Paweł Sajewicz 241314 28.05.2020

Wojciech Śliwa …

Układy cyfrowe i systemy wbudowane 2

- projekt

Temat:  
Obsługa akcelerometru i wyświetlacza LCD  
 na płycie Startan3E

Prowadzący: Dr inż. Jarosław Sugier

**Spis treści**

1. Wstęp 3
   1. Cel i zakres projektu 3
   2. Opis sprzętu 3
   3. Podstawowe informacje 3
2. **Wstęp**
   1. **Cel i zakres projektu**

Zaprojektowanie układu cyfrowego na płytę Spartan3E odczytującego pomiary z akcelerometru i wyświetlającego je na wyświetlaczu LCD.

* 1. **Opis sprzętu**

Rodzina programowalnych macierzy bramek Spartan-3E (FPGA)[1] została specjalnie zaprojektowana w celu zaspokojenia potrzeb dużych, wrażliwych na koszty aplikacji elektronicznych dla konsumentów. W porównaniu do poprzedniej rodziny (Spartan-3) Spartan-3E cechuje się większą ilością logiki na I/O, znacznie zmniejszając koszt na komórkę logiczną. Nowe funkcje poprawiają wydajność systemu i zmniejszają koszty konfiguracji. Te ulepszenia Spartan-3E FPGA, w połączeniu z zaawansowaną technologią 90 nm, zapewniają większą funkcjonalność i przepustowość.

Płyta Spartan-3E[2] pozwala na obsługę układów FPGA. Jest wyposażona w układ programowalny CPLD firmy Xilinx oraz moduł XC3S500E z wyprowadzeniami I/O, złącza portu JTAG, generator kwarcowy sygnału zegarowego, diody LED, klawisze, wyświetlacz LCD i wiele innych elementów.

ADXL345[3] to niewielki akcelerometr, czyli czujnik do pomiaru przyspieszeń w trzech osiach, z wysokiej rozdzielczości (13-bitów) pomiarem w zakresie ± 16 g. Cyfrowe dane wyjściowe są dostępne poprzez interfejs cyfrowy SPI (3- lub 4-przewodowy) lub I2C. Urządzenie mierzy przyspieszenie statyczne grawitacji, a także dynamiczne przyspieszenie wynikające z ruchu lub uderzenia. Jego wysoka rozdzielczość (3,9 mg / LSB) umożliwia pomiar zmian nachylenia mniejszych niż 1,0 °.

* 1. **Podstawowe informacje**

Projekt wykorzystuje szeregową, multi-master-multi-slave magistralę I²C[4], do przesyłu danych z i do akcelerometru.

Z CS połączonym wysoko do VDD I/O, ADXL345 znajduje się w trybie I2C, wymagającym prostego 2-przewodowego podłączenia. Przy spełnieniu odpowiednich parametrów obsługiwane są tryby przesyłania danych: standardowy (100 kHz) i szybki (400 kHz). Obsługiwane są jedno- lub wielo-bajtowe operacje odczytu i zapisu danych. Przy wysokim pinie ALT ADDRESS 7-bitowy adres I2C dla urządzenia to 0x1D, poprzedzający bit R/W. Przekłada się to na 0x3A dla zapisu i 0x3B dla odczytu. Alternatywny adres I2C, 0x53 (poprzedzający bit R/W) można wybrać poprzez uziemienie styku ALT ADDRESS (Pin 12), co przekłada się na 0xA6 dla zapisu i 0xA7 dla odczytu.

Przez brak wewnętrznych rezystorów dla nieużywanych styków, nie ma domyślnego stanu dla styku CS lub ALT ADDRESS, jeśli pozostaną swobodne lub niepodłączone. Dlatego też podczas korzystania z I2C wymagane jest, aby pin CS był podłączony do VDD I/O, a pin ALT ADDRESS do VDD I/O lub GND.

Kluczowe dla działania akcelerometru są rejestry danych, przedstawione w tabeli 1[5]. W projekcie korzystamy z kilku z nich: ID urządzenia – adres 0x00; kontrola funkcji oszczędzania energii – adres 0x2D; kontrola włączania przerwań – adres 0x2E; kontrola szybkości transmisji i trybu zasilania, odpowiadająca za prędkość przesyłania kolejnych pomiarów – adres 0x2C i adres, pod którym zaczynają się rejestry do zapisywania wartości pomiarów – 0x32.

Obraz zawierający zrzut ekranu

Opis wygenerowany automatycznie

Tabela 1. Mapa rejestrówakcelerometru.

1. **Przedstawienie układu**
   1. **Struktura ogólna**
   2. **Opis modułów**
2. **Implementacja**
   1. **Raporty**
   2. **Podręcznik użytkownika urządzenia**
3. **Podsumowanie**
   1. **Ocena krytyczna efektu**
   2. **Ocena pracy**

Umiarkowane problemy sprawiło zaprojektowanie modułu obsługującego akcelero­metr. Pierwszy raz korzystaliśmy z protokołu I2C, więc musieliśmy spędzić trochę czasu na czytaniu dokumentacji, by przyswoić niezbędną wiedzę.

Moduł akcelerometru posiada wiele sygnałów, zarówno wejściowych jak i wyjścio­wych. Konieczne jest również korzystanie z rejestrów, w których urządzenie zapisuje dane. W efekcie maszyna stanów musiała być skomplikowana i zaprojektowanie jej pochłonęło dużo czasu.

Nie obyło się również bez drobnych błędów. Układ sterujący terminalem do wyświetla­nia tekstu, modyfikuje wartość otrzymanego pomiaru, aby otrzymać odpowiedni znak ASCII. Początkowo to rozwiązanie nie działało poprawnie i układ zwracał niewłaściwe znaki. Rozwiązaniem okazało się dodanie do test bench’a instrukcji case, która zwraca odpowiedni znak w zależności od wartości parametru.

* 1. **Możliwy kierunek rozbudowy układu**

Pierwotnie projekt miał być prostą grą, polegającą na kierowaniu samochodem i unikaniu przeszkód. Dokładnie w tym kierunku można rozwinąć układ. Akcelerometr może służyć do sterowania pojazdem. Niestety moduł do wyświetlania tekstu prawdopodobnie nie znajdzie zastosowania i będzie go trzeba zastąpić modułem do wyświetlania grafiki za pomocą VGA.

1. **Literatura**

[1] „Spartan-3E FPGA Family Data Sheet”, <https://www.xilinx.com/support/documentation/data_sheets/ds312.pdf>

[2] „Spartan-3E FPGA Starter Kit Board User Guide”, <https://www.xilinx.com/support/documentation/boards_and_kits/ug230.pdf>

[3] „ADXL345 Data Sheet”, <https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf>

[4] „ADXL345 Data Sheet”, s. 18, <https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf>

[5] „ADXL345 Data Sheet”, s. 23, tabela 19, <https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf>

[6] Moduł „I2C\_Master”, dr inż. Jarosław Sugier, <http://www.zsk.ict.pwr.wroc.pl/zsk_ftp/fpga/#_Toc479592727>

[7] Moduł „VGAtxt48x20”, dr inż. Jarosław Sugier, <http://www.zsk.ict.pwr.wroc.pl/zsk_ftp/fpga/#_Toc479592716>