

Witam. Dzisiaj kilka słów na temat programowania arduino, dokładnie to atmegi 328 P-PU, którą możemy znaleźć w Arduino Uno. Uważam, że zaczęcie od zaprogramowania własnej atmegi jest dobrym przygotowaniem do tworzenia własnych projektów. Oczywiście możemy kupić arduino i na nim projektować, jednak po czasie będziemy musieli i tak przenieść się na własną atmegę, chyba że kogoś stać na wstawianie w każdy projekt arduino, poza tym projektując własną płytkę PCB łatwiej jest wstawić samą atmege niż całe arduino.

Ale zacznijmy od początku. Najpierw czym jest arduino. Wikipedia mówi nam następująco:

Arduino – platforma programistyczna dla <u>systemów wbudowanych</u> oparta na prostym projekcie <u>Open Hardware</u> przeznaczonym dla <u>mikrokontrolerów</u> montowanych w pojedynczym <u>obwodzie drukowanym</u>, z wbudowaną obsługą wejścia/wyjścia oraz standaryzowanym językiem programowania. Język programowania Arduino jest oparty na środowisku <u>Wiring</u> i zasadniczo na języku <u>C/C++</u> (kilka prostych przekształceń kodu wykonywane przed przejściem do avr-gcc). Celem projektu Arduino jest przygotowanie narzędzi – ogólnodostępnych, tanich, nie wymagających dużych nakładów finansowych, elastycznych i łatwych w użyciu przez hobbystów. Częściowo Arduino stanowi również alternatywę dla osób, które nie mają dostępu do bardziej zaawansowanych kontrolerów, wymagających bardziej skomplikowanych narzędzi.

Prostymi słowami arduino to płytka PCB z mikrokontrolerem który możemy dowolnie zaprogramować, oczywiście w miare jego możliwości. Do dyspozycji mamy piny analogowe i cyfrowe. Komunikacje i2c, SPI i kilka innych.

Dziś zajmiemy się wgraniem bootloadera zgodnego z arduino, po to by móc wgrywać programy używając ArduinoIDE. Oczywiście możemy robić to za pomocą terminala czy innego programu do programowania avr'ów.

Zaczniemy od poznania pinoutu (rozłożenia pinów) naszej atmegi.





Powyżej mamy uproszczony układ pinów. Do programowania będzie nam potrzebny poniższy zestaw:



1.a Arduino Uno, którego użyjemy jako programatora. Opcjonalnie możemy użyć



programatora USBASP (1.b) dostępnego na <u>allegro</u>.

2. Płytka stykowa.

- 3. Moduł zasilania do płytek stykowych z opcja ustawienia zasilania na 5V i 3,3V.
- 4. Atmega 328P-Pu, którą kupiłem jakiś czas temu na warszawskim wolumenie.

Oczywiście będziemy też potrzebować kabli do połączeń oraz jednej diody led do testu czy wszystko dobrze się wgrało.

Pozostaje nam połączenie elementów.

-	pin	13	(arduino)	-	pin	19	atmega
-	pin	12	(arduino)	-	pin	18	atmega
-	pin	11	(arduino)	-	pin	17	atmega
-	pin	10	(arduino)	-	pin	1 a	atmega

Schematycznie wygląda to tak:





fritzing

Na powyższym schemacie ujęta jest sytuacja gdy nie dysponujemy modułem zasilającym. Możemy wtedy spokojnie podłączyć zasilanie z arduino. U mnie w rzeczywistości wygląda to tak:





Teraz przygotowujemy nasze IDE. Zaczniemy od podmiany pliku <u>boards</u>.txt. Umieszczamy go w Arduino\hardware\arduino.

Następnie musimy dodać <u>bootloader</u>. Po rozpakowaniu umieszczamy go w Arduino\hardware\arduino\bootloaders\atmega.

Następnym krokiem będzie wgranie na nasze arduino szkicu ArduinoISP.









🙃 A	rduinoISP Arduino 1.0.6								
Plik Edycja Szkic Narzędzia Pomoc									
	Nowy	Ctrl+N				<mark>.</mark>			
	Otwórz	Ctrl+O							
	Szkicownik	•							
	Przykłady	•		01.Basics					
	Zamknij	Ctrl+W		02.Digital	۱.				
	Zapisz	Ctrl+S		03.Analog	۱.				
	Zapisz jako	Ctrl+Shift+S 04.C Ctrl+U 05.C a Ctrl+Shift+U 06.Se Ctrl+Shift+D 07.D		04.Communication	۱.				
	Załaduj		05.Control	۱.					
	Załaduj używając programatora		06.Sensors	۱.					
	Ustawienia strony			07.Display	۱.				
	Drukui	Ctrl+P	30	08.Strings	۱.				
	Drakaj	Carri		09.USB	•				
	Preferencje	Ctrl+Comma		10.StarterKit	•				
	Wyjście	Ctrl+Q		ArduinoISP					
77.9	9: Heartbeat - shows the	programmer		EtherCard	•				
77 8	3: Error - Lights up) if somethi		Keypad	▶ ⁱ	f that makes :			
	7: Programming - In commun	ication wit		MFRC522	•				
11 :	23 July 2011 Randall Bohn			RFID	•				
11 -	-Address Arduino issue 509) :: Portabi							
	<pre>http://code.google.com/p/a</pre>	<u>rduino/issu</u>		EEPROM					
11	October 2010 by Randall Bo	hn		Esplora					
11	HALL AN EPRIDON & OFC has			Ethernet					
				Firmata					
				GSM					
				LiquidCrystal	1				
				Robot_Control	1				
				Kobot_Motor	1				
1				50		28 MHz on COM15			
da.	Son The Str.	148 mar 1		Servo		Co St. Y			
		ten; c)		SoftwareSerial					
		Soi M		SPI					
		Gi Mi		stepper					
		i Mit		IFT MORE					
		C Mill		Wire					
		C Mart		wire	'				

Wgrywamy

NO SYSTEM IS SAFE

na nasze arduino. Następnym krokiem jest wybranie z listy naszej atmegi.

🐵 ArduinoISP Arduino 1.0.6									
Plik Edycja Szkic Narzędzia Pomoc									
Automatyczne formatowanie Ctrl+T ArduinoISP // ArduinoISP // Copyright (Automatyczne formatowanie Ctrl+T Archiwizuj szkic Napraw kodowanie i załaduj ponownie Monitor portu szeregowego Ctrl+Shift+M									
// If you requ Płytka	Arduino Uno								
Port szeregowy	Arduino Duemilanove w/ ATmega328								
// This sketch // using the f // Wypal Bootloader	Arduino Diecimila or Duemilanove w/ ATmega168 Arduino Nano w/ ATmega328								
// pin name: not-mega: mega(1280 and 2560)	Arduino Nano w/ ATmega168								
// slave reset: 10: 53	Arduino Mega 2560 or Mega ADK								
// MISO: 12: 50	Arduino Mega (Al mega 200)								
// SCK: 13: 52	Arduino Esplora								
// Put an LED (with registor) on the following mine:	Arduino Micro								
// 9: Heartbeat - shows the programmer is running	Arduino Mini w/ ATmega328								
// 8: Error - Lights up if something goes wrong (use re	Arduino Mini w/ ATmega168								
// 7: Programming - In communication with the slave	Arduino Ethernet								
// 23 July 2011 Randall Bohn	Arduino Fio								
// -Address Arduino issue 509 :: Portability of ArduinoISP	Arduino BT w/ ATmega328								
// http://code.google.com/p/arduino/issues/detail?id=509	Arduino BT w/ ATmega168								
// October 2010 by Randall Bohn	LilyPad Arduino USB								
Instant as EEDDON'S OF Charter III	LilyPad Arduino w/ ATmega328								
	LilyPad Arduino w/ ATmega168								
	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328								
	Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168								
	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328								
	Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168								
1 ATmega32	Arduino NG or older w/ ATmega168								
dry Service Strategic Militan, Ser	Arduino NG or older w/ ATmega8								
ny service Strategic Millian Ser	Arduino Robot Control								
y Service Strategic Military Service	Arduino Robot Motor								
Convice Strategic Military Sente	ATmega328P @ 8 MHz								
Sen Ce Strategic Millian Sen C	ATmega328P @ 16 MHz								

Z przykładowych szkiców wybieramy Basic -> Blink i modyfikujemy. Zamieniamy 13 pin na A5 – pin analogowy nr 5 który jest pinem nr 28 w naszej atmedze. Zabieg jest zalecany ponieważ do pierwotnego pinu przypisanego pod arduino podpięliśmy przewód





programatora.

Potem już tylko zostaje nam załadowanie naszego szkicu na atmege i sprawdzenie czy wszystko działa. Wiedza dotycząca obsługi ArduinoISP na pewno przyda się nam gdy będziemy tworzyć własne projekty oparte o pcb czy gdy utracimy oryginalny bootloader na naszym arduino.