

Grupa D07

Przemysław Rudowicz	216879	216879@edu.p.lodz.pl - lider
Konrad Jaworski	216782	216782@edu.p.lodz.pl
Jakub Plich	216866	216866@edu.p.lodz.pl

Dokumentacja projektu gry Snake
LPC1768/9

Spis treści

1. Podział obowiązków	3
1.1. Wykorzystane funkcjonalności	3
1.2. Podział obowiązków	3
2. Opis działania programu	4
2.1. Instrukcja użytkownika	4
2.2. Opis algorytmu	4
3. Funkcjonalności	4
3.1. GPIO	4
3.1.1. Głośnik	4
3.1.2. Joystick	5
3.2. Akcelerometr	5
3.3. Timer	5
3.4. OLED	5
3.5. SSP/SPI	5
3.6. Czujnik światła	5
3.7. PCA9532	6
3.8. I ² C	6
3.9.	6
4. Analiza FMEA	6
Literatura	7

1. Podział obowiązków

1.1. Wykorzystane funkcjonalności

Funkcjonalność	Osoba za nią odpowiedzialna
GPIO (joystick)	Konrad Jaworski
Akcelerometr	Konrad Jaworski
Głośnik	Konrad Jaworski
Timer	Przemysław Rudowicz
OLED	Przemysław Rudowicz
SSP/SPI	Przemysław Rudowicz
Czujnik światła	Jakub Plich
pca9532	Jakub Plich
I ² C	Jakub Plich
Rotacyjny przełącznik kwadraturowy	Jakub Plich

1.2. Podział obowiązków

Imię i nazwisko	Procentowy udział w pracy
Konrad Jaworski	33%
Przemysław Rudowicz	34%
Jakub Plich	33%

2. Opis działania programu

2.1. Instrukcja użytkownika

2.2. Opis algorytmu

3. Funkcjonalności

3.1. GPIO

GPIO (oznacza general-purpose input/output) - interfejs wejścia/wyjścia ogólnego przeznaczenia. Należy ustawić kierunki wejścia/wyjścia pinów GPIO (0 - gdy chcemy skonfigurować pin jako wejście, lub 1 - jako wyjście).

3.1.1. Głośnik

Głośnik jest obsługiwany przy pomocy pinów GPIO. Jako, że głośnik nie będzie wysyłał danych, piny ustawiamy na wyjście. W tym celu ustawiamy wartość 1 w rejestrach FIODIR0 i FIODIR2 w miejscach odpowiadających pinom głośnika (każdy bit rejestru odpowiada jednemu pinowi GPIO, każdy port GPIO ma swój rejestr FIODIR). A więc 1 należy ustawić na 28, 27, 26 bicie FIODIR0 i 13 bicie FIODIR2.

Wzmacniacz analogowy LM4811, który znajduje się na płytce LPCXpresso Base Board potrzebuje następujących pinów z mikrokontrolera:

- CLK
- UP/DN
- SHUTDN
- VIN1/VIN2

Ze specyfikacji LM8411 [1] dowiadujemy się, że piny CLK (CLOCK) oraz UP/DN są odpowiedzialne za sterowanie głośnością brzęczyka.

Pin SHUTDN aktywuje funkcję zmniejszającą pobór prądu przez wzmacniacz (Nie korzystamy z tej funkcji).

Piny VIN1/VIN2 odpowiadają za generację sygnału wprawiającego membranę brzęczyka w drgania (generowanie dźwięków).

Sposób połączenie pinów wzmacniacza analogowego do pinów GPIO:

Piny LM4811	Piny GPIO
CLK	P0.27
UP/DN	P0.28
SHUTDN	P2.13
VIN1/VIN2	P0.26

Podczas inicjalizacji głośnika czyszczone są wartości na pinach P0.27, P0.28, P2.13 (ustawiamy 1 w rejestrach FIOCLR dla portu 0 i 2 w miejscach odpowiadających wymienionym pinom).

Generowanie dźwięku przez brzęczyka odbywa się poprzez podawanie zmiennego napięcia na pin P0.26 tak aby wprowadzić membranę brzęczyka w drgania. Pozwala to na generowanie prostych nut.

Aby zagrać nutę 'C', należy wprowadzić membranę brzęczka w drgania o częstotliwości $f = 262\text{Hz}$. A więc okres drgań $T = \frac{1}{f} = 3816\mu\text{s}$. Stąd na pinie P0.26 należy ustawić stan wysoki przez czas równy $\frac{T}{2} = 1908\mu\text{s}$ oraz stan niski analogicznie przez $\frac{T}{2}$. Cykl należy powtarzać w zależności od tego jak długo chcemy odtwarzać dźwięk. Do ustawiania stanów wysokich i niskich na pinach GPIO używamy rejestru FIOSET i FIOCLR. Za generowanie dźwięku odpowiada pin P0.26. Analogicznie postępujemy w przypadku innych nut.

W celu ustawienia stanu wysokiego na pinie P0.26 należy ustawić 1 na 26 bicie rejestru FIOSET (ustawianie zera na tym rejestrze nie ustawia stanu niskiego). Aby odwołać stan wysoki należy wpisać 1 na 26 bicie rejestru FIOCLR.

3.1.2. Joystick

Joystick również jest obsługiwany przy pomocy pinów GPIO. Natomiast w przeciwieństwie do głośnika, joystick wysyła dane do mikrokontrolera, a więc podczas jego inicjalizacji ustawiamy wszystkie piny na wejście.

W tym celu ustawiamy wartość '0' na 15, 16 i 17 bicie rejestru FIODIR0 oraz na 3 i 4 bicie rejestru FIODIR2.

Pozycja joysticka	wartość	Piny GPIO
JOYSTICK_CENTER	0x01	P0.17
JOYSTICK_UP	0x02	P2.3
JOYSTICK_DOWN	0x04	P0.15
JOYSTICK_LEFT	0x08	P2.4
JOYSTICK_RIGHT	0x10	P0.16

Stany podłączonych pinów odpowiadają stanom wciśnięcia joysticka (odpowiednio tak jak w tabeli powyżej). W celu odczytania stanu joysticka sprawdzane są wartości na kolejnych pinach (odpowiednio tak jak w tabeli powyżej) i jeżeli jego wartość to '0', zmienna przechowująca stan joysticka przyjmuje wartość koniunkcji bitowej tego stanu i odpowiadającej wartości (patrz tabela powyżej) przypisanej do pozycji joysticka.

3.2. Akcelerometr

3.3. Timer

3.4. OLED

3.5. SSP/SPI

3.6. Czujnik światła

Czujnik światła jest urządzeniem peryferyjnym przyłączonym do płytki magistralą I²C. Program wykorzystuje odczytane natężenie światła do odwrócenia kolorów na wyświetlaczu w momencie w którym odczytana wartość natężenia światła będzie mniejsza niż 25 luksów. Uruchomienie czujnika

światła polega na przesłaniu przez I2C pod adres (0x44) kolejno wartości (0x00) oraz (1«7) . To powoduje ustawienie wartości 1 na 7 bicie rejestru Command Register(0x00) i w konsekwencji uruchomienie przetwornika analogowo-cyfrowego w czujniku. Zakres odczytu czujnika jest domyślny i wynosi od 0 do 1000 luksów. Odczyt wartości pomiaru czujnika wymaga odczytania zawartości dwóch rejestrów: LSB-Sensor - zawiera dolny bajt ostatniego odczytu sensora(adres 0x04), MSB-Sensor - zawiera górny bajt ostatniego odczytu sensora(adres 0x05). Wynik wyrażony w luksach jest obliczany z następującego wzoru: $E = 973 * \text{odczytana-wartość} / (1\ll 16)$

3.7. PCA9532

Expander PCA9532 wyposażony w 16 diód LED jest wykorzystywany do reprezentacji wyniku w danym momencie gry. Zapalone diody przedstawiają wynik w postaci binarnej. W celu zapalenia odpowiednich diód należy ustalić 16 bitową maskę w której wartości 1 oznaczają zapaloną diodę. Następnie przez I2C pod adres rejestru kontrolnego(0x60) zostaje wysłany bajt kontrolny. Trzeci bit bajtu kontrolnego oznacza flagę inkrementacji która zwiększa o 1 adres podany w pozostałych 4 bitach po każdym przesłanym bajcie. Potem przesłane zostają 4 bajty które zostają kolejno wpisane do 4 8-bitowych rejestrów(LS0,LS1,LS2,LS3) w których każde 2 bity odpowiadają jednej diodzie. Program korzysta jedynie ze stanów OFF(00) oraz ON(01).

3.8. I²C

I2C (Inter-Integrated Circuit) to szeregową, dwukierunkową magistralę do przesyłania danych. Program wykorzystuje ją do komunikacji z PCA9532, czujnikiem światła oraz akcelerometrem. Początek inicjalizacji magistrali zaczyna się od konfiguracji pinów które będą pełniły funkcję linii SCL(linia zegara) i SDA(linia danych). Dla interfejsu I2C2 są one ustawione odpowiednio na pinach P0[11] oraz P0[10]. W tym celu ustawiamy w rejestrze PINSEL0 wartości 1 i 0 kolejno dla bitów 21 i 20(P0.10 SDA2) oraz wartość 1 i 0 dla bitów 23 i 22(P0.11 SCL). Następnie następuje włączenie zasilania dla I2C poprzez ustawienie wartości 1 na 26 bicie rejestru PCONP(Power Control for Peripherals Register). Ustawiamy dzielnik zegara PCLK na 2 ustawiając w rejestrze PCLKSEL 20 i 21 bit na wartości kolejno 0 i 1. Następnie należy ustawić wartości rejestrów I2SCLH i I2SCLL na żadaną ilość cykli zegara PCLK. Obie te wartości są sobie równe.

Na koniec należy w rejestrze I2CONSET ustawiamy wartość 1 na 6 bicie aby włączyć interfejs I2C2.

3.9.

4. Analiza FMEA

Ryzyko	Prawdopodobieństwo	Znaczenie	(Samo)Wykrywalność	Iloczyn	Reakcja
Uszkodzenie joysticka					

Literatura

- [1] *LM4811 Dual 105mW Headphone Amplifier with Digital Volume Control and Shutdown Mode Datasheet*, December 2002, **National Semiconductor**