
Ćwiczenie 20: Gra sterowana czujnikiem ruchu

Instrukcja laboratorium

Mariusz Chilmon <mariusz.chilmon@ctm.gdynia.pl>



2024-06-13

Talk is cheap. Show me the code.

— Linus Torvalds

Cel ćwiczenia

Celem ćwiczenia jest praktyczne ćwiczenie umiejętności odnajdywania w kodzie źródłowym implementacji poszczególnych funkcji systemu mikroprocesorowego i ich poprawy.

Uruchomienie programu wyjściowego

1. Podłącz płytkę *LCD Keypad Shield* do *Arduino Uno*.
2. Podłącz akcelerometr.
3. W pierwszej linii wyświetlacza widoczny jest licznik zdobytych punktów.
4. W drugiej linii wyświetlacza na pierwszej pozycji znajduje się statek kosmiczny, który jest bohaterem gry, a na prawo od niego wrogie pojazdy, których należy unikać, aby kontynuować grę. Zderzenie kończy rozgrywkę.
5. Program wyjściowy nie pozwala poruszać statkiem.



Rysunek 1: Przykładowy stan gry. Uwagę przykuwają, mimo rachitycznej grafiki, znakomicie zamodelowani bohaterowie i bezkompromisowa wizualizacja świata z gatunku *space opera*.

Zadanie podstawowe

Celem zadania podstawowego jest sterowanie pojazdem kosmicznym przez obrót czujnika.

Wymagania funkcjonalne

1. Obrót czujnika powoduje skok pojazdu do góry. Po chwili pojazd automatycznie powraca na pozycję wyjściową.

Modyfikacja programu

W metodzie `Game::loop()` odczytaj pomiar z żyroskopu za pomocą wywołania `gyroscope.measure()`. Określ wartość graniczną prędkości kątowej, przy której uznasz, że należy wykonać skok główną postacią gry, ustawiając zmienną `hop` na wartość `HOP_UP`.



Pamiętaj o użyciu bloku `ATOMIC_BLOCK(ATOMIC_RESTORESTATE){ }`, aby uniknąć problemów z operacjami na zmiennych współdzielonych między pętlą główną, a przerwaniem.

Zadanie rozszerzone

Celem zadania rozszerzonego jest umożliwienie wznowienia rozgrywki po wciśnięciu przycisku `RIGHT`.

Wymagania funkcjonalne

1. Przyciskiem `RIGHT` można rozpocząć nową rozgrywkę.

Modyfikacja programu

W metodzie `Game::loop()` odczytaj stan klawiatury i przy wciśniętym przycisku `RIGHT` wywołaj metodę `start()`. Dodatkowo możesz sprawdzać stan zmiennej `stop` (zerowanej przez `start()`), aby przytrzymanie przycisku nie powodowało wielokrotnego wywoływania restartu gry.



Odczytaj ze schematów ideowych, do którego pinu mikrokontrolera podłączona jest klawiatura.



Nie musisz korzystać z ADC, aby sprawdzić stan przycisku. Ponieważ inne przyciski nie będą wykorzystywane, wystarczy odróżnić stan niski od wysokiego, nie ma potrzeby mierzyć dokładnego napięcia z klawiatury.