

Systemy komputerowe: architektura i oprogramowanie (SYKOM)

Ćwiczenie 3

Politechnika Warszawska, Instytut Telekomunikacji

Prowadzący: Aleksander Pruszkowski

Organizacja ćwiczenia:

- Osoba dla której przygotowano ten dokument: Tłomak Andrzej Marian
- Data wygenerowania dokumentu: 13 grudnia 2022

Przed przystąpieniem do realizacji ćwiczenia należy:

- Podobnie jak w laboratorium 1, pobrać plik z rozszerzeniem OVPN z serwera WWW używając adresu <https://resrepo.tele.pw.edu.pl> i otrzymanych poprzez email danych do zalogowania się do tego serwera WWW.
- Zarezerwować sobie tzw. wirtualny komputer za pomocą serwera WWW i adresu <http://zsutresv.tele.pw.edu.pl/ResourceReservation>.
- Przeczytać ze zrozumieniem dokument SYKOM_lab3.pdf.
- Proszę pamiętać aby zaraz po zalogowaniu się do przydzielonej przez system rezerwacji maszyny wirtualnej należy dokonać klonowania treści przydzielonego Tobie zdalnego repozytorium GIT (a pod koniec czasu rezerwacji dokonać wypchnięcia danych na przydzielone Tobie zdalne repozytorium GIT).

Oczekiwane wyniki pracy: Zgodnie z opisem SYKT_lab3.pdf proszę napisać w "czystym języku" C (czyli Bare-Metal), program testujący działanie komponentu gpioemu (komponent ma być zapisany w języku Verilog - jego treść powinna być utworzona i przetestowana w ramach zajęć lab.2). Na bazie tego komponentu należy zbudować nowy procesor RISC-V zgodnie z instrukcją podaną w SYKT_lab3.pdf.

Dodatkowe informacje opisujące różnice budowy komponentu gpioemu względem zadania w laboratorium 2.: *SYKT_GPIO_ADDR_SPACE*: adres ustalony na podstawie konfiguracji wewnętrznej emulatora QEMU do odkrycia przez narzędzie DTS (zgodnie z procedurą poznaną w ramach laboratorium 1). Adres ten jest globalnym offsetem w 32 bitowej przestrzeni pamięciowej CPU a w której umieszczono podprzestrzeń używaną do komunikacji z modułem gpioemu (o implementacji zgodnej z zadaniem dla lab.2), gdzie poszczególne elementy mają 16-bitowy offset względem początku tej podprzestrzeni:

SYKT_GPIO_SENSOR1_ADDR: 0xF8

SYKT_GPIO_SENSOR2_ADDR: 0xFC

SYKT_GPIO_COUNTER_ADDR: 0x100 (licznik)

SYKT_GPIO_STATUS_ADDR: 0x104 (samodzielnie zaproponuj znaczenia bitów w tym porcie).

Gdybyś zauważył, że w specyfikacji zadania dla lab.2 zamieszczono dodatkowy port nie wyspecyfikowany powyżej - dodaj go samodzielnie i zaproponuj jego lokalizację w opisywanej tu podprzestrzeni.

Dla potrzeb testowania utworzonego narzędziem makeQemuGpioEmu nowego CPU, proszę zaimplementować aplikację w języku C a w niej funkcje: *void my_putchar(unsigned char *s)* która argument wskazany w s wypisuje na konsoli, *void my_ltoa32(unsigned long n, unsigned char *buffer, int max_size)* która wartość w n konwertuje do postaci tekstowej którą zapisuje w zmiennej buffer sprawdzając czy ta postać nie jest dłuższa od max_size znaków (dla dłuższej postaci do buffer ma wpisać wyłącznie litery E), konwersja ta ma być wykonywana formacie: HEX.

Dodatkowo w wyniku działań komponentu GpioEmu na ekranie gdzie uruchomiono program narzędziowy GpioEmuConsole w polu *Gpio wyjsciowe* tego programu powinny pojawiać się informacje z licznika. Zakładamy, że po przekroczeniu wartości 255 będzie liczył dalej ale wartości pokazywane będą zgodne z tylko 8 najmłodszymi bitami.

Zawartość raportu: Proszę zamieścić na przydzielone Tobie zdalnym repozytorium GIT wszystkie elementy źródłowe utworzone lub zmodyfikowane dla potrzeb tego zadania, tam też umieścić a raport. Raport ten ma zawierać również: zacytowany opis przydzielonego zadania dla laboratorium 2., logi i ilustracje (tutaj to tzw. screen'y) ukazujące poprawność działania tego systemu - proszę nie wklejać całych kodów źródłowych do tego raportu. Raport ten proszę utworzyć w dowolnym edytorze tekstowym, ale po jego przygotowaniu należy go skonwertować do formatu PDF. Żadne inne formaty dokumentów elektornicznych np.: DOC, DOCX, ... nie będą przyjmowane. Fianlnie raport i wszelkie pliki źródłowe (nie wynikowe!) będące wynikiem prac nad tym laboratorium proszę umieścić na przydzielonym Tobie indywidualnym repozytorium GIT w jego katalogu lab3 - z tego miejsca (i tylko z tego) prowadzący będzie pobierał te pliki do późniejszego ocenienia i wystawienia oceny z tego laboratorium. Proszę pamiętać, że w tym laboratorium nie są oceniane testy plików verilogowych, budowany jest system z nowym CPU na bazie plików verilogowych oraz tworzone jest oprogramowanie w C i te elementy będą oceniane, nie kopiuj zatem testów z laboratorium 2.