

Uwaga zestawy Arduino można pobierać i oddawać u Marka Matrzoka (pokój 336)
w czwartki w godzinach 14-16.

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

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)

Projektowanie układów elektronicznych

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

Ostateczny termin oddawania zadań:

1-10: 7.XI.2013
11-20: 12.XII.2013
35-44: 5.XII.2013

oznaczenia punktowe:

[a-b]

a - punkty za podstawowe działające rozwiązanie

b - maksymalna ilość punktów (za optymalne rozwiązanie)

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 dodając do niego włącznik i regulator natężenia światła. **[15-40]**
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. Dodaj przyciski min i max pozwalające skalibrować urządzenie (odpowiednio - przy tym świetle jakie jest aktualnie ma być wyłączona dioda/ maksymalne natężenie)**[15-20]**
3. Zbuduj urządzenie opisane w zadaniu 2 bez wykorzystania dodatkowego czujnika natężenia światła, tzn. wykorzystaj do mierzenia natężenia światła diodę, która ma też świecić (trzeba bardzo szybko zmieniać tryb pracy diody).**[20-40]**
4. Zbuduj urządzenie mierzące temperaturę i natężenie światła, a następnie udostępniające stronę html (z użyciem Ethernet Shield dla Arduino) z wynikami tekstowo **[20-40]** lub w postaci wykresów (np. używając canvas z HTML 5 i javascript) **[0-30]**
5. Magnetyczne ciepło-zimno - zbuduj urządzenie, które losuje sobie wartość, a następnie bada natężenie pola magnetycznego, im bliżej tym wartości tym częściej załączony głośniczek powinien "pikać", a w przypadku osiągnięcia wylosowanej wartości, sygnał dźwiękowy powinien się zmienić i powinna zapalić się dioda. Dodaj też przycisk resetujący urządzenie. **[20-30]**
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. **[30-50]**
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 głośniczka podłączonych pod arduino. **[20-30]**
8. Zrób z arduino z podłączonym głośniczkiem odtwarzacz numerów telefonów tonowo ([DTMF](#)). Numery telefonów mają być pobierane od użytkownika na komputerze. **[20-30]**
9. Użyj arduino do policzenia ilości bitów ustawionych na jedynkę w wybranym na komputerze pliku (liczenia powinno się odbyć po stronie arduino) **[15-30]**

Ostateczny termin oddawania zadań:

1-10: 7.XI .2013

11-20: 12.XII.2013

35-44: 5.XII.2013

10. Napisz program, który policzy sumę kontrolną crc32 dla zadanego pliku znajdującego się w pamięci arduino, oraz wyświetli ją na konsoli wraz z długością pliku **[15-30]**
11. 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 może się odbywać za pomocą potencjometru. **[30-50]**
12. Oddychająca dioda LED RGB, zasymuluj efekt oddychającej diody led (więcej informacji na google - Breathing LED). Po każdym cyklu wylosuj nowy docelowy kolor. **[20-30]**
13. Zbuduj następujący układ: na początku losowany jest kolor i pokazywany użytkownikowi przy użyciu diody RGB (używając tylko dwóch z podstawowych kolorów) albo dwóch diód (dodaj przycisk, który pozwoli użytkownikowi ponownie obejrzeć tej 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 mają zamrugać i ma zostać wylosowany nowy kolor. **[20-40]**
14. Skonstruuj choinkę z kolorowymi diodami (zielony kartonik z wpiętymi diodami), które lampki będą świecić w różny sposób a dołączony głośnik będzie grać kolędę lub inną świąteczną piosenkę. **[20-30]**
15. 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. **[20/30]**
16. Zbuduj sterowany przez ethernet układ odtwarzający muzykę. Minimalne opcje sterowania to zmiana tempa odtwarzania, start/stop i przesłanie nowego utworu. **[40-60]**
17. **[projekt, 2-4 osób]** Zbuduj sześcián LED 8x8x8 **[100-200]**. Stwórz program wykorzystujący ten sześcián w ciekawy sposób **[0-100]** i/lub wtyczkę do winampa używającą sześciánu **[0-50]**. (UWAGA: chęć wykonania tego zadania należy wcześniej zgłosić prowadzącemu ze względu na dużą ilość potrzebnych części).
18. Zaprojektuj i zbuduj urządzenie generujące dźwięk i muzykę na podstawie ruchu sieciowego (bluetooth, wifi i/lub ethernet). Możesz też dodać ciekawą wizualizację. **[30-50]**
19. Zbuduj urządzenie sterujące prędkością wiatraczka w zależności od temperatury.
Uwaga: w wiatraczek należy się zaopatrzyć we własnym zakresie - może być stary wiatraczek wymontowany z komputera **[20-30]**.
20. **[lutowanie]** Zbuduj układ udający świerszcza, opisany pod adresem: <http://www.extremecircuits.net/2010/01/cricket-chirping-generator-circuit.html> dodając fotorezystor, w sposób opisany na powyższej stronie **[12-40]**
Uwaga: to zadanie może być uznane za zadanie wstępne z lutowania zamiast zadania

Ostateczny termin oddawania zadań:

1-10: 7.XI.2013
 11-20: 12.XII.2013
 35-44: 5.XII.2013

21. **[lutowanie]** Zlutuj krokomierz opisany pod adresem: <http://www.extremecircuits.net/2009/12/digital-step-km-counter-circuit.html> **[20-50]**
22. **[lutowanie]** Zbuduj elektroniczną kostkę opisaną pod adresem: <http://www.extremecircuits.net/2010/04/dicing-with-leds.html> **[20-50]**
23. Zbuduj latającą myszkę, tzn. przeprogramuj kontroler USB w Arduino, tak żeby komputer widział go jako mysz USB. O ruchu kursora ma decydować akcelerometr, dodaj także dwa przyciski służące jako lewy i prawy przycisk myszki. **[20-50]**
Uwaga: Po oddaniu tego zadania, należy wgrać z powrotem oryginalny firmware do kontrolera USB
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. **[20-50]**
Uwaga: Po oddaniu tego zadania, należy wgrać z powrotem oryginalny firmware do kontrolera USB
25. **[lutowanie]** Zbuduj podłączany pod arduino theremin, opisany pod adresem: <http://interface.khm.de/index.php/lab/experiments/theremin-as-a-capacitive-sensing-device/> **[20-60]**. Dodatkowe punkty za inne ciekawe wykorzystanie tego układu niż pokazane na podanej stronie. **[0-15]**
26. Stwórz następującą grę dla wielu graczy: jedno arduino z podłączoną diodą i głośnikiem będzie odpowiedzialne za rozpoczęcie rundy, zliczanie punktów i wyświetlanie wyników na komputerze. Pozostałe arduino (po jednym na gracza) mają być wyposażone w przycisk i diodę. Rozgrywka przebiega w następujący sposób: przed pierwszą rundą główne arduino wykrywa wszystkie podłączone urządzenia (zlicza graczy), następnie w każdej rundzie w losowym momencie zapala się dioda na głównym urządzeniu, a gracze ścigają się kto pierwszy naciśnie przycisk. Gracz który pierwszy wciśnie przycisk po zapaleniu diody dostaje punkty, jest to sygnalizowane sygnałem dźwiękowym, gaśnięciem głównej diody i mruganiem diody przy danym graczu. Naciśnięcie przycisku w momencie, gdy nie świeci się główna dioda skutkuje punktami ujemnymi (i odpowiednim sygnałem dźwiękowym). **[20-60]** Istnieje także możliwość zrobienia tego zadania na jednym arduino **[20-30]**.
27. Zbuduj urządzenie odgrywające melodie sterując częstotliwością wiatraczka. **[15-25]**
Uwaga: w wiatraczek należy się zaopatrzyć we własnym zakresie - może być stary wiatraczek wymontowany z komputera
28. Z użyciem Ethernet shield, wykonaj sprzętowy program "ping" (do adresu 8.8.8.8) z efektem dźwiękowym lub diodą LED. **[20-30]**
29. Zaprogramuj mikrokontroler USB na Arduino tak by konwertowało sygnały z joysticka lub pada od jakiejś starej konsoli (Nintendo, Sega) lub komputera (Commodore, Atari) i pozwalało grać pod emulatorem. **[80-120]**
30. Przy pomocy Arduino wygeneruj sygnał VGA, a następnie zaprogramuj prostą animację według uznania. **[50]** Ew. zaprogramuj grę Pong dla dwóch graczy. **[50]**

Ostateczny termin oddawania zadań:

1-10: 7.XI.2013
 11-20: 12.XII.2013
 35-44: 5.XII.2013

31. Procesory AVR nie posiadają instrukcji dzielenia liczb całkowitych. Zaprogramuj w assemblerze wydajny algorytm dzielenia z resztą liczby 16-bitowej przez 8-bitową przy pomocy natywnych instrukcji (można użyć avr studio z debugerem). **[30-40]**
32. Procesory AVR nie posiadają instrukcji działających na liczbach zmiennopozycyjnych (float). Zaprogramuj w assemblerze dodawanie dwóch liczb float w standardzie IEEE754 przy pomocy natywnych instrukcji. **[40-60]**
33. Wykonaj układ według schematu na [tej](#) stronie (lft-phasor), a następnie skompiluj i zaprogramuj ten układ dostarczonymi źródłami. **[60-80]**
34. Zbuduj programator USBasp (<http://www.fischl.de/usbasp/>). Do zaprogramowania mikrokontrolera Atmega8 będącego elementem programatora można użyć Arduino (<http://arduino.cc/en/Tutorial/ArduinoISP>). Gotowy programator może zostać użyty do wykonania kolejnych zadań bez użycia Arduino. **[40-50]**

Zadania z optymalizacji (do uruchomienia na komputerach pc)

uwaga: należy mierzyć czasy wykonania zadań i używać wygenerowanych przez siebie danych wejściowych.

35. Napisz program, który wypisze wszystkie liczby z podanego zakresu posiadające wszystkie jedynki na miejscach o tej samej parzystości (tzn. tylko parzyste bądź tylko nieparzyste). Dodatkowe punkty są za jakość rozwiązania **[20]**
 Uwaga: warunek dotyczy zapisu binarnego, przykładowe liczby:
 pasujące: 10101010, 10000, 010101, 01000101
 niepasujące: 11111111, 10100100, 1100000, 00100100
 Powyższe liczby są tylko przykładami - proszę się nie kierować ich rozmiarem.
36. Napisz program, który z ciągu liczb bez znaku podanych na standardowe wejście (ciąg kończy się EOF) wypisze wszystkie te, które mają co najmniej dwa bity ustawione na jedynkę. **[20]**
37. Napisz program wypisujący znakami '*' prostokątny trójkąt równoramienny o podanej przez użytkownika długości przyprostokątnej. Program napisz w dwóch wersjach - minimalizując czas i minimalizując pamięć.
 Przykład dla długości 3:
 *
 **

[30]
38. 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]**
39. * Napisz własne funkcje do alokacji i dealokacji pamięci na prealokowanym dużym bloku pamięci (tzn. alokacja polega na przyznaniu miejsca w dużej globalnie dostępnej tablicy). Przygotuj dwie wersje - minimalizującą zużycie pamięci (dodatkowej, tzn.

Ostateczny termin oddawania zadań:

1-10: 7.XI .2013

11-20: 12.XII.2013

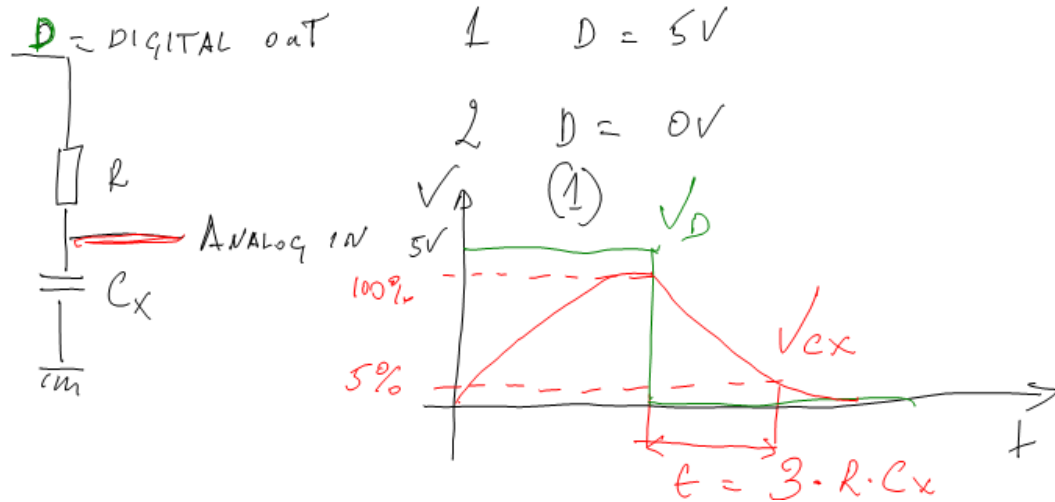
35-44: 5.XII.2013

- nie licząc tablicy w której alokujemy) i minimalizującą czas. W obu wersjach zadбай o możliwie optymalne wykorzystanie tablicy, w której następuje alokacja. **[80]**
40. Napisz programy kompresujący i dekompresujący pliki metodą run length encoding. **[40]**
Uwaga: [algorytm RLE](#) chodzi o algorytm opisany od "Ciekawszym..."
41. Napisz program, który doda do pliku sumy kontrolne co zadaną liczbę bitów i o zadanym rozmiarze. Dodaj możliwość wyboru algorytmu sumy kontrolnej - suma modularna bądź parzystość. Sumy mają znaleźć się w pliku zaraz po bitach, których dotyczą. Dodaj też możliwość sprawdzenia i usunięcia sum kontrolnych. **[40]**
42. Napisz program konwertujący plik w dwie strony pomiędzy [PDP-endian](#) i formatem stosowanym w Twoim komputerze. Kolejność bajtów w komputerze sprawdź wewnątrz programu nie używając funkcji i makr bibliotecznych. **[40]** **Uwaga:** Traktujemy plik jako ciąg długich słów (wartości 32-bitowych). W szczególności oznacza to, że wielkość pliku wyrażona w bajtach jest podzielna przez 4.
43. Napisz funkcje zwracającą n-tą liczbę Fibbonaciego. Przygotuj dwie wersję naiwną i zoptymalizowaną. Zoptymalizuj zamotywowany czas zwracania dziesiątek tysięcy takich wartości. **[30]**
44. Napisz funkcję wypełniającą tablicę kolorów poczynszysy od zadanego pixela (flood fill). Naiwna wersja powinna nadpisywać zadany pixel nowym kolorem, oraz wywoływać się rekurencyjnie dla wszystkich sąsiadów o tym samym kolorze. Wersja zoptymalizowana powinna działać iteracyjnie. **[40]**
45. Zbuduj wykrywacz metalu według opisu: <http://dzlsevilgeniuslair.blogspot.com/2013/07/diy-arduino-based-metal-detector.html> **[50]**
46. Multimetr: Arduino potrafi zmierzyć napięcie. Zbuduj prosty multimetr:
- Pomiar napięcia - używając Analog In zmierz napięcie i wysyłaj na serial wartość w Voltach. Do czego służy analog reference? **[5]**
 - Napisz proste GUI do komunikacji z Arduino, zawierające pole odczytu i selektor funkcji i zakresu pomiarowego. Dane mają się odświeżać najrzadziej co sekundę. Komunikacja z Arduino ma następować przez serial. **[10]**
 - Dodaj dzielnik napięciowy umożliwiający pomiar napięc do 50V. Jaka jest rozdzielczość (w V) pomiaru? Na tym zakresie można mierzyć np. ładowarki do laptopów. **[5]**
 - Za pomocą wzmacniacza operacyjnego LM358 zbuduj wzmacniacz nieodwracający (http://pl.wikibooks.org/wiki/Wzmacniacze_operacyjne#Wzmacniacz_nieodwracaj.C4.85cy) umożliwiający pomiar napięć 0-200mV z rozdzielczością około 0.1mV. **[10]**
 - Pomiar prądu można wykonać mierząc spadek napięcia na znanym małym rezystorze. Dołóż rezystor 1ohm do wzmacniacza żeby móc mierzyć prądy do 200mA. **[5]**

Ostateczny termin oddawania zadań:

1-10: 7.XI.2013
 11-20: 12.XII.2013
 35-44: 5.XII.2013

- f. Pojemność możemy mierzyć licząc czas rozładowania kondensatora przez znaną oporność, np. tak:



Do jednego z wyjść cyfrowych DOUT podłączamy znany rezystor i kondensator. Ustawiamy wyjście DOUT=HIGH (5V) co powoduje naładowanie kondensatora. Kiedy napięcie na kondensatorze zbliży się do 5V ustawiamy wyjście DOUT=LOW. Napięcie na kondensatorze maleje wykładniczo ze stałą czasową $R \cdot C_x$. Po upływie trzech stałych czasowych napięcie spadnie do około 5%. Można wtedy przerwać pomiar obliczając $C_x = t/3/R$. Zaimplementuj pomiar pojemności i zmierz jakiś kondensator. **[10]**.

W tym zadaniu można mierzyć: napięcie i prąd różnokolorowych LEDów, baterie, ładowarki do komórek i laptopów. Dla LEDów spadek napięcia na słabo świecącej diodzie powinien odpowiadać energii światła w eV, np. dla światła czerwonego długość fali to około 700nm, energia to $h \cdot c / \text{długość} = 4.1 \cdot 10^{-15} \text{ eV} \cdot \text{s} \cdot 3 \cdot 10^8 \text{ m/s} / 700 \text{ nm} = 1.7 \text{ eV}$, ponieważ w diodzie LED na kwant światła emitowany jest przez zmianę energii jednego elektronu spadek napięcia wynosi około 1.7V.

Ostateczny termin oddawania zadań:

1-10: 7.XI.2013

11-20: 12.XII.2013

35-44: 5.XII.2013