

Uwaga zestawy Arduino można pobierać i oddawać w pokoju 337

w środy w godzinach 12:15-14:00 lub po zajęciach.

## Progi punktowe oceny:

80% - 5.0

70% - 4.5

60% - 4.0

50% - 3.5

40% - 3.0

## Podstawowe materiały do pracowni

### Mikrokontrolery i sprzęt

- [8-bit AVR Instruction Set](#)
- [ATmega328P documents](#)
- [ATmega8U2 documents](#)
- [http://www.ti.com/ww/en/launchpad/msp430\\_head\\_usb.html](http://www.ti.com/ww/en/launchpad/msp430_head_usb.html)

### Środowisko programisty

- [AVR libc](#)
- [AVR gcc](#)
- [AVRDUDE](#)
- [AVaRICE](#) (GNU gdb po JTAGu)
- [Arduino IDE](#)

### Blogi i tutoriale

- [mikrokontrolery.blogspot.com](#)
- [linki odnośnie lutowania na ladyada.net](#)
- [wytrawianie płytek \(termotransfer\)](#)
- [tutoriale z adafruit](#)
- [tutoriale z sparkfun](#)
- [elportal.pl](#)
- [elektroda.pl](#)
- [Lessons in electric circuits](#) (dobry otwarty podręcznik do nauki elektroniki)
- <http://www.robotyka.net.pl/123d-circuits-symulacja-arduino-czesc-1-dioda-led-rgb-przyciski-potencjometri/> (artykuł o symulatorze Arduino)

### Projektowanie układów elektronicznych

- [Eagle Light](#) (bezpłatna wersja do użytku niekomercyjnego)
- [circuits.io](#)
- [Circuit Simulator Applet](#)
- [Circuit Lab](#)

Ostateczny termin oddawania zadań:

1 - 5 : 27.X .2016  
6 - 10 : 24.XI .2016  
11 - 15 : 15.XII.2016  
16 - 20 : 19. I.2017  
20 - 25 : 2. II.2017

## Zadania na pracownię do wykonywania w grupach

1. **[Lutowanie]** Zlutuj układ opisany pod adresem:  
[http://www.extremecircuits.net/2010/06/led-light-pen\\_19.html](http://www.extremecircuits.net/2010/06/led-light-pen_19.html)  
dodając do niego włącznik i regulator natężenia światła. **[40]**  
**Uwaga:** Zrobienie tego zadania jest wymagane do dopuszczenia do innych zadań wymagających lutowania
2. Zbuduj oparty na Arduino układ dostosowujący natężenie światła diody do światła panującego w pomieszczeniu (im ciemniej tym dioda powinna jaśniej świecić). Dodaj przyciski min i max pozwalające skalibrować urządzenie (odpowiednio - przy tym świetle jakie jest aktualnie ma być wyłączona dioda/maksymalne natężenie) **[20]**
3. Oddychająca dioda LED RGB, zasymuluj efekt oddychającej diody led (więcej informacji na google - Breathing LED). Po każdym cyklu wylosuj nowy docelowy kolor. **[30]**
4. Magnetyczne ciepło-zimno - zbuduj urządzenie, które losuje sobie wartość, a następnie bada natężenie pola magnetycznego, im bliżej docelowej wartości tym częściej załączony głośniczek powinien "pikać", a w przypadku osiągnięcia wylosowanej wartości, sygnał dźwiękowy powinien się zmienić na ciągły i powinna zapalić się dioda. Dodaj też przycisk resetujący urządzenie. **[40]**
5. Zbuduj urządzenie które zapamiętuje naciśnięcia i zwolnienia przycisku przez użytkownika, i odtwarza je zapalając i gasząc diodę z jedno sekundowym opóźnieniem. **[30]**
6. Zbuduj i zaprogramuj układ tłumaczący kod Morse'a ma tekst. Kod Morse'a ma być wprowadzany poprzez naciśnięcia przycisku, które dodatkowo powodują zapalenie diody i wydanie sygnału dźwiękowego. Wynik należy wyświetlić na ekranie komputera. **[30]**
7. Zbuduj i zaprogramuj układ tłumaczący tekst na kod Morse'a. Tekst ma być wprowadzany na komputerze, a wynik ma zostać wyemitowany przy użyciu diody i głośnika podłączonych pod arduino. **[30]**
8. Zbuduj układ z 4 diodami i joystickiem i wykorzystaj go do stworzenia następującej gry: diody symbolizują cztery kierunki i zapalają się losowo, a zadaniem użytkownika jest skierowanie joysticka w odpowiednią stronę, po czym dostaje odpowiednią liczbę punktów (im szybciej tym więcej) i dioda gaśnie. Wyświetlanie punktów, ustalanie czasu rozgrywki i rozpoczynanie nowej gry powinno się odbywać na ekranie komputera. Zadbaj o system anti-cheatowy. **[40]**
9. Używając co najmniej pięciu diod LED zbuduj wyświetlacz widmowy (ang. persistence of vision display) wyświetlający jakiś napis lub obrazek - np. bieżącą godzinę. Diody należy ułożyć jedna obok drugiej w rządęk (np. na płytce stykowej) i zapalać je i gasić z dużą szybkością tak, aby poruszanie nimi w powietrzu skutkowało ukazaniem się pożądanego obrazu. Obraz może być przerwany lub powtarzający się - układ nie musi wiedzieć, w której fazie ruchu jest rządęk diod. Zadbaj, aby układ nie rozsypał się w trakcie poruszania nim. **[50]**

Ostateczny termin oddawania zadań:

1 - 5 : 27.X .2016  
6 - 10 : 24.XI .2016  
11 - 15 : 15.XII.2016  
16 - 20 : 19. I.2017  
20 - 25 : 2. II.2017

- 10. Dzieńdobruino** - Zbuduj oparty na Arduino układ odgrywający na zewnętrznym głośniku głos "dzień dobry" z losowym zakresem prędkości (0d 0.5 do 2.0, gdzie 1.0 to prędkość oryginalna) i głośności (mnożymy wartości z tablicy przez wylosowaną liczbę z zakresu 0.8-1.2). Układ powinien aktywować się za każdym razem kiedy czujnik odległości wykryje ruch (użyj ultradźwiękowego czujnika lub czujnika natężenia oświetlenia). Tablice opisującą dźwięk zaszyj (makro PROGMEM) w kodzie programu. **[30]**

#### **Zadania optymalizacyjne:**

Zadania wykonujemy pod Windowsem lub Linuxem z włączoną pełną opcją optymalizacji przez kompilator. W każdym zadaniu robimy dwie wersje (brute force i optymalna), różnicę czasu mierzymy dokładnie (QueryPerformanceCounter - Windows, clock - Linux).

**Uwaga zadania 11-15 oddajemy indywidualnie.**

- 11.** Przygotuj przynajmniej 300MB plik binarny. Twój program na starcie powinien go w całości wczytać do pamięci. Następnie za pomocą minimalnie trzech metod (jedna bazująca na 256 elementowej tablicy precalc, druga na 65536 elementowej tablicy precalc i trzecia operująca na maskach bitowych) program powinien policzyć ilość zapalonych bitów i wypisać ją na konsoli. Zmierz i wypisz na konsoli czas działania poszczególnych funkcji. **[30]**
- 12.** Napisz funkcję, która w zadanym jako parametr napisie (tablicy typu char, jej rozmiar jest drugim parametrem) przesunie wszystkie spacje na początek zachowując względny porządek pozostałych znaków, np. "Ala ma kota" => " Alamakota". **[30]**
- 13.** Zaimplementuj kilka (minimalnie 3) metod odwracania kolejności bitów w 32-bitowej liczbie (tzn. zamienić bit 1 z 32, 2 z 31 itd.) Zmierz na wybranym komputerze rzeczywistą szybkość wykonania zaimplementowanych metod. **[30]**
- 14.** Napisz program, który z tablicy liczb typu int bez znaku (wczytaj z pliku lub wylosuj przynajmniej 300MB tablice) policzy wszystkie te, których negacja bitowa ma co najmniej dwa bity ustawione na jedynkę. **[30]**
- 15.** Z użyciem [wstawek assemblerowych](#) (ale nie używając operacji mnożenia) napisz operację mnożenia dwóch liczb nieujemnych. Twoja wstawka ma przyjmować dwie 16-bitowe liczby, i zwracać liczbę 32 bitową w dwóch 16-bitowych kawałkach. Napisz test, który będzie czytał przez łącze szeregowe dwie liczby, wykonywał działanie i drukował wynik na Arduino. **[50]**
- 16.** Zbuduj układ symulujący zapalanie diód i odgrywający dźwięk (równocześnie) z filmu "knight rider". **[30]**
- 17.** Zbuduj magnetyczny (z hall sensor) lub świetlny (z fotorezystorem) [teremin](#) z możliwością regulacji podstawowej częstotliwości, nagrywaniem i odtwarzaniem. Możesz generować po prostu falę prostokątną o częstotliwości zależnej od wskazań sensora. Kontrola głośności powinna odbywać się za pomocą potencjometru. **[35]**
- 18.** Używając dwóch Arduino zbuduj urządzenie transmisji bezprzewodowej. Jedno Arduino powinno nadawać (i powtarzać) tekst kodem Morsa włączając i wyłączając diodę led, oraz odgrywając dźwięk na głośniczku. Drugie urządzenie korzystając z czujnika światła powinno odczytać nadawany tekst i wyświetlić go na ekranie komputera. Sprawdź jak bardzo możesz przyspieszyć transmisję nie wprowadzając

Ostateczny termin oddawania zadań:

1 - 5 : 27.X .2016  
6 - 10 : 24.XI .2016  
11 - 15 : 15.XII.2016  
16 - 20 : 19. I.2017  
20 - 25 : 2. II.2017

- błędów. Alternatywnie do transmisji można użyć pary głośnik/mikrofon. [50]
19. Zbuduj następujący układ: na początku losowany jest kolor i pokazywany użytkownikowi przy użyciu diody RGB (używając tylko dwóch z podstawowych kolorów) albo dwóch diod (dodaj przycisk, który pozwoli użytkownikowi ponownie obejrzeć jej kolor). Po pokazaniu docelowego koloru urządzenie ma przejść w tryb pokazywania koloru ustawionego przez użytkownika przy pomocy czujnika przyspieszenia w następujący sposób - każda oś odpowiada jednemu kolorowi, po zakończeniu ruchu kolory są modyfikowane o największą wartość przyspieszenia podczas ruchu. Po osiągnięciu przez użytkownika docelowego koloru diody powinny migać trzy razy, następnie losowany jest nowy kolor. [30]
20. Zbuduj układ z 5-8 diodami LED i 3 guzikami. Naciśnięcie jednego guzika powinno spowodować zgaszenie wszystkich diod. Naciśnięcie drugiego powinno wyświetlić kolejną liczbę w kodzie Graya. Wciśnięcie trzeciego guzika powinno wyświetlić poprzednią liczbę kodu Graya. Nie korzystaj z Arduino IDE i bibliotek arduino. Dodatkowo obsługę wciśnięcia umieść w obsłudze przerwania. Pamiętaj, że ATmega328p posiada tylko dwa porty przerwań zewnętrznych. Postaraj się maksymalnie uśpić mikrokontroler w czasie, kiedy nie obsługuje on zdarzeń. [30]
21. Zrób z arduino z podłączonym głośniczkiem odtwarzacz numerów telefonów wybieranych tonowo ([DTMF](#)). Numery telefonów powinny być wprowadzane przez użytkownika na komputerze. [35]
22. Zaimplementuj następującą grę dla trzech graczy. Każdy z graczy ma odpowiadający mu przycisk oraz diodę led. Dodatkowo istnieje główna dioda, której zapalenie oraz sygnał dźwiękowy na głośniczku oznacza rozpoczęcie rundy. Po rozpoczęciu rundy gracz który najszybciej naciśnie swój przycisk otrzymuje punkty a jego dioda zaczyna mrugać 3x i odgrywana jest krótka muzyczka. W przypadku zbyt szybkiego naciśnięcia przycisku (przed rozpoczęciem rundy) przyznawane są punkty ujemne. Falstart sygnalizowany jest krótkim dźwiękiem oraz zapaleniem diody użytkownika na 2 sekundy. Aktualną punktację i stan gry pokazuj w oknie serial komputera. Losuj czas rozpoczęcia kolejnej rundy. [40]
23. Zbuduj latającą myszkę, tzn. przeprogramuj kontroler USB w Arduino, tak żeby komputer widział go jako mysz USB. O ruchu kursora ma decydować akcelerometr, poza tym mają być dwa przyciski służące jako lewy i prawy przycisk myszki.  
**Uwaga:** Po oddaniu tego zadania, należy wgrać z powrotem oryginalny firmware do kontrolera USB. [extra 40]
24. Zbuduj układ pozwalający odtwarzać zadane dwie sekwencje klawiszy i przerw między nimi. Sekwencje mają być uruchamiane przez naciśnięcia przycisków podłączonych do Arduino, a samo Arduino widziane jako klawiatura USB. Daj możliwość łatwego ustawienia (w kodzie, nie w czasie działania) tych sekwencji, np. jako stringi.  
**Uwaga:** Po oddaniu tego zadania, należy wgrać z powrotem oryginalny firmware do kontrolera USB. [extra 40]
25. Zaprogramuj mikrokontroler USB na Arduino tak, aby konwertowało sygnały z joysticka lub pada od jakiejś starej konsoli (Nintendo, Sega) lub komputera (Commodore, Atari) i pozwalało grać pod emulatorem na PC.  
**Uwaga:** Po oddaniu tego zadania, należy wgrać z powrotem oryginalny firmware do

Ostateczny termin oddawania zadań:

1 - 5 : 27.X .2016  
6 - 10 : 24.XI .2016  
11 - 15 : 15.XII.2016  
16 - 20 : 19. I.2017  
20 - 25 : 2. II.2017

kontrolera USB.[extra 50]

**Pomysły na duży projekt:**

1. Komputer rowerowy.
2. <https://hackaday.io/project/1887-open-bicycle-computer>
3. Zegar "nieskończony".
4. <https://hackaday.io/project/1852-infinity-mirror-clock>
5. Robot do podlewania kwiatków.
6. <https://hackaday.io/project/10138-lowcost-full-automated-watering-robot>
7. Stacja pogodowa.
8. <https://hackaday.io/project/1878-modular-weather-station>
9. Robot typu line follower.
10. <https://hackaday.io/project/6655-arduino-based-desktop-line-follower-jollibot>
11. Sześćian LED.
12. <https://hackaday.io/project/9233-jollicube-an-8x8x8-led-cube-spi>
13. Instrument muzyczny ze stacji dysków.
14. [https://www.youtube.com/watch?v=cM\\_sAxAu7Q](https://www.youtube.com/watch?v=cM_sAxAu7Q)
15. Zegar z dysku twardego. (dostępne części)
16. <https://www.youtube.com/watch?v=Xy103O3qAQc>

Ostateczny termin oddawania zadań:

- |         |   |        |       |
|---------|---|--------|-------|
| 1 - 5   | : | 27.X   | .2016 |
| 6 - 10  | : | 24.XI  | .2016 |
| 11 - 15 | : | 15.XII | .2016 |
| 16 - 20 | : | 19. I  | .2017 |
| 20 - 25 | : | 2. II  | .2017 |