

## Návrh počítačových systémů 2013: Projekt č. 1

Název: Řízení maticového LED displeje

Odevzdání: dle termínu Projekt 1 v IS FIT

Hodnocení: max. 13 bodů

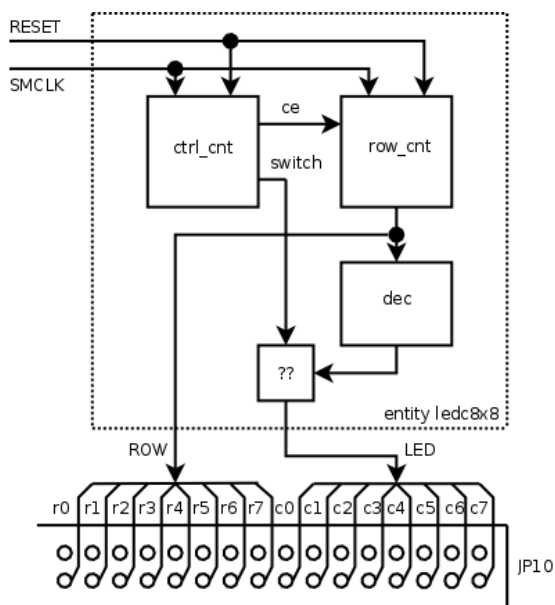
Dotazy: [bidlom@fit.vutbr.cz](mailto:bidlom@fit.vutbr.cz), osobně - M. Bidlo (L330)

### Zadání

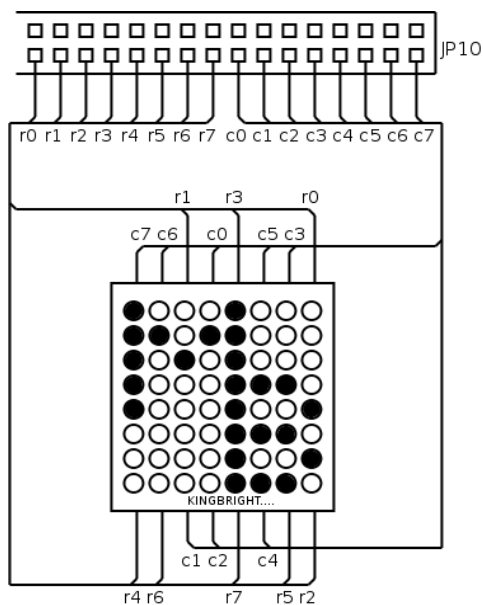
Vytvořte behaviorální model obvodu, který periodicky bliká s iniciály Vašeho jména a příjmení současně zobrazenými na maticovém LED displeji externě připojeném k FPGA na FITkitu. Zvolte si zobrazení každého iniciálu tak, aby byly znaky jasně čitelné. V případě pocitu nedostatku místa na displeji můžete sdílet vybrané linie obou znaků. Schéma obvodu pro FPGA a připojení maticového displeje k FITkitu ukazují obrázky 1 a 2. **Dodržte pojmenování entity a jejího rozhraní, neoznačené vnitřní signály obvodu si vhodně pojmenujte sami.**

Odzkoušení řešení bude možné **prezenčně** v CVT v učebně M105. K dispozici tam bude několik identických přípravků s kompletním zapojením maticového displeje včetně rozhraní s konektorem pro připojení přípravku k FITkitu. Doneste si FITkit, USB kabel a NB s řešením projektu, případně využijte PC v učebně. **S přípravky zacházejte ohleduplně, neměňte jejich zapojení, ani je neodnášejte z učebny. Ačkoliv je možné vypracovat projekt s využitím simulátoru, doporučuji ověřit finální řešení na poskytnutém přípravku, který bude využit i při hodnocení.**

**Odevzdává se: archiv xlogin00.tar nebo xlogin00.zip (nahraďte Vaším loginem), jenž bude obsahovat adresář opět pojmenovaný Vaším loginem malými písmeny a v něm pouze to, co bylo součástí zadání s doplněným řešením v souborech ledc8x8.vhd a ledc8x8.ucf.**



Obr. 1



Obr. 2

## Popis obvodu

(1) 22-bitový synchronní cyklický čítač `ctrl_cnt` generuje povolovací signál `ce` (`clock enable`) pro synchronizaci čítače řádků displeje (`row_cnt`) na kmitočet `SMCLK/256`. Smyslem snížení kmitočtu čítání `row_cnt` je zvýšení svítivosti LED displeje při jeho dynamickém řízení a napájení přímo z pinů sběrnice `X` na FPGA. Dynamickým řízením displeje rozumíme postupnou periodickou aktivaci jednotlivých jeho řádků v čase, kdy je pro každý aktuálně aktivní řádek specifikována sestava svítících LED diod. Signálem `switch`, odvozeným od bitu 21 (MSB) čítače `ctrl_cnt`, zajistíte realizaci efektu blikání zobrazení. Signál `RESET` provede asynchronní vynulování čítače `ctrl_cnt`. `Clock enable` nastavte do `log. 1` při dosažení hodnoty `0xFF` ve spodních 8 bitech `ctr_cnt`, jinak bude `ce` v `log. 0`.

(2) Čítač aktivace řádků displeje (`row_cnt`) představuje synchronní osmibitový rotační registr, který cyklicky aktivuje **právě jeden řádek** displeje. **Aktivace řádku se provádí hodnotou `log. 1` na příslušném bitu tohoto čítače, zbývající bity (neaktivních řádků) musí být nastaveny do `log. 0`!** Signálem `RESET` asynchronně nastavte registr tak, aby byl aktivován první řádek displeje, ostatní řádky jsou neaktivní. Nezapomeňte, že k rotaci hodnoty čítače dochází pouze při `ce` v `log. 1` (`clock enable`), jinak je tento čítač fundamentálně taktován hodinami `SMCLK`.

(3) Dekodér `dec` určuje sestavu rozsvícených LED v aktivním řádku displeje na základě aktuálního stavu čítače `row_cnt`. **Dioda svítí, je-li na příslušné pozici signálu LED `log. 0`, v případě `log. 1` je odpovídající dioda zhasnutá (tj. signál LED je přiváděn na katody diod).** Dekodér představuje čisté kombinační obvod; v části „others“ zajistíte stav „LED zhasnuta“ pro všechny diody v řádku přiřazením signálu LED hodnoty „11111111“.

(4) Dle schématu doimplementujte příslušný obvodový prvek pro realizaci efektu blikání (v obr. 1 označen ??).

(5) Maticový displej LED 8x8 je standardní zobrazovací součástka připojená externě k FITkitu prostřednictvím vybraných pinů sběrnice `X` (viz obr. 2, je již součástí poskytnutého přípravku). Jelikož anody diod v každém řádku displeje sdílejí jediný vodič, je nutné provádět řízení displeje (aktivaci řádků) dynamicky v tzv. časovém multiplexu, což zajišťuje obvod z obr. 1, jehož VHDL popis je předmětem tohoto projektu. Podíváte-li se na displej ze strany, na které je uveden popisek, odpovídá význam a zapojení jeho vývodů schématu na obr. 2 a řádky, resp. sloupce jsou číslovány od 0 do 7 shora dolů, resp. zleva doprava (na obr. 1 a 2 symboly `r0-r7`, resp. `c0-c7`).

## Pokyny k řešení

(1) Doplňte kód dle uvedené specifikace obvodu do souboru `ledc8x8.vhd`.

(2) Doplňte mapování výstupních signálů `ROW` a `LED` do souboru `ledc8x8.ucf` tak, aby tyto signály byly přivedeny na příslušné bity sběrnice `X`, jak je uvedeno na obr. 1. Fyzicky budou na všech verzích FITkitu namapovány na spodní řadu 16-ti pinů zcela vpravo na rozhraní `JP10`. Inspirujte se tím, co je již v tomto souboru uvedeno a použijte schéma FITkitu k určení mapování zbývajících signálů (schémata naleznete na webu <http://merlin.fit.vutbr.cz/FITkit/hardware.html>). Ve schématu `FITKIT-INPUT/OUTPUT INTERFACE` je uvedena struktura rozhraní `JP10`, kde naleznete, které piny sběrnice `X` je třeba použít (viz též obr. 1 a 2) a podle toho pak ze schématu `FITKIT-FPGA INTERFACE` určíte čísla pinů FPGA, na které je třeba v souboru `ledc8x8.ucf` namapovat signály `ROW` a `LED` entity obvodu.

(3) Pro simulaci obvodu a zobrazení časového diagramu byl vytvořen skript `isim.tcl`, který se nachází v adresáři `fpga/sim`. Zadáním `make isim` z příkazového řádku v kořenovém adresáři projektu (nebo volbou *Spustit simulaci (ISIM)* z kontextového menu projektu v QdevKitu) se spustí simulátor ISIM a zobrazí se průběh sledovaných signálů po dobu  $123\ \mu\text{s}$ . Je-li model správně vytvořen, zachytí simulace tři periody postupné aktivace všech řádků displeje (náhled zachycující jen malou část počátku zobrazení iniciálů).

### Upozornění

Pracujte samostatně, nikomu nedávejte práci k opsání. Plagiátorství se hodnotí 0 body a adekvátním postihem dle platného Disciplinárního řádu FIT VUT v Brně.