### Programator uniwersalny mikroprocesorów ATMEL.

Zestaw do samodzielnego montażu.

#### 1.Opis ogólny.

Funkcje programatora:

- Programowanie równoległe procesorów ATMEL AT89C1051/2051, w podstawce 20 pinowej przy użyciu firmowego oprogramowania ATMELA (freeware) pod DOS i WINDOWS. Należy zadeklarować procesor AT89C1051 jako AT90S1200 D/E/F, a AT89C2051 jako AT90S2313. Tryb "emu89"
- 2. Programowanie równoległe procesorów ATMEL AT90S1200, AT90S2313 w podstawce 20 pinowej przy użyciu firmowego oprogramowania ATMELA (freeware) pod DOS i WINDOWS. Pozwala uaktywnić funkcje niedostępne przy programowaniu ISP np. ustawienie RCEN włącza oscylator wewnętrzny procesora zmniejszając liczbę elementów zewnętrznych aplikacji podstawowej nawet do zera . Skasowanie SPIEN uniemożliwia późniejsze programowanie w trybie ISP. Tryb "AVR PAR"
- 3. Programowanie w trybie ISP (In System Programming) procesorów AVR oraz AT89C8282 firmy ATMEL przez 6 pinowe złącze ISP. Obsługuje procesory AVR AT90S1200, AT90S2313, AT90S4414, AT90S8515 i inne. Zgodnie z filozofią ISP programowanie procesora odbywa się w układzie aplikacyjnym czyli w normalnym układzie pracy. Płyta powinna być wyposażona w złącze na które wyprowadzono sygnały procesora MISO, MOSI, SCK, /RESET, masę a dodatkowo plus zasilania. Linia /RESET procesora nie może być bezpośrednio zwarta z plusem zasilania (konieczny rezystor np. 10k). Linie MOSI, MISO, SCK nie mogą być zwarte z masą ani plusem zasilania. Warunki powyższe spełniają płytki eksperymentalne E100 i E200. Przy uruchamianiu układu programator może być spięty z płytą programowanego mikroprocesora. Po zakończeniu cyklu programowania układ rusza tak jak po włączeniu zasilania. Programator P200 w tym trybie reaguje na komendy protokołu firmy ATMEL i obsługiwane są przez firmowe oprogramowanie ATMELA (freeware) pod DOS i WINDOWS. Tryb "AVR ISP"
- 4. Zaimplementowane funkcje elementarne umożliwiające ustawianie i kasowanie każdego pinu programatora z poziomu komputera PC jak również ustawianie napięcia programowania, co umożliwia stworzenie własnego programu do obsługi dowolnego procesora. Tryb "generic".

Przewidziano możliwość rozbudowy programatora przez dodanie dodatkowych wymiennych płytek umieszczonych "piętrowo" w istniejących złączach, np. do programowania procesorów 40 nóżkowych 89C5x jak i pamięci EPROM i EEPROM o napięciu programującym do 12V.

Programator P200 jest przystosowany do współpracy z komputerem PC przez port szeregowy COM1..4. Wymaga zewnętrznego zasilacza o napięciu 15 V.

#### 2. Opis działania:

Schemat ideowy programatora przedstawiono na rys. 1. Układ komunikacji szeregowej RS232 z komputerem nadrzędnym PC tworzą elementy J1, Q1,Q3,Q6,D4,D6,R1,R2, R9, R4, R7, R8, R10, które dopasowują poziomy logiczne RS232 do poziomów logicznych mikroprocesora U1. Tranzystor Q3 wraz z rezystorami R1, R2, R9 zamienia poziomy napięcia linii RxD +12 V na 0V i –12V na +5V mikroprocesora. Tranzystor Q1 (nadajnik) zmienia poziomy napięć mikroprocesora +5V na ok. –7V i 0V na +12V na linii TxD. Napięcie ujemne ok 7V pobierane jest z linii RxD przez diodę D4 i kondensator C10. Napięcie dodatnie +12V pobierane jest z kondensatora C11 przez układ R16 i diody Zenera D6. Port P0 jest wejściem / wyjściem danych a porty P1 P2 P3 mikroprocesora U1 są liniami sterującymi i adresowymi . Linie P2.4 do P2.7 tworzą interfejs ISP. Linie P3.6 i P3.7 wykorzystywane są do sygnalizacji stanów programatora – dioda zielona sygnalizuje prawidłowość pracy, a dioda czerwona tryb programowania.

W trybie ISP programator łączony jest z układem programowanym przewodem ISP wykonanym wg schematu. Napięcia programujące 0V, 5V, 12V dostarcza układ U4 LM317S w typowej aplikacji wraz z tranzystorami Q2,Q4,Q5. Rezystory R11,R12, R13 tworzą przełączany dzielnik napięcia odniesienia dla U4 i należy je dobrać.

# 3. Lista podzespołów programatora P200.

lp	sztuk	nazwa	wartość / typ	oznaczenia
1	1	Gniazdo RS232 kątowe do druku	DB9M	J1
2	1	Gniazdo zasilające		JP1
3	1	Złącze ISP męskie 3x2		JP13
4	2	Wtyk 3x2 żeński		JP14 JP15
5	1	Stabilizator napięcia	LM317S	U4
6	1	Stabilizator napięcia	LM78S05	U2
7	1	Mikroprocesor AT89C52	P200 1.0	U1
8	1	Podstawka	DIP20	U3
9	4	Tranzystor NPN	BC548B*	Q3 Q4 Q5 Q6
10	2	Tranzystor PNP	BC557B*	Q1 Q2
11	1	Dioda LED 3mm	czerwona	D8
12	1	Dioda LED 3mm	zielona	D7
13	1	Dioda	1N4004	D9
14	2	Dioda	1N4148	D4 D10
15	1	Dioda Zenera 200mW	12V0	D6
16	1	Rezonaror kwarcowy HC12	18.432Mhz	X1
17	1	Kondensator ceramiczny	47nF	C7
18	2	Kondensator ceramiczny	33pF	C1 C2
19	2	Kondensator tantalowy	1uF	C9 C10
20	3	Kondensator monolit	100nF	C4 C6 C8
21	1	Elektrolit	100uF/16	C5
22	1	Elektrolit	10uF/16	C3
23	1	Elektrolit	4,7uF/50	C11
24	12	Rezystor metalizowany 0.125W	10k	R2 R3 R5 R6 R21-R28
25	5	Rezystor metalizowany 0.125W	4k7	R1 R4 R7 R8 R17
26	6	Rezystor metalizowany 0.125W	1k	R14 R15 R16 R18 R30
27	1	Rezystor metalizowany 0.125W	130	R10
28	1	Rezystor metalizowany 0.125W	240	R11
29	2	Rezystor metalizowany 0.125W	620	R19 R20
30	1	Rezystor metalizowany 0.125W	1106**	R13
31	1	Rezystor metalizowany 0.125W	2164**	R12
32	1	Złącze pinowe męskie	SIP10	JP6
33	2	Złącze pinowe męskie	SIP8	JP2 JP5
34	1	Złącze pinowe męskie	SIP4	JP3
35	1	Jumper		
36	1	Płyta drukowana P200		
37	1	Przewód	6LY0.14	
38	1	Podstawka	DIP40	

- \* dopuszcza się zamienniki
- \*\* dobrać wg opisu

#### 4. Schemat ideowy.







#### Rys.1. Schemat programatora P200.

#### 5. Protokół komunikacji komputera z programatorem:

Komunikacja programatora z komputerem nadrzędnym odbywa się poprzez port szeregowy RS232 komputera, pracujący w trybie transmisji 19200, 8, n, 1 (19200 bitów na sekundę, 8 bitów danych, brak parzystości, 1 bit stopu), Half Duplex bez potwierdzeń. Komputer wysyła kod rozkazu po którym mogą występować jeden lub dwa bajty danych. Programator w odpowiedzi na rozkaz wysyła kod potwierdzenia <CR> - 13 dziesiętnie lub jeden do kilku bajtów danych – opis w punkcie 12.

#### 6. Zasady montażu.

- Końcówki elementów wyginaj w odległości min 1 mm od korpusu (szczególnie szklane diody), zachowując promień gięcia ok. 1mm.
- Nóżki elementów od strony lutowania powinny być zagięte w kierunku ścieżki na długość ok. 2 mm, i obcięte przed lutowaniem. Elementy przed lutowaniem nie powinny wypadać z płytki odwróconej "do góry nogami".
- Do lutowania używaj lutownicy ze stabilizacją temperatury grota z ostrym stożkowym końcem. Nie używaj lutownicy transformatorowej – grozi to odklejaniem ścieżek.
- Stosuj lut cynowy CYNEL LC60 w postaci drutu 0.6 –0.8 mm z topnikiem w środku, lub podobny. Nie stosuj kalafonii a pastę lutowniczą zostaw dekarzom.
- Lutuj dwoma rękami: płytkę połóż na stole, lewą ręką podawaj drut, w prawej trzymaj lutownicę (osoby leworęczne na odwrót). Ostry koniec grota przykładaj w miejsce styku nóżki elementu i ścieżki płytki drukowanej i w to miejsce podaj lut. Przytrzymaj grot do czasu rozpłynięcia się cyny wokół nóżki (1 2 sek). Jeśli nóżka nie zwilża się cyną pocieraj ją lekko grotem lutownicy podając drugą ręką odrobinę cyny.
- Używaj jak najmniej lutu tak aby nóżki były oblane dookoła. Nie może być widać części otworu w płytce. Nie może występować widoczna granica na obwodzie styku nóżki elementy i lutu. Powierzchnia lutu powinna być błyszcząca.
- Pracując z układami scalonymi używaj odzieży bawełnianej. Zanim weźmiesz scalak do ręki dotknij ręką uziemionego metalu w celu rozładowania elektryczności statycznej.

#### 7. Montaż płytki.

- Sprawdź stan ścieżek trzymając płytkę "pod światło". Ewentualne zwarcia niedotrawienia usuń ostrym nożem.
- Sprawdź kompletność zestawu.
- Wykonaj zwory 7 szt.
- Zamontuj rezystory R21- R28, R30 (pod podstawką U1).
- Zamontuj podstawki i gniazda.
- Uwaga jeśli w przyszłości będziesz używał płyt rozszerzeń: kondensatory C1, C2 zamontuj w pozycji leżącej w stronę gniazda zasilania, a nad nimi również w pozycji leżącej rezonator X1. Na końcówki rezonatora naciągnij koszulki izolacyjne aby uniknąć zwarcia do obudowy.
- Zamontuj pozostałe rezystory, kondensatory zwracając uwagę na biegunowość (C10 tantal 1uF ma plus na masie - to nie pomyłka). Obetnij nóżki obcinaczkami.
- Zamontuj półprzewodniki. Obetnij nóżki obcinaczkami.
- Polutuj zamontowane elementy.
- Wykonaj przewód ISP wg schematu.
- Sprawdź lupą jakość lutowania na płytce , usuń zwarcia , popraw zimne luty.
- Sprawdź czy wtyk zasilacza pewnie siedzi w gnieździe. Występowanie luzów może powodować zaniki zasilania i zawieszanie pracy mikroprocesora. Doraźnym sposobem usunięcia luzu w gnieździe jest delikatne wygięcie kolca gniazda w stronę płytki.

#### 8. Uruchomienie układu.

- Zanim włożysz mikroprocesor do podstawki, włącz wtyk zasilacza do gniazda zasilającego (napięcie 15V plus w środku) sprawdź wartość napięcia zasilania mikroprocesora U1 tj pomiędzy 20 (GND) a 40 (VCC) pinem podstawki. Napięcie powinno zawierać się w przedziale 4.8 – 5.2 V. Wyłącz zasilanie.
- Sprawdź napięcie na pinie 1 podstawki U3 zwierając piny 17 i 20 podstawki U1. Dobierz wartość R12 tak aby napięcie na pinie 1 U3 wynosiło 11.7 do 12.5 V. Rozewrzyj piny 17 i 20 U1 .Dobierz wartość R13 tak aby napięcie programujące na pinie 1 podstawki U3 wynosiło 5.0 do 5.3 V.
- Umieść mikroprocesor w podstawce U1 .
- Połącz płytkę programatora z komputerem kablem RS232 Null Modem.
- Podłącz zasilacz.- powinna zapalić się zielona dioda LED.
- Ustaw tryb pracy AVR ISP jumperem według naklejki na mikroprocesorze U1.
- Uruchom program firmowy ATMEL a. Jeśli program ruszył to świadczy o prawidłowej komunikacji programatora z komputerem.
- Wybranie opcji "ADVANCED" powoduje błyśnięcie zielonej diody LED. Podobnie czytanie pamięci "FLASH" lub "EEPROM" do wcześniej nazwanego zbioru spowoduje jaśniejsze zaświecenie diody LED zielonej na czas kilku sekund i utworzenie zbioru .HEX zawierającego wartości FF heksadecymalnie.
- Pracę programatora można sprawdzić dowolnym emulatorem terminala ustawionym na 19200,8, n,1 wysyłając kody sterujące wg punktu 12. Np. wysłanie 'P' powoduje zapalenie czerwonej diody LED a 'L' zgaszenie (wejście i wyjście z trybu programowania).

#### 9. Schemat montażowy.



D1, D2 opcja

Rys. 2. Schemat montażowy programatora P200.

#### 10. Praca z programatorem

- Ustaw tryb pracy wg. naklejki na mikroprocesorze U1.
- Utwórz zbiór wynikowy w formacie INTEL Hex np. firmowym asemblerem ATMELA WAVRASM.
- Połącz programator z komputerem przewodem RS232 Null Modem.
- Umieść w podstawce programowany mikroprocesor lub połącz programator z uruchamianym układem mikroprocesorowym (np. na płycie eksperymentalnej E100 lub E200) przewodem ISP.
- Włącz wtyk zasilacza do gniazda programatora
- W trybie ISP układ uruchamiany pobiera napięcie 5V z programatora i jeśli jego pobór prądu przekracza 80 mA należy zastosować oddzielne źródło zasilania i w takim przypadku należy rozłączyć zasilanie układu z programatora tj odlutować odpowiedni przewód lub obciąć odpowiedni pin w listwie ISP układu uruchamianego.
- Uruchom program obsługi programatora np. PROGWIN, wybierz opcję "Advanced", sprawdź poprawność odczytu sygnatury, wróć do poprzedniego okna "close", opcją "Browse" otwórz zbiór wynikowy asemblera WAVRASM w formacie HEX. Wybierz w panelu "FLASH" przycisk "PROGRAM". Po skasowaniu pamięci programu, zapisie i weryfikacji następuje samoczynne uruchomienie programu mikroprocesora.
- Zmiany w programie dokonujesz w uruchomionej sesji asemblera, następnie kompilujesz, przechodzisz do okna PROGWIN-a "wybierasz opcję "RELOAD" a następnie "PROGRAM" i tak w koło aż do osiągnięcia poprawności działania układu.
- Kończąc pracę przed wyłączeniem zasilania programatora, najpierw rozłącz przewód ISP

AvrProg     Low     Hex file     C:\HOBBY\\ABK\Impuls.hex     Browse     Rejoad	Advanced  Lock bits  Mode 1 LB1:U LB2:U No program lock features
Flash Program Verify Read EEPROM Program Verify Read	Fuse bits         RC Oscillator Enable         Serial Programming Enable         Bead       Write         Chip Erase
Device	Device signature FF FF FF Target board AVR ISP Target SW rev. 1.5

Rys. 3. Okna programu PROGWIN.

#### 11. Oprogramowanie .

Przydatne oprogramowanie jest dostępne w sieci INTERNET oraz na płytach "Elektroniki Praktycznej" Asembler oraz symulatory <u>http://www.atmel.com/atmel/products/prod203.htm</u> **asmpack.**exe asembler okienkowy i pod DOS , oraz prosty symulator. (Także na płycie CD EP1) **astudio.exe** (także na płycie CD EP2) lub **astudio2.exe** – symulatory pod Windows.

Pliki do obsługi programatora. <u>http://www.atmel.com/atmel/products/prod203.htm</u>
• Aprogwin.exe (Rys. 3.)

#### Aprogdos.exe

Pod tym adresem można znaleźć źródła do wielu interesujących not aplikacyjnych między innymi AVR 910 na której oparto kod programatora. Opisy not aplikacyjnych można znaleźć pod adresem <a href="http://www.atmel.com/atmel/products/prod201.htm">http://www.atmel.com/atmel/products/prod201.htm</a> Karty katalogowe mikroprocesorów oraz opisy instrukcji są dostępne pod adresem <a href="http://www.atmel.com/atmel/products/prod200.htm">http://www.atmel.com/atmel/products/prod201.htm</a>

#### 12. Rozkazy sterujące programatorem P200 ver.1.0

Tryb AVR AT90S1200 / AT90S2313 równoległy – "AVR PAR".

Kod	Rozkaz	dane wyjścio	we	dane wejściowe	
Δ	ustaw adres	adresH adres	1	CB	
R	czytaj pamieć programu			baitH baitI	
d	czytaj bait z pamieci EEprom			bait	
c	zapisz pamieć programu bait niski	baitL		CR	
C	zapisz pamieć programu bajt wysoki	baitH		CR	
	kolejnośc	ć C bajtH c bajtL			
D	zapisz bajt do pamięci EEprom	bajt		CR	
е	kasuj mikroprocesor			CR	
Р	inicjacja programowania			CR	
F	czytaj bity i bezpieczniki			BAJTR	
f	ustaw bezpieczniki	BAJTF		CR	
S	czytaj sygnaturę			bajt bajt bajt	
<u> </u>	ustaw BITY	BITY		CR	
L	koniec programowania			CR	
Т	podaj typ procesora	TYP		CR	
V	wersja oprogramowania			bajt bajt	
V	wersja sprzętu			bajt bajt	
S	identyfikator			'AVR ISP'	
р	typ programatora			<u>'S'</u>	
t	lista procesorów			lista , 0H	
CR – kod 13 dziesiętnie					
BA	JTF: x x SPIEN x x x x RCEN; warto	sć '0' odpowiada us	tawieniu;	x bit bez znaczenia	
E	BITY: x x x x x BIT2 BIT1 x wartość	: '0' odpowiada ustav	vieniu ;x	bit bez znaczenia	
BAJTR : BIT1 BIT2 SPIEN x x x x RCEN watrość '0' oznacza ustawienie bitu ; x bit bez znaczenia					
TYP:					
	AI90S1200C	-	12h		
	A190S1200D	- 10	13n		
	AI89C1051	- 13n			
	A190S2313A	- 20h	20n		
	ATO054444A	i - ∠∪n	70h		
	AT 30344 14A ATAAQQ515A	- 4	2011 28h		
	AT89 S8252	_	S6h		

#### • tryb AVR AT90S1200 / AT90S2313 szeregowy ISP - "AVR ISP".

Kod sterujący	Rozkaz	dane wyjściowe	dane wejściowe
A	ustaw adres	adresH adresL	CR
R	czytaj pamięć programu		bajtH bajtL

d	czytaj bajt z pamięci EEprom			bajt
С	zapisz pamięć programu bajt niski		bajtL	CR
С	zapisz pamięć programu bajt wysoki		bajtH	CR
	kolejnoś	ć C bajtH c	bajtL	
D	zapisz bajt do pamięci EEprom		bajt	CR
е	kasuj mikroprocesor			CR
P	inicjacja programowania			CR
S	czytaj sygnaturę			bajt bajt bajt
<u> </u>	ustaw BITY		BITY	CR
L	koniec programowania			CR
Т	podaj typ procesora		TYP	CR
V	wersja oprogramowania			bajt bajt
v	wersja sprzętu			bajt bajt
S	identyfikator			'AVR ISP'
р	typ programatora			'S'
t	lista procesorów			lista , 0H
	CR – ko	d 13 dziesię	etnie	
BITY= x x x x x bit2 bit1 x ; wartość '0' odpowiada ustawieniu; x bit bez znaczenia				
		TYP:		
	AT90S1200C	-	12h	
	AT90S1200D	-	13h	
	AT89C1051	1 –	13h	
	AT90S2313A	-	20h	
	AT89C2051	1 -	20h	
	AT90S4414A	-	28h	
	AT90S85515A	-	38h	
	AT89 S8252	-	86h	

## • tryb programowania AT89C1051/ AT89C2051 – emu 89.

Kod	Rozkaz	dane wyjściowe	dane wejściowe		
sterujący					
A	ustaw adres	adresH adresL	CR		
R	czytaj pamięć programu		bajt bajt		
С	zapisz pamięć programu	bajt	CR		
С	zapisz pamięć programu	bajt	CR		
	kolejno	ść C bajt c bajt			
P	inicjacja programowania		CR		
S	czytaj sygnaturę		bajt bajt bajt		
I	ustaw BITY	BITY	CR		
L	koniec programowania		CR		
Т	podaj typ procesora	TYP	CR		
е	kasuj mikroprocesor		CR		
CR – kod 13 dziesiętnie					
BITY= x x x x x BIT2 BIT1 wartość '1' odpowiada ustawieniu x – bit bez znaczenia					
Тур					
	AT89C1051 - 13h				
AT89C2051 - 20h					

• tryb dostępu do portów U1 - "generic"

Kod sterujacy	Rozkaz	dane wyjściowe	dane wejściowe		
A	ustaw adres	adres	CR		
R	czytaj port		bajt		
С	zapisz do portu	bajt	CR		
а	P = P AND bajt	bajt	CR		
0	P = P OR bajt	bajt	CR		
Х	P = P XOR bajt	bajt	CR		
h	ustaw bit	bit	CR		
	kasuj bit	bit	CR		
L	restart		CR		
adres: P0 = 0 P1 = 1 P2 = 2 P3 = 3					
bit0 = 1 bit1 = 2 bit3 = 4 bit7 = 128					
CR – kod 13 dziesiętnie					

13.Kabel łączący programator z komputerem PC typu modem zerowy (null modem) 2 druty + ekran wykonać według schematu:



Modem zerowy minimalny



14.Wsparcie techniczne: http://www.perform.cc.pl