

Návrh číslicových systémů (INC): Projekt: Představení

Lukáš Kekely

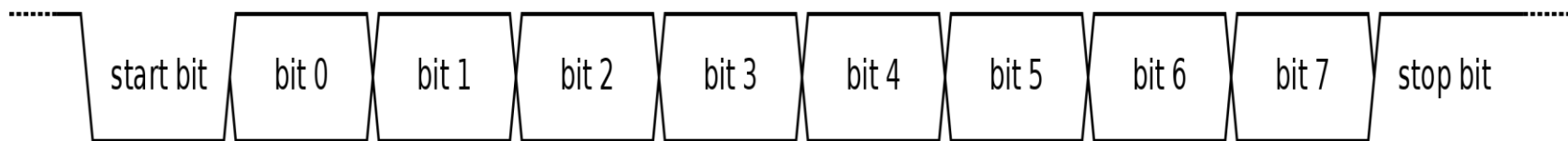
Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2, 612 00 Brno - Královo Pole
ikekely@fit.vutbr.cz



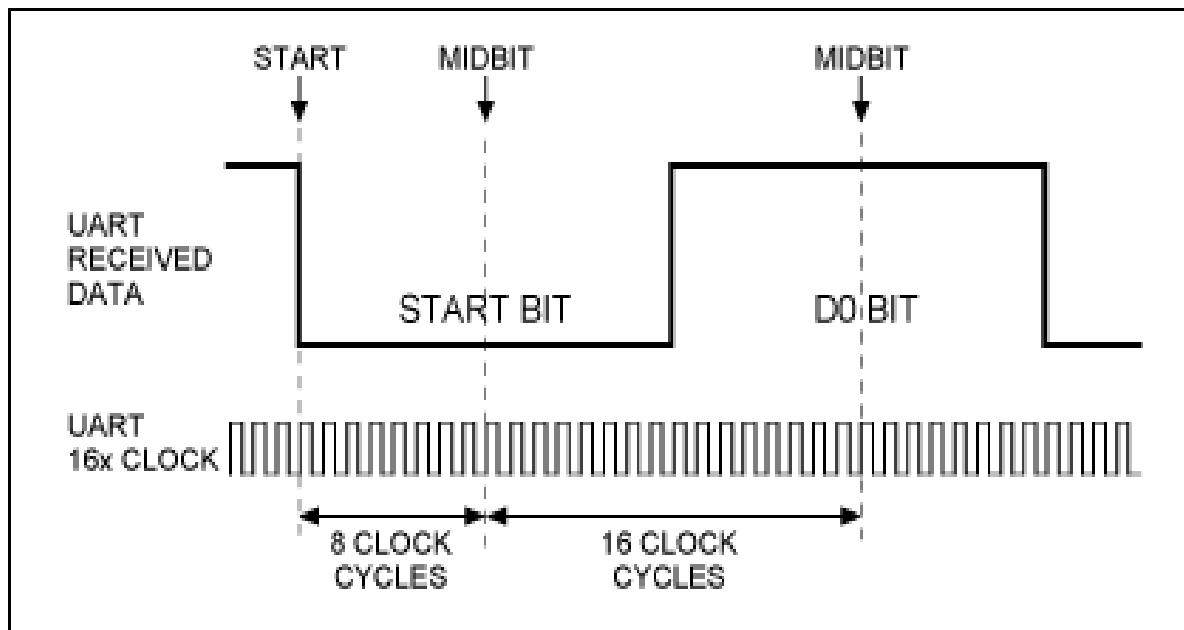
01. 02. 2023

- Získané dovednosti:
 - Vyzkoušet návrh číslicového obvodu (datová a kontrolní cesta)
 - Naučit se číslicové obvody vhodně popisovat v jazyce VHDL
 - Získat zkušenosti s překladem a simulací vytvořených obvodů
- Použitý příklad:
 - Příjem asynchronního komunikačního signálu rozhraní UART (běžně používaný protokol v oblasti vestavěných systémů)
- Formát vypracování:
 - Tři části – 1. návrh, 2. implementace a 3. testování
 - Návrh „na papír“, volitelné odevzdání RTL a FSM schémat
 - **Povinná implementace v jazyce VHDL a její dokumentace**
 - Dobrovolné testování implementace na reálném hardware

- Protokol na přenos vícebitového datového slova po vodiči
 - neaktivní stav – log. 1
 - START bit – log. 0 (přechodem 1\0 začíná přenos)
 - DATA – n-bitů od LSB po MSB
 - 1-2 STOP bity – log. 1
- Příklad konkrétního UART přenosu:
 - START bit, 8 bitů dat, 1x STOP bit

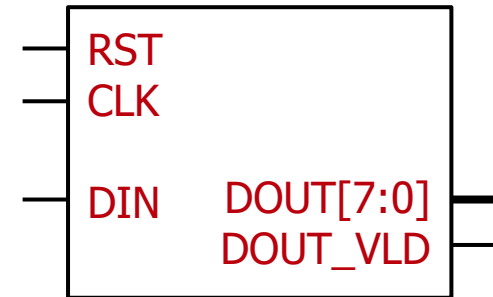


- Metoda pro příjem vstupních dat z UART vodiče
 - Výchozí hodnota neaktivní linky – vodič v log. 1
 - Čeká se na START bit – moment přechodu vodiče do log. 0
 - Obě strany znají přenosovou rychlost (např. 9600 baudů)
 - Pro vzorkování vodiče je použit 16x rychlejší hodinový signál
 - Hodnota se vzorkuje uprostřed intervalu přenosu daného bitu

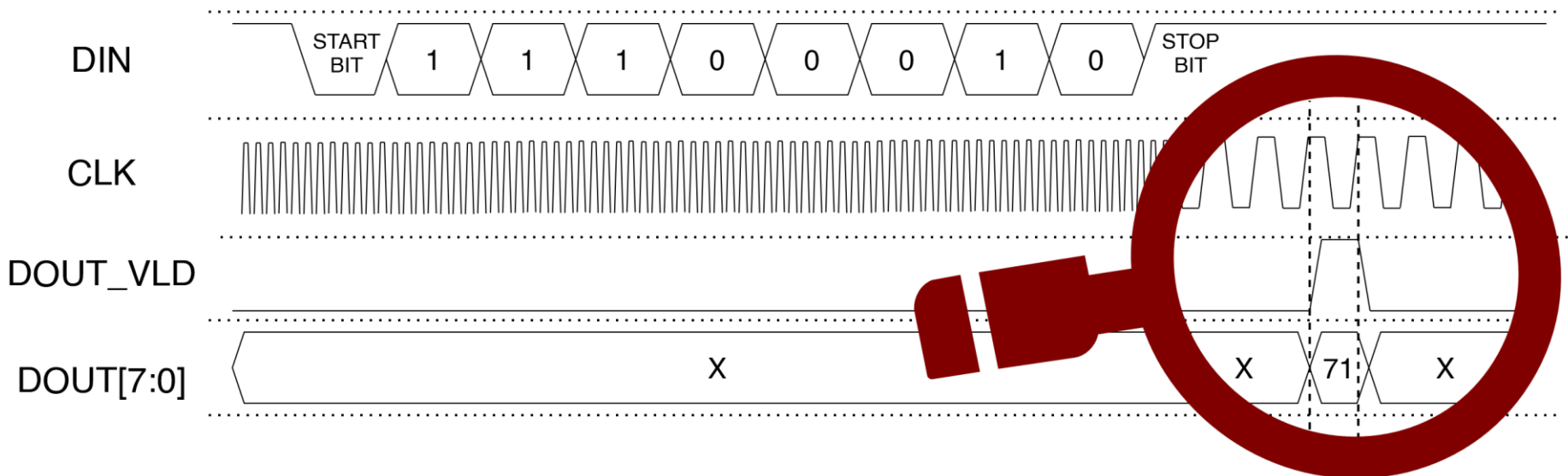


- **DIN** – vstupní vodič asynchronní sériové UART komunikace, slova v pevném formátu:
1x START bit '0', 8x DATA bit, 1x STOP bit '1',
konstantní přenosová (baudová) rychlost,
16x rychlejší CLK a vzorkování uprostřed

Schématická značka



- **DOUT[7:0]** – deserializovaný 8-bitový datový výstup
- **DOUT_VLD** – potvrzení platnosti dat, v log. 1 na jeden takt hodin



- Přesné informace dostupné ve VUT IS u předmětu INC
 - na eLearning (Moodle) stránce předmětu je sekce Projekt
 - texty zadání uloženy v datovém skladu a odkazovány
 - hlavní termín v IS pro odevzdání druhé části (implementace)
 - přihlašování na volitelnou první a třetí část (návrh a testování)
- Přehled bodování a odevzdání:
 - **První část:** extra 1b, deadline cca. konec března, volitelné odevzdání návrhu obvodu a získání zpětné vazby
 - **Druhá část:** max. 20b, získat min. 5b pro zápočet, **deadline cca. konec dubna**, implementace a dokumentace obvodu
 - **Třetí část:** extra 2b, možné samostatně nebo na cvičení, na konci semestru budou vypsány termíny

1. Nastudujte specifika fungování UART komunikace
2. Uvažujte specifický formát vstupních a výstupních dat
 - jeden START bit (log. 0), 8 bitů dat, jeden STOP bit (log. 1)
 - rychlost 9600 baudů za sekundu, 16x vyšší frekvence CLK
 - snímání bitů uprostřed přenášeného intervalu
 - paralelní 8 bitový výstup, potvrzení log. 1 v jednom taktu
3. Navrhněte obvod na RTL úrovni včetně řídicího automatu

Dobrovolně:

4. Vytvořte technickou zprávu s vaším návrhem obsahující:
 - RTL schéma architektury obvodu a její stručný popis
 - Nákres grafu přechodů automatu a jeho stručný popis
5. Odevzdejte do IS zprávu ve formátu PDF jako ***zprava.pdf***

1. Stažení virtuálního stroje nebo lokální instalace nástrojů
2. Stažení ZIP archivu se šablonou zdrojových souborů
 - *uart_rx.vhd* – VHDL definice rozhraní a prázdné architektury
 - *uart_rx_fsm.vhd* – prázdná VHDL komponenta pro FSM
 - *testbench.vhd* – soubor pro základní test funkčnosti obvodu
 - *uart.sh*, *wave.tcl* – pomocné soubory pro překládání a simulaci
3. Navržený obvod implementujte v jazyce VHDL
4. Proveďte syntézu a simulaci vašeho kódu, ověřte jeho správnou funkčnost a odstraňte nalezené chyby
5. Doplňte a aktualizujete technickou zprávu projektu
6. Odevzdejte vaše ***uart_rx.vhd***, ***uart_rx_fsm.vhd***, ***zprava.pdf*** soubory v ZIP archivu ***xlogin00.zip***

- Zadání 1. a 2. části obsahuje ukázkou formátu a obsahu zpráv

Příloha: Výstupní zpráva (Ukázka)

Jméno:
Login:

Architektura navrženého obvodu (na úrovni RTL)

Schéma obvodu

Poznámky:

- Pro přehlednost CLK a RST signály ve schématech uvádíme, ale nemusíme zapojovat.
- Několik jednotlivých D-KO můžete pro přehlednost spojit do jednoho vícebitového registru, pokud tedy sdílí všechny kontrolní signály jako CLK, RST, nebo CE.
- Jednotlivé vodiče můžete spojit do vícebitových sběrnic.

Popis funkce

Stručný slovní popis struktury a funkce obvodu (max. polovina strany A4).

Návrh automatu (Finite State Machine)

Schéma automatu

Legenda:

- Stavy automatu: S1, S2, S3
- Vstupní signály: X, Y, Z
- Mělyho výstupy: A, B, C
- Moorovy výstupy: P, Q, S

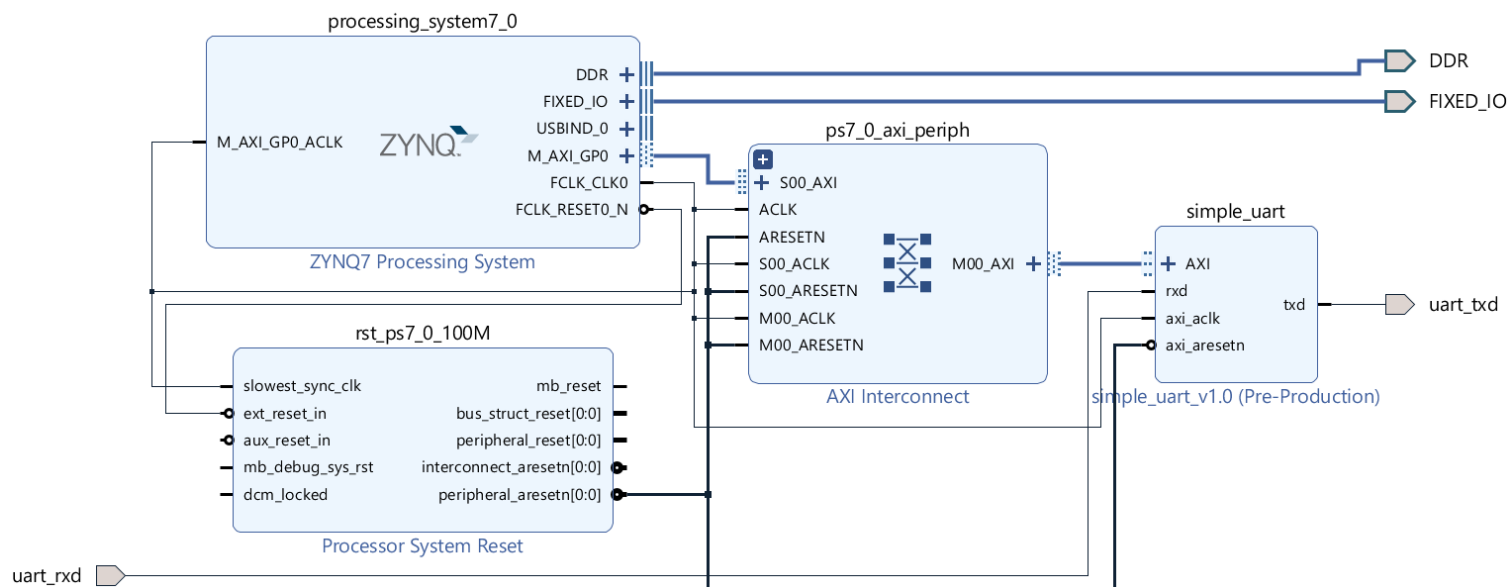
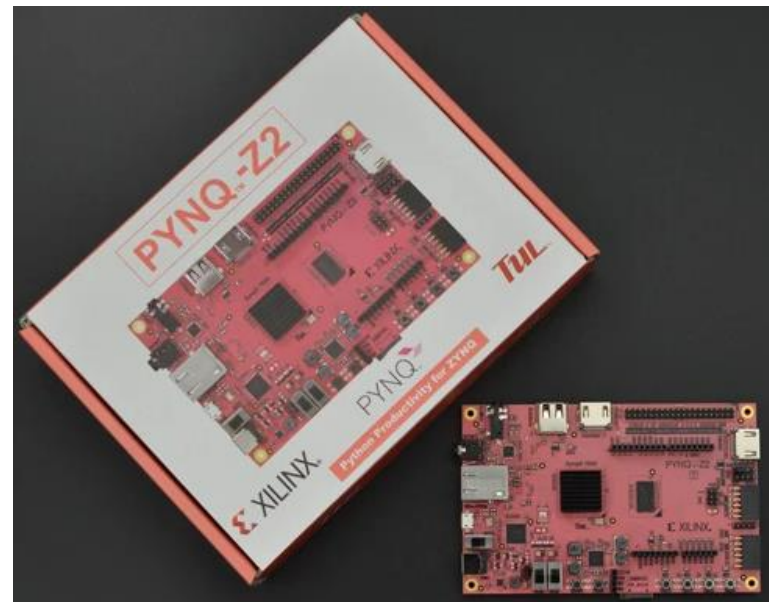
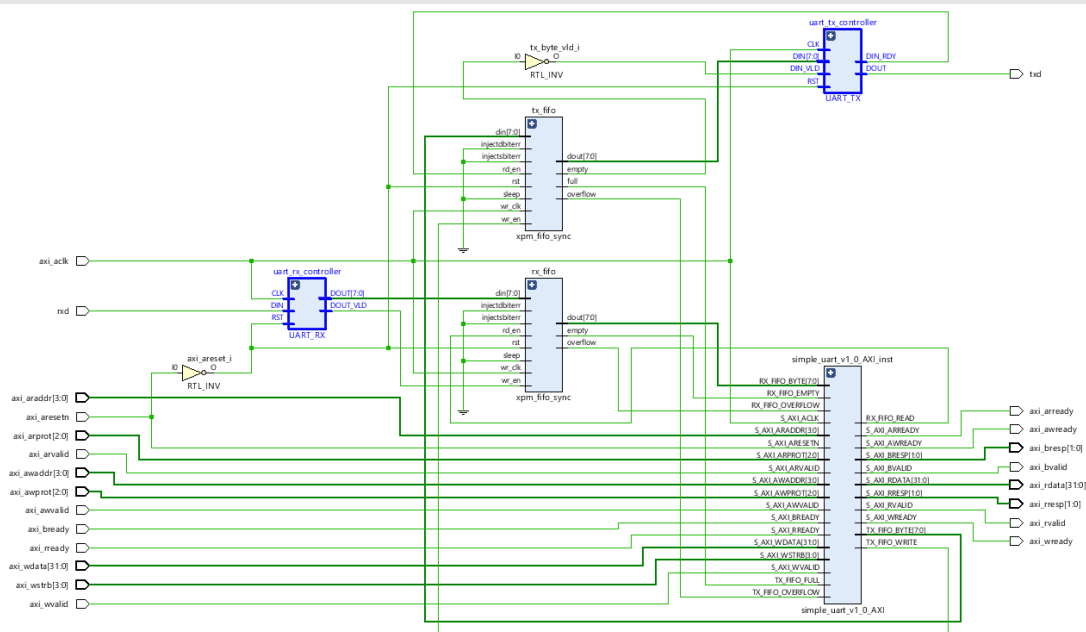
Poznámky:

- Pečujte vhodné názvy pro stavy, vstupní a výstupní signály tak, aby byl snáz pochopitelný jejich význam.
- Za vstupní/výstupní signály XYZ, ABC a PQS dosadíte do grafu přímo hodnoty 0, 1 nebo X (don't care).
- Signály CLK a RST neuvádíme mezi vstupy automatu ani je nekreslíme do schémat.
- Automat může vhodné kombinovat jak Mělyho, tak Moorovy výstupy.
- Pokud je vstupním signálem vektor bitů, můžete s ním na hranách pracovat jako s vektorem.
- Připomeňte si konvence pro kreslení grafu automatu probírány na přednáškách.
- Nezapomeňte na označení počátečního stavu automatu.

Popis funkce

Stručný slovní popis funkce automatu (max. polovina strany A4).

- Před odevzdáním implementace si její funkčnost otestujte skriptem z archivu **test.zip**, podrobný návod v **README**
- Důležité upozornění:** Plagiátorství nebo nedovolená spolupráce vede na neudělením zápočtu z předmětu INC, případně získáte také předvoláním před disciplinární komisí



- Praktická **ukázka návrhu** obdobného číslicového obvodu
 - na **cvičení za týden** navrhnu a nakreslím RTL schéma podobně jednoduchého obvodu a jeho řídicí konečný automat
 - upozorním na formát a potřebné náležitosti těchto schémat
- **Ukázka implementace** obdobného obvodu v jazyce VHDL
 - na **cvičení za cca. měsíc** naimplementuju v jazyce VHDL obvod podle RTL a FSM schémat, předvedu také jeho simulaci
 - upozorním na vhodný postup a pravidla psaní HDL kódu
- Zvolený obvod: **vysílací strana UART protokolu**
 - realizace serializace datových bajtů na jednobitový vodič
 - správný formát přenosů (START bit, 8 bitů dat, STOP bit)
 - zajištění časování vysílaných bitů (každý bit držet 16 taktů)

Děkuji za pozornost!