

Projekt: Přístupový terminál

1. Zadání

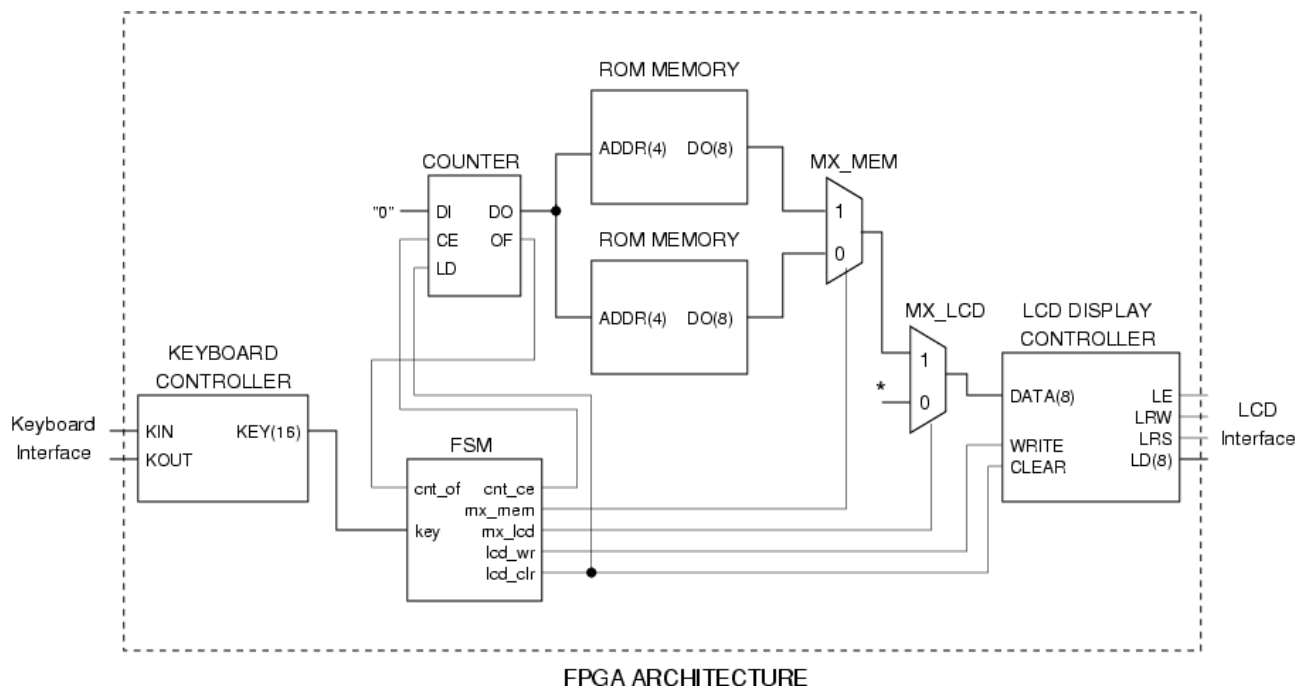
1. Seznamte se s přípravkem FITKit a způsobem připojení jeho periférií, zejména klávesnice a LCD displeje.
2. Prostudujte si zdrojové kódy projektu v jazyce VHDL.
3. Navrhněte řídicí jednotku (konečný automat) jednoduchého přístupového terminálu.
4. Navrženou řídicí jednotku implementujte v jazyce VHDL a ověřte její funkčnost přímo na přípravku FITKit.

2. Architektura přístupového terminálu

Přístupový terminál je jednoduché elektronické zařízení, které povoluje oprávněným uživatelům vstup do chráněných objektů. Obvykle je složeno z klávesnice, LCD displeje a řídicí jednotky. Každý uživatel musí před vstupem do objektu vyťukat na klávesnici přístupového terminálu aktivační kód a na jeho základě získá nebo nezíská přístup do objektu.

Cílem tohoto projektu je realizovat takovýto přístupový terminál na přípravku FITKit, kde je k dispozici klávesnice, LCD display a FPGA hradlové pole připojené k oběma těmto perifériím. Úlohou FPGA čipu bude sledovat vstupy klávesnice, vyhodnocovat zadaný vstupní kód a vypisovat příslušné odezvy na LCD displeji. Architektura FPGA čipu je uvedena na následujícím obrázku.

Obrázek 1. Architektura aplikace uvnitř FPGA



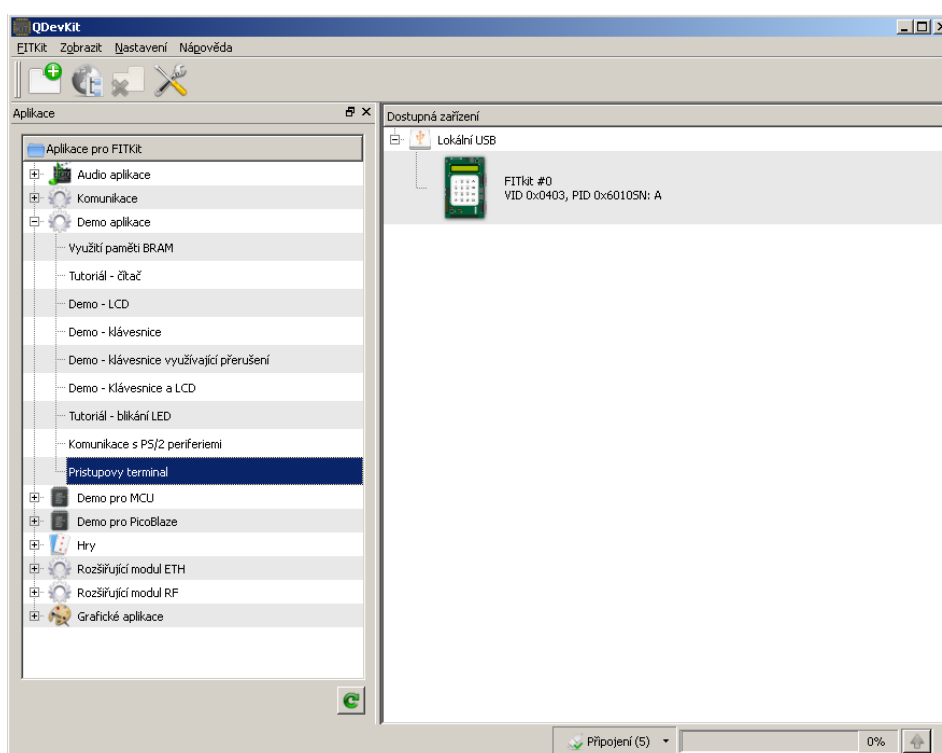
Funkce obvodu je následující:

- Vstupy z klávesnice jsou pravidelně testovány pomocí řadiče klávesnice (Keyboard Controller). Jakmile je detekován stisk některé klávesy, nastaví řadič na svém výstupu KEY jeden z 16-ti signálů odpovídající číslu stisknuté klávesy (signály KEY(0..9) odpovídají klávesám 0..9, signály KEY(10..13) odpovídají klávesám A..D, signál KEY(14) odpovídá klávese "*", signál KEY(15) odpovídá klávese "#").
- Výstupní signály z klávesnice jsou dále připojeny ke konečnému automatu (FSM), který ovládá zbývající části obvodu. Automat sleduje posloupnost stisknutých kláves, v průběhu zadávání kódu vypisuje na LCD displeji znak "*" a po potvrzení kódu klávesou "#" vypíše na displej zprávu "Přístup povolen" nebo "Přístup odepren". Opětovným stisknutím klávesy "#" přechází obvod opět do stavu čekajícího na vstupní kód.
- Textové zprávy "Přístup povolen" resp. "Přístup odepren" jsou uloženy ve dvou paměťových modulech typu ROM. Každý z modulů obsahuje 16 osmi-bitových položek obsahujících jednu ze zpráv.
- V okamžiku potvrzení vstupního kódu klávesou "#", aktivuje automat Clock Enable signál 4-bitového čítače (COUNTER), který je připojen na adresové vstupy paměťových modulů a způsobí tak vyčtení jejich obsahu na výstup.
- Výstupy paměťových modulů jsou dále připojeny na dvou-vstupý multiplexor (MX_MEM). V případě, že byl kód správně zadán, potom automat vybere skrze tento multiplexor zprávu "Přístup povolen", v opačném případě vybere zprávu "Přístup odepren".
- Výstup multiplexoru (MX_MEM) je připojen na další dvou-vstupý multiplexor (MX_LCD). Tento multiplexor přepíná na svůj výstup buď zprávu uloženou v paměťových modulech nebo znak "*" podle toho, zda se přístupový terminál nachází ve stavu zadávání kódu nebo ve stavu vypisujícím výstupní zprávu.
- Výstup multiplexoru (MX_LCD) je připojen na datový vstup řadiče LCD displeje (LCD DISPLAY CONTROLLER) a reprezentuje znaky, které se budou zobrazovat na displeji. Řadič je ovládán pomocí dvou signálů WRITE a CLEAR. Aktivace signálu WRITE způsobí zápis znaku na displeji, zatímco aktivace signálu CLEAR displej vymaže a připraví na zápis nové sekvence znaků.

Architektura FPGA čipu je připravena tak, že vše kromě konečného automatu FSM je již naimplementováno v jazyce VHDL. Cílem toho projektu je proto správně navrhnout a implementovat právě tento automat řídící zbývající části obvodu.

3. Postup práce

1. Seznamte se s přípravkem FITKit a nainstalujte si na svůj počítač potřebné nástroje pro práci s tímto přípravkem (viz informace v sekci [návod](#)). Alternativně můžete pro řešení projektu využít také počítače v centru [CVT](#), kde již jsou potřeby nástroje nainstalovány, popř. si můžete stáhnout a na osobním počítači spustit obraz [virtuálního stroje](#), kde jsou pro Vás rovněž všechny potřebné nástroje připraveny.
2. Stáhněte si aktuální obsah SVN repozitáře FITKitu (viz informace v sekci [SVN](#)).
3. Z informačního systému si stáhněte archiv zdrojových souborů úlohy (*projekt.zip*) a rozbalte si jej do podadresáře `app/demo/terminal/` v SVN repozitáři FITKitu. Po spuštění aplikace QDevKit by se Vám měla objevit v záložce *Demo aplikace* nová položka s názvem *Přístupový terminál* (viz obrázek). Od tohoto okamžiku můžete aplikaci překládat, simulovat, nahrávat do přípravku FITkit a následně spouštět.



4. V informačním systému také naleznete soubor *kod.txt* se seznamem přístupových kódů. Uvnitř tohoto souboru vyhledejte svůj login a u něj naleznete také dva přístupové kódy pro Váš projekt.
5. Prostudujte si zdrojové kódy projektu v jazyce VHDL a způsob zapojení jeho jednotlivých částí.
6. Navrhněte řídicí jednotku (konečný automat) jednoduchého přístupového terminálu. Při návrhu dbejte na následující požadavky:
 - Přístupový terminál musí akceptovat pouze Vaše dva přístupové kódy (tzn. terminál vypíše hlášení „Přístup povolen“, pokud uživatel zadá správně libovolný z obou kódů).
 - Pokud uživatel v průběhu zadávání kódu stiskne špatnou klávesu, nesmí to Váš automat nijak dát najevo, dokud není stisknuta potvrzovací klávesa "#".
 - Identifikujte Mealyho a Moorovy výstupy.
 - Sestavte si graf přechodů automatu.

- Při sestavování grafu přechodu automatu vzniká obecně velké množství přechodů mezi stavy, neboť je potřeba reagovat na všechny možné stisknuté klávesy (celkem 16 možností). Pro zjednodušení a přehlednost prosím využijte v grafu označení typu KEY=X (pro očekávanou klávesu) a KEY\diamondX (pro všechny ostatní případy).
 - Při návrhu respektujte rozhraní automatu (tj. názvy vstupních a výstupních signálů), které je připraveno v souboru *fsm.vhd*. Rovněž zachovejte název proměnných automatu: *present_state* a *next_state*.
 - Nejednodušujte si prosím práci vložím přístupového kódu do proměnné typu pole a jejím cyklickým testováním. Takovéto projekty nebudou hodnoceny.
 - **Příklad automatu v souboru *fsm.vhd* slouží pouze pro inspiraci a v žádném případě nemusí souhlasit s Vámi navrženým automatem.**
7. Navržený konečný automat implementujte v jazyce VHDL a uložte do předpřipraveného souboru *fsm.vhd* v adresáři *fpga*.
 8. Proveďte simulaci VHDL kódu pomocí programu ModelSim. Simulaci lze spustit pomocí nástroje *qdevkit*.
 9. Jakmile je ověřena funkce přístupového terminálu v simulacích, je možné přistoupit k testování funkce přímo na FITKitu. Nejprve proveďte překlad zdrojových souborů do binární podoby (pomocí nástroje *qdevkit*).
 10. Vytvořený binární soubor s příponou **.bin* nahrajte do FPGA čipu (pomocí nástroje *qdevkit*).
 11. Správnou funkci přístupového terminálu ověřte přímo na FITKitu.

4. Výstupy projektu

Výstupem projektu bude:

1. Soubor *fsm.vhd* se zdrojovým kódem konečného automatu.
2. Soubor *zprava.pdf* s výstupní zprávou (ve formátu PDF) obsahující následující informace:
 - Jméno a příjmení, login, přístupové kódy
 - Graf přechodu konečného automatu
 - Seznam výstupů s identifikací, zda se jedná o Mealyho nebo Moorovy výstupy.
 - Ukázku výstupní zprávy naleznete v příloze č. 1.

Rozsah zprávy nesmí překročit jednu stranu formátu A4.

Oba soubory zabalte do archívu s názvem <login>.zip a odevzdejte prostřednictvím informačního systému nejpozději do data uvedeného na privátních stránkách předmětu INC. Pozdější odevzdání projektu nebude bráno v úvahu. Při odevzdání dbejte na správné názvy souborů a všechny tyto názvy uvádějte malými písmeny.

Příloha č. 1: Ukázka výstupní zprávy

Jméno:

Login:

Přístupové kódy:

Vstupní/výstupní signály

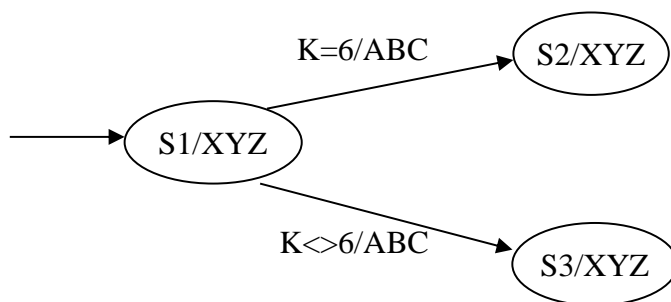
Legenda vstupních signálů:

- K : KEY
- CO : CNT_OF

Identifikace výstupních signálů

- Mealyho výstupy: A,B,C
- Moorovy výstupy: X,Y,Z

Graf přechodů (ukázka)



Poznámka: Za výstupní signály ABC a XYZ dosad'te do grafu přímo hodnoty 0, 1 nebo X (don't care)