

Knock Knock

Systemy Mikroprocesorowe II

Projekt końcowy

Agnieszka Lupierz, Adrian Barnaś Elektronika III rok

1. Założenia projektowe

1.1. Wstęp

Projekt realizowany jest w ramach laboratorium projektowego przedmiotu Systemy Mikroprocesorowe II.

1.2. Opis urządzenia

Urządzenie będzie wzorowane na urządzeniu KNOCKI. Zadaniem systemu będzie detekcja stuknięć przez akcelerometry, Interpretowanie ich i w zależności od wykrytej sekwencji wykonanie jednej z zaprogramowanych akcji. Do detekcji stuknięć wykorzystamy układ MPU6050, komunikujący się z mikrokontrolerem poprzez magistralę I²C. Wykorzystanym przez nas mikrokontrolerem będzie Kinetis L na płycie NXP Freedom KL46Z.

1.3. Logika działania

Przygotowane mamy dwie koncepcje logiki działania urządzenia. Preferowanym sposobem wykrywania stuknięć będzie generowanie przerwań kiedy zostanie wykryte przyspieszenie przez układ MPU6050 następnie odczytanie danych z akcelerometrów i sprawdzenie czy wykryty ruch to rzeczywiście stuknięcie. Zaletą takiego podejścia do problemu byłoby ograniczenie zużycia energii przez procesor który mógłby być uśpiony jeśli rządzenie pozostawało by bez ruchu. Trudne może być określenie progu zadziałania programu. Drugim podejściem byłoby ciągła analiza pobieranych z czujnika informacji i rozpoznawanie kiedy wykrytym przez urządzenie ruchem jest wstrząs podłoża spowodowany stuknięciem.

1.4. Faza prototypu

Docelowo zamierzamy stworzyć prototyp urządzenia, który wykryte sekwencje będzie sygnalizował zmianą stanu na wyjściu kontrolera(np. różne diody dla różnych sekwencji).

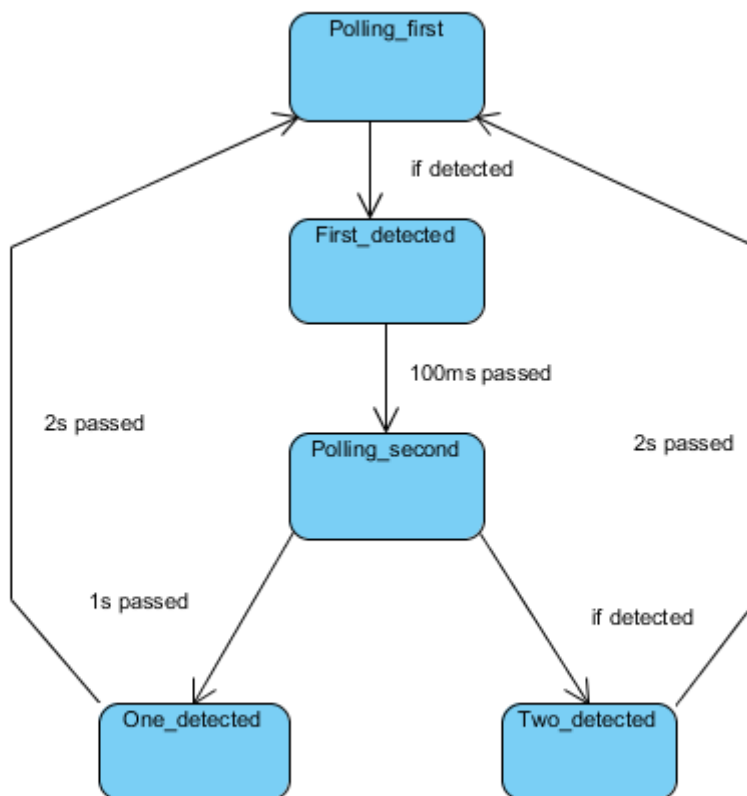
2. Realizacja projektu

2.1. Wykorzystane narzędzia

Do realizacji projektu wykorzystaliśmy środowisko programistyczne Keil uVision. Program był napisany w całości w C++, a kontrola wersji była wykonywana z użyciem narzędzia GIT. Repozytorium git było zdalnie umieszczone na serwerze serwisu GitHub gdzie wspólnie umieszczaliśmy wprowadzane zmiany. Do stworzenia dokumentacji wykorzystaliśmy program WORD z pakietu MS Office, do stworzenia diagramu stanów narzędzie Visual Paradigm, do wizualizacji danych środowiska MATLAB.

2.2. Struktura programu

Program został zrealizowany jako maszyna stanów. Maszyna przyjmowała jeden z pięciu stanów zgodnie z logiką zamieszczoną na diagramie:

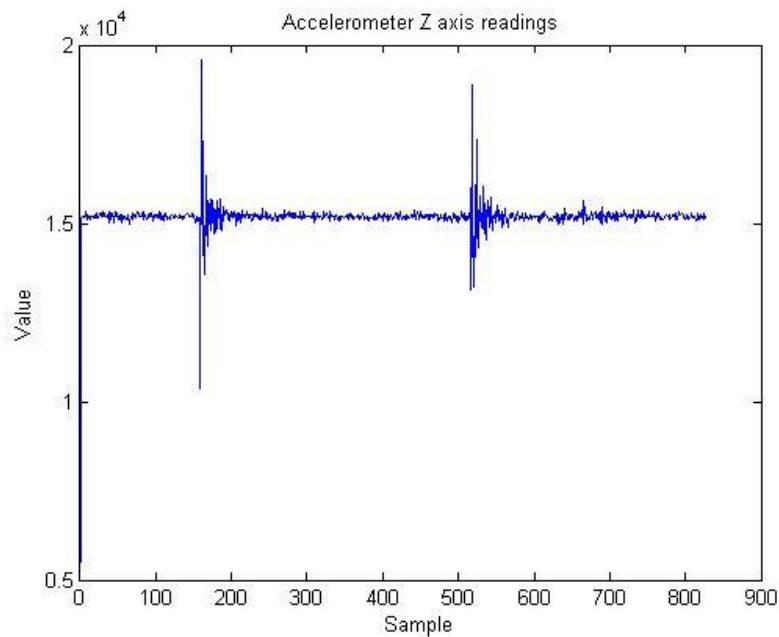


- W stanie Polling_first odpytujemy czujnik o surowe dane przyspieszenia w osi Z czujnika i porównujemy z zadany m poziomem odniesienia a  do przekroczenia tego poziomu (detected)
- W stanie First_detected sygnalizujemy wykrycie pierwszego stukni cia oraz oczekujemy na ustanie drga .
- W stanie Poling_second zn w odpytujemy czujnik przez maksimum 1000ms i je li w tym czasie zn w wykryjemy uderzenie to nast puje „detected”
- W stanach One_detected oraz Two_detected programujemy akcje odpowiednie dla ka dej z sekwencji. Stan trwa 2000ms.

Do realizacji obs ugi czasu w programie u yli my licznika systemowego SysTick, do komunikacji z akcelerometrem u yli my spr towego I²C, do wy wietlania wyników u ywali my diod i wy wietlacza slcd, u ywali my r wnie  UART w celu analizy danych wyj sciowych z akcelerometru.

2.3. Wykrywanie stukni c

Wykrywanie stukni c odbywa si  przez por wnanie odpytywanych warto ci przyspieszenia w osi 'z' z ustalonym progiem zadzia ania. Na potrzeby wyznaczenia progu zadzia ania zebrali my surowe dane akcelerometru, wys lali my na komputer z pomoc  UART i wykre ili my ich wykres w programie MATLAB:



Wykres przedstawia dwa stuknięcia mocniejsze i słabsze. Oba dały się wyraźnie oddzielić od normalnych odczytów akcelerometru a próg zadziałania ustaliliśmy na połowę piku wywołanego słabszym stuknięciem.

2.4. Sygnalizacja wyników

Na potrzeby projektu przyjęliśmy dwie sekwencje. Pierwsza to pojedyncze uderzenie w powierzchnię na której leży czujnik, druga to podwójne uderzenie w powierzchnię w przeciągu 1,1s. Sekwencje w podanej kolejności sygnalizowane są na wyświetlaczu slcd jako 0001 i 0002. Stan Polling_first sygnalizowany jest na wyświetlaczu jako AAAA. Jednocześnie wykrycie pierwszego uderzenia sygnalizowane jest diodą zieloną, natomiast wykrycie drugiego stuknięcia diodą czerwoną.

3. Podsumowanie

Efektem naszej pracy jest działające urządzenie (prototyp) oraz aplikacja w trakcie pisania wersja V1.1.

Do obsługi każdego z peryferii napisaliśmy osobną klasę będącą uniwersalnym i łatwym do rozwinięcia „biblioteką”. Kod sam w sobie jest podstawą, na której może zostać zbudowany większy projekt rozwijający stworzoną przez nas funkcjonalność. Efekty naszej pracy można obejrzeć i wykorzystać w ramach otwartego kodu na serwerze github pod adresem:

<https://github.com/designLaboratory/KnockKnock>
