

Programator JuPic+

Emulator programatora PICSTART Plus
Współpracuje z aplikacją MPLAB IDE i PICP
<http://ajpic.zonk.pl/>

Opis programatora

Programator zbudowany został w oparciu o nowy procesor PIC16F87. Urządzenie w połączeniu ze zintegrowanym środowiskiem uruchomieniowym MPLAB™ IDE (pracującym pod kontrolą systemu operacyjnego Windows 98SE, Windows ME, Windows NT 4.0 SP6a WS, Windows 2000 SP2 lub Windows XP) lub PICP (Windows, Linux) tworzy niezwykle sprawne i profesjonalne narzędzie programisty. Funkcjonalność tego zestawu powoduje, że może być przeznaczony do małych i średnich zastosowań, a polecany jest zarówno dla użytkowników początkujących jak i zaawansowanych. Urządzenie jest w pełni kompatybilne z programatorem PICSTART Plus® i w pełni integruje się z programem MPLAB. Programator przeznaczony jest do obsługi rodziny mikrokontrolerów serii „F” z pamięcią FLASH oraz serii „C” z pamięcią EPROM:

PIC10F2XX, PIC12F6XX, PIC16F6XX
PIC16F7X, PIC16F8X, PIC16F81X, PIC16F7X7
PIC16F8XA, PIC16F8XX, PIC16F8XXA, PIC18FXXXX

PIC12C5XX, PIC12C6XX, 16C4XX, 16C5XX,
16C6X, 16C6XX, 16C7X, 16C7XX, 16C9XX

Firma Microchip udostępnia program MPLAB IDE jako darmowy i można go pobrać ze strony internetowej <http://www.microchip.com/>. Najnowsze oprogramowanie jest dostępne w wersji „8.xx”, natomiast programator pracuje również ze starszą wersją „5.70.40”

Program **MPLAB** daje możliwość pracy zarówno z pojedynczymi plikami jak i z całymi **projektami**. Grupowanie programów w projekty znacznie ułatwia tworzenie aplikacji, a zarazem uzyskujemy zwiększenie komfortu pracy. Program z wbudowanymi wieloma funkcjami pozwala m. in. na:

- ✓ tworzenie i edycję plików źródłowych
- ✓ grupowanie plików w projekty
- ✓ importowanie gotowych plików Intel HEX
- ✓ wyszukiwanie błędów kodu
- ✓ asemblację, kompilację i linkowanie kodu źródłowego
- ✓ wyznaczanie zależności czasowych
- ✓ podglądanie zmiennych w czasie pracy programu
- ✓ edycję zawartości pamięci
- ✓ symulację programu
- ✓ przesyłanie kodu wynikowego do procesora
- ✓ debugowanie za pomocą protokołu ICD
- ✓ eksportowanie i importowanie danych
- ✓ rozwiązywanie problemów z wykorzystaniem podręcznej pomocy

Instalacja Programatora

1. Programator umieścić na stabilnym nieprzewodzącym podłożu
2. Podłączyć przewód sygnałowy **RS232** do komputera i programatora
3. Podłączyć zasilacz do sieci, a następnie przewód zasilania **12V** do programatora
4. Umieścić programowany procesor w **podstawce** lub podłączyć przewód programowania zewnętrznego **ICSP** (rozdział **Złącze ICSP**)
5. Uruchomić program **MPLAB IDE** na komputerze (program jest darmowy i można go pobrać ze strony <http://www.microchip.com/> lub <http://ajpic.zonk.pl/> dokumentacja do programu znajduje się również na stronie internetowej). Interfejs użytkownika zależy od zainstalowanej wersji programu, programator JuPic pracuje zarówno ze starszą wersją programu **5.70.40** jak również z nowszymi wersjami od **6.00** wzwyż.
6. Przed przystąpieniem do pracy należy zapoznać się z rozdziałem „**Konfiguracja programatora**”

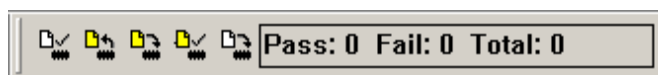
Obsługa programu MPLAB w wersji 8.xx

Uruchomienie programatora następuje po wybraniu poniższych opcji:

1. Ustawić typ programatora: Programmer → Select Programmer → **PICSTART Plus**
2. Ustawić port, na którym będzie pracował programator: Programmer → Settings... → Communication → **COMx**
3. uaktywnić programator: Programmer → **Enable Programmer**

Czynność wybierania i ustawiania parametrów programatora wykonywana jest jednokrotnie, ponieważ MPLAB zapamiętuje te ustawienia. Jeśli zachodzi potrzeba ich zmiany można wykonać je w dowolnym momencie.

Po wykonaniu powyższych czynności zostanie nawiązana łączność pomiędzy programem **MPLAB** i programatorem **JuPic**. Informacją o poprawnym zainicjowaniu programatora jest pojawienie się ikonki na pasku obsługi:



Rys. 1 Funkcje programatora

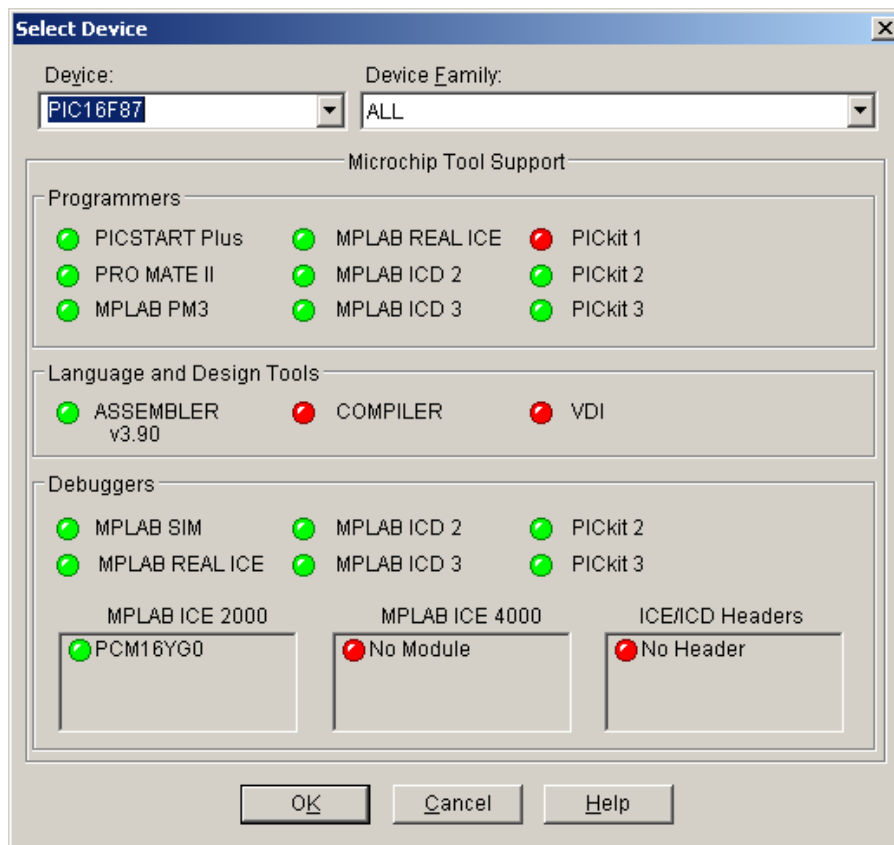
Wywoływane funkcje oznaczają kolejno od lewej:

- **Blank Check** — test wyzerowania procesora
- **Read** — odczyt kodu z procesora
- **Program** — zapis kodu do procesora
- **Verify** — weryfikacja kodu
- **Erase Flash Device** — wyzerowanie procesora

Na pasku po prawej stronie ikonki przedstawiona jest liczba udanych (**Pass**), nieudanych (**Fail**) i wszystkich (**Total**) zapisów pamięci procesora.

Od tego momentu można budować obszar pracy tzw. „Work Space” i rozpocząć pracę z programatorem. Przed przystąpieniem do programowania należy ustawić typ procesora, który będzie używany w projekcie Configure → Select Device... → **Device** (Rys. 2)

Zielona lampka przy napisie PICSTART Plus sygnalizuje, że dany procesor obsługiwany jest przez ten programator (lista procesorów obsługiwanych przez programator JuPic podana jest w ostatnim rozdziale). Żółta lampka sygnalizuje ograniczone możliwości obsługi. Czerwona lampka sygnalizuje brak obsługi danego procesora.

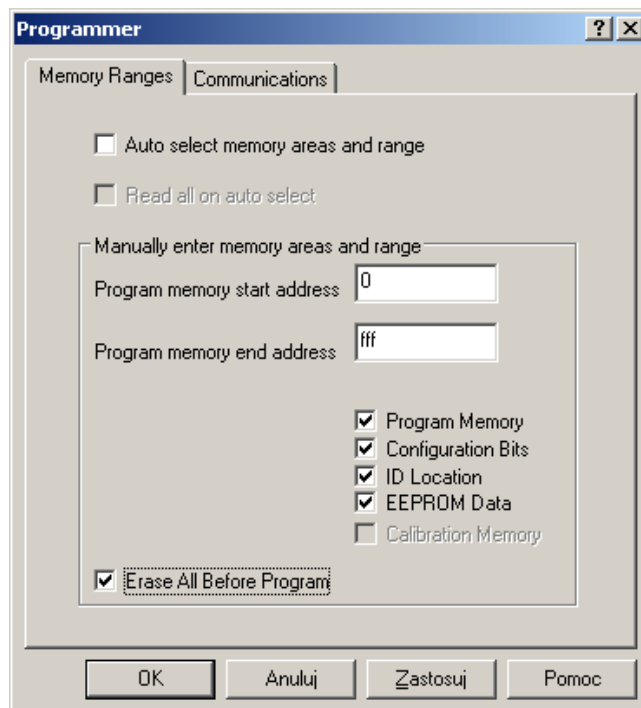


Rys. 2 Okno wybierania procesora

Po skonfigurowaniu ustawień program MPLAB jest gotowy do programowania przyłączonego procesora.

Zaprogramowanie procesora następuje po naciśnięciu funkcji „**Program**”, zmianie zawartości ulegną wszystkie obszary pamięci ustawione w sekcji „**Settings**”.

Jeśli zaprogramowana ma zostać tylko część pamięci należy wybrać funkcję z menu Programmer → Settings... → **Memory Ranges** (Rys. 3), a następnie ustawić wymagane parametry. Interfejs daje również możliwość ustawienia zakresu programowania pamięci programu. Wszystkie ustawienia są zapamiętywane w projekcie dlatego są aktywne przy kolejnych operacjach zapisu i odczytu wykonywanych na procesorze.



Rys. 3 Ustawianie zakresu programowania

Przed przystąpieniem do programowania można odpowiednio ustawić bity konfiguracyjne: Configure → **Configuration Bits...** (Rys. 4 i Rys. 5)

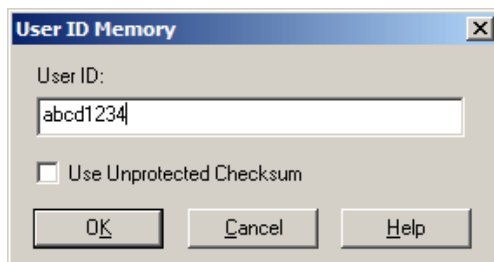
Address	Value	Category	Setting
2007	3F72	Oscillator	HS
		Watchdog Timer	Off
		Power Up Timer	On
		Brown Out Detect	On
		Low Voltage Program	Disabled
		Flash Program Write	Enabled
		Background Debug	Disabled
		Data EE Read Protect	Off
		Code Protect	Off

Rys. 4 Ustawienia bitów konfiguracyjnych 16F876

Address	Value	Category	Setting
300001	26	Oscillator	HS-PLL Enabled
		Osc. Switch Enable	Disabled
300002	0F	Power Up Timer	Disabled
		Brown Out Detect	Enabled
		Brown Out Voltage	2.0V
300003	0F	Watchdog Timer	Enabled
		Watchdog Postscaler	1:128
300006	05	Low Voltage Program	Enabled
		Background Debug	Enabled
		Stack Overflow Reset	Enabled

Rys. 5 Ustawienia bitów konfiguracyjnych 18F458 (część)

Zapisanie bajtów pamięci identyfikacji jest dostępne dla użytkownika w menu: Configure → **ID Memory...** (Rys. 6)



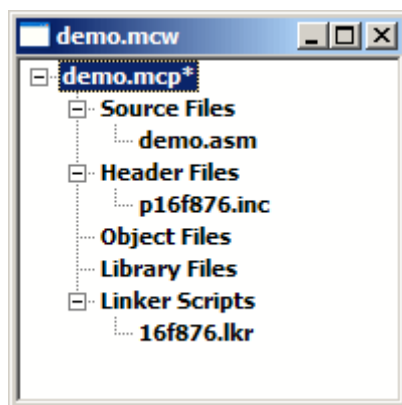
Rys. 6 Pamięć ID procesora

Dowolnie modyfikować można również pamięć nieulotną EEPROM, która jest dostępna w menu: View → **EEPROM** (Rys. 7)

Address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	ASCII
0000	F8	EB	FC	82	73	03	5C	A6	74	F9	CE	8F	3B	5C	B7	DD	...s.\. t...;\..
0010	78	70	5C	90	BC	D1	F7	CA	2E	58	09	D6	7F	42	F1	6A	xp\..... .X...B.j
0020	20	8F	BE	29	B2	A9	42	6C	BE	46	67	42	A6	8F	C7	B6	...)..B1 .FgB....
0030	DB	82	76	50	51	95	22	DE	30	9C	5F	73	CD	EC	3C	B1	..vPQ.". 0_s_<.
0040	D8	C1	1A	44	D7	DD	BE	AF	D2	72	A4	4B	50	42	96	8C	...D.... .r.KPB..
0050	83	05	7E	87	C0	0B	79	B1	32	22	2C	EA	CE	BA	5B	BA	...~...y. 2",...[.
0060	89	47	B7	DA	C9	E7	F8	F6	1A	45	2C	B1	23	BD	4F	EA	.G..... .E,.#.O.
0070	D8	C1	1B	3D	F1	7B	21	CD	9A	B3	19	41	6C	F4	77	0D	...=.{!.. ...Al.w.
0080	9C	EA	3F	F3	72	0F	19	C9	FD	87	A8	7C	06	47	18	55	..?.r...G.U

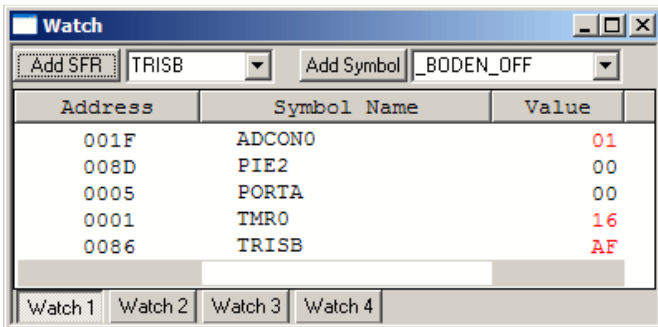
Rys. 7 Pamięć EEPROM procesora

Program MPLAB pozwana na bardzo zaawansowaną pracę z projektami, które znacznie ułatwiają poruszanie się po strukturze plików. Przyłączanie plików do projektu przedstawia Rys. 8

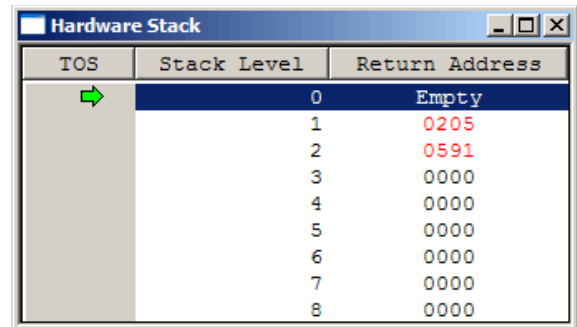


Rys. 8 Budowanie projektu

Podczas pracy z programem MPLAB dostępnych jest również wiele pomocnych narzędzi do debugowania i symulowania kodu między innymi podglądanie rejestrów procesora i stosu programu:

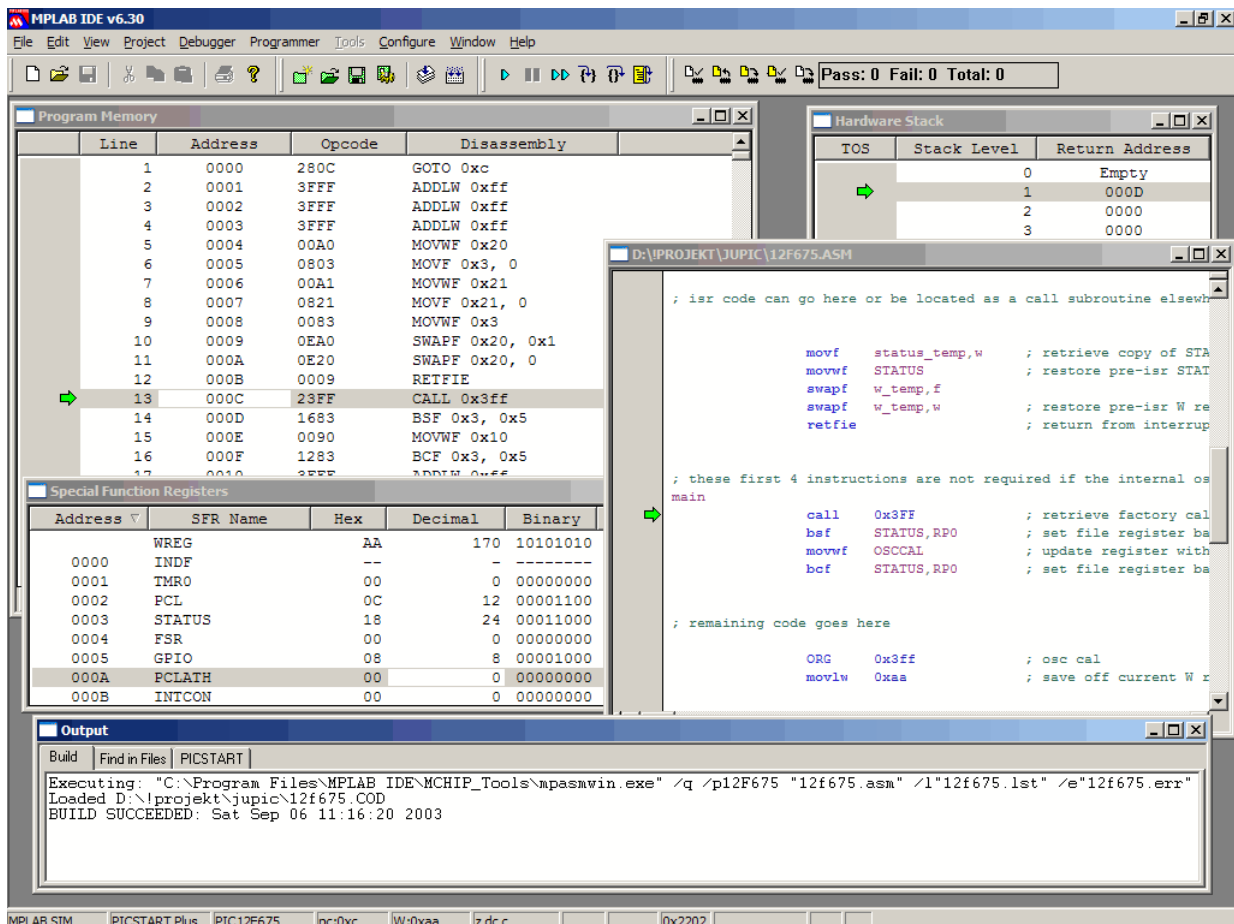


Rys. 9 Podglądanie rejestrów Procesora



Rys. 10 Podglądanie stosu programu

Program posiada nowy edytor który podświetla składnię, dzięki temu kod programu staje się bardziej przejrzysty. Podczas pracy można również wykorzystać wbudowany symulator: Debugger → Select Tool → **MPLAB SIM**



Rys. 11 Pełny interfejs użytkownika

Funkcje programatora

Programator wyposażony jest w szereg funkcji i usprawnień rozszerzających jego możliwości:

- ✓ układ posiada przycisk kasowania, który bez konieczności podłączenia programatora do komputera umożliwia wyzerowanie pamięci procesora (także Code Protection), aby wykasować pamięć należy nacisnąć i przytrzymać klawisz „ERASE” przez 2 sekundy — dioda informacyjna zasygnalizuje wyzerowanie procesora.
- ✓ wprowadzono diodę informacyjną „PROG”, która pokazuje aktualny stan urządzenia:
 - dioda sygnalizuje dwa podwójne krótkie błyski — załączenie zasilania i inicjacja programatora
 - dioda świeci światłem ciągłym — programator gotowy do pracy
 - dioda pulsuje równomiernie szybko — nawiązywana jest komunikacja z programem **MPLAB IDE**
 - dioda pulsuje równomiernie wolno — programator jest w trakcie wymiany danych z programem **MPLAB IDE** (zapis/odczyt)
 - dioda gaśnie — reakcja na przyśnięcie klawisza
 - dioda sygnalizuje 3 krótkie błyski — procesor został wykasowany
 - dioda okresowo nadaje 3 krótkie błyski — zawieszenie programatora, naruszenie struktury programu, konieczna wymiana oprogramowania
- ✓ wprowadzono złącze **ICSP** (In-Circuit Serial Programming), czyli możliwość programowania w obwodzie bez konieczności wyjmowania procesora z uruchamianego urządzenia
- ✓ wprowadzono możliwość programowania procesorów w trybie **LVP!**
- ✓ wyprowadzono podstawkę **DIP18**, która umożliwia szybkie zaprogramowanie procesora bez konieczności podłączania przewodów. Do podstawki można włożyć procesory, które mają kompatybilne wyjścia z układami **DIP18**, a także **DIP14** oraz **DIP8**
- ✓ wyprowadzono zworki konfiguracyjne, które pozwalają przystosować programator do własnych potrzeb sprzętowych
- ✓ programator można podłączyć bezpośrednio do portu **COM** komputera bez konieczności posiadania przewodu **RS232**
- ✓ układy scalone są umieszczone na podstawkach co umożliwia ich ewentualną wymianę bez konieczności lutowania

- ✓ procedury programujące zostały poddane optymalizacji co powoduje przyspieszenie programowania procesorów. Każda komórka przed zaprogramowaniem jest porównywana z wartością, która ma być do niej wpisana i jeśli jest taka sama, operacja zapisu jest pomijana. Taki sposób programowania znacznie przyspiesza cykl zapisu pojedynczej komórki, dla procesorów, które w jednym cyklu zapisywanych jest więcej komórek niż jedna procedura ta nie jest stosowana.
- ✓ możliwość upgrade'owania nowej wersji firmware'u po RS.

Przewód łączący RS-232

Programator **JuPic** łączy się z komputerem poprzez szeregowo łącze RS-232C kablem prostym typu „modem” (tzw. straight through), który jest także wykorzystywany przy łączeniu oryginalnego programatora PICSTART Plus.

Siedmiożyłowy przewód łączący zakończony jest męską wtyczką **DB-9** od strony programatora i żeńską **DB-9** lub **DB-25** od strony komputera. Przewód można wykonać we własnym zakresie; wykaz sygnałów używanych przez programator przedstawiony jest w poniższej tabeli.

Sygnał	DB-25	DB-9	Kierunek	DB-9	Sygnał
	Żeński			Męski	
TX	2	3	→	3	RX
RX	3	2	←	2	TX
DTR	20	4	→	4	Data Ready
GND	7	5	—	5	GND
DSR	6	6	←	6	pull up +5V
RTS	4	7	→	7	CTS
CTS	5	8	←	8	RTS

Tabela 1 Sygnały przewodu łączącego komputer z programatorem

Do komunikacji łączem szeregowym programator wykorzystuje standardowy protokół komunikacyjny o prędkości **19200Kb/s** i ramce **8N1**. Przesyłanie danych odbywa się w trybie półdupleksowym poprzez sprzętowe sterowane przepływem (flow control) liniami **RTS** i **CTS** (handshaking).

Konfiguracja Programatora

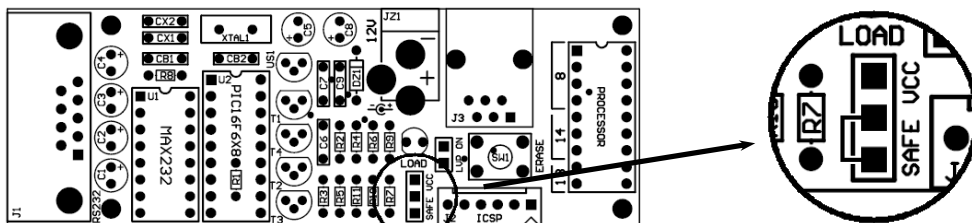
Programowanie może odbywać się na dwa sposoby:

- ✓ napięciem wysokim HVP (14V) — zworka „LVP ON” rozłączona
- ✓ napięciem niskim LVP (5V) — zworka „LVP ON” zwarta

Możliwe są 4 tryby pracy pod względem zasilania układu:

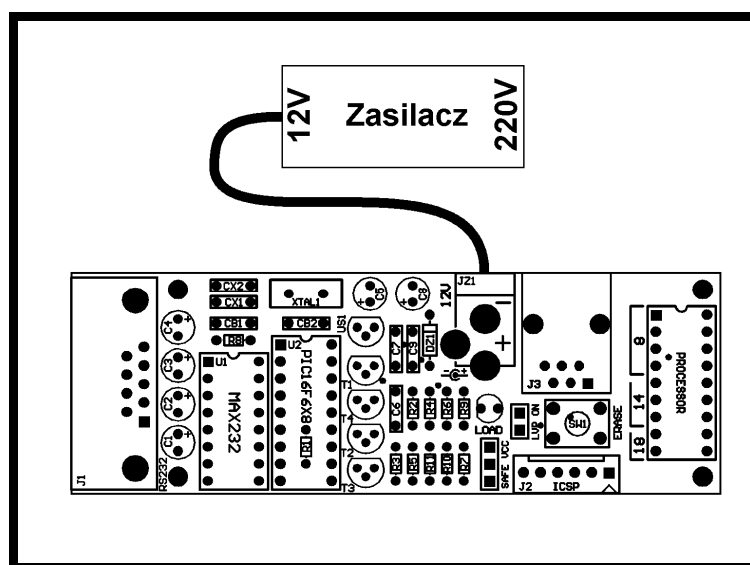
1 Tryb SAFE — bezpieczny

Uaktywnienie trybu następuje po załączeniu zworki „SAFE”



Rys. 12 Konfiguracja trybu SAFE

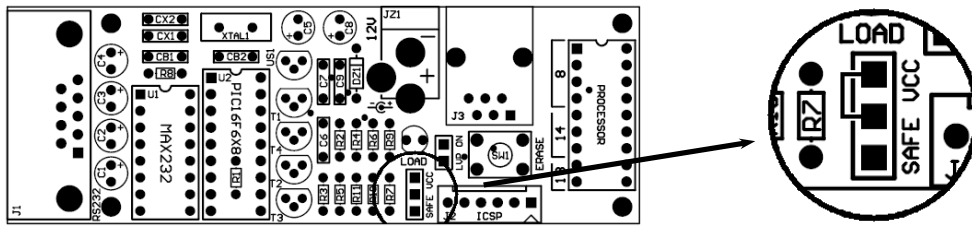
Tryb ten pozwala na programowanie procesora najbezpieczniejszym sposobem. Napięcie do procesora podawane jest tylko podczas jego programowania (zapis/odczyt), natomiast po operacji wymiany danych jest odcinane. Zalecane jest aby w tym trybie nie zasilać układu zewnętrznego ze złącza **ICSP** ze względu na małą obciążalność klucza tranzystorowego.



Rys. 13 Podłączenie układu w trybie SAFE

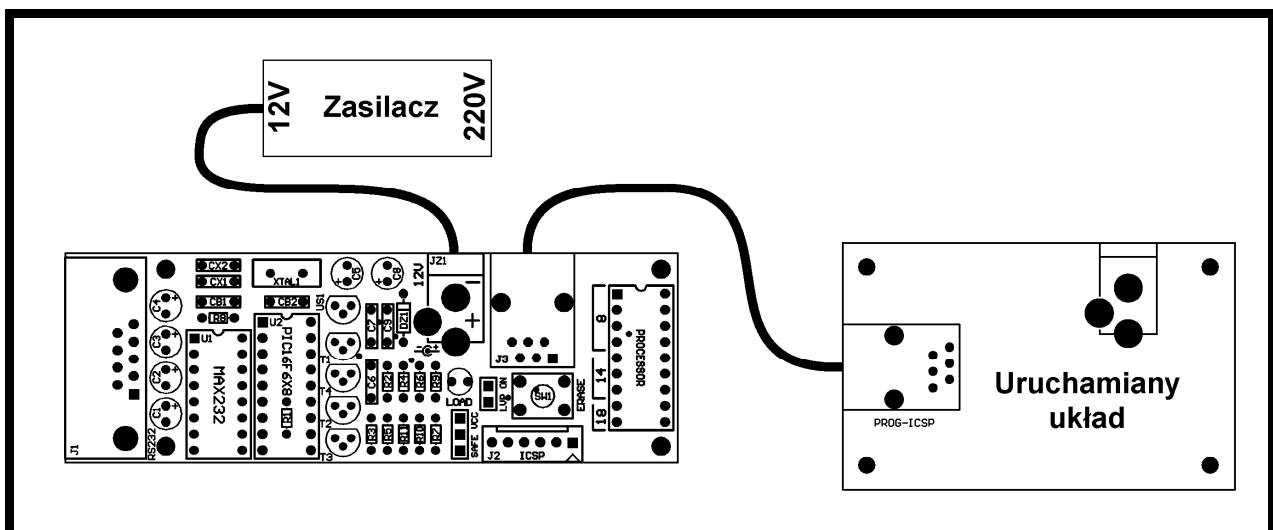
2 Tryb VCC – z bezpośrednim zasilaniem

Uaktywnienie trybu następuje po załączeniu zworki „VCC”



Rys. 14 Konfiguracja trybu VCC

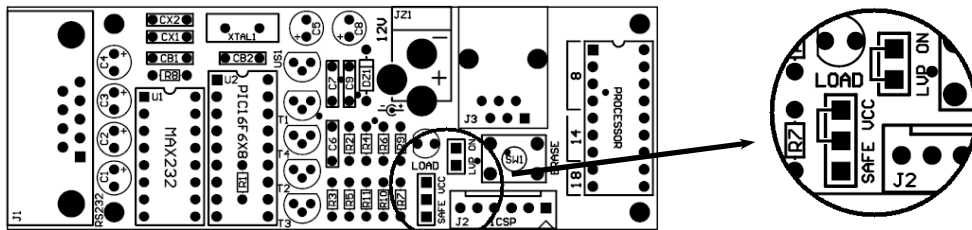
Tryb ten pozwala bezwarunkowo podawać napięcie do przyłączonego procesora (układu) z zasilacza umieszczonego na płytce. Napięcie jest podawane cały czas na nóżki procesora dlatego należy zachować ostrożność podczas jego wkładania lub wyjmowania z podstawki. Jeśli procesor jest programowany nie w podstawce lecz poprzez złącze zewnętrzne protokołem **ICSP**, napięcie podawane jest również na przyłączony układ. W ten sposób można wykorzystać zasilacz programatora do zasilania układu programowanego. Jedynym ograniczeniem jest tu wydolność prądowa stabilizatora w programatorze, dlatego należy wziąć ten fakt pod uwagę aby nie spalić tego elementu.



Rys. 15 Podłączenie układu w trybie VCC

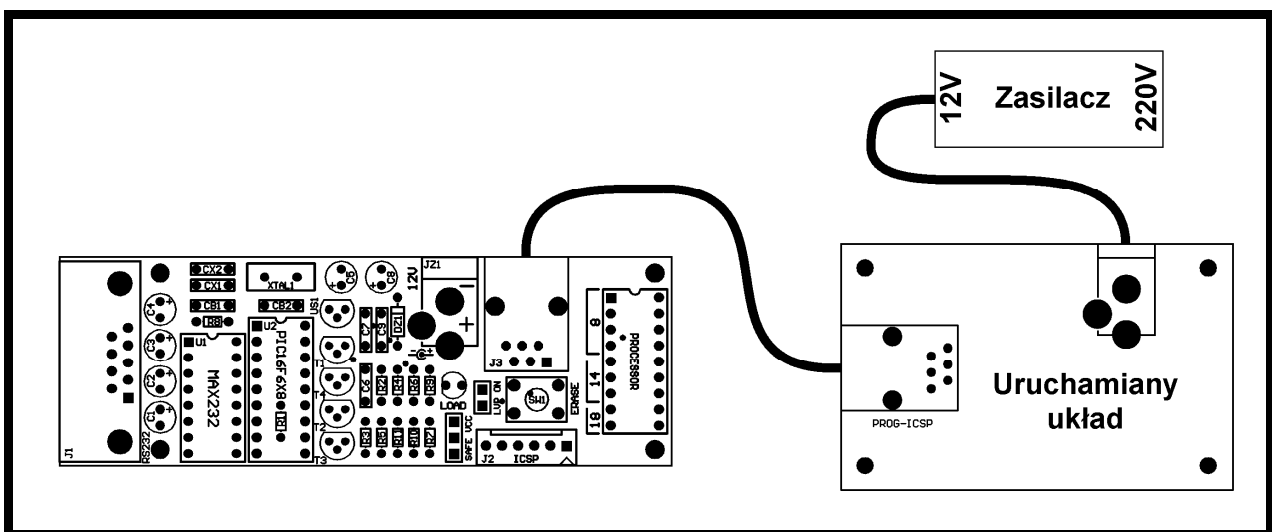
3 Tryb zależny – bez zasilacza

Uaktywnienie trybu następuje po załączeniu zworki „VCC” i „LVP ON”



Rys. 16 Konfiguracja trybu zależnego

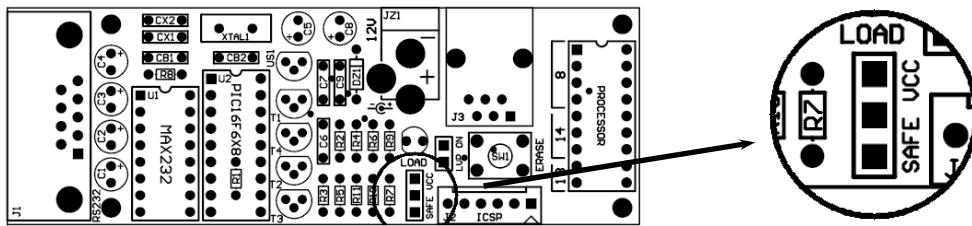
Tryb ten pozwala na pracę programatora bez przyłączonego zasilania. Napięcie do pracy programatora (5V) podawane jest pośrednio z układu programowanego przez złącze **ICSP**. W tej konfiguracji możliwa jest tylko praca z procesorami, które mogą być programowane niskim napięciem (**LVP**). Podczas pracy w tym trybie nie wolno podłączać zasilania do programatora.



Rys. 17 Podłączenie układu w trybie zależnym

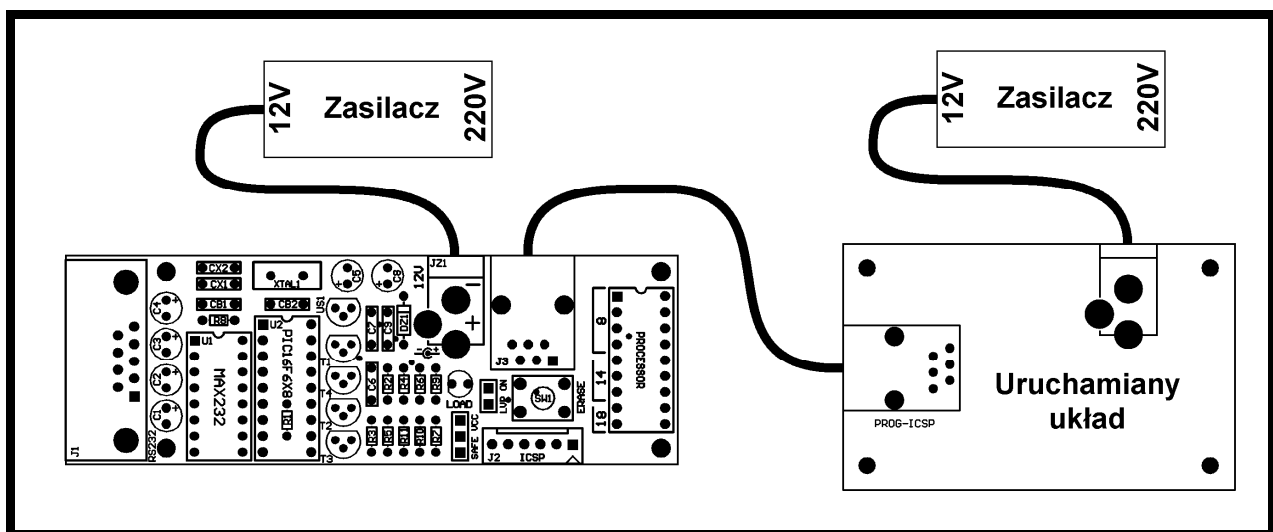
4 Tryb niezależny – z podwójnym zasilaniem

Uaktywnienie trybu następuje po rozłączeniu zworki „SAFE” i „VCC”



Rys. 18 Konfiguracja trybu niezależnego

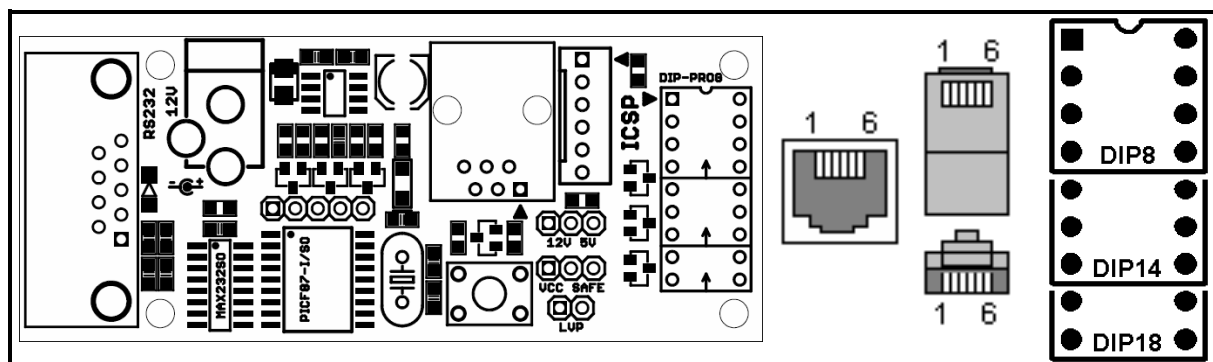
Tryb ten pozwala na podawanie zasilania z dwóch różnych źródeł. Programator zasilany jest z własnego stabilizatora, natomiast programowany procesor zasilany jest z uruchamianego układu. Tryb ten jest najczęściej stosowany przy pracy w trybie ICSP ponieważ daje możliwość zastosowania peryferiów o znacznie większym poborze prądu w uruchamianym układzie.



Rys. 19 Podłączenie układu w trybie niezależnym

Złącze ICSP

Programator wyposażony został w dwa 6 pinowe złącza programujące **ICSP** typu: „**SIP6P**” oraz „**RJ-12**”, poprzez które możliwe jest programowanie procesorów w uruchamianym układzie bez konieczności jego wyjmowania (Rys. 21). Zainstalowana podstawka precyzyjna umożliwia bezpośrednie programowanie procesorów 8 pinowych: PIC12FXXX, PIC12CXXX, 14 pinowych: PIC16F6XX oraz 18 pinowych: PIC16CXXX, PIC16FXXX, PIC18F1X20. Sposób umieszczenia układów oraz rozkład wyprowadzeń przedstawia Rys. 20. Dokładny opis złącza ICSP przedstawia Tabela 2. Do programowania większych procesorów należy dołączyć zewnętrzny adapter lub wykorzystać podłączenie układów poprzez ICSP. W miejsce podstawki do programowania można zainstalować również złącze **ZIF**.

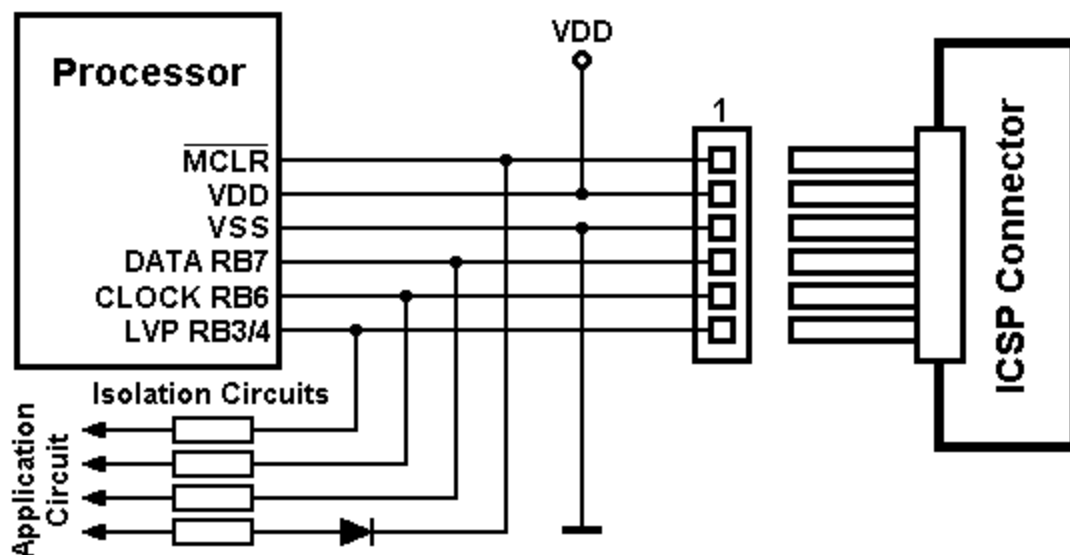


Rys. 20 Sposób podłączenia łącz ICSP, ustalenie pin'u 1.

Pracując ze złączem **ICSP** trzeba zwrócić uwagę na kilka szczegółów dotyczących „spinania” się z budowanym układem:

- ✓ sygnał „**MCLR**” powinien być przyłączony bezpośrednio do procesora, jeśli wymagane jest zewnętrzne zerowanie (układ opóźniający RC) linia ta musi być bezwzględnie odseparowana diodą sygnałową (np. 1N4148) (Rys. 21) ponieważ w przeciwnym wypadku procesor nie będzie w stanie wejść w tryb programowania.
- ✓ Linie danych „**DATA**” i zegar „**CLOCK**” powinny być również podłączone bezpośrednio do układu. Wykorzystanie tych linii jako porty I/O jest możliwe gdy budowany układ nie będzie zakłócał transmisji programatora. Najprostszym rozwiązaniem w tym wypadku jest podłączenie do portów **RB6** i **RB7** przycisków, które podczas programowania zostawiają te linie otwarte. Po cyklu zapisu/odczytu linie są odłączane od układu.

- ✓ Linia **LVP/PGM** jest wykorzystywana przy programowaniu procesora w trybie niskiego napięcia. Port RB3, 4 lub 5 (w zależności od procesora) nie może być wtedy zastosowany jako linia wejścia/wyjścia.
- ✓ podłączenie zasilania omówione zostało dokładnie w rozdziale „Konfiguracja Programatora”



Rys. 21 Sposób podłączania programatora przez łącze ICSP

Pin	Sygnal	Port
1	MCLR	MCLR
2	VCC	VDD
3	GND	VSS
4	DATA	RB7
5	CLOCK	RB6
6	PGM/LVP	RB3/4/5

Tabela 2 Opis wyprowadzeń złącza ICSP według standardu Microchip'a

Protokół programujący zaimplementowany w urządzeniu spełnia wszystkie wymagania elektryczno-czasowe jakie narzucone są do prawidłowego zaprogramowania procesora i jest całkowicie zgodny z dokumentacją techniczną firmy **Microchip®**

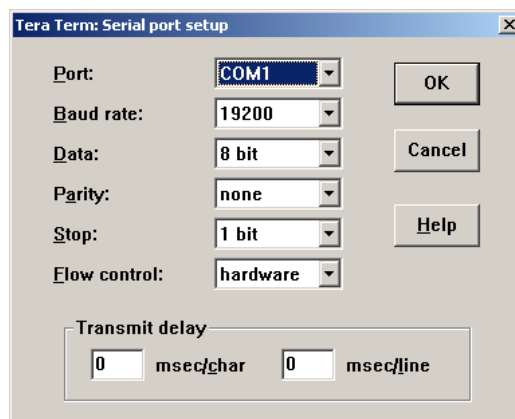
Pełny opis urządzenia w formie elektronicznej znajduje się na stronie internetowej <http://ajpic.zonk.pl/>

Upgrade programatora

Programator wyposażony w moduł bootloader'a, który umożliwia w prosty sposób ładowanie nowego kodu do procesora. Procedura uaktualniania firmware'u składa się z dwóch kroków:

Krok 1 Przygotowanie i konfiguracja sprzętu oraz oprogramowania PC

- ✓ Zainstalować terminal szeregowy (np. **Tera term Pro**, do pobrania ze strony internetowej: <http://ajpic.zonk.pl/download/tterm23.zip>)
- ✓ Uruchomić terminal i skonfigurować parametry portu COMx: Menu → Setup → Serial port... → **COMx, 19200, 8n1, hardware**



Rys. 22 Ustawienia portu terminala

- ✓ Zapisać ustawienia: Menu → Setup → Save setup... → '**teraterm.ini**'

Do wykonania upgrade'u można również wykorzystać wbudowany program **Hyper Terminal**, ustawiając parametry w ten sam sposób (może działać nieco wolniej z powodu zmienionych parametrów bufora FIFO).

Po skonfigurowaniu wszystkich wymaganych parametrów, należy przeprowadzić test połączenia:

1. Podłączyć programator do komputera i włączyć zasilanie programatora
2. Uruchomić terminal
3. Nacisnąć klawisz „s” (serial) na klawiaturze komputera

Jeśli wyświetlony zostanie numer seryjny programatora połączenie pracuje poprawnie i można przystąpić do instalowania oprogramowania.

Krok 2 Instalacja nowego oprogramowania do programatora

1. Włączyć zasilanie programatora
2. Uruchomić terminal (skonfigurować do pracy jak w Kroku 1)
3. Nacisnąć przycisk „**ERASE**” i przytrzymać
4. Nacisnąć klawisz „u” (upgrade) na terminalu
5. Wejście w tryb programowania zostanie zasygnalizowane zapaleniem się diody „**PROG**” oraz na terminalu pojawia się znak „:”.
6. Nacisnąć klawisz „l” (load), aktualny kod procesora zostanie wykasowany, a na terminalu pojawi się znak oczekiwania na plik „>”, od tego momentu programator jest pozbawiony aktywnego kodu.
7. Wysłać plik: Menu → File → Send file... → 'jupic-x.xx.xx.hex', jeśli jest wykorzystywany inny terminal należy wysłać plik w trybie tekstowym (**ASCII**).

Podczas ładowania nowego kodu w oknie terminala widoczny będzie pasek progresu, a na programatorze będzie migotała równomiernie dioda informacyjna „**PROG**”. Po zakończonej operacji poprawnego zapisu pojawi się napis „**OK!**” i programator automatycznie wystartuje z wprowadzonym programem. Od tego momentu upgrade jest zakończony, a programator gotowy do pracy.

Jeśli podczas ładowania kodu programu wystąpią jakiegokolwiek kłopoty z zapisem (np. zerwana transmisja, wyłączenie zasilania, załadowanie niewłaściwego kodu oraz inne nieprzewidziane sytuacje) i programator nie uruchomi się należy uruchomić tryb **Recovery** opisany poniżej. Brak poprawnego kodu w procesorze programator sygnalizuje trzema błyskami diodą **PROG**, a na terminalu pojawia się napis „**ERROR!**”. Prawidłowe załadowanie kodu musi zakończyć się pojawieniem się napisu „**OK!**” oraz błyskami diody **PROG** oznaczającymi poprawne wystartowanie kodu.

Tryb Recovery:

1. Wyłączyć zasilanie programatora
2. Nacisnąć przycisk „**ERASE**” i przytrzymać
3. Włączyć zasilanie programatora
4. Po tej operacji programator wchodzi w standardowy tryb aktualizacji oprogramowania jak opisany w Kroku 2 pkt. 5

Pełna lista obsługiwanych procesorów

PIC10F200	PIC16F819	PIC18F2550	PIC16C55A
PIC10F202	PIC16F83	PIC18F2580	PIC16C56
PIC10F204	PIC16F84	PIC18F2585	PIC16C56A
PIC10F206	PIC16F84A	PIC18F2610	PIC16C57
PIC10F220	PIC16F87	PIC18F2620	PIC16C57C
PIC10F222	PIC16F870	PIC18F2680	PIC16C58A
	PIC16F871	PIC18F2682	PIC16C58B
PIC12F508	PIC16F872	PIC18F2685	PIC16C620
PIC12F509	PIC16F873	PIC18F4220	PIC16C620A
PIC12F510	PIC16F873A	PIC18F4221	PIC16C621
PIC12F519	PIC16F874	PIC18F4320	PIC16C621A
PIC12F609	PIC16F874A	PIC18F4321	PIC16C622
PIC12F615	PIC16F876	PIC18F4331	PIC16C622A
PIC12F629	PIC16F876A	PIC18F4410	PIC16C62A
PIC12F635	PIC16F877	PIC18F4420	PIC16C62B
PIC12F675	PIC16F877A	PIC18F4423	PIC16C63
PIC12F683	PIC16F88	PIC18F4431	PIC16C63A
	PIC16F882*	PIC18F4431	PIC16C63A
	PIC16F883	PIC18F4450	PIC16C642
PIC16F505	PIC16F883	PIC18F4455	PIC16C64A
PIC16F506	PIC16F884	PIC18F4480	PIC16C65A
PIC16F526*	PIC16F886	PIC18F4510	PIC16C65B
PIC16F54	PIC16F887	PIC18F4515	PIC16C66
PIC16F57	PIC16F913	PIC18F4520	PIC16C662
PIC16F610	PIC16F914	PIC18F4523	PIC16C67
PIC16F616	PIC16F916	PIC18F4525	PIC16C71
PIC16F627	PIC16F917	PIC18F4550	PIC16C710
PIC16F627A	PIC16F946	PIC18F4580	PIC16C711
PIC16F628		PIC18F4585	PIC16C712
PIC16F628A	PIC18F242	PIC18F4610	PIC16C715
PIC16F630	PIC18F248	PIC18F4620	PIC16C716
PIC16F631	PIC18F252	PIC18F4680	PIC16C717
PIC16F636	PIC18F258	PIC18F4682	PIC16C72
PIC16F639	PIC18F442	PIC18F4685	PIC16C72A
PIC16F648A	PIC18F448	PIC18F6620	PIC16C73A
PIC16F676	PIC18F452	PIC18F6720	PIC16C73B
PIC16F677	PIC18F458	PIC18F8620	PIC16C745
PIC16F684	PIC18F1220	PIC18F8720	PIC16C74A
PIC16F685	PIC18F1230		PIC16C74B
PIC16F687	PIC18F1320	PIC12C508	PIC16C76
PIC16F688	PIC18F1330	PIC12C508A	PIC16C765
PIC16F689	PIC18F2220	PIC12C509	PIC16C77
PIC16F690	PIC18F2221	PIC12C509A	PIC16C770
PIC16F72	PIC18F2320	PIC12C671	PIC16C771
PIC16F73	PIC18F2321	PIC12C672	PIC16C773
PIC16F74	PIC18F2331	PIC12CE518	PIC16C774
PIC16F76	PIC18F2410	PIC12CE519	PIC16C781
PIC16F77	PIC18F2420	PIC12CE673	PIC16C782
PIC16F716	PIC18F2423	PIC12CE674	PIC16C923
PIC16F722*	PIC18F2431		PIC16C924
PIC16F727*	PIC18F2450	PIC14C000	PIC16C925
PIC16F737	PIC18F2455		PIC16C926
PIC16F747	PIC18F2480	PIC16C505	PIC16CE623
PIC16F767	PIC18F2510	PIC16C54	PIC16CE624
PIC16F777	PIC18F2515	PIC16C54C	PIC16CE625
PIC16F785	PIC18F2520	PIC16C55	
PIC16HV785	PIC18F2523	PIC16C55A	
PIC16F818	PIC18F2525	PIC16C558	

Notatki