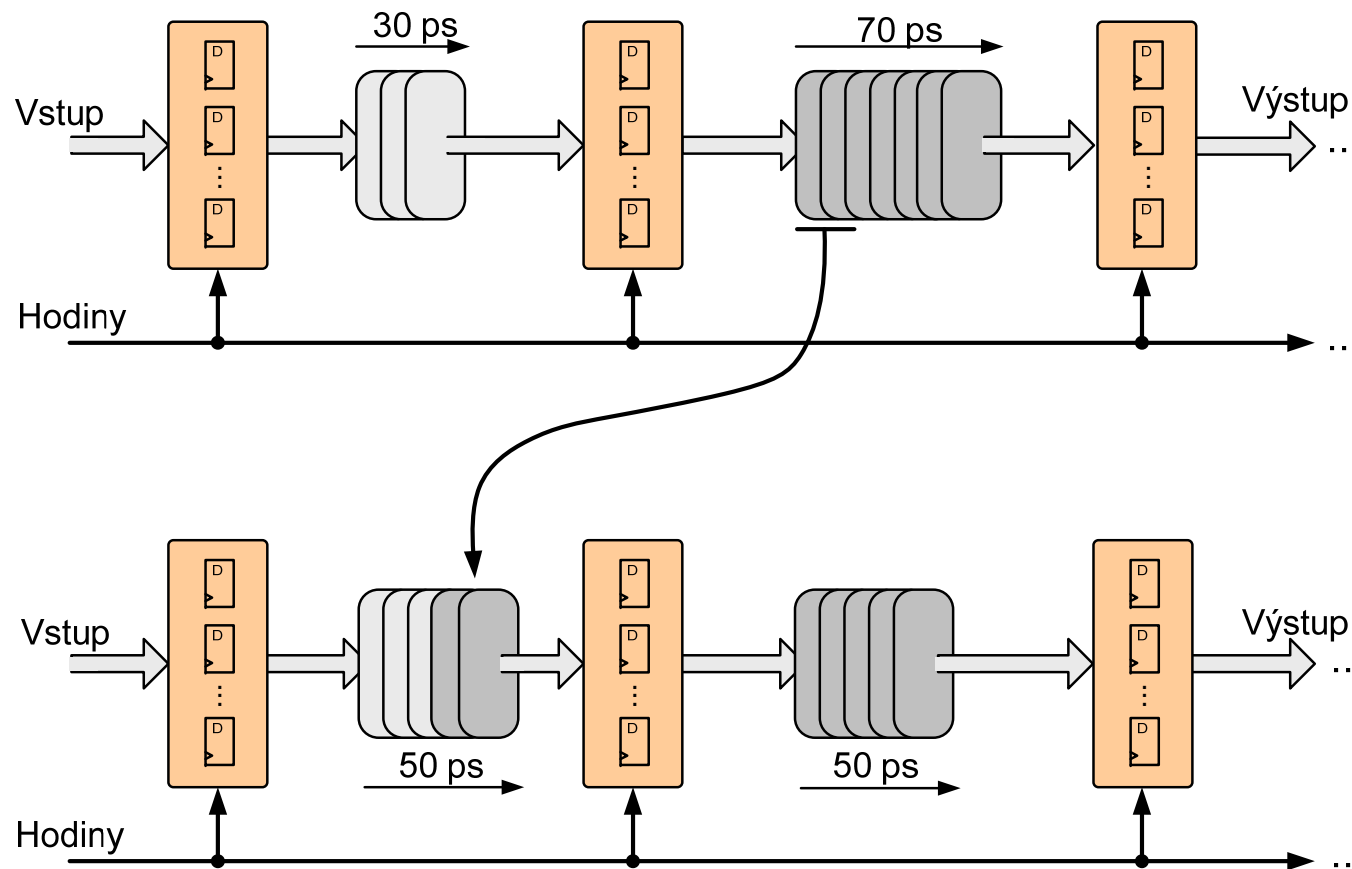
Umožňuje zvyšování pracovního kmitočtu v systému



Největší zpoždění 7 ns => $F_{MAX} = 143 \text{ MHz}$



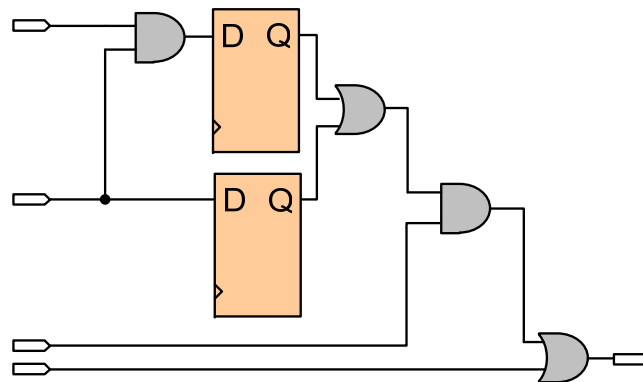
Synchronizace registrů (retiming)



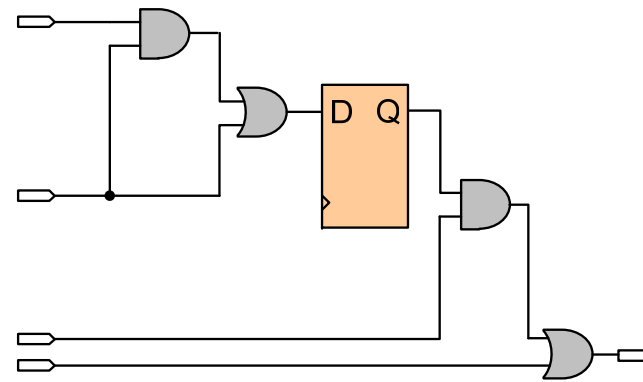


Příklad synchronizace registrů

Vyrovnání kombinačních logických cest mezi registry



Před synchronizací



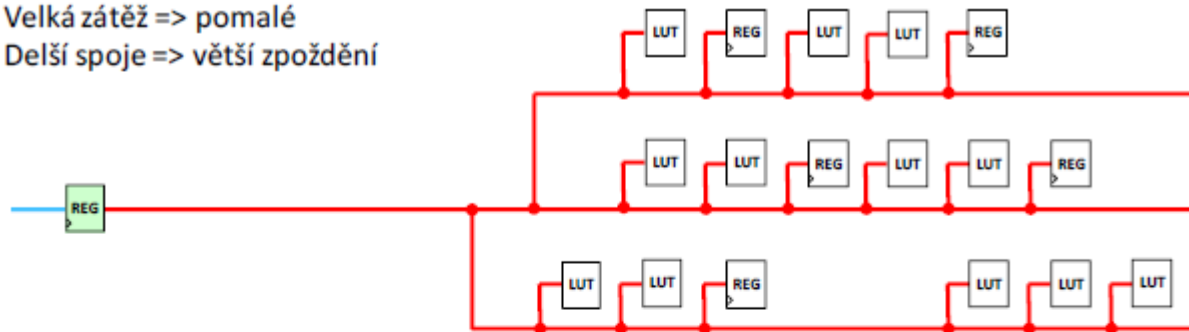
Po synchronizaci



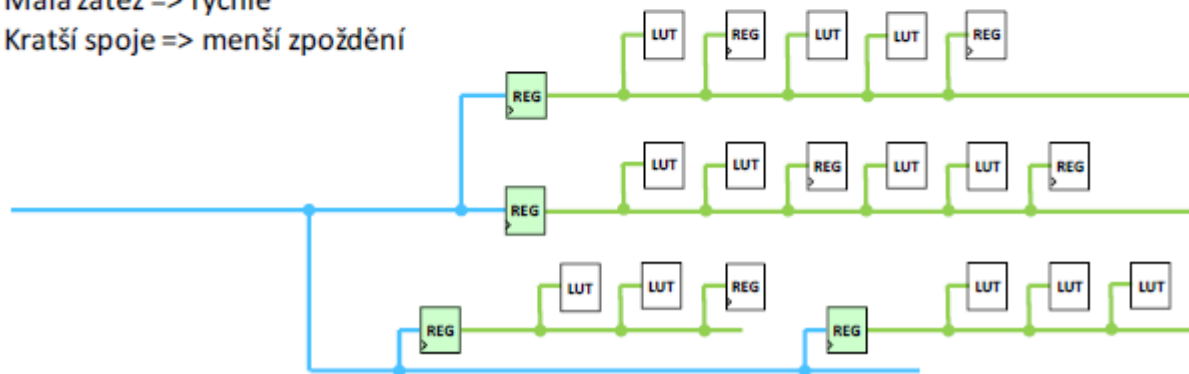
Metoda replikace registrů

Replikace a rozmístění registrů do míst, kde jsou cílové obvody

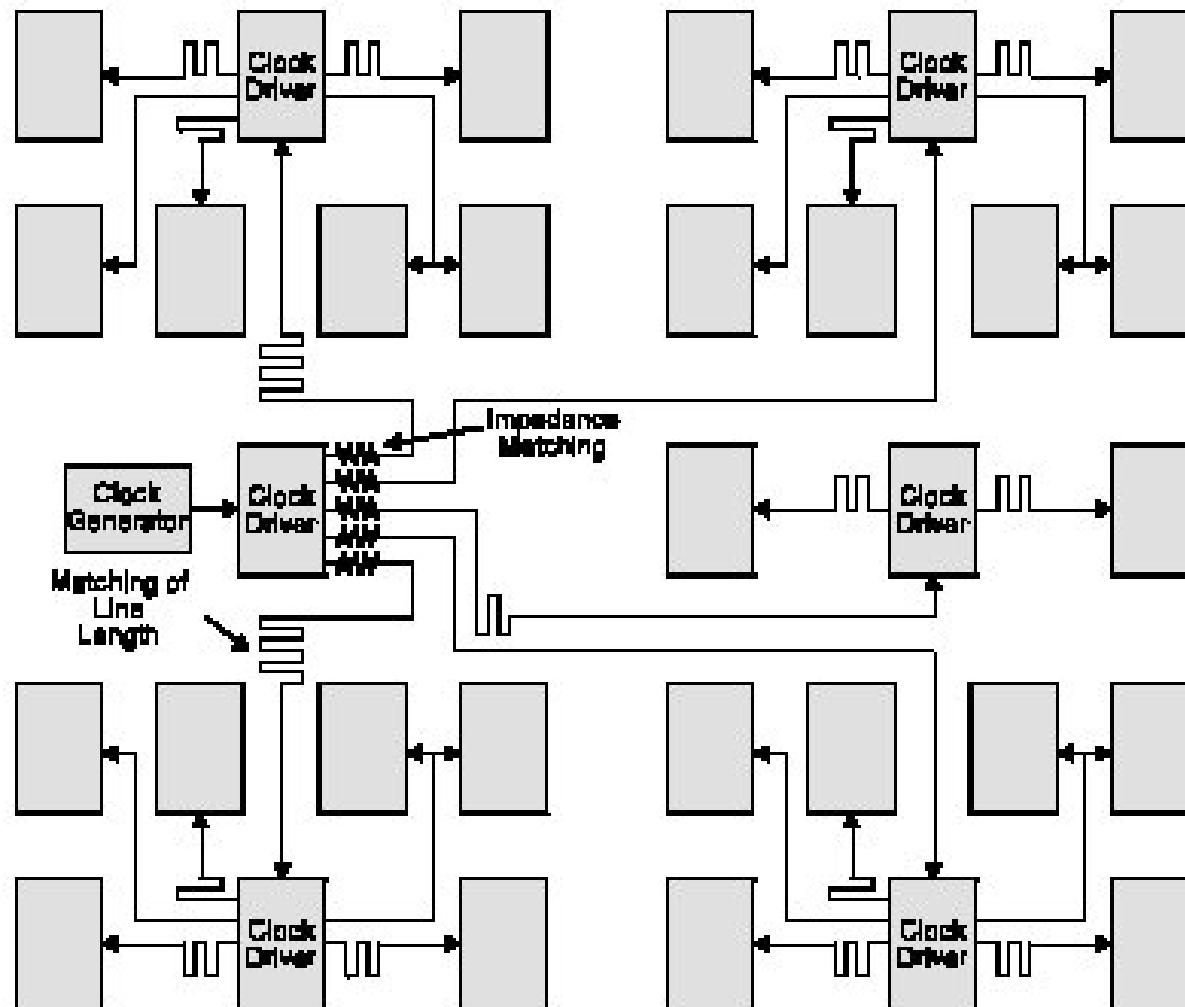
Velká zátěž => pomalé
Delší spoje => větší zpoždění



Malá zátěž => rychlé
Kratší spoje => menší zpoždění



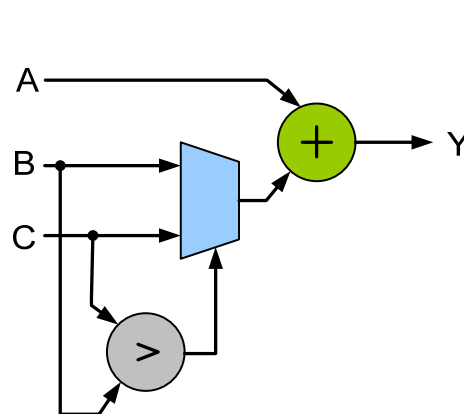
Hodinový signál s budiči



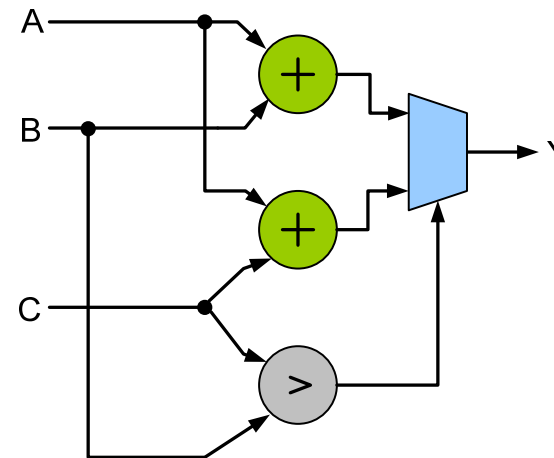
Sdílení prostředků

Sdílení prostředků snižuje množství potřebné logiky,
ale snižuje se také maximální pracovní frekvence

Příklad realizace popisu: `if (B > C)`
`then Y = A + B;`
`else Y = A + C;`
`end if;`



Se sdílením prostředků



Bez sdílení prostředků



Synchronní návrh

Časová doména – část obvodu, jehož registry jsou buzeny stejným hodinovým signálem.

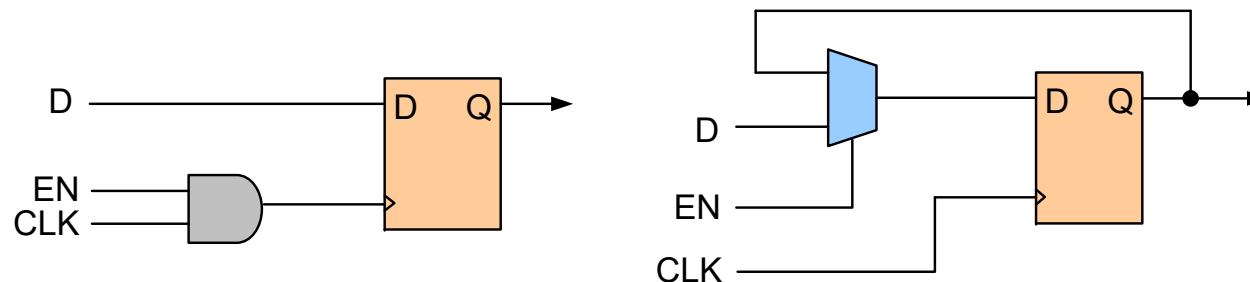
Vše navrhovat synchronně (nejlépe 1 časová doména, příp. důsledně oddělit jednotlivé časové domény)

- podpora v návrhových systémech,
- snadno testovatelné,
- odolnější vůči šumům (přeslechy, odrazy, hazardy, ...),
- funkčnost nezávisí na konkrétním rozmístění a propojení,
- funkční simulace souhlasí se simulací časovou.



Pravidla synchronního návrhu

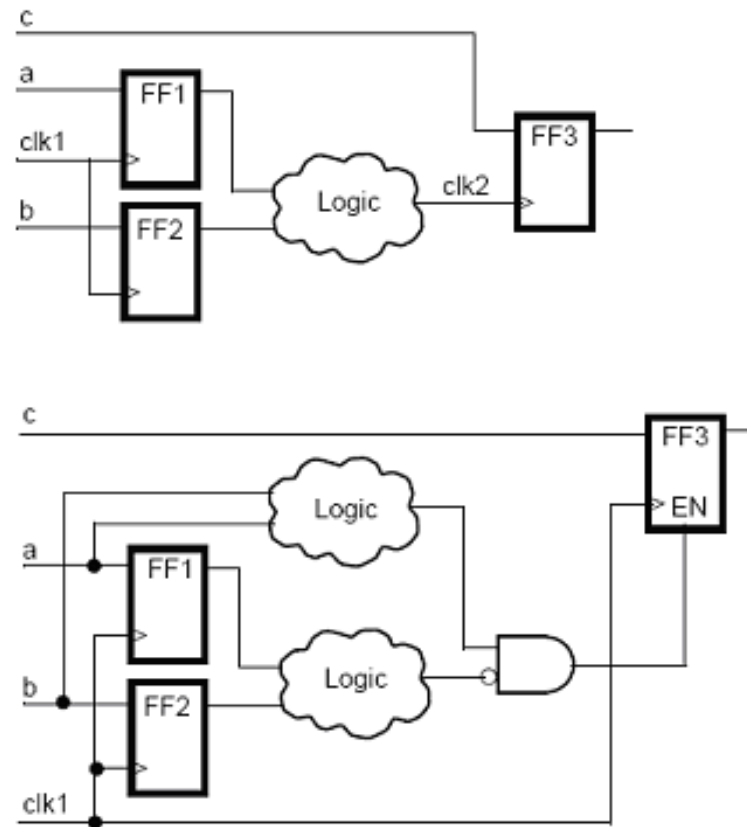
- na hodinové vstupy všech prvků jsou přivedeny pouze hodinové signály (bez přídavné logiky), případné „zastavení“ hodin řešíme klopnými obvody se vstupy „clock enable“;
- všechny klopné obvody jsou hranové řízené (ne hladinově řízené obvody - latche);
- zpětné vazby v kombinačních obvodech se v návrhu nevyskytují (ZV vedou na asynchronní sekvenční logiku);
- asynchronní signály jsou synchronizovány (viz dále).





Jednotné hodinové signály

Výstupy z logiky (přenosy čítačů, výstupy dekodérů a apod.) nejsou použity jako hodinové signály



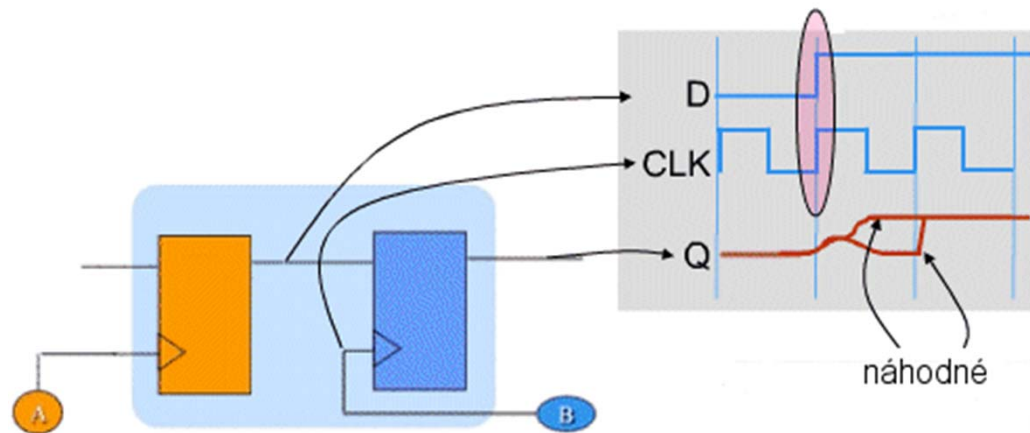


Přenos signálů mezi čas. doménami

Data se na vstupu KO mohou měnit i v době příchodu aktivní hrany hodinového signálu

=> vznikají kritická časová okna, kdy není dodržena doba předstihu t_s (t_{set-up}), doba přesahu t_h (t_{hold}), příp. doba zotavení po resetu t_{rr}

=> na výstupu KO může vzniknout nedefinovaný stav, který mohou následné vstupy vyhodnotit rozdílně.



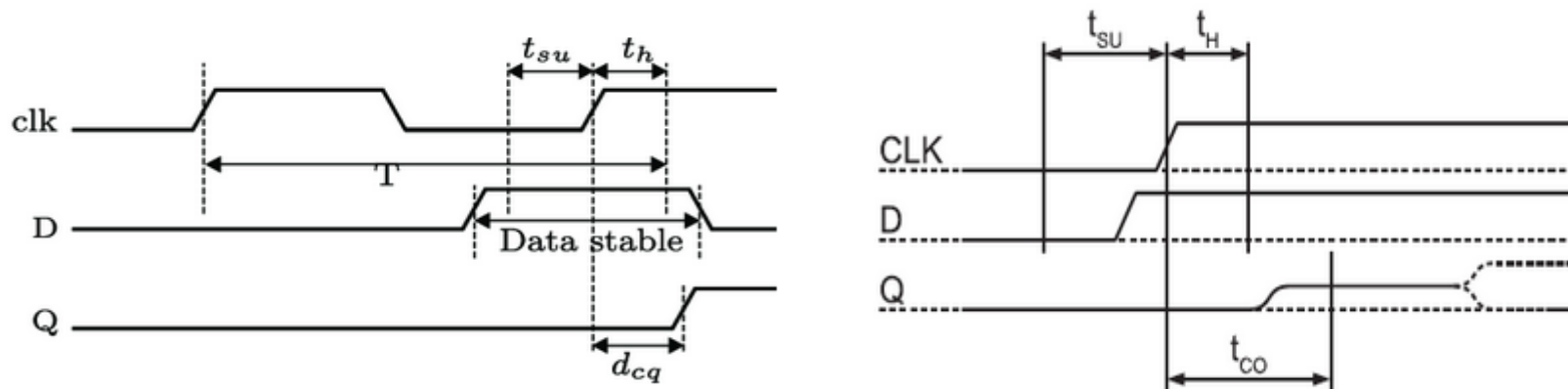


Metastability

Metastabilitou nazýváme neschopnost výstupu registru ustálit se na definované logické úrovni v přesně definovaném čase, obvykle za jednu hodinovou periodu.

Důsledkem metastabilit mohou vznikat:

- proudové špičky na napájení,
- nekorektní přechody mezi stavy stavových automatů,
- nekorektní hodnoty na sběrnicích.

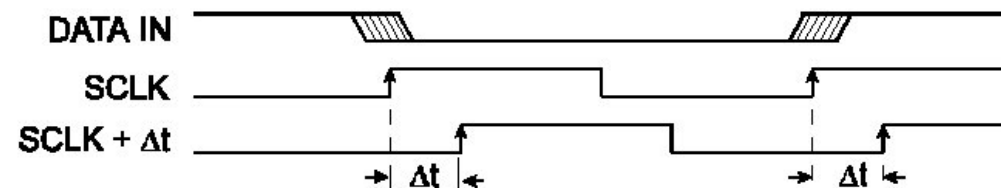




MTBF

MTBF (Mean-time-between-failures) - střední doba bezporuchového provozu (mezi chybami v synchronizaci):

$$MTBF = \frac{e^{T \cdot \Delta t}}{f_{in} \cdot f_{clk} \cdot T_0}$$



T , T_0 ... pravděpodobnosti výskytu chyby charakteristické pro daný typ logického obvodu (souvisí s technologií a architekturou daného obvodu)

f_{in} ... kmitočet vstupního signálu

f_{clk} ... kmitočet synchronizačního signálu

Δt ... doba zpožděného vzorkování výstupu KO (doba pro stabilizaci)



Poruchovost návrhu

Poruchovost celého návrhu Q_C je dána:

$$Q_C = \frac{1}{MTBF_C} = \sum_{i=1}^n \frac{1}{MTBF_i}$$

⇒ časových domén by mělo být co nejméně a $MTBF_i$ by v návrhu měly být všechny srovnatelné.

Příklad:

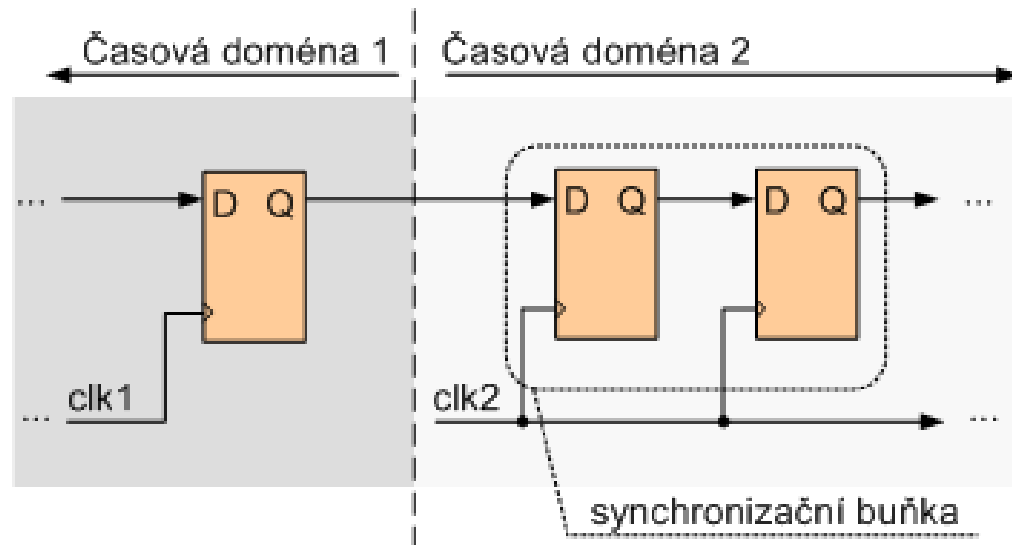
U obvodu Altera Flex10K ($T_0 = 1,01 \cdot 10^{-13}$ s, $T = 1,27 \cdot 10^{10}$ s⁻¹) chceme, aby k chybě došlo jednou za 100 let ($MTBF = 3,2 \cdot 10^9$ s) při $f_{in} = 100$ kHz a $f_{clk} = 10$ MHz. Testování signálu musí být prováděno se zpožděním:

$$\Delta t = \frac{\ln(MTBF \cdot f_{in} \cdot f_{clk} \cdot T_0)}{T} = 2 \text{ ns}$$



Synchronizátory

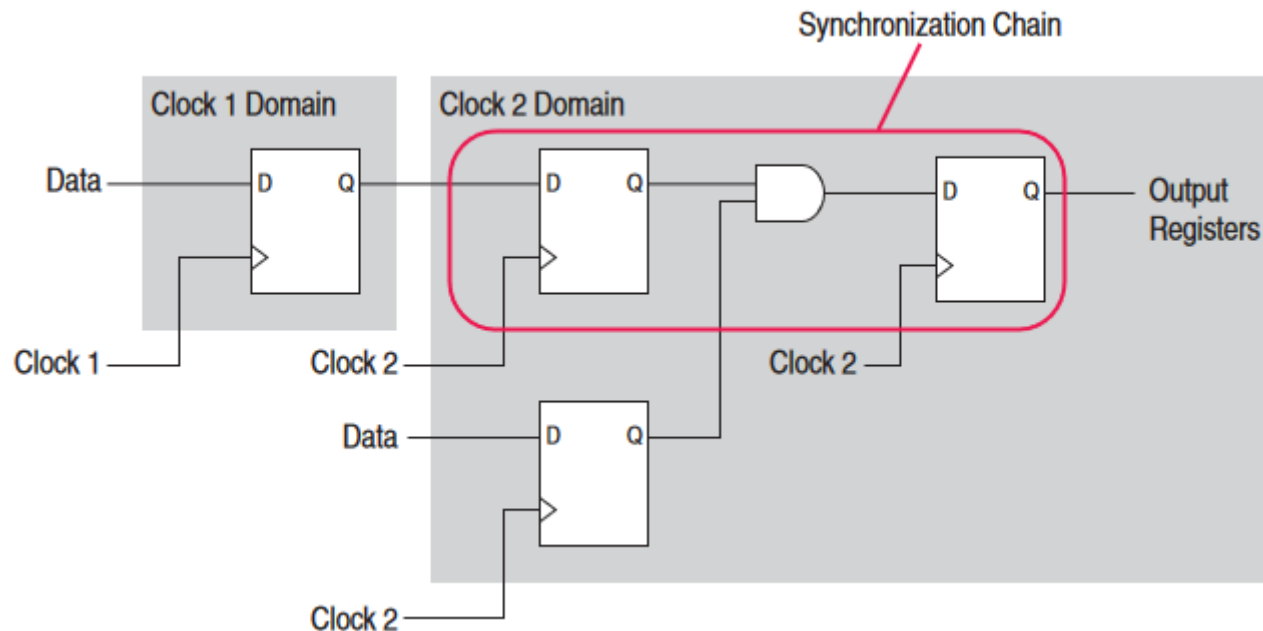
Pro snížení pravděpodobnosti vzniku metastability se používají dvojnásobné (výjimečně i trojnásobné) synchronizátory (synchronizační buňky) – pro jednotlivé signály.





Synchronizátory

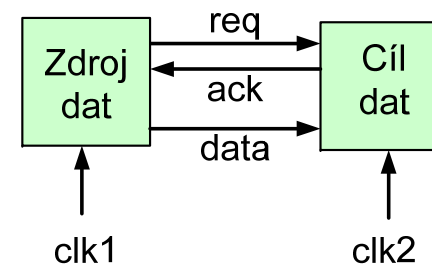
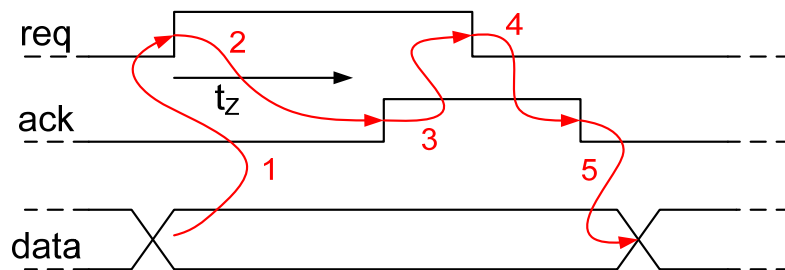
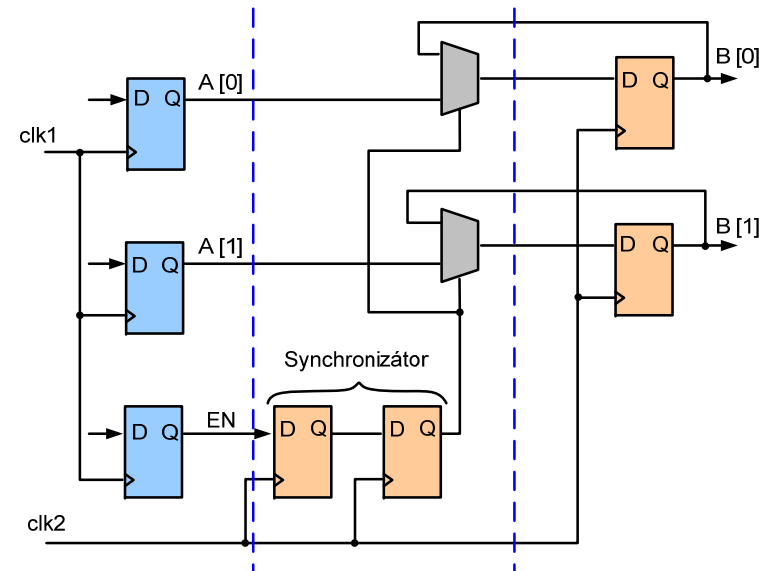
Cesta mezi synchronizačními registry může obsahovat kombinační logiku, pokud jsou všechny registry řetězce ve stejné časové doméně (nevětvit ale signál mezi synchronizátory).



Synchronizace sběrnic

Problém se synchronizací n-bit. sběrnic

- vlivem nestejných zpoždění jednotlivých bitů by mohlo dojít k zápisu nesprávné hodnoty;
- je-li $f_{clk1} \leq f_{clk2}$, lze signálem *EN* potvrzovat platnost dat;
- je-li $f_{clk1} \geq f_{clk2}$, nutno zavést zpětnou vazbu (handshake) – data se nesmí ztratit ani zdublikovat (*req* a *ack* je třeba přesynchronizovat) => pomalejší.

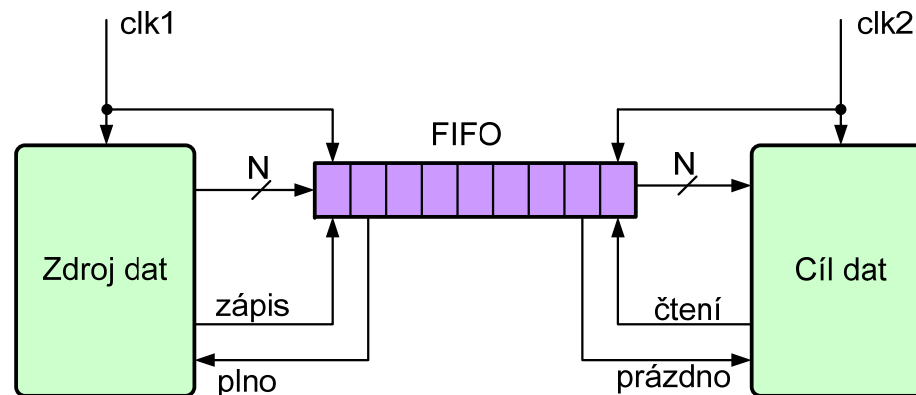




Přenos dat mezi čas. doménami

Pro přenos dat mezi časovými doménami lze použít fronty s využitím paměti FIFO s asynchronními hodinami (dual clock FIFO)

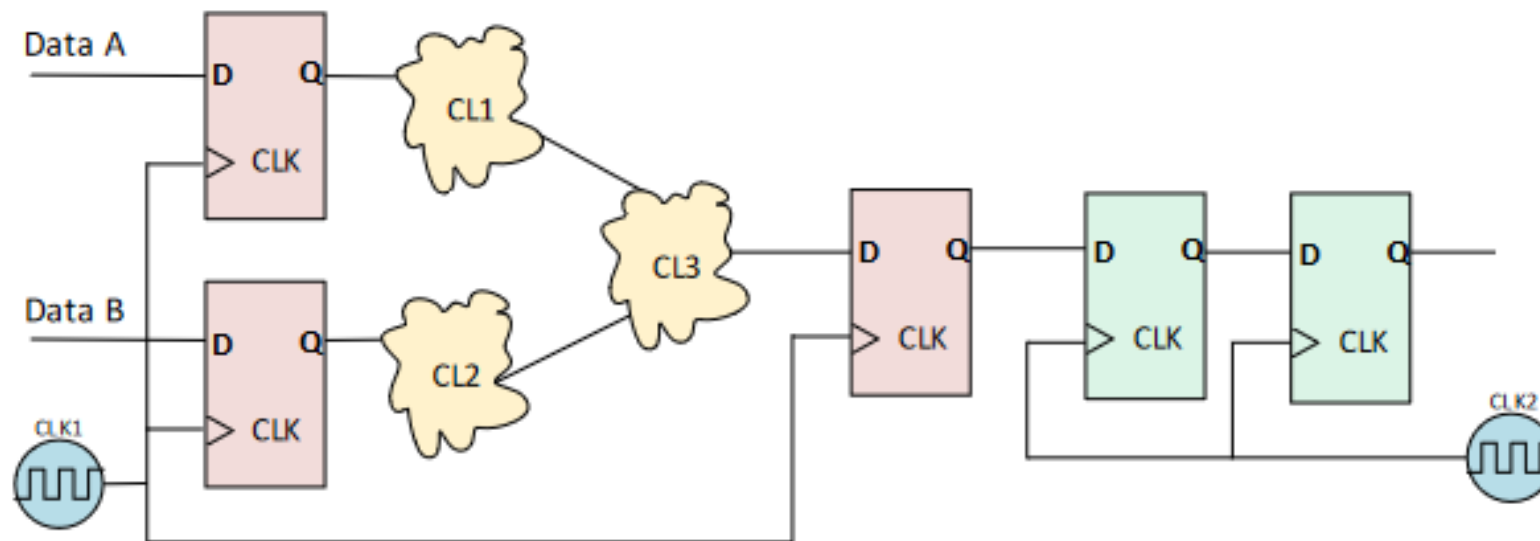
- využíváme zejména v případě, kdy je zpoždění spojené s resynchronizací nepřijatelné





Krátké pulsy v signálu

Velmi krátký puls (glitch) v datovém signálu (v době t_{set-up} nebo t_{hold}) vznikající např. vlivem hazardů, příp. krátký rušivý puls v hodinovém signálu mohou způsobit také metastabilní chování. Proto případnou výstupní kombinační logiku z časové domény doplňujeme KO.

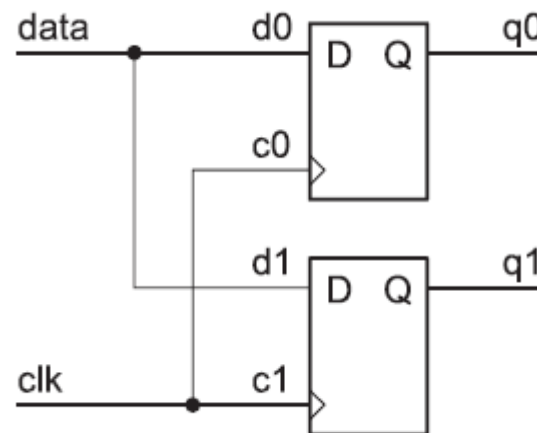




Nekoherence

Nekoherence vzniká pokud přivádíme asynchronní signál do vstupů několika synchronních prvků (klopných obvodů) současně. Zejména v FPGA obvodech může být zpoždění jednotlivých signálů (především datových) dosti rozdílné – to může způsobit rozdílné hodnoty na výstupech (nemusí teoreticky dojít k metastabilitě).

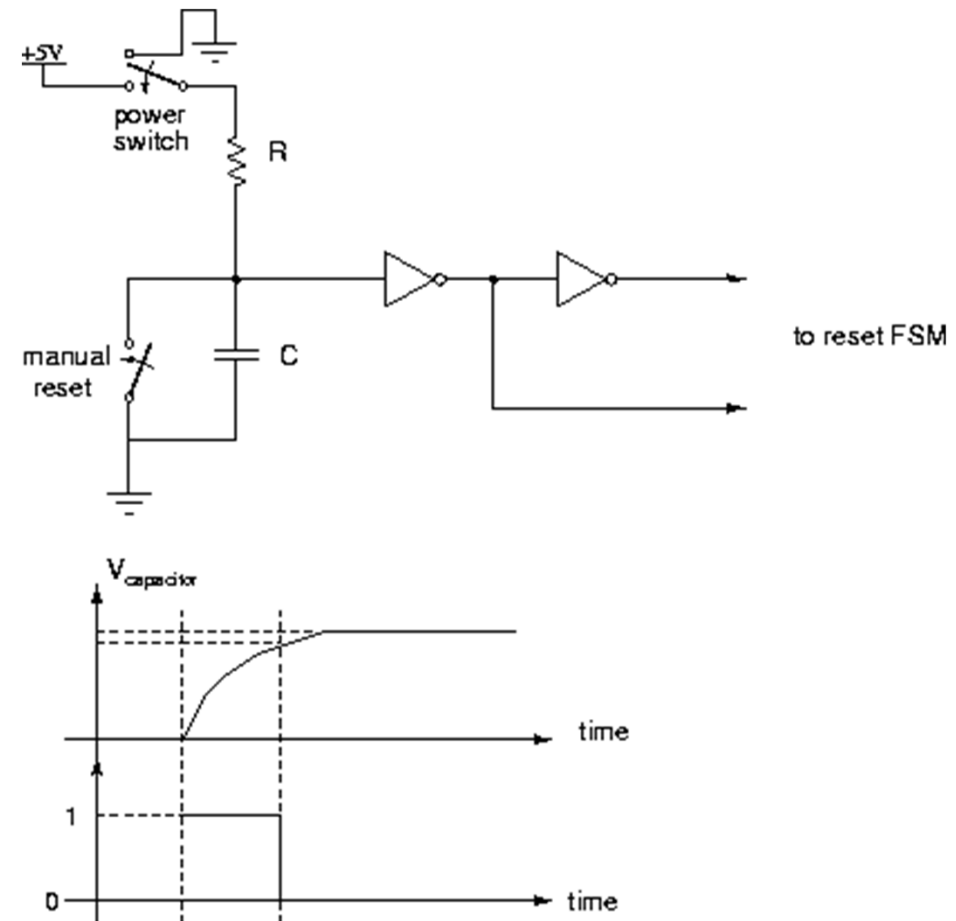
Řešením je opět synchronizace asynchronních signálů.



Inicializace obvodů (globální reset)

Důležité pro počáteční nastavení zejména stavových automatů.

Asynchronní nulování a přednastavení může být použito pouze k nastavení počátečního stavu klopných obvodů (ne v průběhu jejich běžné funkce).

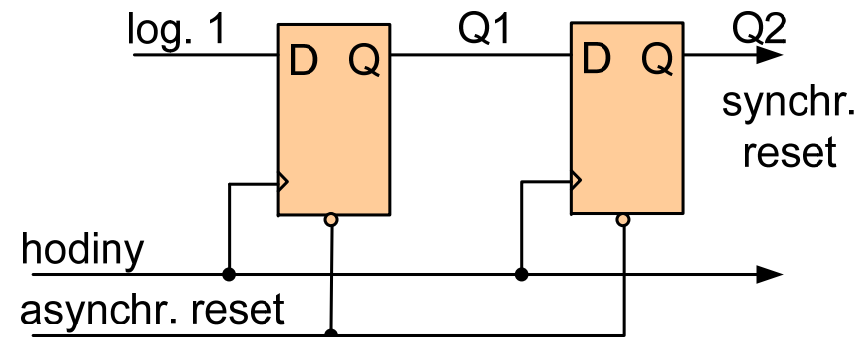




Synchronizace asynchronního resetu

Problém s uvolněním resetu s aktivní hranou hodin (nutno dodržet dobu zotavení po resetu – reset recovery time t_{rr}) – může vést na metastabilní chování;

⇒ reset synchronizer

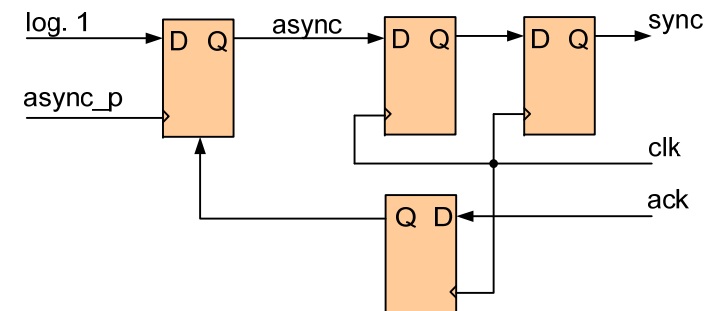
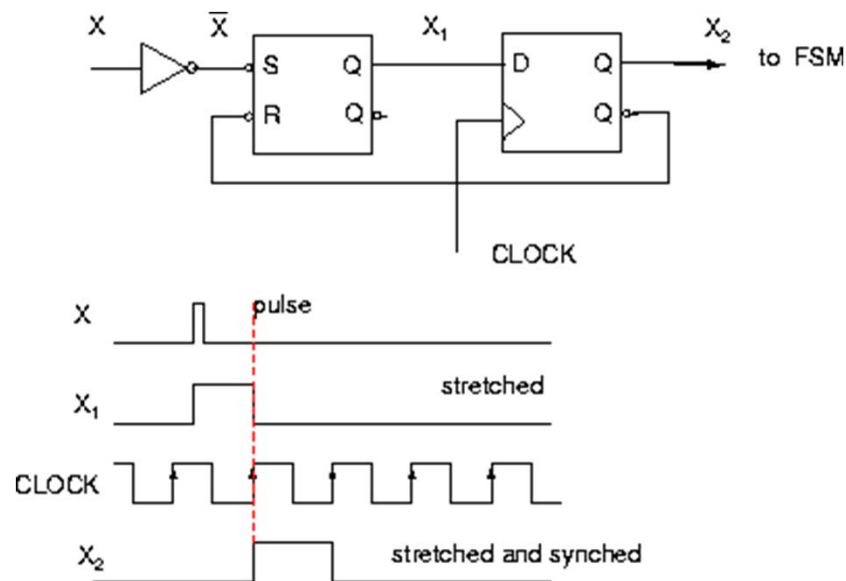


Zastavení hodinového signálu vkládáním logiky není vhodné, lze použít bloky DCM (Digital Clock Manager) či PLL.



Resynchronizace asynchr. vstupů

- synchronizace jednotným hodinovým signálem
- detekce úzkých pulsů



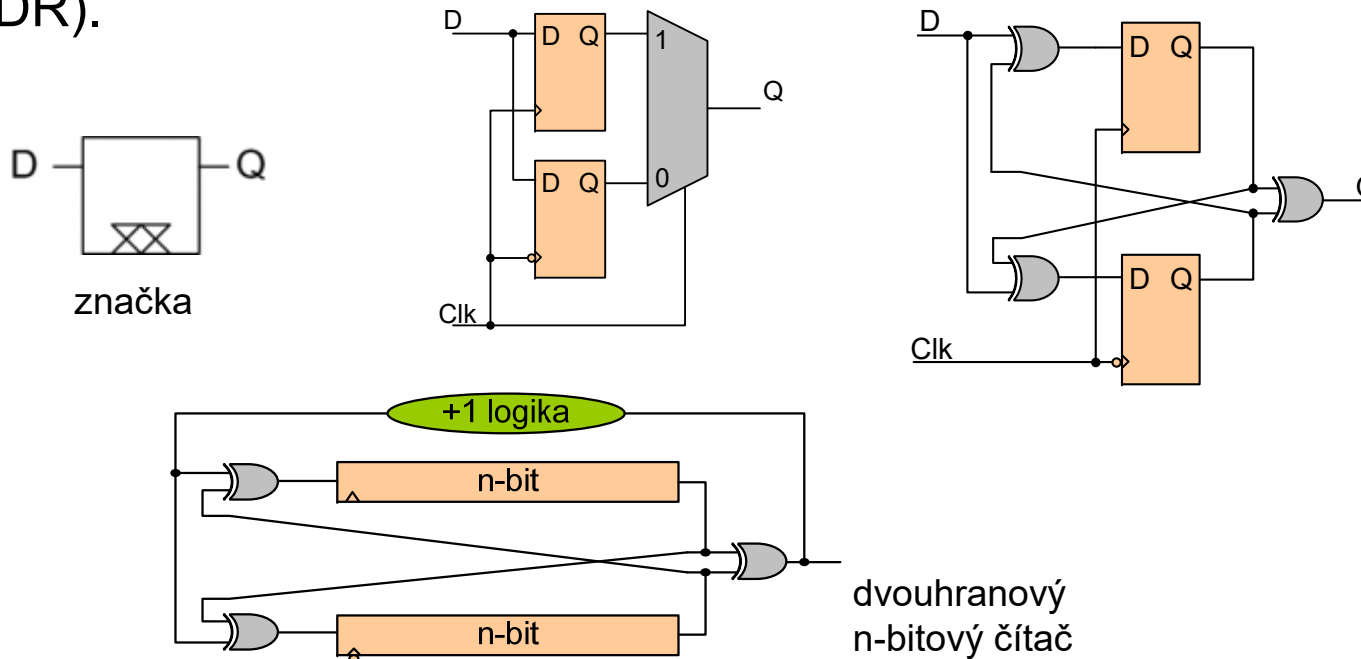
Asynchronní vstupní události nesmí být častější než $4 T_{clk}$

Dvouhranové klopné obvody

V architekturách FPGA jsou většinou jen KO reagující na náběžnou (po negaci *clk* na sestupnou) hranu.

Ani HDL jazyky většinou nepodporují dvouhranové KO.

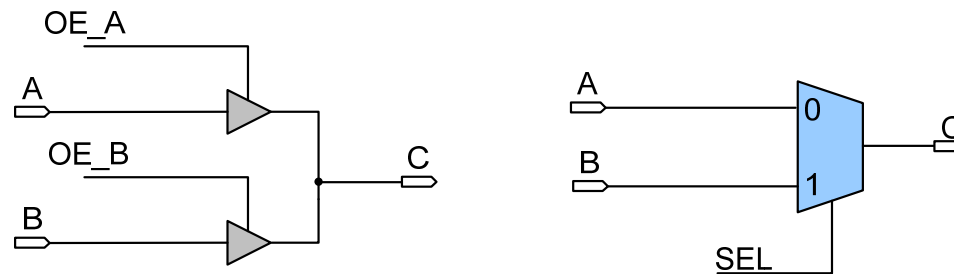
U některých zařízení je třeba reagovat na obě hrany (např. paměti DDR).



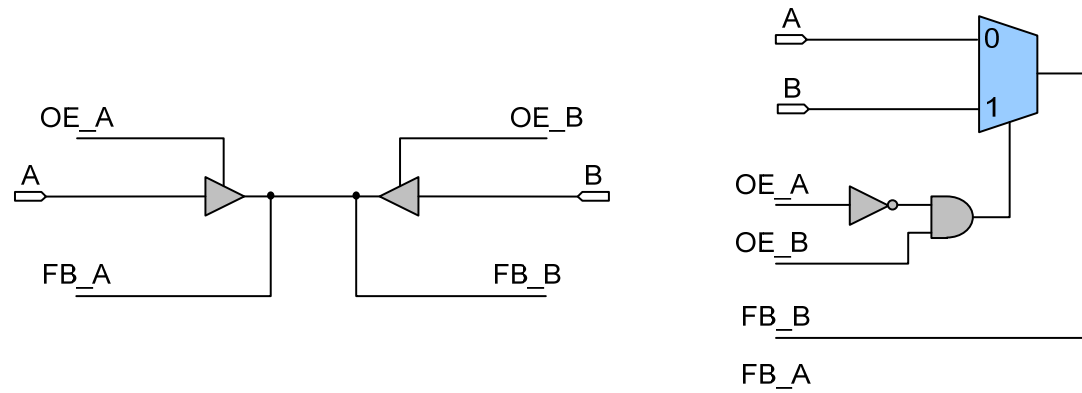
Náhrada sběrnic multiplexorem

Náhrada jednosměrné třístavové sběrnice multiplexorem

- třístavové výstupy jsou pomalejší, někdy uvnitř FPGA nedostupné
- vhodné zejména pro menší počet větvení



Náhrada obousměrné třístavové sběrnice multiplexorem:





Makrobloky, jádra (cores)

Makro je definováno jako část navrhovaného systému, která je použitelná jako samostatný stavební blok.

Používání maker výrazně zkracuje a zlevňuje návrh.

Dva typy makrobloků:

- **soft core** – forma syntetizovatelného RTL kódu;
- **hard core** – výstupem je hotový layout (závislé na technologii),
 - úspora místa na čipu (ve srovnání s realizací v log. buňkách),
 - vyšší pracovní kmitočet,
 - snížení spotřeby bloku (malá plocha, optimalizace),
 - možnost implementovat i nestandardní (např. analogové) bloky,
 - funkce pevně dána, nelze upravovat.



Makrobloky (pokračování)

Základní vlastnosti makrobloků:

- souhlasí s příslušnými normami (standard compliant),
- vesměs verifikované pomocí FPGA obvodů, bloky pro testování,
- parametrické (např. v tabulkách uvedená různá velikost makrobloku pro proměnnou šířku sběrnice),
- integrované, spolehlivé a snadno použitelné,
- dobře zdokumentovatelné pomocí kompletní technické specifikace,
- podpora použití (od výrobce), aj.

IP bloky – navržené makrobloky, často nutné zakoupit licenci (Intellectual Property – intelektuální vlastnictví).

Metodika návrhu s IP bloky

