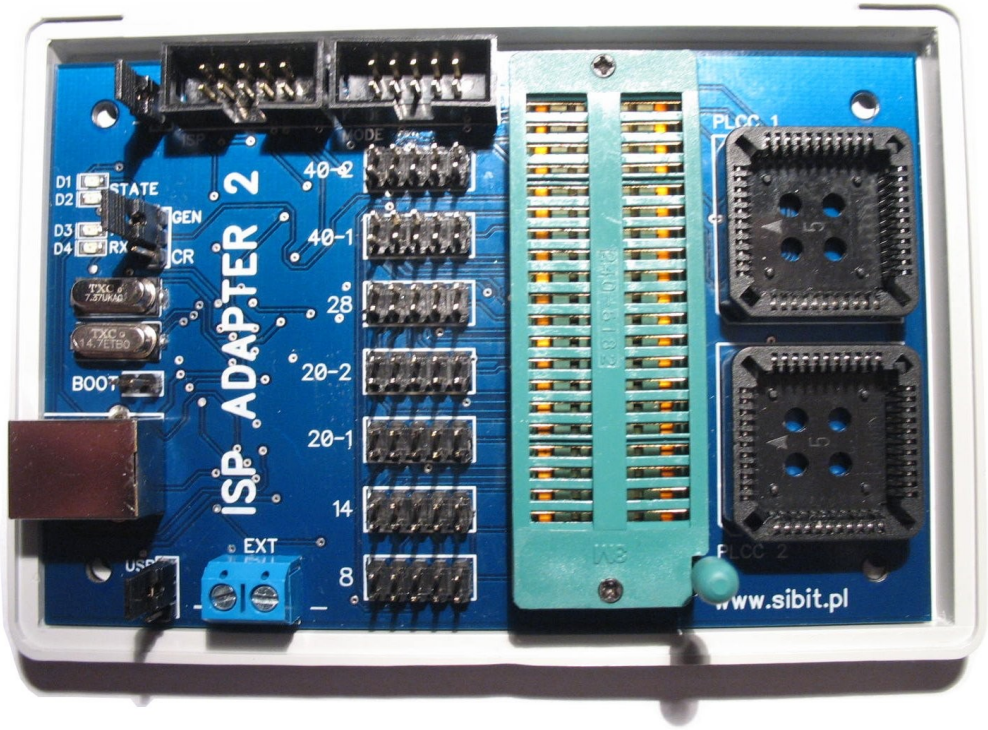
ISP ADAPTER 2

Instrukcja obsługi

rev.1.0



Copyright © 2011 SIBIT www.sibit.pl

Spis treści

1.Wprowadzenie..... 3
2. Rozmieszczenie elementów.....4
3. Wybór źródła taktowania.....6
4. Konfiguracja adaptera do pracy z podstawką ZIF.....7
4.1 DIP 8..... 8
4.2 DIP 14.....9
4.3 DIP 20A.....10
4.4 DIP 20B.....11
4.5 DIP 28.....12
4.6 DIP 40A.....13
4.7 DIP 40B.....14
5. Konfiguracja adaptera do pracy z podstawkami PLCC.....15
6. Programator ISP - programowanie w systemie.....16
6.1 Opis wyprowadzeń złącza ISP.....16
6.2 Podłączenie programatora do układu.....18
6.3 Lista obsługiwanych mikrokontrolerów.....20
7. Instalcja sterowników.....24
8. Konfiguracja USB Serial Port – zmiana numeru COM.....29
9. Konfiguracja USB Serial Port – zmiana czasu opóźnienia.....31
10. Zmiana firmware programatora - AVR Prog.....32
11. Konfiguracja oprogramowania.....34
11.1 AVRISP (AVR Studio).....34
11.2 AVR Prog (AVR Studio).....36
11.3 AVRDUDE – AVRDUDE GUI.....37
11.4 AVRDUDE – AVR8 Burn-O-Mat.....38
11.5 BASCOM-AVR.....39
11.6 AVR OSP II.....43
11.7 CodeVision.....45
12. Kontakt - wsparcie techniczne.....46

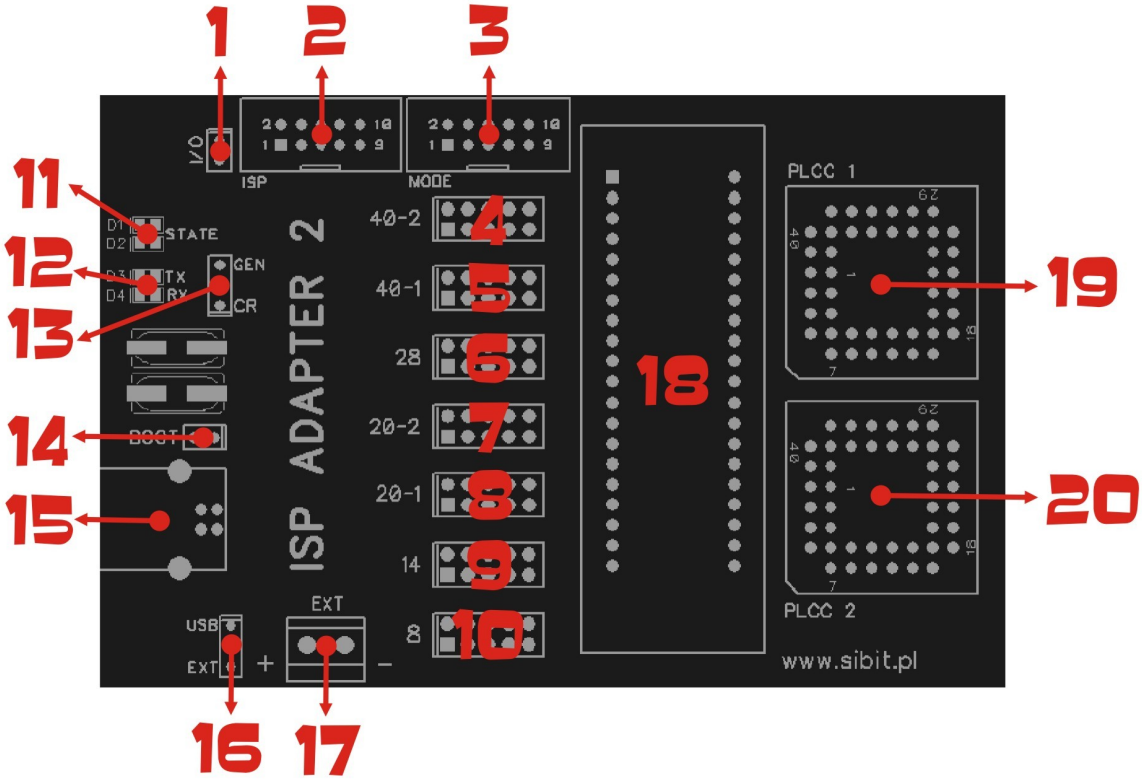
1. Wprowadzenie

ISP ADAPTER 2 jest programatorem USB mikrokontrolerów AVR. Programator umożliwia programowanie w systemie (ISP - złącze zgodne ze standardem KANDA) oraz poza systemem (mikrokontrolery w obudowach DIP od 8 do 40 wyprowadzeń oraz PLCC). Układ wyposażony jest w profesjonalną podstawkę ZIF (zero insercion force) dzięki czemu programowany mikrokontroler może być z łatwością wkładany i wyjmowany bez ryzyka uszkodzenia pinów. Programator obsługuje protokoły STK500v2 i AVR910. Dzięki zaimplementowaniu obydwu standardów urządzenie obsługuje praktycznie wszystkie ośmiobitowe mikrokontrolery AVR. Poprzez odpowiednią konfigurację (zworka) urządzenie może zasilać (z USB komputera) programowany układ co eliminuje potrzebę stosowania dodatkowych źródeł napięcia. Programowany mikrokontroler może być taktowany oscylatorem wewnętrznym, zewnętrznym oscylatorem kwarcowym lub sygnałem zegarowym (GEN). Wybór źródła sygnału dokonywany jest za pomocą zworki.

W urządzeniu zastosowano buforowanie sygnałów linii SPI co zwiększyło odporność układu na zakłócenia oraz rozszerzyło zakres tolerowanych napięć zasilania programowanego układu (1,8V - 6V).

Programator współpracuje m.in. z AVRStudio (AvrProg, AVRISP, STK500), AVR-OSP II, CodeVision, BASCOM-AVR, AVRDUDE.

2. Rozmieszczenie elementów



1 – zworka I/O (input/output). Założona zworka I/O dezaktywuje wbudowany programator a do złącza ISP (2) można podłączyć programator zewnętrzny. Zdjęcie zworki uaktywnia wbudowany programator. Złącze ISP (2) jest wówczas złączem wyjściowym adaptera i może służyć do programowania mikrokontrolerów w systemie.

2 – złącze ISP które w zależności od konfiguracji zworki I/O (1) jest wyjściem programatora (zworka I/O zdjęta) lub wejściem do podłączenia programatora zewnętrznego (zworka I/O założona).

3 – złącze MODE, służy do konfiguracji podstawki ZIF adaptera.

4 – złącze konfiguracyjne podstawki ZIF DIP40-2

5 – złącze konfiguracyjne podstawki ZIF DIP40-1

6 – złącze konfiguracyjne podstawki ZIF DIP28

7 – złącze konfiguracyjne podstawki ZIF DIP20-2

8 – złącze konfiguracyjne podstawki ZIF DIP20-1

9 – złącze konfiguracyjne podstawki ZIF DIP14

10 – złącze konfiguracyjne podstawki ZIF DIP8

11 – diody STATE oznaczające stan pracy programatora. W zależności od wgranego firmware stan pracy wyraża się następująco:

- Tryb STK500 – w stanie spoczynkowym świeci się dioda czerwona. Podczas programowania zaświeca się dioda zielona.
- Tryb AVR910 – zielony kolor oznacza stan spoczynkowy programatora. Podczas programowania świeci się dioda czerwona.
- Tryb BOOTLOADER – niezależnie od wgranego firmware, przejście programatora do sekcji bootloadera sygnalizowane jest miganiem zielonej diody.

12 – diody LED Rx i Tx. Diody sygnalizują transmisję w obydwu kierunkach : komputer->programator oraz programator->komputer.

13 – zworka CR/GEN – wybór źródła taktowania. Zworka w pozycji CR to taktowanie programowanego mikrokontrolera z oscylatora kwarcowego na adapterze. Zworka w pozycji GEN to taktowanie sygnałem zegarowym.

14 – zworka BOOT. Założona zworka w chwili podłączenia programatora do komputera powoduje uaktywnienie bootloadera. Sygnalizowane jest to miganiem zielonej diody STATE. Podczas normalnej pracy zworka powinna być zdjęta.

15 – złącze USB programatora / adaptera. Złącze komunikacyjne oraz zasilające.

16 – zworka – konfiguracja zasilania programowanego układu. Zworka powinna być założona na pozycję USB w przypadku zasilania adaptera z gniazda USB B (15). W przypadku korzystania z zewnętrznego źródła zasilania podłączonego do gniazda EXT (17) zworka powinna być ustawiona na pozycję EXT.

17 – gniazdo EXT do którego można podłączyć zewnętrzne źródło zasilania. Ten sposób zasilania może być wykorzystany w przypadku programowania mikrokontrolerów niskonapięciowych. UWAGA ! Należy zwrócić uwagę na właściwą polaryzację doprowadzonego napięcia.

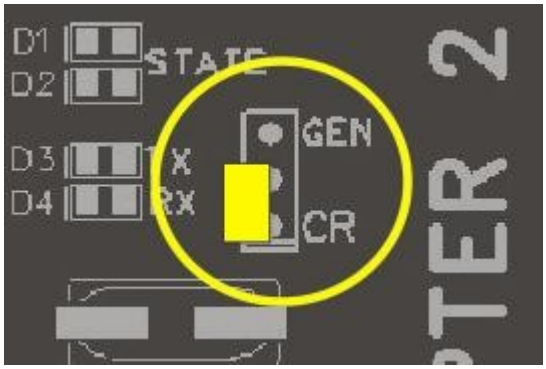
18 – podstawka ZIF

19 – podtawka PLCC1

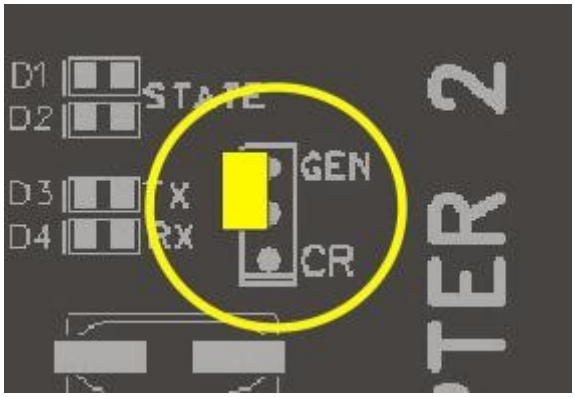
20 – podstawka PLCC2

3. Wybór źródła taktowania

Aby dołączyć umieszczony na adapterze kwarc do programowanego układu DIP lub PLCC zworka GEN/CR (13) powinna być ustawiona na pozycję CR zgodnie z poniższym rysunkiem:



W razie potrzeby taktowania programowanego mikrokontrolera zewnętrznym sygnałem zegarowym, zworka GEN/CR (13) powinna być ustawiona na pozycję GEN zgodnie z niżej zamieszczonym rysunkiem:



4. Konfiguracja adaptera do pracy z podstawką ZIF

Przed umiejscowieniem programowanego układu w podstawce ZIF, adapter należy skonfigurować do pracy z danym mikrokontrolerem. Do konfiguracji służy załączona tasiemka, której jeden koniec powinien być umieszczony w złączu MODE (3) a drugi w złączu szpilkowym (DIP8 - DIP40) właściwym dla danego mikrokontrolera.
Sposób konfiguracji i umiejscowienie układów w podstawce ZIF w zależności od typu mikrokontrolera zostało przedstawione poniżej.

Sposób taktowania programowanego mikrokontrolera określamy zworką CR/GEN. Taktowanie z kwarca - zworka w pozycji CR, taktowanie z generatora - zworka w pozycji GEN.

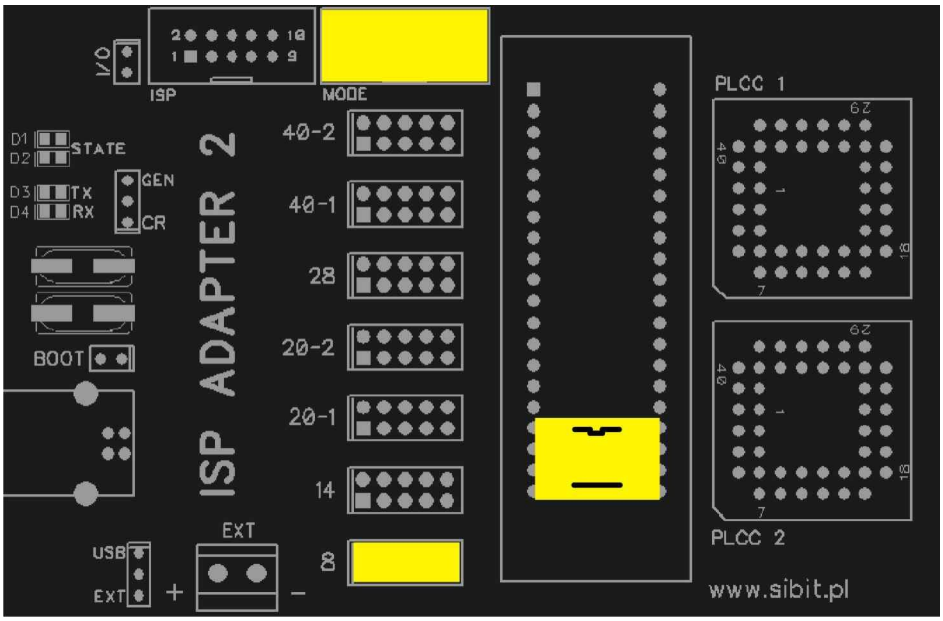
Do obsługi podstawki ZIF można wykorzystć wbudowany programator STK500v2 / AVR910 - w tym celu zworka I/O (1) powinna być zdjęta. Złącze USB (15) w tym przypadku jest złączem komunikacyjnym i zasilającym. Z USB zasilany jest wbudowany programator i programowany układ (jeśli zworka USB/EXT założona będzie na pozycję USB). Wysokość napięcia zależy od komputera i zgodnie ze standardem USB może zawierać się w przedziale 4,3V – 5,25V. Przy tym należy pamiętać o ograniczonej wydajności prądowej portu USB (ok. 500mA).
Jeżeli programowany mikrokontroler ma być zasilony napięciem o mniejszej wartości np. 1,8V - 3,3V, zworka USB/EXT powinna być ustawiona w pozycji EXT a alternatywne źródło zasilania należy doprowadzić do gniazda EXT zachowując właściwą polaryzację.

Jeśli zaistnieje potrzeba podłączenia zewnętrznego programatora (np. ze względu na nieobsłgiwany przez STK500/AVR910 typ mikrokontrolera) zworka I/O (1) powinna być założona a zewnętrzny programator (np. STK200/300) powinien być podłączony do gniazda ISP (2).

4.1 DIP 8

Lista mikrokontrolerów
obsługiwanych w trybie DIP8 :

ATTINY 12
ATTINY 13
ATTINY 15
ATTINY 22
ATTINY 25
ATTINY 45
ATTINY 85
AT90S2323
AT90S2343
AT90S2433*

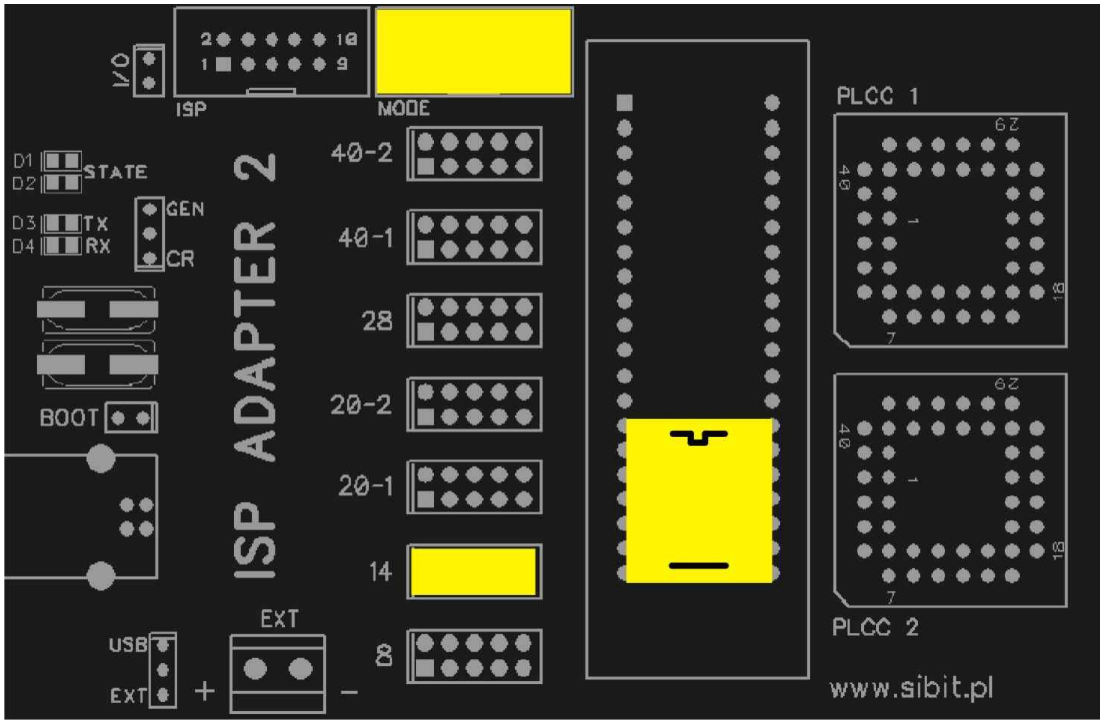


* - nieobsługiwane przez wbudowany programator. Do zaprogramowania niezbędny jest zewnętrzny programator STK200/300 podłączony do gniazda ISP (2) adaptera.

4.2 DIP 14

Lista mikrokontrolerów obsługiwanych w trybie DIP14 :

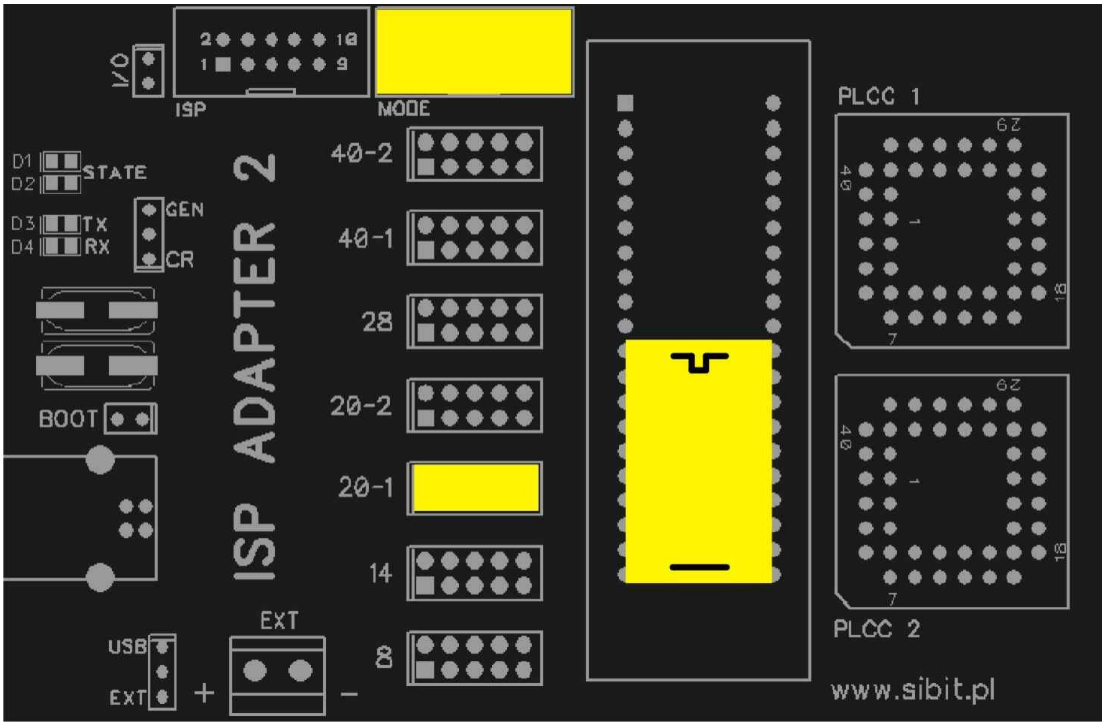
ATTINY 24
ATTINY 44
ATTINY 84



4.3 DIP 20-1

Lista mikrokontrolerów obsługiwanych w trybie DIP20-1 :

ATTINY 2313
AT90S2313
AT90S1200
AT89S2051*
AT89S4051*

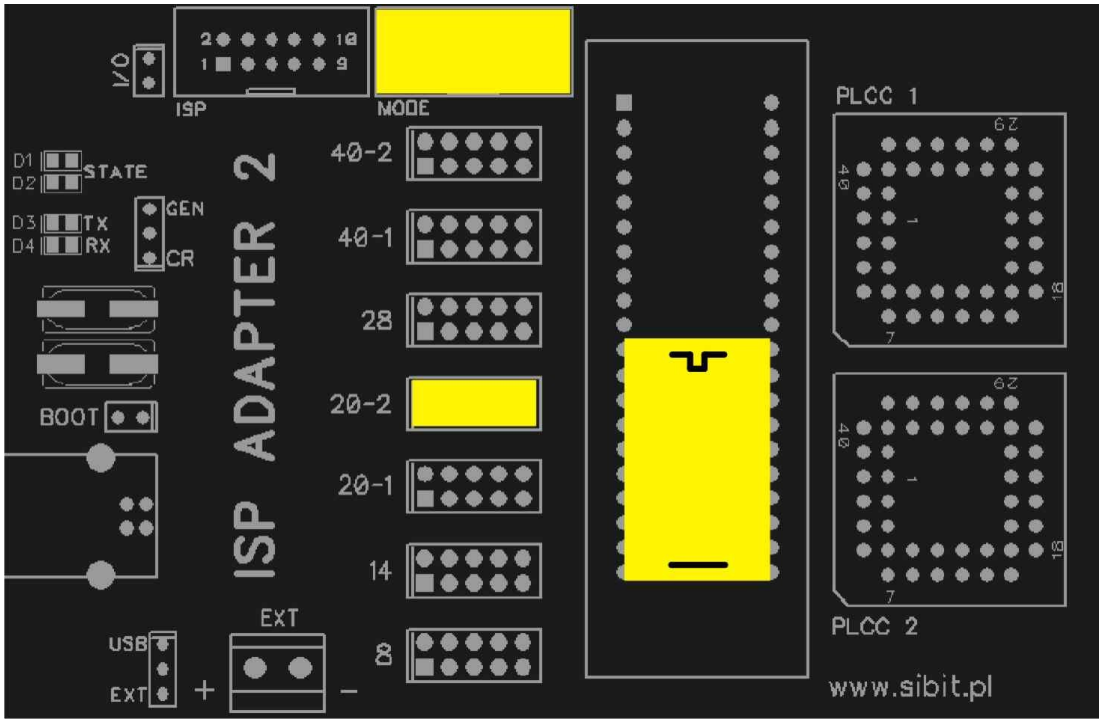


* - nieobsługiwane przez wbudowany programator. Do zaprogramowania niezbędny jest zewnętrzny programator STK200/300 podłączony do gniazda ISP (2) adaptera.

4.4 DIP 20-2

Lista mikrokontrolerów obsługiwanych w trybie DIP20-2 :

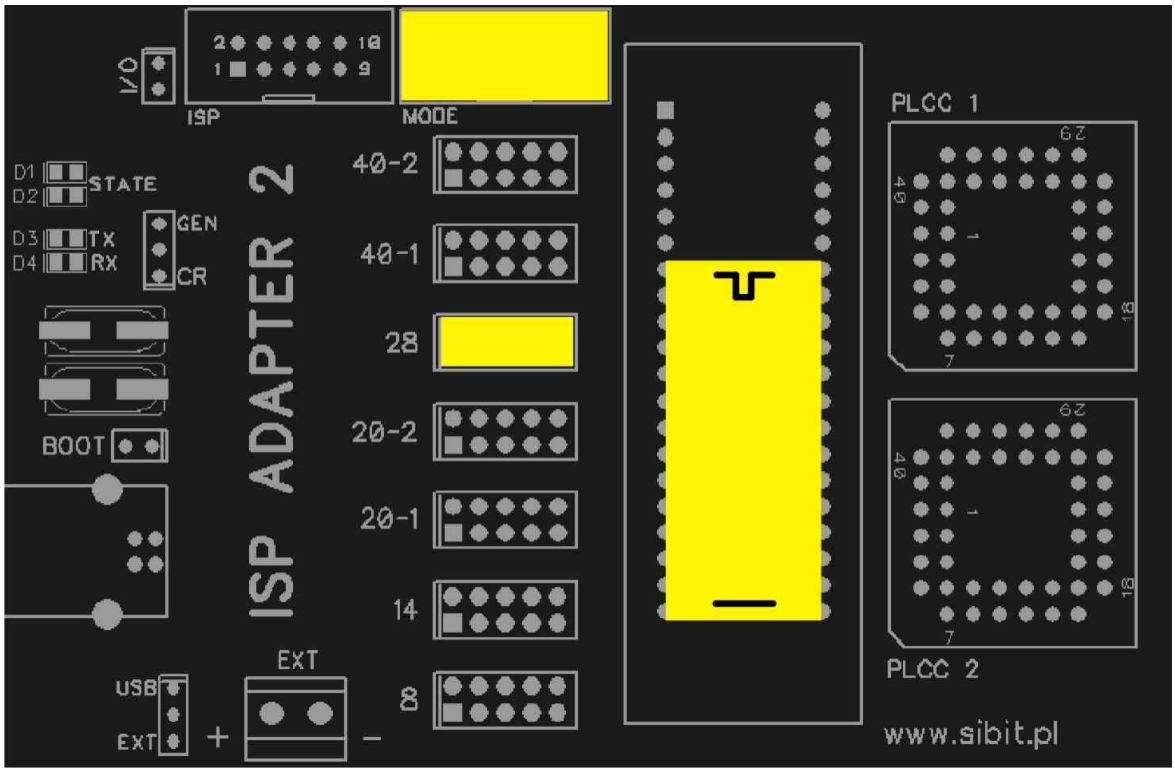
ATTINY 26
ATTINY 261
ATTINY 461
ATTINY 861



4.5 DIP 28

Lista mikrokontrolerów obsługiwanych w trybie DIP28 :

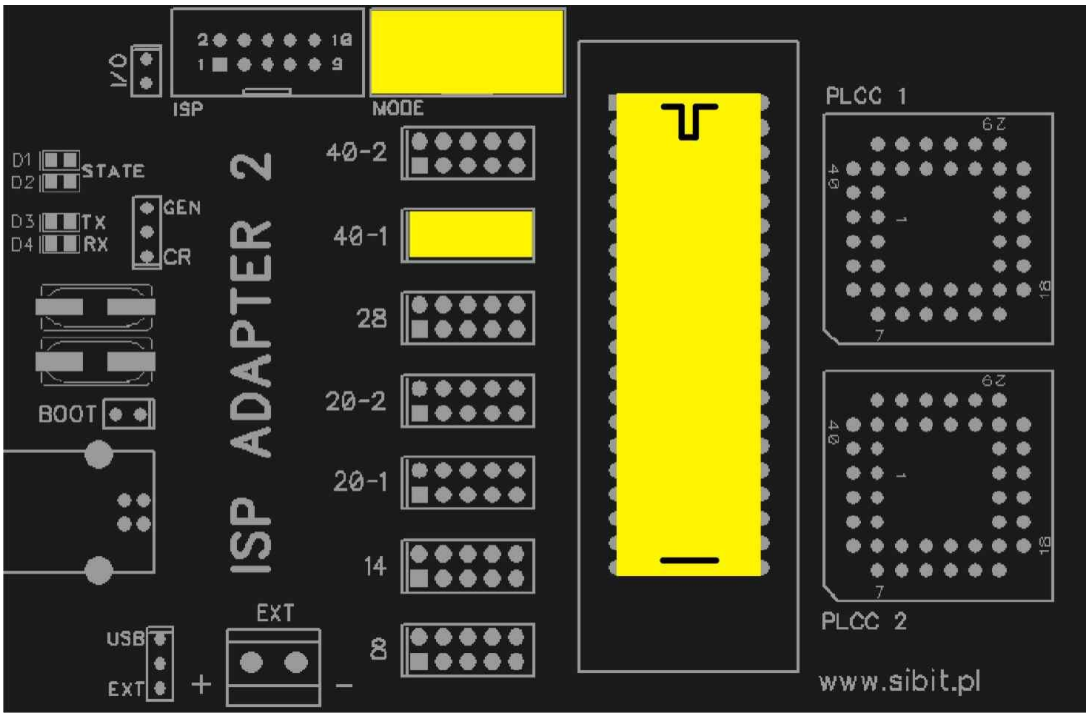
ATMEGA 8
ATMEGA 48
ATMEGA 88
ATMEGA 168
ATMEGA 328
AT90S2333
AT90S4433



4.6 DIP 40-1

Lista mikrokontrolerów obsługiwanych w trybie DIP40-1 :

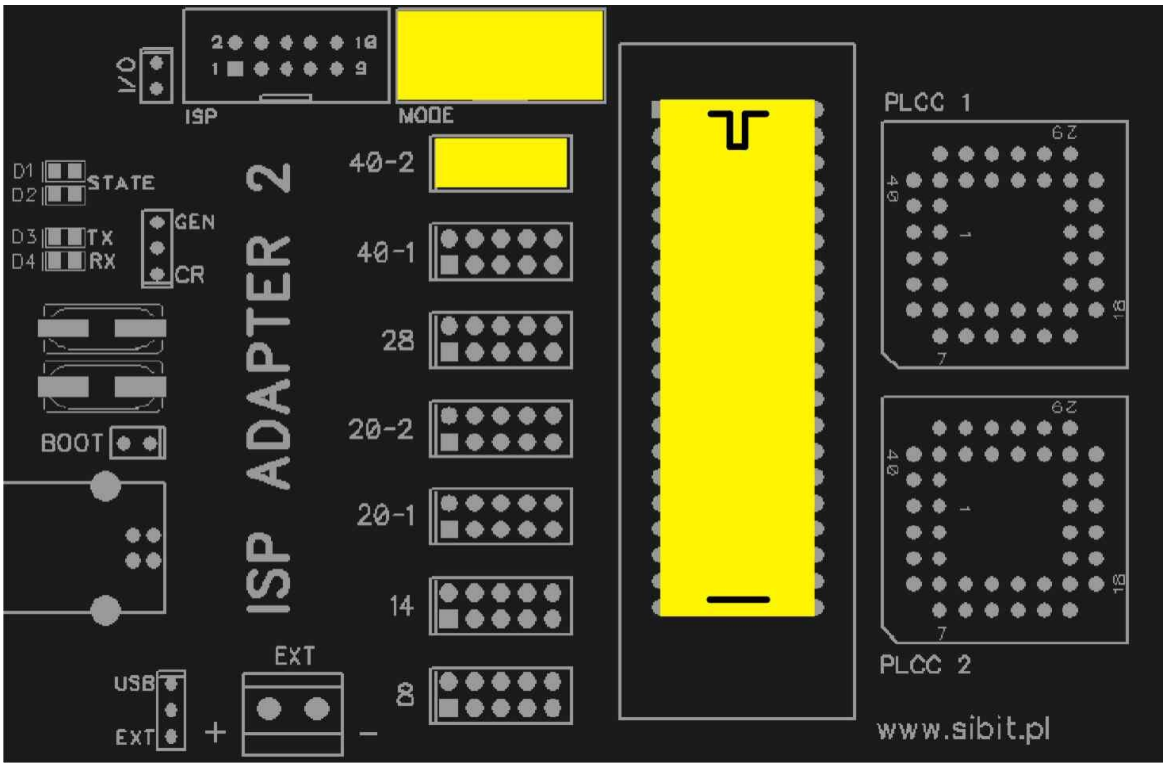
ATMEGA 16
ATMEGA 32
ATMEGA 323
ATMEGA 644
ATMEGA 8535
ATMEGA 163



4.7 DIP 40-2

Lista mikrokontrolerów obsługiwanych w trybie DIP40-2 :

ATMEGA 162
ATMEGA 8515
AT89S52*
AT89S53*
AT89S8253*
AT89S8515*



* - nieobsługiwane przez wbudowany programator. Do zaprogramowania niezbędny jest zewnętrzny programator STK200/300 podłączony do gniazda ISP (2) adaptera.

5. Konfiguracja adaptera do pracy z podstawkami PLCC

Magistrala ISP dla podstawek PLCC1 i PLCC2 jest podłączona na sztywno tzn. nie jest wymagana konfiguracja złączem MODE. Wszystkie sygnały niezbędne do programowania są podłączone na stałe. W kwesti taktowania i zasilania obowiązują te same zasady jak dla podstawki ZIF.

Lista mikrokontrolerów obsługiwanych w podstawce PLCC1:

ATMEGA 8515
AT90S4414
AT90S4434

AT89S52*
AT89S53*
AT89S8253*

* - nieobsługiwane przez wbudowany programator. Do zaprogramowania niezbędny jest zewnętrzny programator STK200/300 podłączony do gniazda ISP (2) adaptera. Przed podłączeniem zewnętrznego programatora należy pamiętać o założeniu zworki I/O (1).

Lista mikrokontrolerów obsługiwanych w podstawce PLCC2:

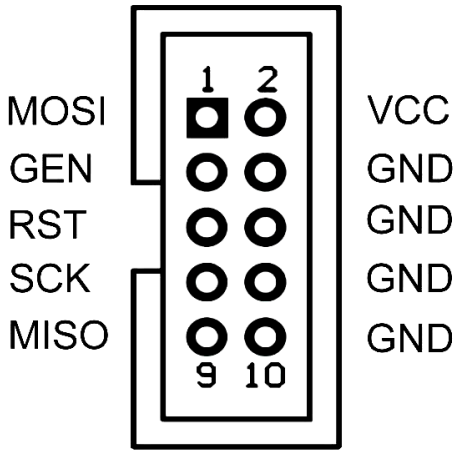
ATMEGA8535

6. Programator ISP - programowanie w systemie.

ISP ADAPTER 2 posiada wbudowany programator USB zgodny ze standandardem STK500v2 lub AVR910 w zależności od wgranego wsadu. Programator umożliwia programowanie mikrokontrolerów AVR znajdujących się w podstawkach ZIF i PLCC lub zewnętrznym systemie mikroprocesorowym - bezpośrednio w docelowym urządzeniu lub w zestawie ewaluacyjnym.

6.1 Opis wyprowadzeń złącza ISP

Złącze ISP (2) adaptera w zleżności od konfiguracji jest złączem wejściowym lub wyjściowym. Przy zdjętej zworce I/O jest to wyjście programatora, natomiast przy założonej zworce I/O jest to gniazdo wejściowe do podłączenia zewnętrznego programatora (o ile zajdzie taka potrzeba). Niezależnie od konfiguracji rozkład sygnałów jest taki sam, kompatybilny ze standardem 10 wyprowadzeniowym firmy ATMEL i KANDA.



MOSI – sygnał magistrali ISP

SCK – sygnał magistrali ISP

MISO – sygnał magistrali ISP

RST – sygnał RESET

GEN – dodatkowy, niezależny sygnał zegarowy. Może być wykorzystywany podczas przeprogramowywania mikrokontrolerów z ustawionym fuse-bitem na taktowanie zewnętrzne (SUT_CKSEL → Ext. Clock).

VCC – Przy założonej zworce USB/EXT (16) pin VCC jest pinem wyjściowym i może służyć do zasilenia programowanego w zewnętrznym systemie mikrokontrolera. Dla zworki w pozycji USB wysokość napięcia na pinie VCC zgodnie ze standardem USB w komputerach może zawierać się w przedziale 4,3V – 5,25V. Przy tym należy pamiętać o ograniczonej wydajności prądowej portu USB (ok. 500mA). Dla zworki w pozycji EXT napięcie na nodze VCC uzależnione jest od wysokości napięcia doprowadzonego do gniazda EXT.
Jeśli zworka USB/EXT jest całkowicie zdjęta, do pinu VCC niezbędne jest doprowadzenie napięcia zasilania z programowanego układu. Napięcie to służy do zasilenia bufora wyjściowego programatora i powinno zawierać się w przedziale od 1.8V do 6V.

GND – masa układu

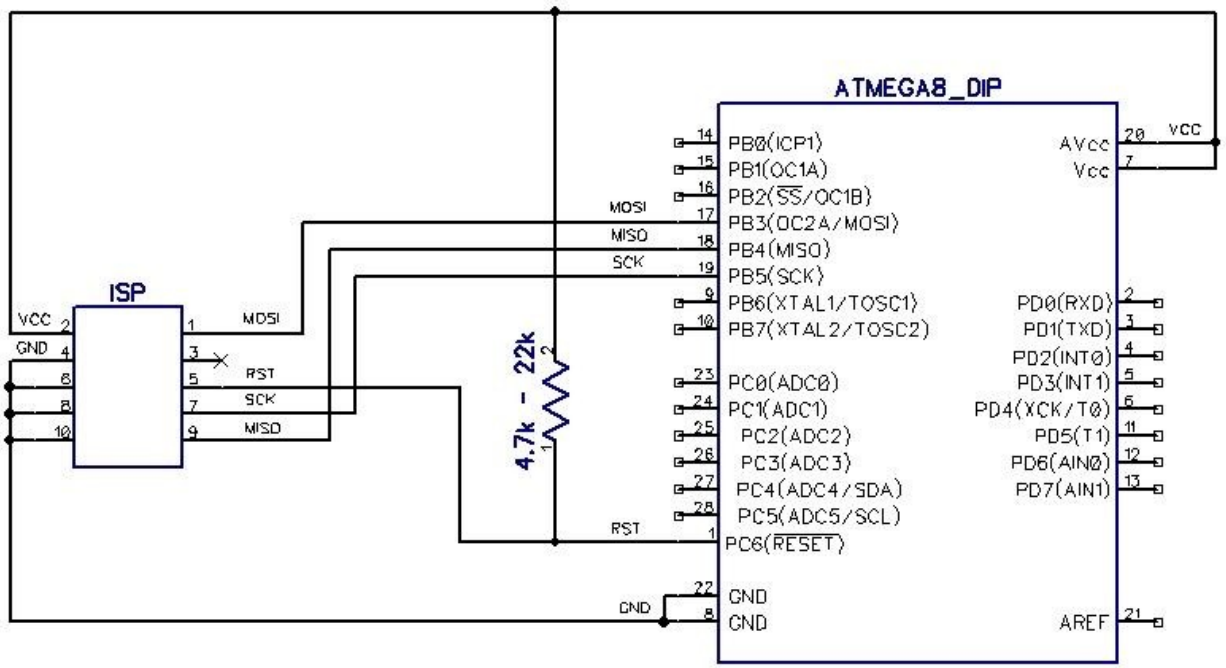
6.2 Podłączenie programatora do układu

Sygnały z programatora należy podłączyć z odpowiednimi liniami interfejsu ISP programowanego układu. Rozkład portów odpowiedzialnych za programowanie w systemie należy wcześniej sprawdzić w dokumentacji do swojego mikrokontrolera (Serial Programming Pin Mapping). W niektórych mikroprocesorach magistrala ISP pokrywa się ze sprzętową magistralą SPI (np. Atmega8), w innych magistrala do programowania ulokowana jest pod „regularnymi" portami (np. Atmega 128). Podłączając mikrokontroler do programatora należy więc zwrócić uwagę że nie zawsze ISP jest tożsame z SPI.

Łączymy odpowiednio:

Złącze ISP programatora	Magistrala ISP mikrokontrolera
MOSI	MOSI
MISO	MISO
SCK	SCK
RST	RESET
VCC*	VCC*
GND	GND

*** Należy pamiętać o konfiguracji zworki USB/EXT. W przypadku zasilania programowanego układu z USB zworka ma być założona w pozycji USB. Zworka USB/EXT musi zdjęta jeśli mikrokontroler zasilany jest ze swojego źródła.**



Połączenie programatora z mikrokontrolerem na przykładzie Atmega8 DIP

Przy pierwszym podłączeniu, w celu poprawnego zainstalowania sterowników, do komputera należy podłączyć sam programator (bez programowanego układu). Dopiero po ukończonym procesie instalacji i restarcie programatora można przejść do dalszego etapu.

W trybie zasilania z USB (zworka USB/EXT w pozycji USB) korzystnie jest w pierwszej kolejności zestawić połączenie między programatorem a układem programowanym (przy odłączonym USB), upewnić się co do poprawności a dopiero później podłączyć całość do USB komputera.

6.3 Lista obsługiwanych mikrokontrolerów

Protokół STK500v2 jest niezależny od sprzętu. Algorytmy obsługi poszczególnych mikrokontrolerów zaszyte są w oprogramowaniu sterującym na komputerze. Co za tym idzie, lista obsługiwanych przez programator układów jest tożsama z „pulą” mikrokontrolerów obsługiwanych przez daną aplikację.

Podstawowa lista wygląda następująco:

MIKROKONTROLER	TRYB PRACY (FIRMWARE)	
	STK500V2	AVR910
AT90CAN128	X	X
AT90CAN32	X	
AT90CAN64	X	
AT90PWM2	X	X
AT90PWM216	X	
AT90PWM2B	X	
AT90PWM3	X	X
AT90PWM316	X	
AT90PWM3B	X	
AT90S1200	X	X
AT90S2313	X	X
AT90S2323	X	X
AR90S2333		X
AT90S2343	X	X
AT90S4414	X	X
AT90S4433	X	X
AT90S4434	X	X
AT90S8515	X	
AT90S8515A		X
AT90S8534		X
AT90S8535	X	X
AT90S8544		X
AT90USB1286	X	

AT90USB1287	X	
AT90USB162	X	
AT90USB646	X	
AT90USB647	X	
AT90USB82	X	
ATmega103	X	X
ATmega128	X	X
ATmega1280	X	X
ATmega1281	X	X
ATmega16	X	X
ATmega161	X	X
ATmega162	X	X
ATmega163	X	X
ATmega164		X
ATmega164P	X	
ATmega165	X	
ATmega165P	X	
ATmega168	X	X
ATmega168P	X	
ATmega169	X	X
ATmega169P	X	
ATmega16HVA	X	
ATmega2560	X	
ATmega2561	X	
ATmega32	X	X
ATmega323	X	
ATmega324		X
ATmega324P	X	
ATmega325	X	
ATmega3250	X	
ATmega3250P	X	
ATmega325P	X	
ATmega328P	X	
ATmega329	X	X

ATmega3290	X	X
ATmega3290P	X	
ATmega329P	X	
ATmega406	X	
ATmega48	X	X
ATmega48P	X	
ATmega64	X	X
ATmega640	X	X
ATmega644	X	
ATmega644P	X	
ATmega645	X	
ATmega6450	X	
ATmega649	X	X
ATmega6490	X	X
ATmega8	X	X
ATmega8515	X	X
ATmega8535	X	X
ATmega88	X	X
ATmega88P	X	
ATtiny10		X
ATtiny11	X	
ATtiny12	X	X
ATtiny13	X	X
ATtiny15	X	X
ATtiny22	X	
ATtiny2313	X	X
ATtiny24	X	X
ATtiny25	X	X
ATtiny26	X	X
ATtiny261	X	
ATtiny28	X	
ATtiny43U	X	
ATtiny44	X	X
ATtiny45	X	X

ATtiny461	X	
ATtiny48	X	
ATtiny84	X	X
ATtiny85	X	X
ATtiny861	X	

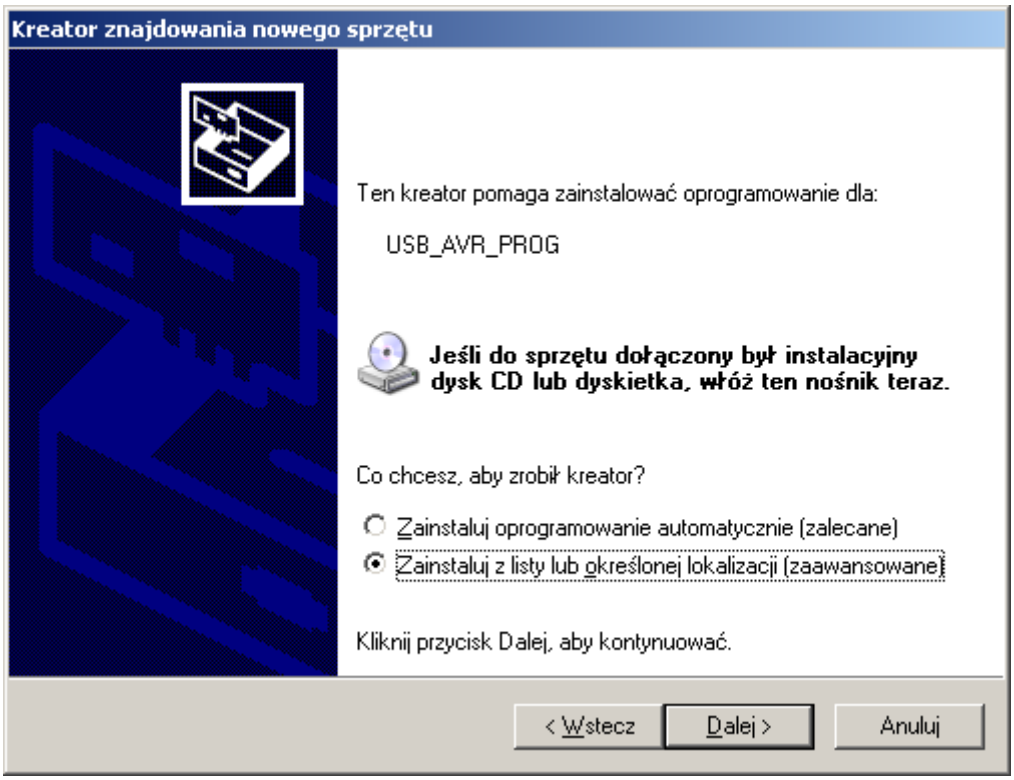
7. Instalacja sterowników

Dokument ten zawiera opis procesu instalacji sterowników w systemie Windows XP.

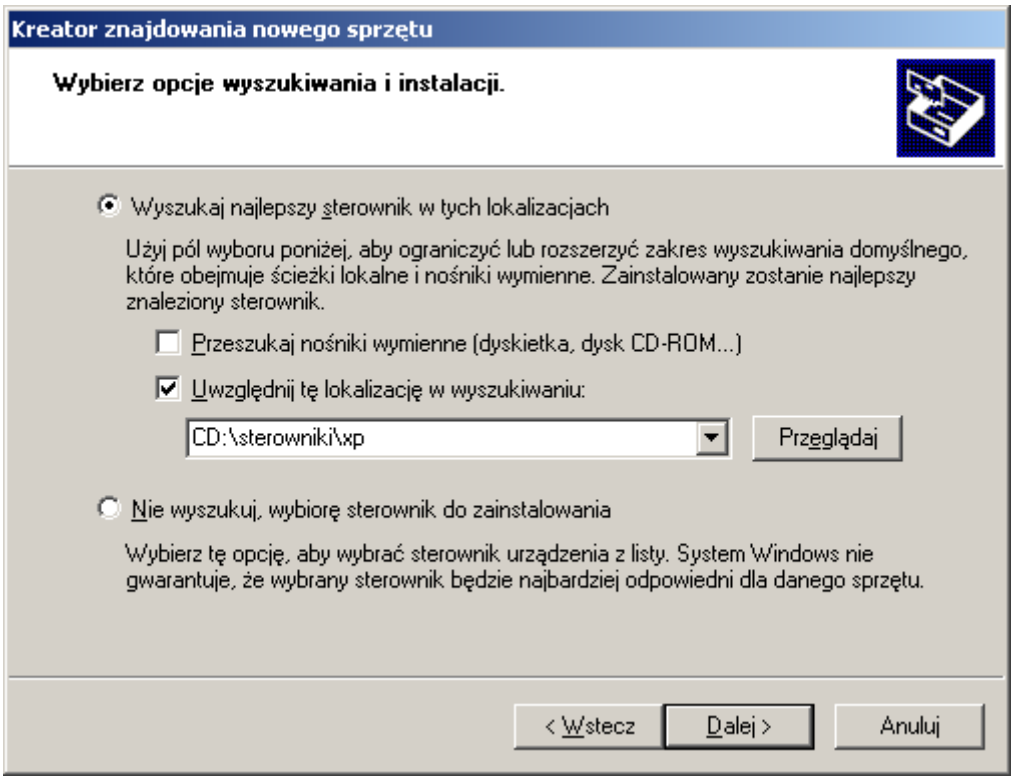
Instrukcje instalacji sterowników FTDI dla innych systemów (Windows Vista, MAC, Windows 2000, Windows 98, Linux) znajdują się na dołączonym CD (CD:\sterowniki\installation guide).
Zawsze aktualne wersje instrukcji instalacji dostępne są również na stronie producenta układu FT232R pod adresem:
http://www.ftdichip.com/Documents/InstallGuides.htm

Programator podłączamy do komputera. Po wykryciu urządzenia zostanie uruchomiony kreator znajdowania nowego sprzętu. W kreatorze wybieramy opcję '*Nie, nie tym razem*' i klikamy '*Dalej*'.

W następnej planszy wybieramy opcję '*Zainstaluj z listy lub określonej lokalizacji (zaawansowane)*' i przechodzimy dalej.

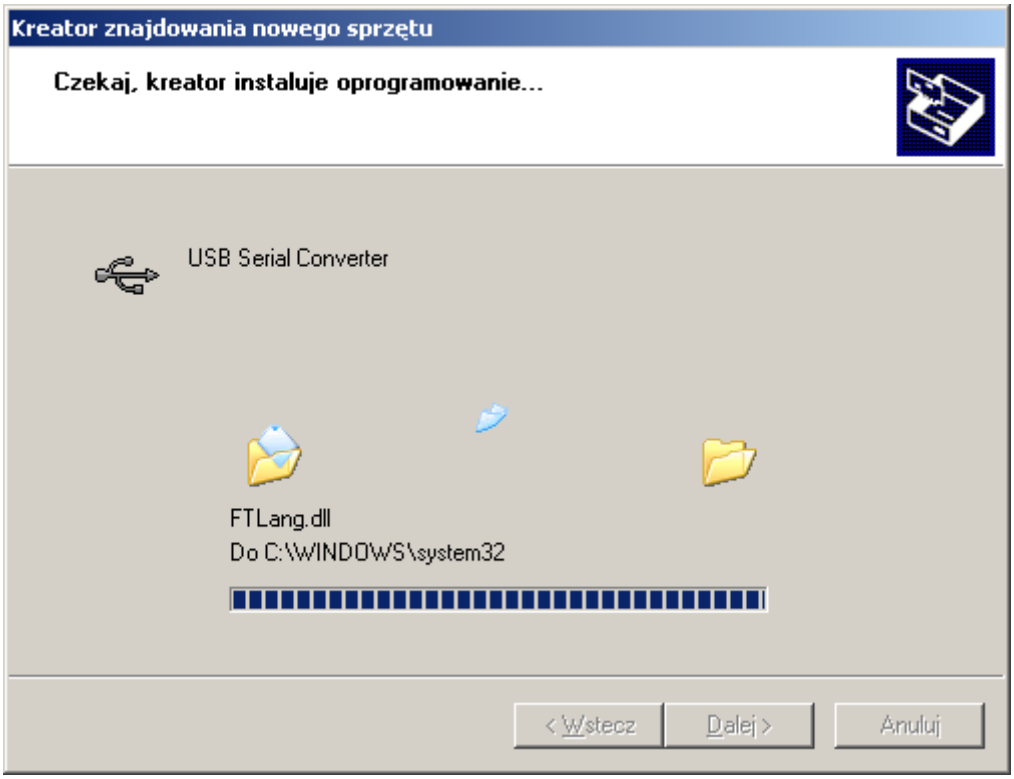


Następnie należy wskazać lokalizację sterowników

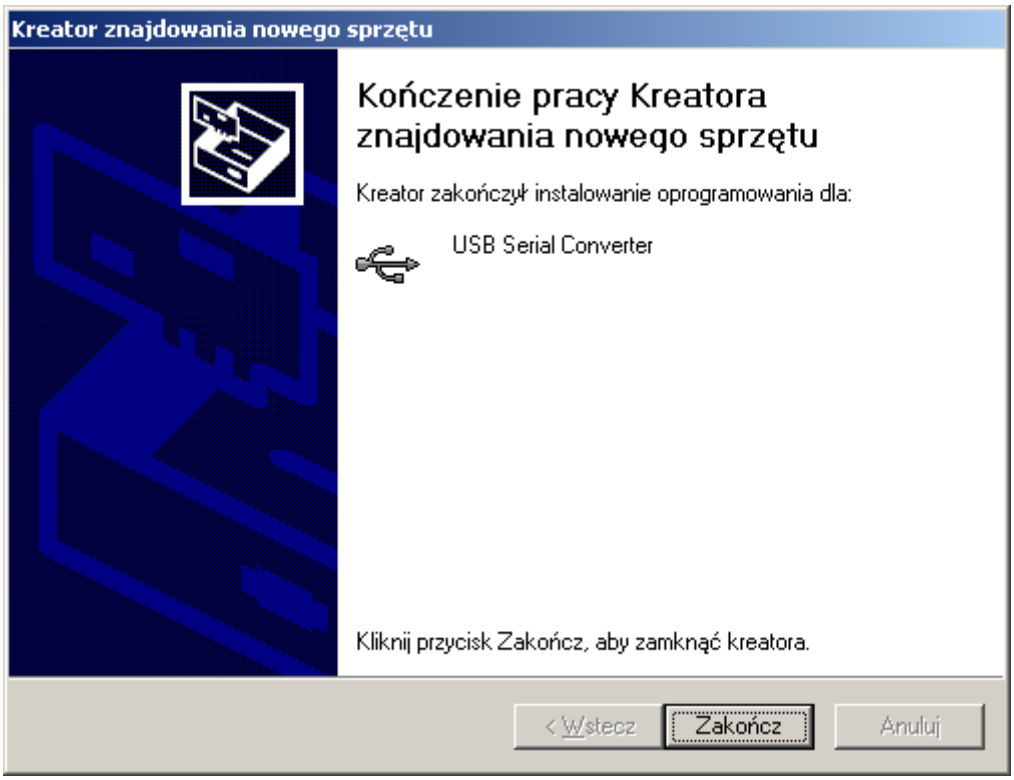


W planszy która się pojawi wybieramy opcję '*Wyszukaj najlepszy sterownik w tych lokalizacjach*' oraz zaznaczamy '*Uwzględnij tę lokalizację w wyszukiwaniu* '. Wciskamy '*Przeglądaj* ' i wskazujemy lokalizację z płyty – \sterowniki\xp

System powinien znaleźć i zainstalować sterowniki z podanej lokalizacji.



Po wyświetleniu okna końca pracy kreatora należy wcisnąć przycisk '*Zakończ*'.



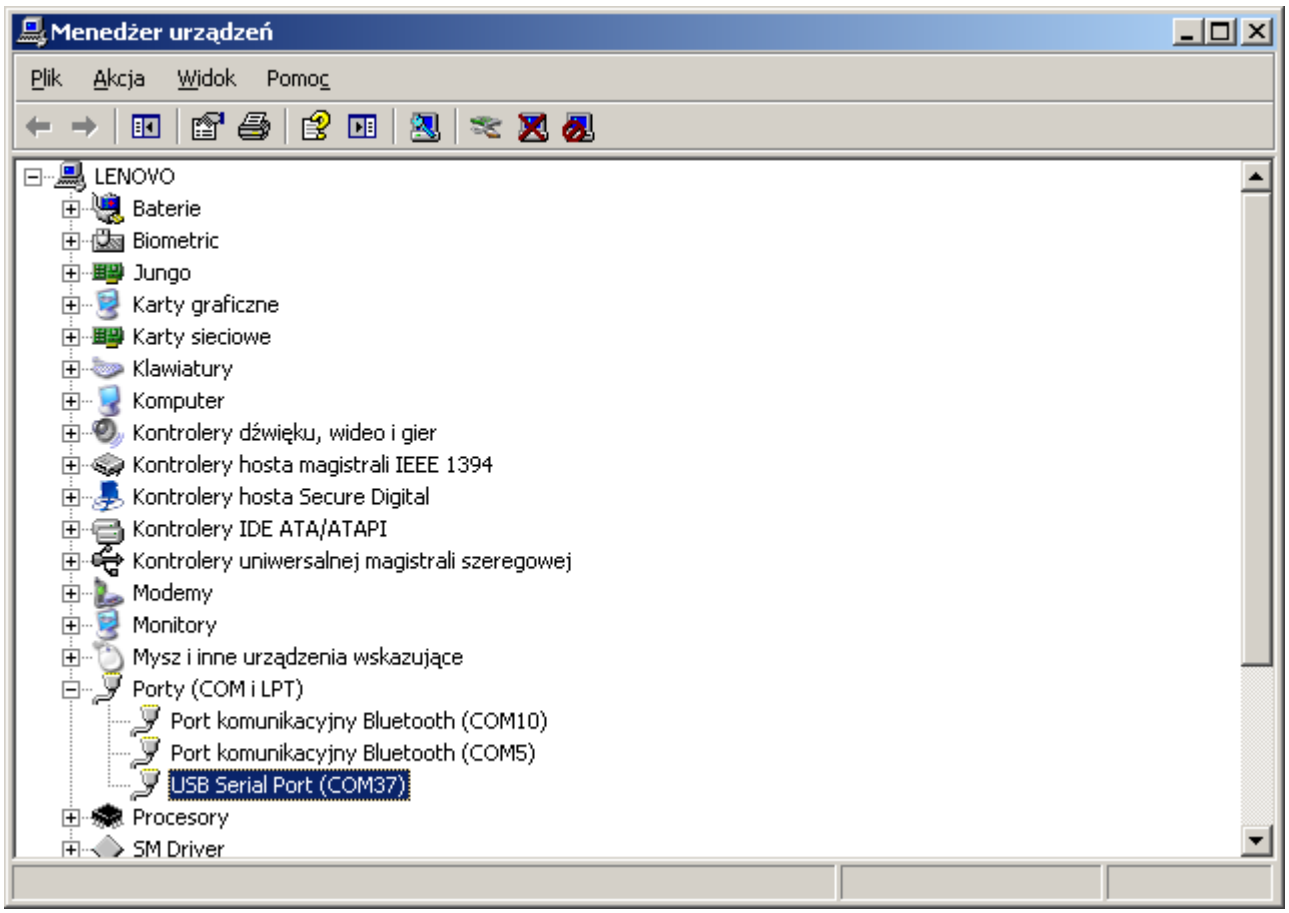
Chwilę potem system wykryje kolejne urządzenie – '*USB Serial Port*'. Proces instalacji tego urządzenia przebiega identycznie jak poprzednio. W kolejnych etapach pracy kreatora zaznaczamy te same opcje co poprzednio, a do sterowników podajemy tą samą lokalizację z płyty.

O prawidłowym zainstalowaniu sterowników system poinformuje nas komunikatem '*Twój nowy sprzęt jest zainstalowany i gotowy do użycia*'.

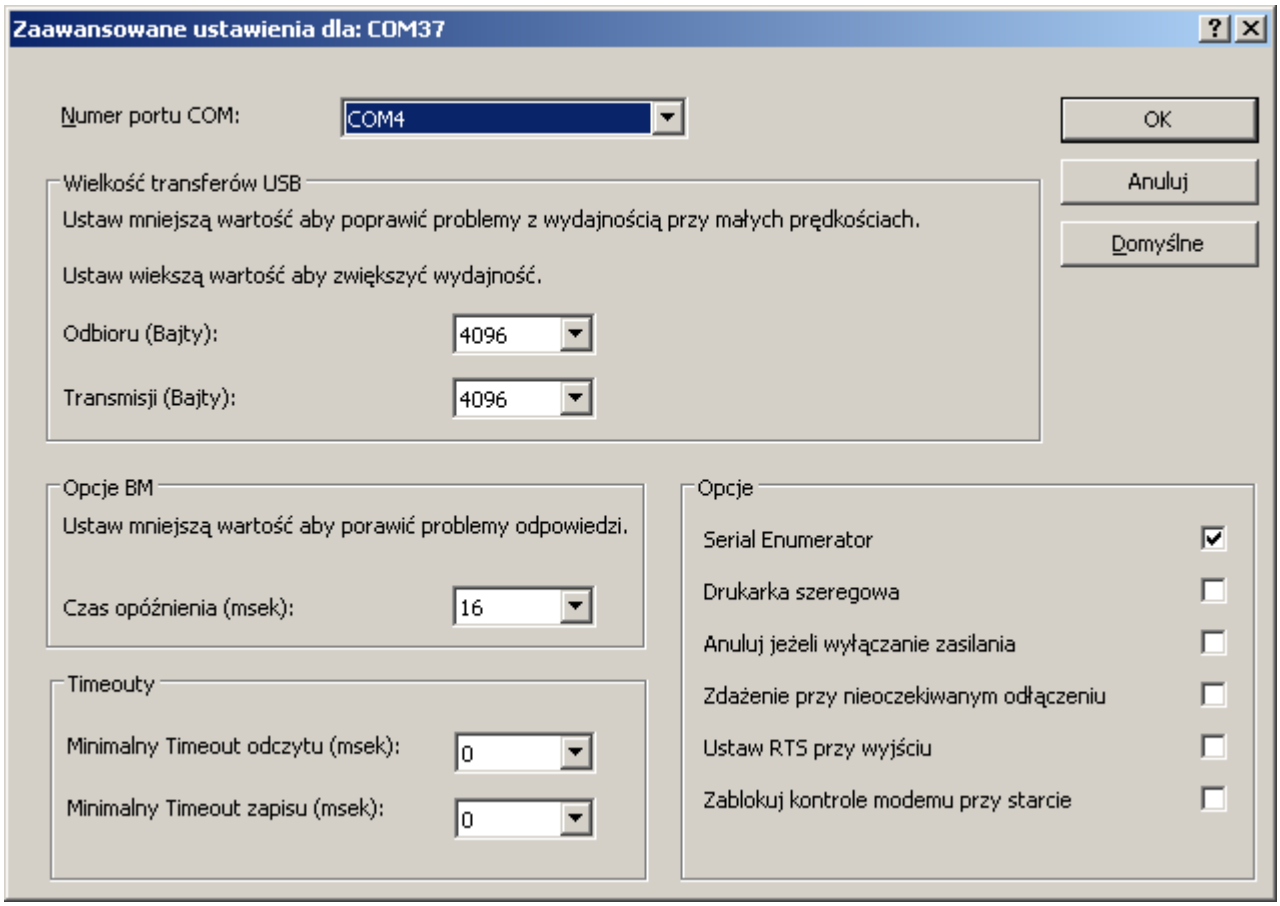
8. Konfiguracja USB Serial Port – zmiana numeru COM

Przy współpracy z niektórymi programami może zaistnieć potrzeba zmiany numeru portu COM który zajmuje programator. Np AVR Studio odpytuje tylko początkowe porty komputera COM1 – COM4. Chcąc zarejestrować programator na którymś z początkowych portów COM należy wykonać następujące czynności.

W pierwszej kolejności należy otworzyć menadżer urządzeń i rozwinąć gałąź '*Porty(COM i LPT)*'. Interesuje nas pozycja '*USB Serial Port*' z numerem COM-a w nawiasie. Jeśli występujący tu numer jest dosyć wysoki, korzystnie jest zmienić na jeden z pierwszych tj. COM1 – COM4.



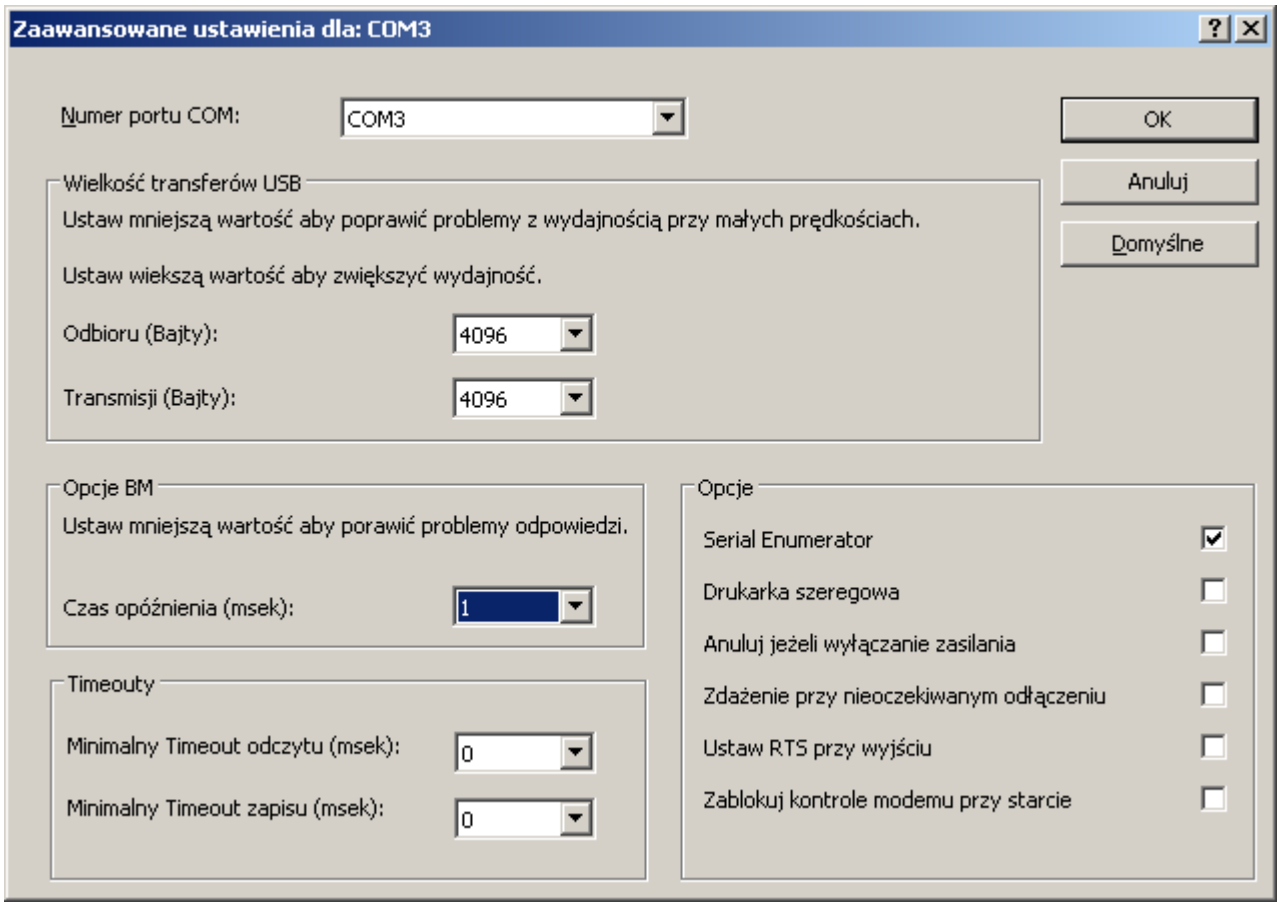
Chcąc zmienić numer COM-a otwieramy prawym przyciskiem myszy właściwości USB Serial Port-u. Przechodzimy do zakładki '*Ustawienia Portu*' i naciskamy '*Zawansowane*'. W polu '*Numer portu COM*' wybieramy któryś z początkowych portów, najlepiej wolny. Jeśli obok interesującego nas numeru widnieje napis '*(w użyciu)*' możemy to zignorować pod warunkiem że fizycznie na tym COM-ie nie znajduje się żadne urządzenie.



Klikamy '*OK*' i po kilku sekundach odłączamy interfejs od komputera. Po ponownym podłączeniu konwerter powinien ulokować się na wskazanym numerze portu COM.

9. Konfiguracja USB Serial Port – zmiana czasu opóźnienia

Znaczną poprawę wydajności programatora można uzyskać zmniejszając w konfiguracji portu COM opóźnienia. W tym celu należy otworzyć menadżer urządzeń i rozwinąć gałąź '*Porty(COM i LPT)*'. Interesuje nas pozycja '*USB Serial Port*'. Otwieramy (prawy przyciskiem myszy) właściwości USB Serial Port-u. Przechodzimy do zakładki '*Ustawienia Portu*' i naciskamy '*Zawansowane*'.
W sekcji '*Opcje BM*' zmniejszamy czas opóźnienia z wartości domyślnej (16ms) na niższą.

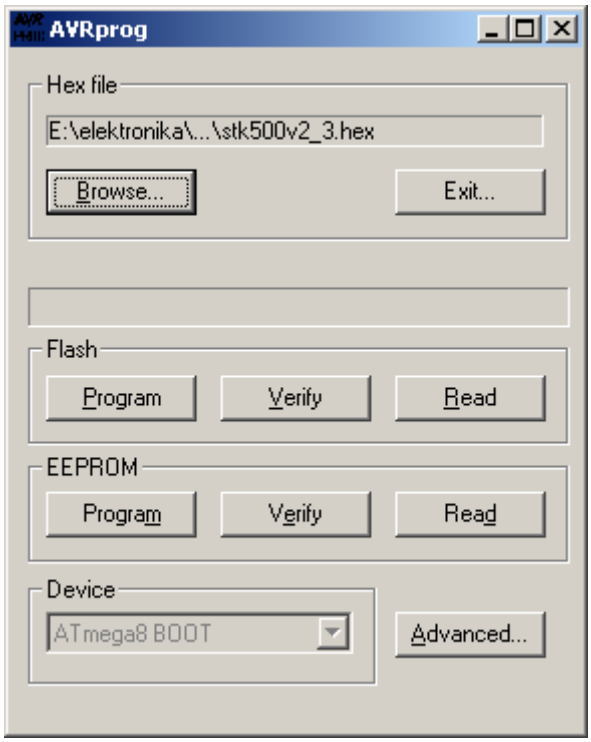


Wartość tą należy dobrać doświadczalnie – testując po każdej zmianie komunikację z programatorem. Teoretycznie, minimalna wartość przy której uda się uzyskać poprawną transmisję może zależeć od konkretnego programu, w praktyce jednak na nowszych komputerach udaje się zejść z opóźnieniem do1ms na stałe.
W przypadku pojawienia się problemów z komunikacją należy wrócić do ustawień domyślnych.

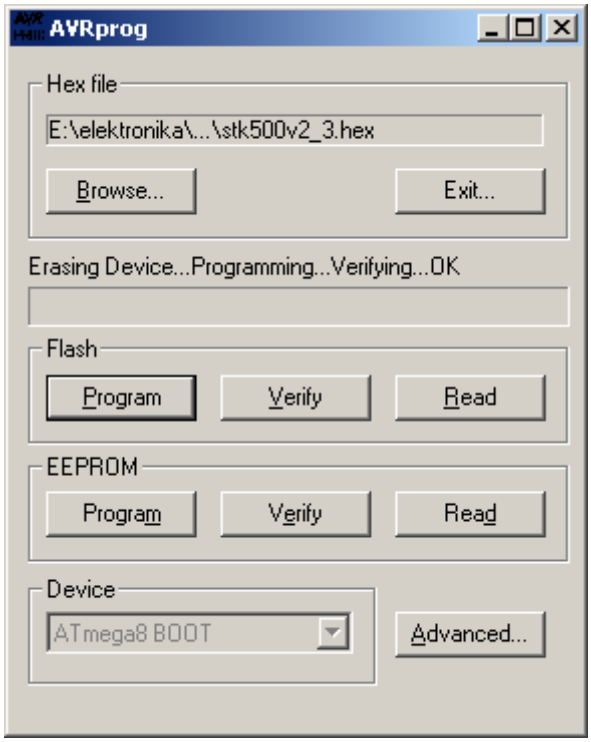
10. Zmiana firmware programatora – AVR Prog

Firmware programatora może być zmieniany programem AVR Prog wchodzącym w skład pakietu AVR Studio.
W celu wgrania wsadu należy wykonać następujące czynności:

- odłączyć (jeśli był podłączony) ISP ADAPTER od komputera
- założyć zworkę BOOT (14)
- podłączyć ISP ADAPTER do komputera
- programator zostanie uruchomiony w trybie bootloadera – powinna migać zielona dioda STATE
- uruchomić program AVR Prog (‘*CD:\SOFTWARE\AvrProg.exe*’). Po uruchomieniu programu powinna przestać migać zielona dioda. Jeśli zamiast programu pojawia się okno z informacją *'No supported board found !'* należy sprawdzić na którym porcie COM zainstalowany jest programator. Aplikacja AVR Prog wymaga umiejscowienia programatora na COM1, COM2 lub COM3 (zmiana konfiguracji opisana jest w punkcie *Konfiguracja USB Serial Port).* Po uruchomieniu programu pojawi się okno:



- klikając przycisk *'Browse...'* należy wskazać odpowiedni wsad czyli np. plik stk500v2_12.hex (*CD:\FIRMWARE\stk500v2_12.hex*) lub avr910_6.hex (*CD:\FIRMWARE\avr910_6.hex*)
- w sekcji *'Flash'* programu nacisnąć przycisk *'Program'*
- o pomyślnym przebiegu operacji flashownia poinformuje nas komunikat *'Erasing Device...Programming...Veryfing..OK'* nad wskaźnikiem postępu.



- po skończonej operacji upgrade'u zamykamy program (krzyżyk w górnym prawym rogu), zdejmujemy zworkę BOOT oraz odłączamy programator od USB komputera.
- po podłączeniu do USB, programator jest gotowy do pracy.

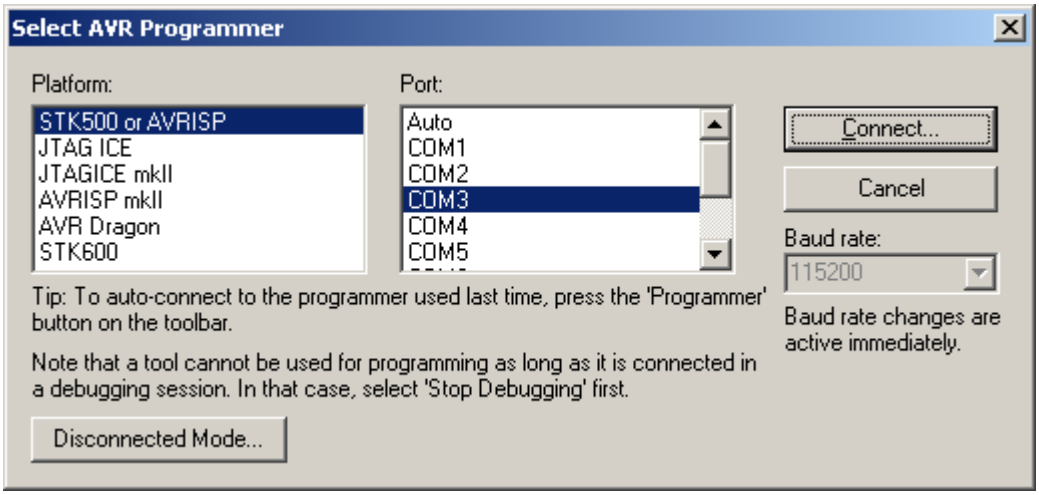
11. Konfiguracja oprogramowania

11.1 AVRISP (AVR Studio)

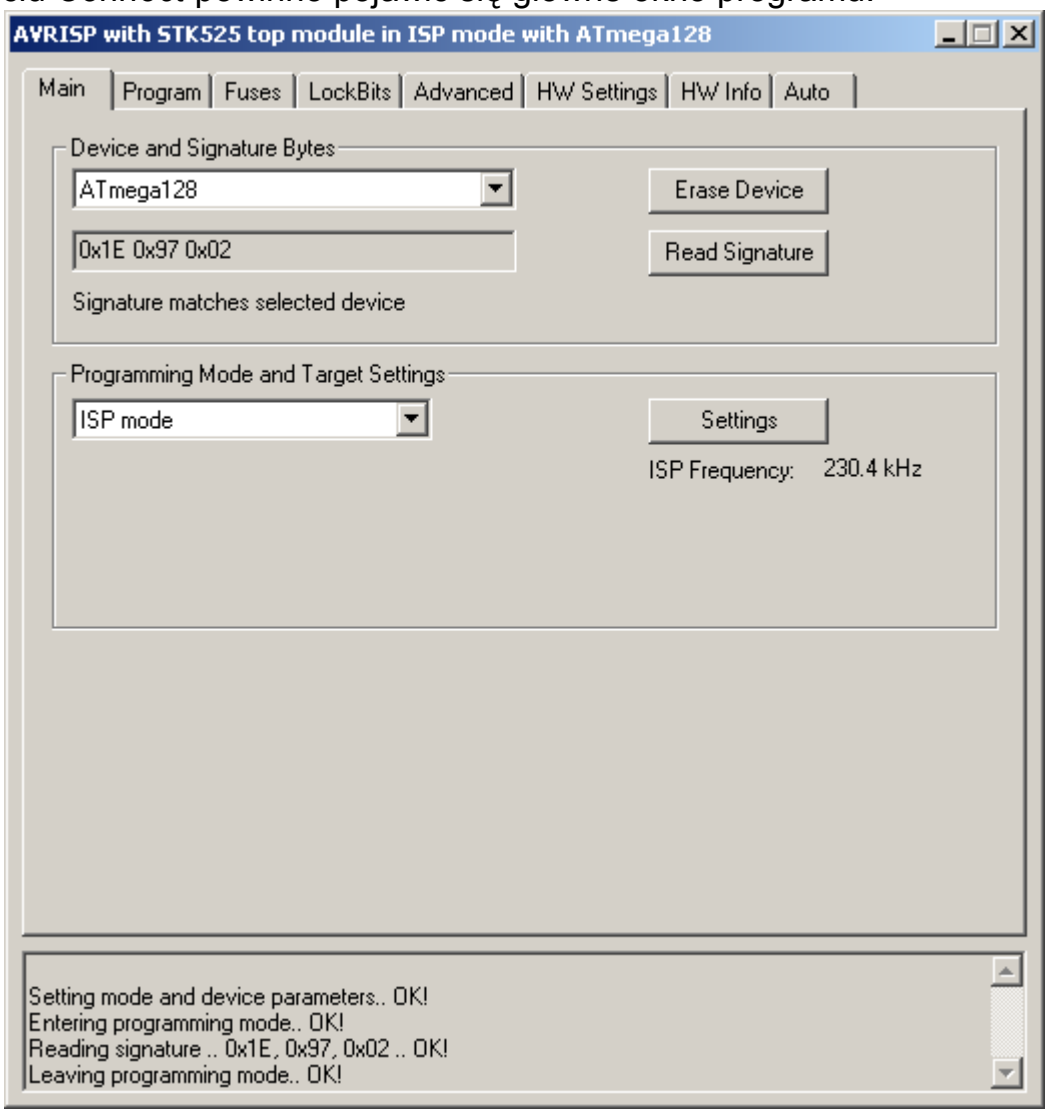
Do programatora należy wgrać firmware STK500v2.

W celu połączenia z programatorem wybieramy AVR Studio -> Tools -> Program AVR -> Connect.

W oknie *Select AVR Programmer* jako *Platform* wybieramy pierwszą pozycję czyli *STK500 or AVRISP* oraz numer portu na którym zainstalowany jest programator.



Po kliknięciu *Connect* powinno pojawić się główne okno programu.

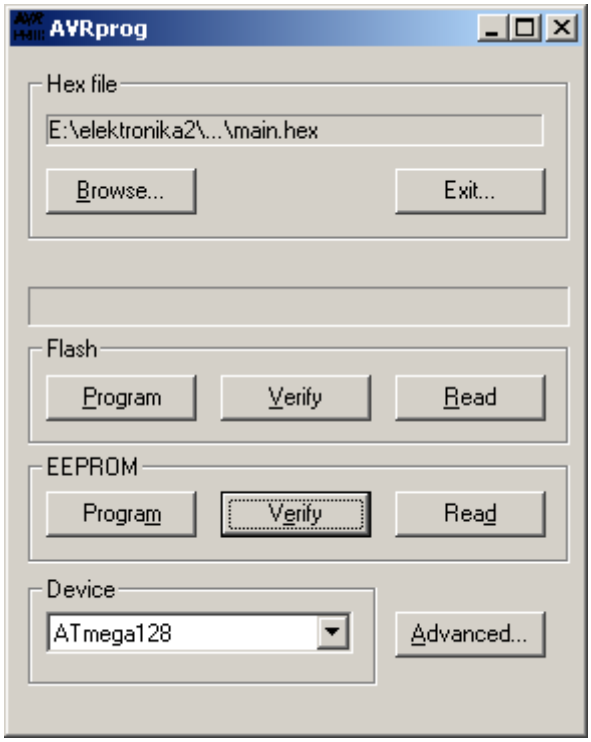


Uwaga ! Na chwilę obecną AVR Studio nie posiada autodetekcji typu podłączonego do programatora mikrokontrolera, dlatego bardzo ważne jest, żeby przed przystąpieniem do jakichkolwiek operacji związanych z programowaniem, w polu *Device and Signature Bytes* zakładki *Main* ręcznie wybrać z rozwijanej listy nazwę docelowego mikroprocesora.
Wciskając przycisk *Read Signature* należy sprawdzić czy prawidłowo jest rozpoznawana sygnatura – *Signature matches selected device*. Jeśli odczytana sygnatura nie pasuje do typu wybranego mikrokontrolera – *WARNING:Signature does not match selected device!,* należy upewnić się czy prawidłowo został wybrany typ mikrokontrolera. W następnej kolejności należy zwrócić uwagę na właściwe zasilanie układu oraz zweryfikować poprawność połączeń mędzy programatorem a mikrokontrolerem.

11.2 AVR Prog (AVR Studio)

Do programatora należy wgrać firmware AVR910.

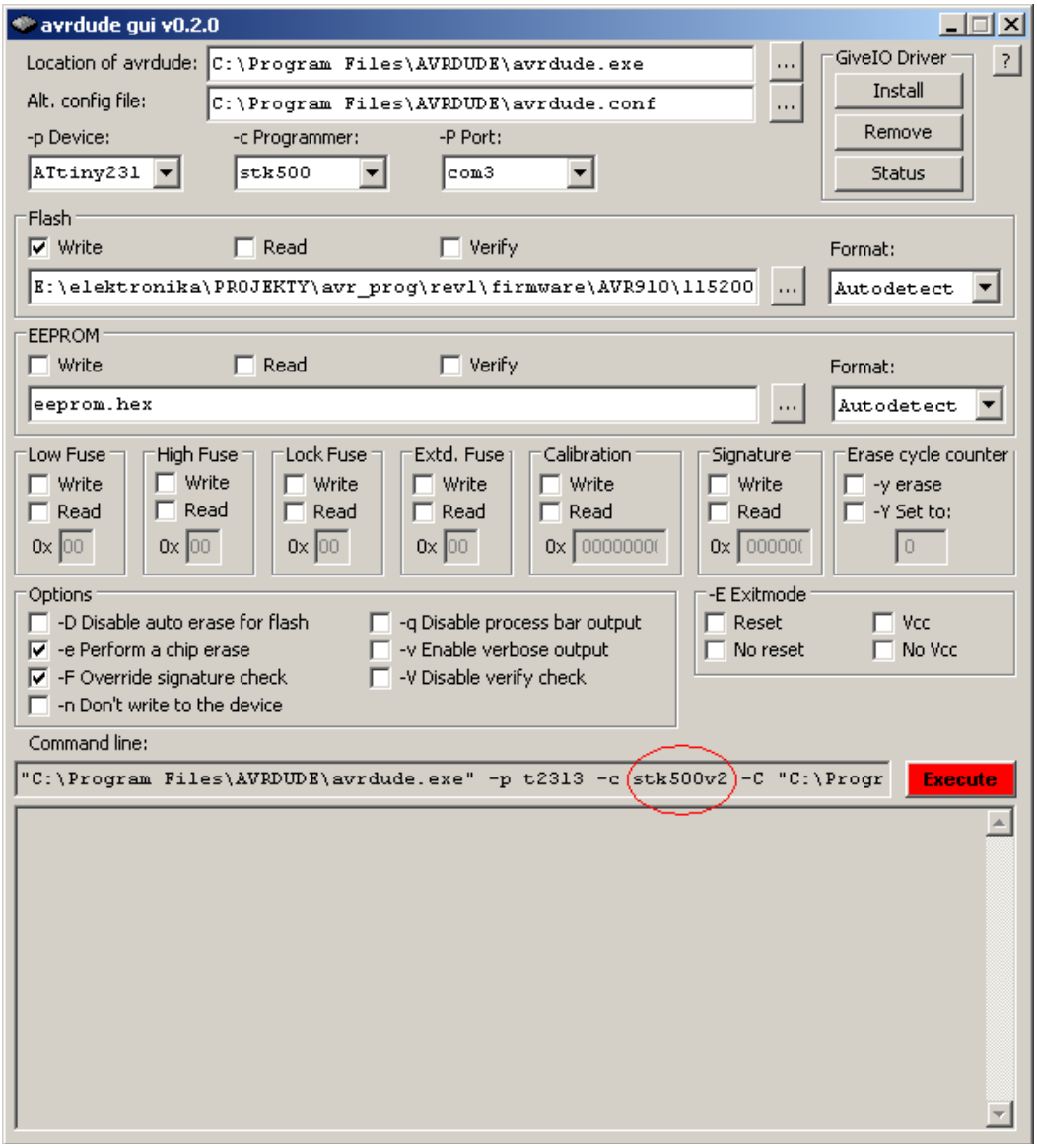
Program AVR Prog może być wywoływany z poziomu środowiska lub jako samodzielna aplikacja. Z poziomu AVR Studio, AVRprog uruchamiamy wybierając pierwszą od góry pozycję w zakładce *Tools* (AVR Studio -> Tools -> AVR Prog...). Aplikacja przed uruchomieniem odpytuje porty COM1 do COM4 komputera w poszukiwaniu programatora. Programator musi być zatem podłączony i prawidłowo skonfigurowany czyli zainstalowany na którymś z początkowych portów COM. Zmiana numeru COM opisana jest w rozdziale *Konfiguracja USB Serial Port – zmiana numeru COM*.



Bardzo ważne jest, żeby przed przystąpieniem do jakichkolwiek operacji związanych z programowaniem, w polu *Device* wybrać z rozwijanej listy nazwę podłączonego mikrokontrolera.

11.3 AVRDUDE – AVRDUDE GUI

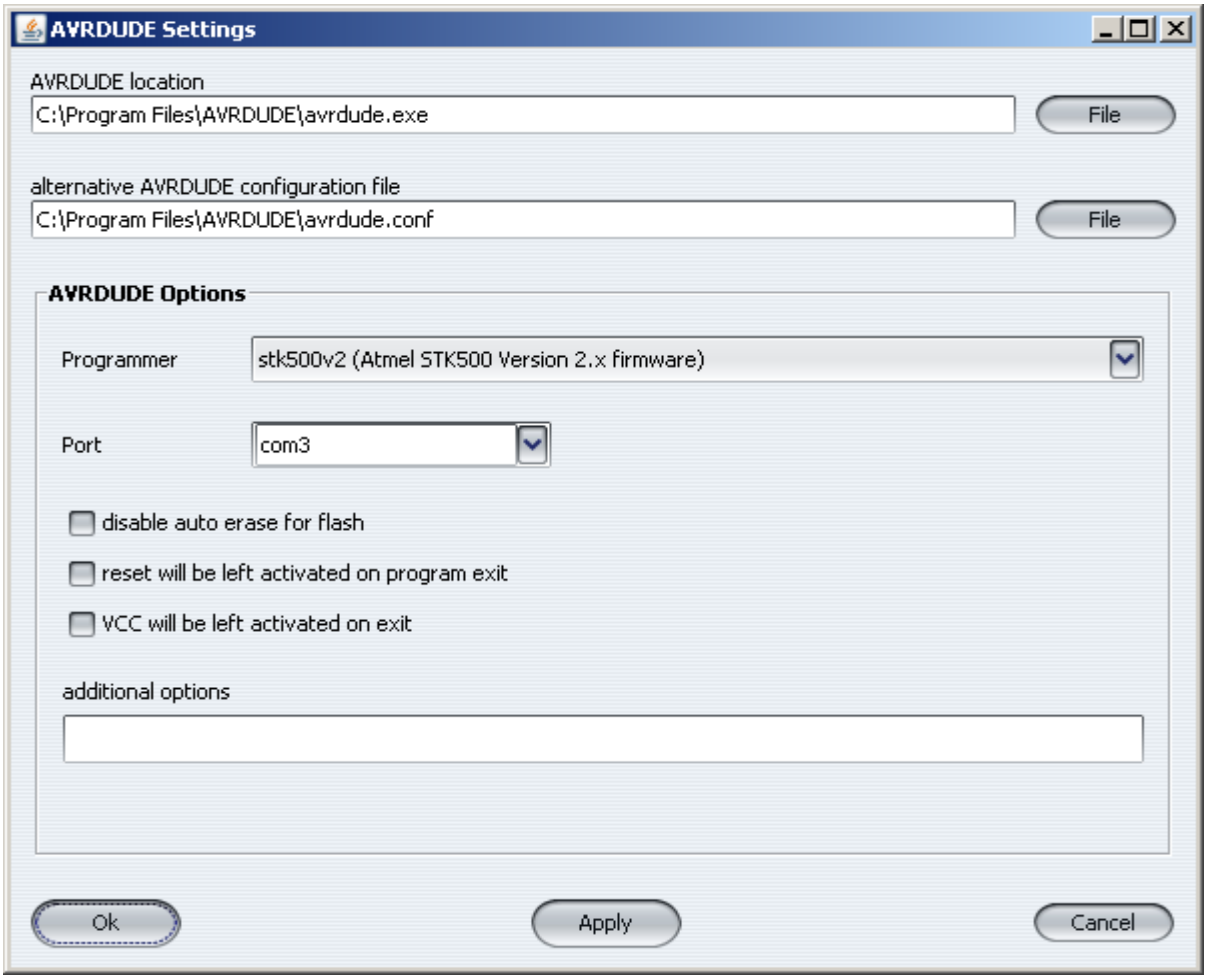
Program AVRDUDE może współpracować zarówno z firmware STK500 jak i AVR910. Programowanie w trybie STK500 przebiega jednak szybciej więc w instrukcji skupimy się na tym wariancie – programator z firmware STK500.
Na początek należy skopiować katalogi AVRDUDE oraz AVRDUDE_GUI na dysk twardy komputera. Następnie uruchamiamy program *avrdude-gui.exe*
W linii *Location of avrdude* wskazujemy lokalizację pliku „avrdude.exe", natomiast niżej, w linii *Alt. Config file* podajemy ścieżkę do pliku konfiguracyjnego „avrdude.conf".
Wybieramy rodzaj programowanego mikrokontrolera (*-p Device*) , numer portu COM (*-P Port*) na którym jest programator a jako „*-c Programmer*" wybieramy „stk500". Uwaga ! Przed finalnym naciśnięciem czerwonego przycisku „*Execute*" w „*Command line*", do „stk500" dostawiamy „v2"- tak jak poniżej.



11.4 AVRDUDE – AVR8 Burn-O-Mat

Programator z wsadem STK500.

Aby uruchomić program AVRDUDE z GUI AVR8 Burn-O-Mat należy w pierwszej kolejności zainstalować środowisko JAVA (CD\SOFTWARE\jre-6u4-windows-i586-p). Następnie należy przekopiować na dysk twardy katalog AVRDUDE (CD\SOFTWARE\AVRDUDE) oraz katalog AVR8_Burn-O-Mat (CD\SOFTWARE\AVRDUDE_GUI\AVR8_Burn-O-Mat). Z katalogu AVR8_Burn-O-Mat uruchomić plik wsadowy *start.bat*. Z menu programu wybrać Settings->AVRDUDE



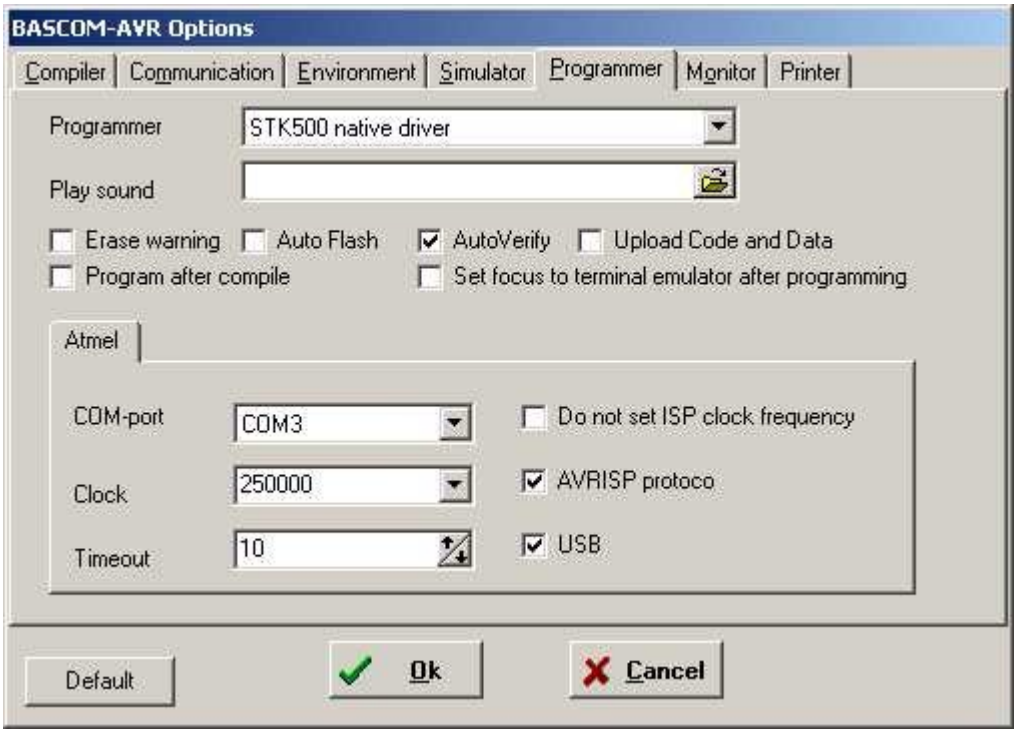
W '*AVRDUDE location'* podać lokalizację pliku avrdude.exe
W 'alternative AVRDUDE configuration file' należy podać lokalizację pliku avrdude.conf.
Następnie zamknąć program. Po ponownym uruchomieniu gui jeszcze raz przejść do AVRDUDE Settings i w „Programmer" wybrać '*stk500v2...'* oraz numer portu com zajmowanego przez programator. Zaakceptować ustawienia i nacisnąć „Ok".
W głównym oknie programu należy wybrać rodzaj programowanego mikrokontrolera.

11.5 BASCOM-AVR

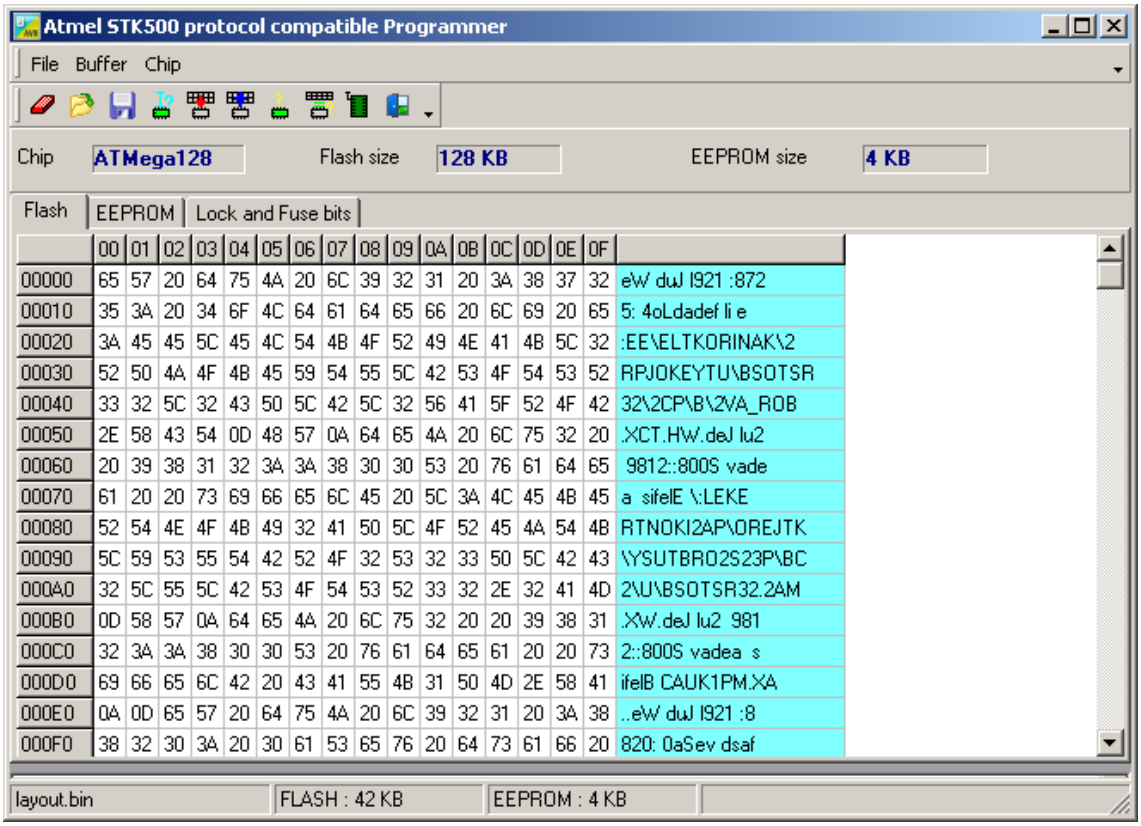
Jest kilka możliwości współpracy AVR PROG USB ze środowiskiem Bascom.
We wszystkich opisanych tu wariantach do programatora powinien być załadowany firmware STK500.

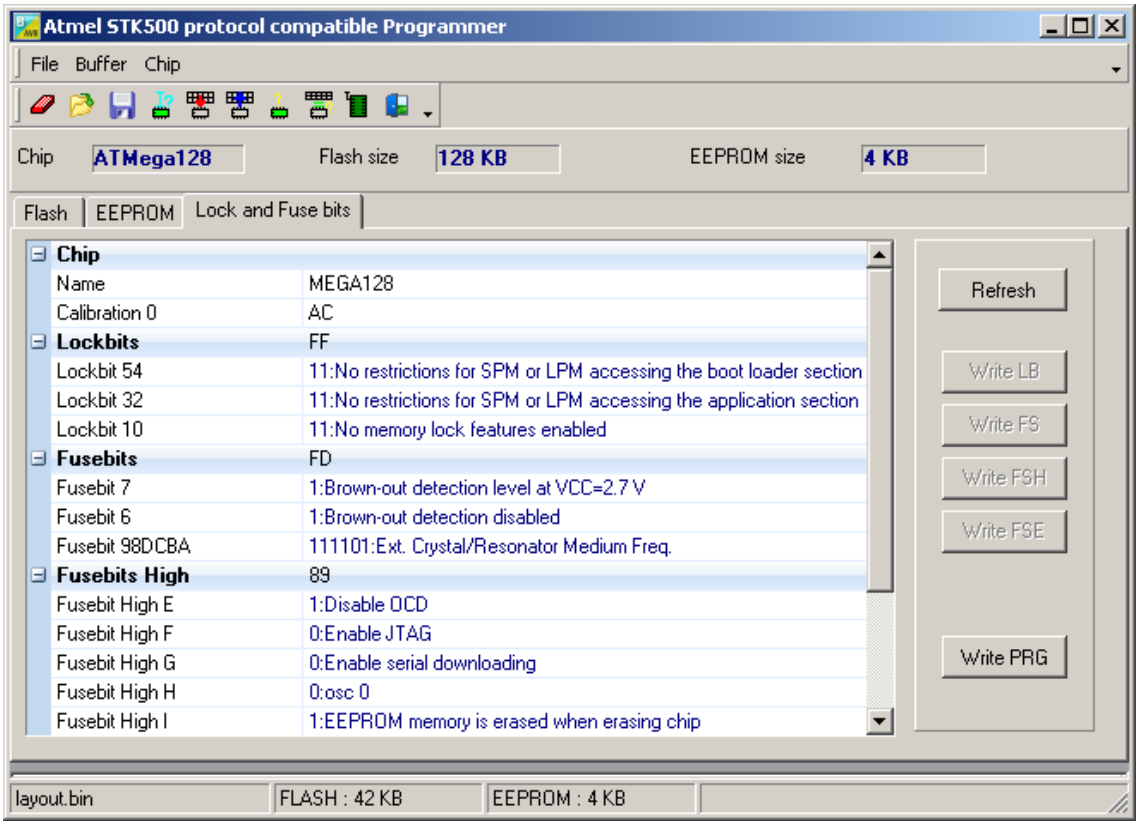
STK500 native driver

Począwszy od wersji 1.11.9.3 w środowisku Bascom zaimplementowana została bezpośrednia obsługa programatorów zgodnych z stk500.
Jest to ukłon w stronę użytkowników przyzwyczajonych do korzystania z programatora STK200 - obsługa programowanego mikrokontrolera w tym trybie wygląda wówczas bardzo podobnie jak w STK200.
W oknie konfiguracji programatora *(BASCOM-AVR->Options->Programmer)* jako *Programmer* trzeba wybrać *STK500 native driver.* Następnie należy zaznaczyć pola *AVRISP protoco* i *USB* oraz wprowadzić numer portu COM na którym zainstalowany jest programator (pole *COM-port*). Sprawdzenie oraz zmiana numeru portu COM na którym znajduje się programator opisane jest w rozdziale *Konfiguracja USB Serial Port – zmiana numeru COM*.



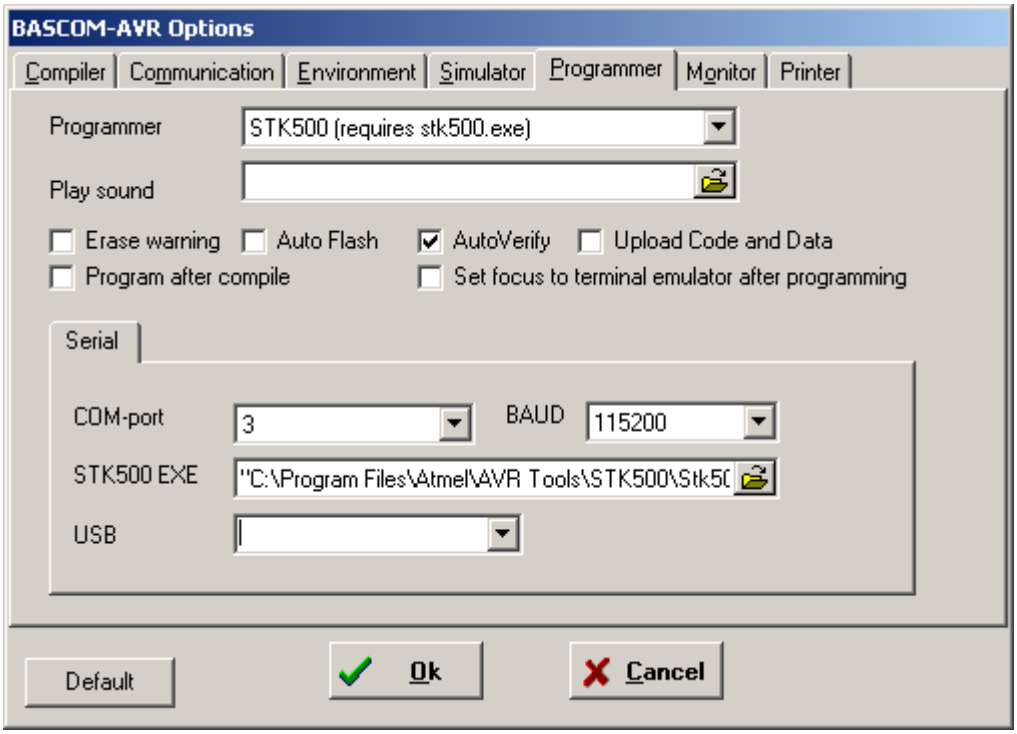
Poniżej znajdują się przykładowe zrzuty ekranu z pracy w tym trybie.





STK500

Aby pracować w tym trybie na komputerze musi być zainstalowane środowisko AVR Studio (bascom do programowania będzie korzystał z bibliotek tego środowiska).



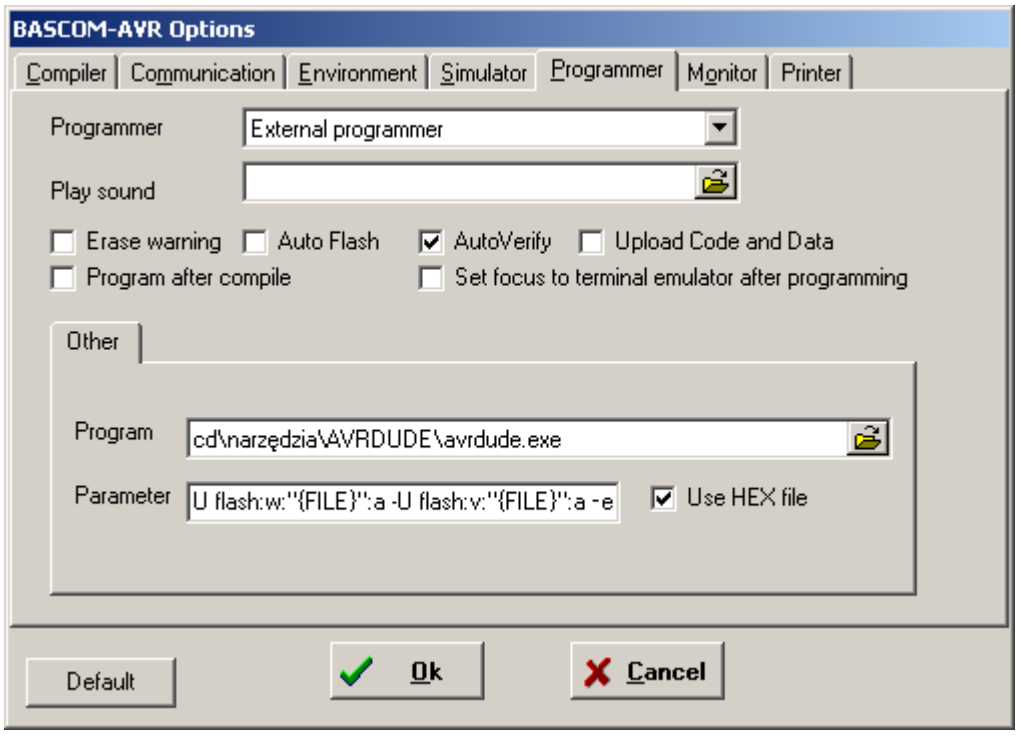
W oknie konfiguracji programatora *(BASCOM-AVR->Options->Programmer)* jako *Programmer* należy wybrać *STK500 (requires stk500.exe).* Następnie należy wprowadzić numer portu COM na którym zainstalowany jest programator (pole *COM-port*) oraz ustawić prędkość *BAUD* na 115200 (Sprawdzenie oraz zmiana numeru portu COM na którym znajduje się programator opisane jest w rozdziale *Konfiguracja USB Serial Port – zmiana numeru COM*). W trybie tym należy również uzupełnić ścieżkę do pliku stk500.exe (pole *STK500 EXE*). Zazwyczaj plik ten znajduje się w lokalizacji "C:\Program Files\Atmel\AVR Tools\STK500".

External programmer

W trybie tym mamy możliwość wskazania zewnętrznej aplikacji którą BASCOM będzie wywoływał podczas programowania. W przykładzie posłużę się programem AVRDUDE. Programator powinien mieć wgrany wsad STK500.
Ważne jest by korzystać z AVRDUDE dostarczonego z programatorem ze względu na specyficzne ustawienia pliku konfiguracyjnego. Tak więc katalog AVRDUDE z płyty CD należy skopiować na dysk twardy.
W polu '*Program*' podajemy ścieżkę do programu avrdude.exe, natomiast w '*Parameter*' wpisujemy (można wkleić):

"avrdude" -p m128 -c stk500v2 -P com3 -U flash:w:"{FILE}":a -U flash:v:"{FILE}":a –e

Oczywiście zamiast m128 musi być wpis odpowiedni dla programowanego procesora np.dla Atmega 8 będzie m8 , dla ATmega16 będzie m16 , itd.
W miejscu com3 wpisujemy numer portu na którym zainstalowany jest programator.
Ponadto zaznaczamy '*Use HEX file*'



Po takiej konfiguracji programator jest gotowy do pracy.
Tradycyjnie, w celu zaprogramowania wybieramy '*Send to chip*' lub naciskamy F4.

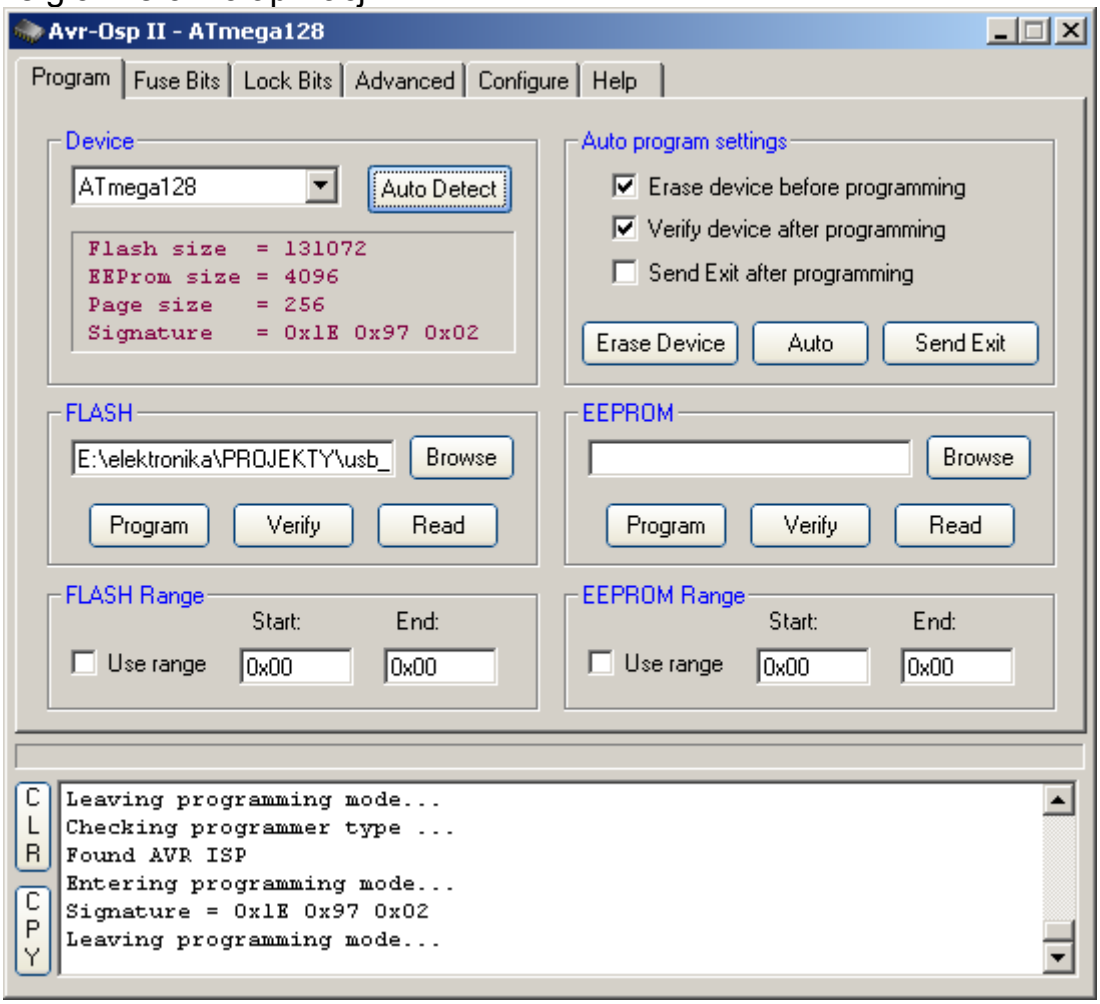
11.6 AVR OSP II

Programator z wsadem AVR910.

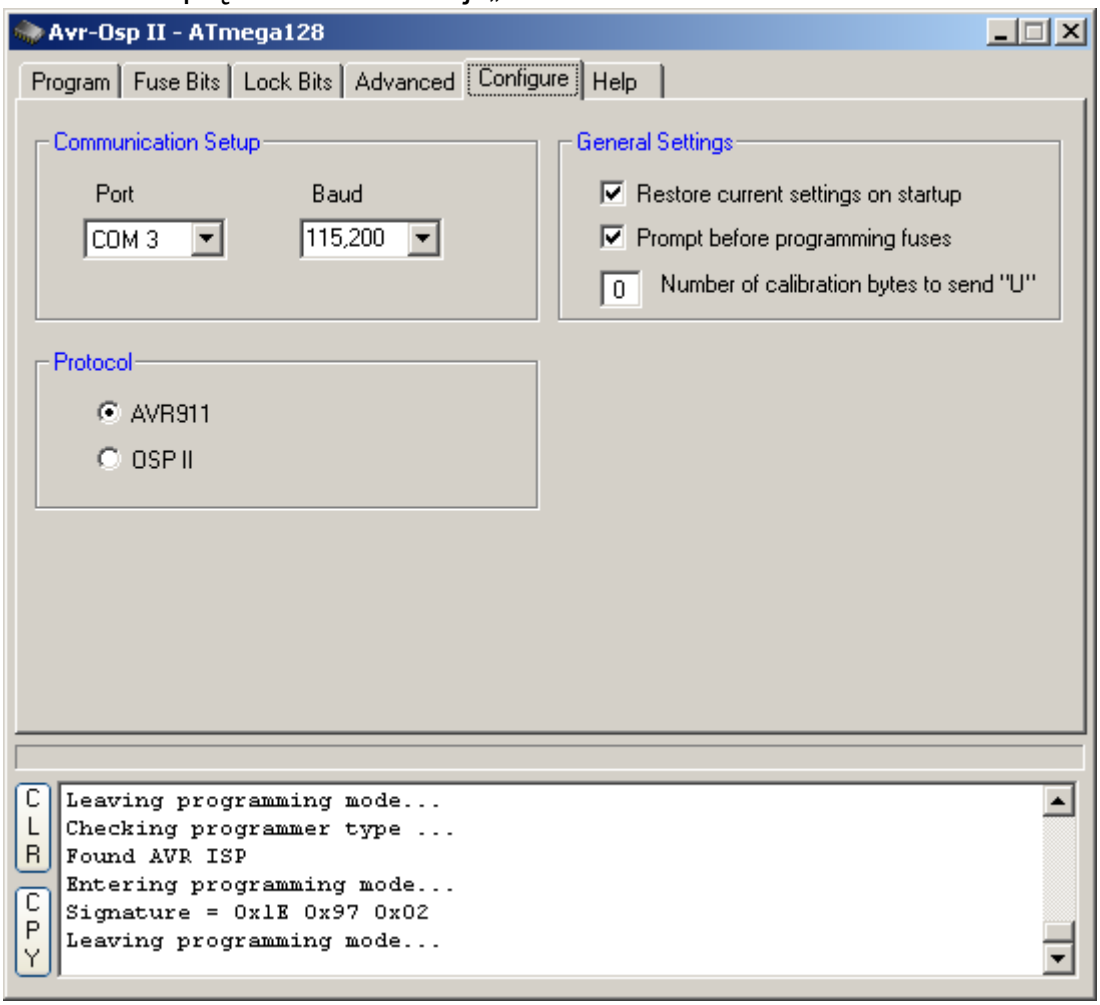
Bardzo przyjemny i intuicyjny interfejs oraz rozbudowana lista obsługiwanych mikrokontrolerów sprawiają iż warto od czasu do czasu sięgnąć do programu AVR OSP II. Aplikacja obsługuje programatory z implementacją protokołu AVR910.

Jedynym mankamentem aplikacji jest trochę powolne (w porównaniu do np. AVR Studio) programowanie. Jeśli jednak plik wsadowy jest niewielkich rozmiarów różnica może być mało odczuwalna a korzyści płynących z przejrzystej obługi zmiany fusebitów nikomu nie trzeba tłumaczyć.

Przed uruchomieniem, folder z aplikacją należy skopiować na dysk twardy komputera (CD\SOFTWARE\AvrOspII). Po kliknięciu na plik AvrOspII.exe otwarte zostanie główne okno aplikacji.



Przed przystąpieniem do programowania należy przejść do zakładki „*Configure*” i w polu „*Communication Setup*” ustawić numer portu na którym zainstalowany jest programator oraz prędkość transmisji „*Baud”* na 115200.



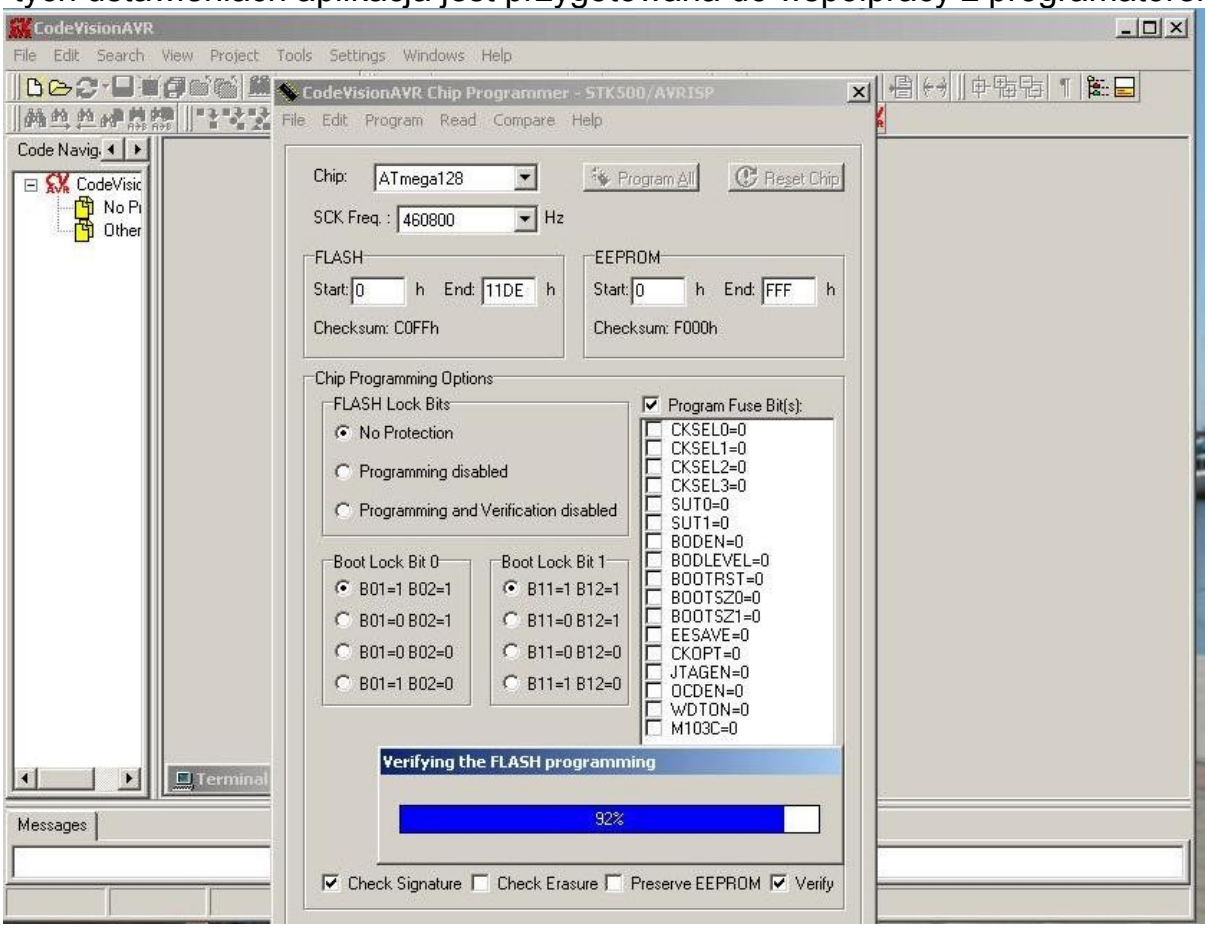
Warto również w ustawieniach portu COM zmniejszyć czas opóźnienia – poprawi to znacznie prędkość programowania. Zmiana ta opisana jest w rozdziale *Konfiguracja USB Serial Port – zmiana czasu opóźnienia.*

Poprawność zestawionego połączenia można sprawdzić wciskając w zakładce "*Program*" przycisk "*Auto Detect*". W polu "*Device*" powinna pojawić się nazwa podłączonego do programatora mikrokontrolera.

11.7 CodeVisionAVR

Do programatora należy wgrać firmware STK500.
W *Programmer Settings* w polu *AVR Chip Programmer Type* wybieramy opcję *Atmel STK500/AVRISP.* W polu *Communication Port* należy wskazać numer portu COM na którym zainstalowany jest programator.

Po tych ustawieniach aplikacja jest przygotowana do współpracy z programatorem.



12. Kontakt - wsparcie techniczne

W celu uzyskania dodatkowej pomocy technicznej skontaktuj się z nami
serwis@sibit.pl

Windows is a registered trademark of Microsoft Corporation.
ATMEL, AVR, XMEGA, FLIP and AVR Studio are the registered trademarks of Atmel.
CodeVisionAVR is trademark of HP InfoTech.
All other brand names or product names are the property of their respective holders.