

# AVR JTAG-ICE

## Interfejs debugera dla mikrokontrolerów AVR


**AVT  
5322**

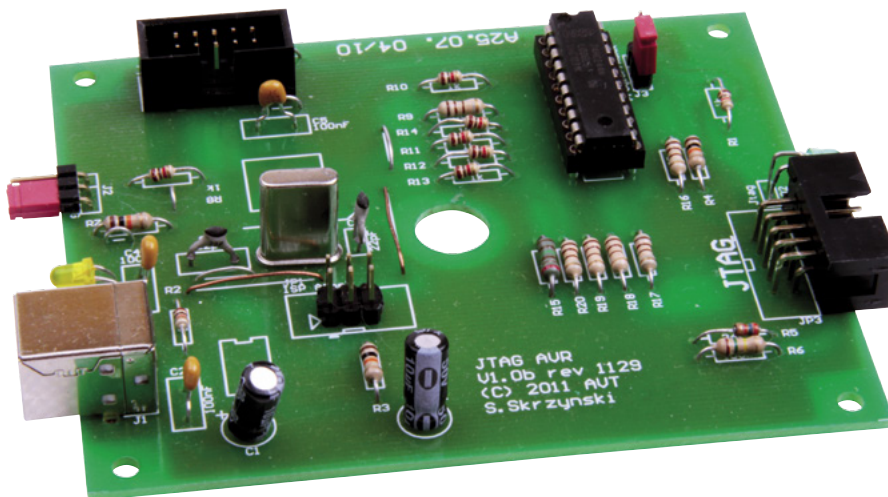
Bardzo przydatnym, a w niektórych sytuacjach wręcz niezbędnym narzędziem do uruchamiania systemów mikroprocesorowych jest debuger. Dla mikrokontrolerów AVR dobrym i tanim rozwiązaniem jest AVR Dragon. Niestety, jeśli ulegnie on uszkodzeniu, to naprawa jest najczęściej nieopłacalna.

Na stronie firmy Atmel umieszczono opis programatora/debugera zgodnego z JTAG-ICE, który można wykonać samodzielnie, ale jego wadą jest skomplikowana budowa. W artykule zaprezentowano opis programatora/debugera zgodnego i JTAG-ICE o nieskomplikowanej konstrukcji. Ponadto, wyposażono go w bufor I/O, dzięki czemu szanse uszkodzenia programatora są dość małe, a jeśli uda nam się coś zepsuć, to koszt naprawy jest bardzo niski.

**Rekomendacje:** JTAG przyda się każdemu zajmującemu się programowaniem lub uruchamianiem układów wyposażonych w mikrokontrolery AVR.

Na projekt programatora natknąłem się na stronie <http://www.m2uu.com/elektronika-avrijtag>. Jednak aby nieco uodpornić JTAG na różne sytuacje, które zdarzają się podczas uruchamiania urządzeń z mikrokontrolerami, zdecydowałem się na zmodyfikowanie oryginalnej konstrukcji. W porównaniu z oryginalnym wprowadziłem następujące zmiany:

- dodałem bufor 74HC244 zabezpieczony rezystorami szeregowymi,
- konwerter USB-RS232 zasililem z magistrali USB,
- pozostałe układy programatora zasililem z uruchamianego systemu, dzięki czemu



programator może pracować zasilany napięciem innym niż 5 V.

### Budowa i zasada działania

Schemat ideowy JTAG'a pokazano na rysunku 1. Do połączenia z komputerem służy interfejs USB. Komunikacja odbywa się poprzez UART za pośrednictwem popularnego układu konwertera, układu U1 typu FT232RL. Jest on zasilany z portu USB. Pozostałe obwody programatora są zasilane z uruchamianego systemu, dlatego wyprowadzenie VccIO układu U1 dołączono do wyprowadzenia 4 złącza JP3, a nie do portu USB.

Mikrokontroler U2 przyjmuje polecenia od komputera PC i steruje interfejsem JTAG, który jest emulowany przez port SPI. Wszystkie linie JTAG są buforowane układem U3 typu 74HC244. Sam bufor dodatkowo zabezpieczono rezystorami szeregowymi. Dla wejść są to rezystory o rezystancji 1 k $\Omega$ , natomiast dla wyjść 100  $\Omega$ . Dzięki temu podanie napięcia w zakresie  $-20...+25$  V na wejście nie powinno spowodować uszkodzenia bufora (prąd wejścia nie przekroczy 20 mA). Wyjście jest zabezpieczone przed napięciami w zakresie  $-2$  V...Vcc+2 V. A jeśli mimo wszystko bufor uszkodzi się, łatwo go wymienić, ponieważ jest zamontowany w podstawce.

W stosunku do rozwiązania dostępnego w Internecie zmieniłem wartość rezystora R6 w dzielniku napięcia. Dodałem także źródło napięcia odniesienia 2,5 V. Spowodowane było to tym, że mikrokontroler jest zasilany

**AVT-5322 w ofercie AVT:**  
AVT-5322A – płytka drukowana  
AVT-5322B – płytka drukowana + elementy

- Podstawowe informacje:**
- Lista obsługiwanych układów: ATmega128, ATmega128L, AT90CAN128, ATmega64, ATmega64L, ATmega32, ATmega32L, ATmega323, ATmega323L, ATmega16, ATmega16L, ATmega162, ATmega162L, ATmega162V, ATmega165, ATmega165V, ATmega169, ATmega169L, ATmega169V
  - Zasilanie JTAG'a z uruchamianego urządzenia.
  - Napięcie pracy 2,7...5,25 V.
  - Współpraca z AVR Studio.

**Dodatkowe materiały na CD/FTP:**  
<ftp://ep.com.pl>, user: 17692, pass: 4yv87ftn  
• wzory płytek PCB  
• karty katalogowe i noty aplikacyjne elementów oznaczonych w Wykazie elementów kolorem czerwonym

**Projekty pokrewne na CD/FTP:**  
(wymienione artykuły są w całości dostępne na CD)

- AVT-5279 Programator PIC (EP 2/2011)
- AVT-5172 Uniwersalny programator mikrokontrolerów AVR (EP 2/2009)
- AVT-5153 Uniwersalny programator JTAG/ISP (EP 10/2008)
- AVT-5125 Programator USB AVR (STK500) (EP 2/2008)
- AVT-1462 Uniwersalny adapter dla programatorów AVR-ISP (EP 2/2008)
- AVT-2855 Ulepszony programator STK200 (EdW 2/2008)
- AVT-988 Programator AVRISP z interfejsem USB (STK500) (EP 7/2007)
- AVT-1452 Adapter dla programatorów AVR ISP (EP 7/2007)
- AVT-947 Programator JTAG dla mikrokontrolerów STR9 (EP 9/2006)
- AVT-937 Programator ISP/ICP dla mikrokontrolerów ST7 (EP 7/2006)
- AVT-921 Flash z ISP – JTAG (EP 3/2006)
- AVT-451 Programator z interfejsem USB dla Bascom AVR (EP 11/2005)
- AVT-1409 Programator JTAG dla układów MSP430 (EP 3/2005)
- AVT-540 Miniprogramator AT89C051 (EP 11/2004)

z uruchamianego systemu, a napięcie odniesienia JTAG-ICE pochodzi z wejścia Aref. Gdyby mikrokontroler był zasilany napięciem 5 V, to pomiar napięcia występującego na złączu JTAG byłby prawidłowy. Niestety, konieczne byłby zasilanie JTAG'a z jakiegoś stabilizatora, ponieważ złącze USB komputera PC nie gwarantuje napięcia zasilającego 5 V. Może ono wynosić np. tylko 4,4 V. Dzięki podzieleniu napięcia ze złącza JTAG przez 2 w stosunku do oryginału (rezystor R6 o rezystancji 75 kΩ, a nie 150 kΩ) oraz zastosowa-

niu napięcia Aref o wartości 2,5 V zamiast 5 V, pomiary napięcia wykonywane przez AvrStudio są prawidłowe. Możliwe jest oczywiście zrezygnowanie z U4. Wtedy rezystor R6 musi mieć rezystancję 150 kΩ.

**Uwaga! Jeśli nie korzystamy z U4 (R6 = 150 kΩ), pomiar może być obciążony dużym błędem. Wynika to z faktu, że napięcie w złączu USB może się zawierać w granicach 4,4...5 V. W skrajnym przypadku (4,4 V) błąd wynosi +12%. W takiej sytuacji faktyczne napięcie 5 V będzie przedstawio-**

ne jako 5,6 V, natomiast 3,3 V jako 3,69. Do błędów tego należy doliczyć jeszcze tolerancję rezystorów R6 i R7. W prototypie bez U4, 5 V było obrazowane w AvrStudio jako 6,2 V!

### Montaż i uwagi odnośnie do komponentów

Schemat montażowy JTAG'a pokazano na rysunku 2. Montaż jest typowy i nie wymaga szczegółowego opisu. Na początku warto wzlutować układy SMD znajdujące się od strony ścieżek. Wzlutowanie U2 nie sprawi większego kłopotu, natomiast U1 jest już bardziej kłopotliwy w montażu ze względu na mały raster wyprowadzeń. Osobiście z U1 poradziłem sobie w ten sposób, że po pozycjonowaniu układu i przylutowaniu skrajnych nóżek, załamałem stopem lutowniczym wszystkie nóżki, a następnie nadmiar cyny zebrałem za pomocą plecionki „WIK”. Po wzlutowaniu elementów SMD montujemy zwory. Pod U3

#### Wykaz elementów

##### Rezystory:

- R1, R2, R8, R10...R15, R21: 1 kΩ
- R5: 36 kΩ
- R6: 75 kΩ (150 kΩ – opis w tekście)
- R9, R16...R20: 100 Ω
- R3, R4, R7: 10 kΩ

##### Kondensatory:

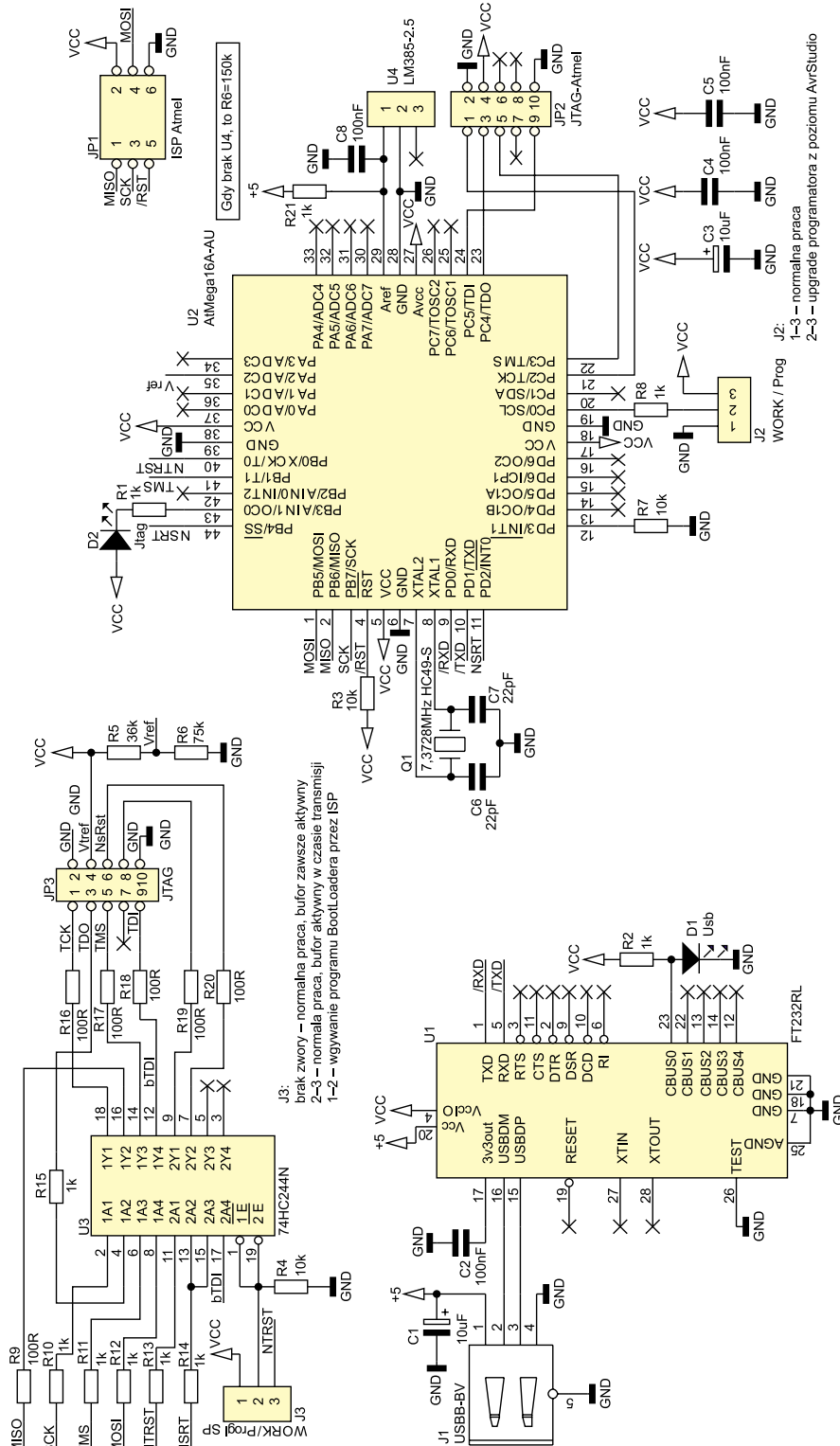
- C1, C3: 10 μF/10 V
- C6, C7: 22 pF/50 V
- C2, C4, C8: 100 nF/50 V

##### Półprzewodniki:

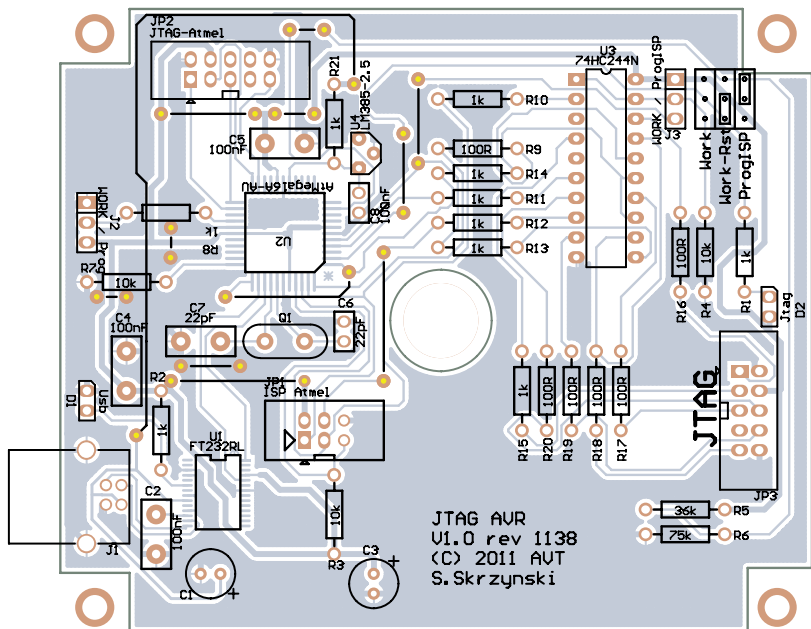
- U2: ATmega16A-16AU (QFP44)
- U1: FT232RL (SSOP-28)
- U4: LM385-2.5 (TO-92, opis w tekście)
- U3: 74HC244N (DIP-20)
- D1: dioda LED żółta 3 mm
- D2: dioda LED zielona 3 mm
- Inne:
- Q1: kwarc 7,3728 MHz (HC-49S)
- JP1: ZL231-06PG (6 pin proste) lub ZL311-2×3 (listwa goldpin prosta 2×3)
- JP2: ZL231-10PG (10 pin proste)
- JP3: ZL231-10GK (10 pin kątowe)
- J1: USB-B-BV (gniazdo USB-B do druku kątowe)
- J2: ZL201-3 (listwa goldpin prosta 1×3) + zworka
- J3: ZL211-3 (listwa goldpin kąтова 1×3) + zworka

REKLAMA

Na CD: karty katalogowe i noty aplikacyjne elementów oznaczonych w wykazie elementów kolorem czerwonym



Rysunek 1. Schemat ideowy JTAG'a AVR



Rysunek 2. Schemat montażowy JTAG'a AVR

należy zastosować podstawkę. Płytkę jest przystosowana do umieszczenia w obudowie KM-35.

Jako układ U3 można zastosować 74HC244 lub 74AC244. Nie mogą to być układy typu HCT lub ACT, ponieważ nie będą one pracować przy napięciach różniących się od 5 V o więcej niż 10%. Nie może to też być układ LS, nawet gdy programator miałby pracować przy napięciach 5 V, a to ze względu na prąd wejścia (konieczna byłaby korekta wartości rezystorów zabezpieczających).

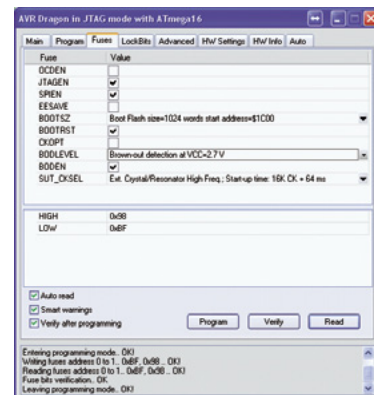
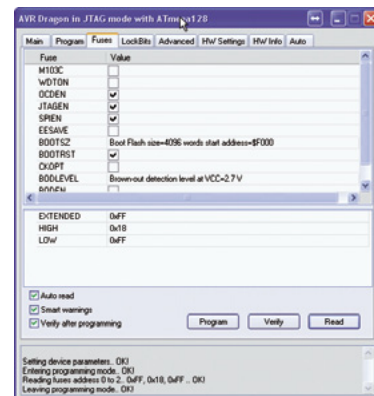
Jako U2 można użyć AtMega16A-AU lub AtMega16L-8AU. Nie może to być AtMega16-AU, ponieważ napięcia pracy zo-

staną ograniczone do zakresu 4,5...5,25 V (górną granicę napięcia wprowadza układ U1).

### Uruchomienie

W pierwszej kolejności należy podłączyć nasze urządzenie do portu USB komputera (wystarczy zasilanie z USB). Sposobu instalacji sterowników nie będę opisywał, ponieważ ten temat był już poruszony wielokrotnie. Ze względu na zastosowanie układu FT232RL można przy użyciu programu „MProg.exe”, ustawić funkcje wyprowadzenia C0 na „TX & RX LED#” (rysunek 3).

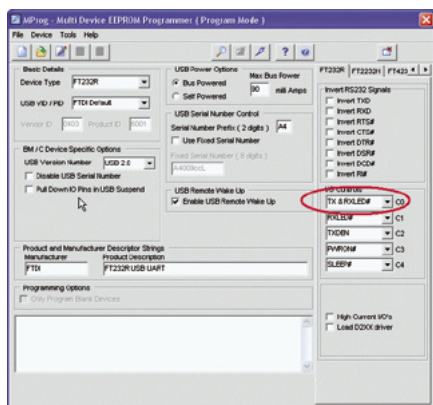
Następnym krokiem będzie zaprogramowanie U2, o ile nie korzystamy z już zaprogramowanego mikrokontrolera. Na



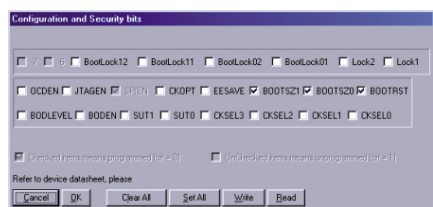
Rysunek 5. Ustawienie fuse bitów mikrokontrolera za pomocą AVR Studio

plytce są dostępne dwa złącza programujące: JP2 dla programatora JTAG oraz JP1 dla programatora ISP. Jeśli używamy programatora ISP, to zwórkę na J3 należy ustawić w pozycji „PROG isp” (zwarłe piny 1-2). Przy programowaniu za pomocą JTAG'a ustawienie J3 nie ma znaczenia.

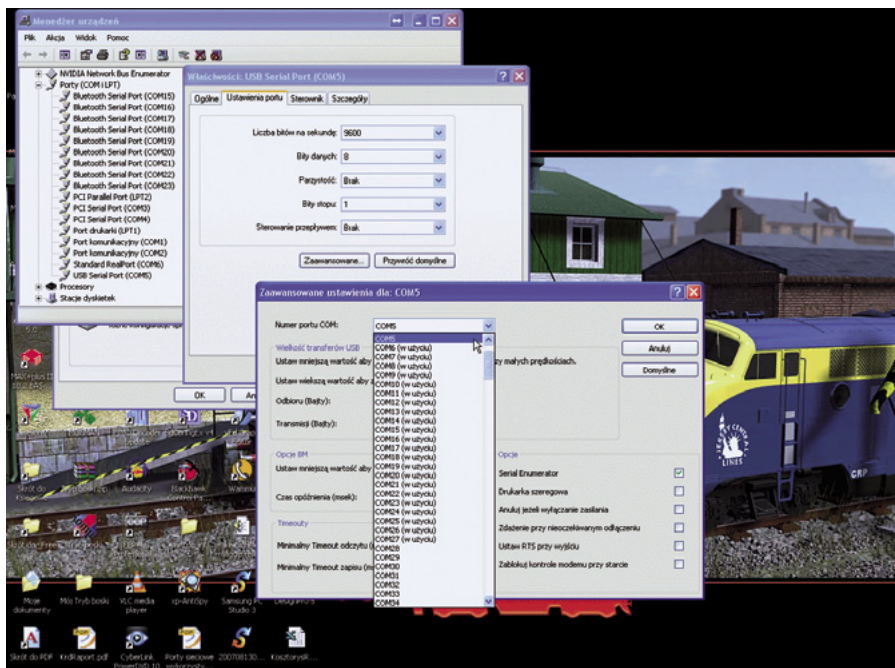
W materiałach dodatkowych do artykułu znajduje się plik *avr\_jtag\_m16.hex*. Jest to plik firmware działającego programatora JTAG-ICE. Po podłączeniu zasilania



Rysunek 3. Ustawienie funkcji wyprowadzenia C0

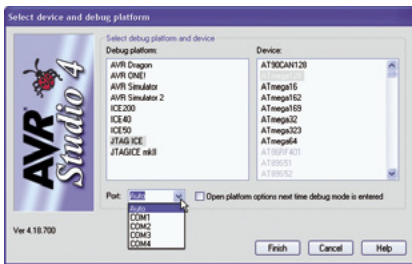


Rysunek 4. Ustawienie fuse bitów mikrokontrolera za pomocą PonyProg



Rysunek 6. Ustawienie numeru portu COM

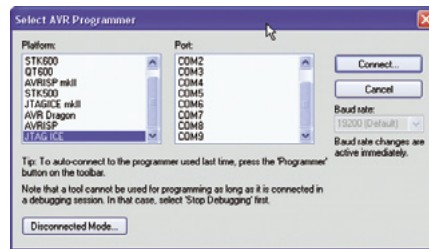




Rysunek 7. Numery portów COM w trybie debug AVR Studia

do programatora przez złącze JP3 „JTAG”. Po zapisaniu pamięci Flash mikrokontrolera, należy ustawić bity konfiguracyjne zgodnie z **rysunkiem 4** (PonyProg). Dla programatora kompatybilnego z AvrStudio ustawienia będą takie, jak na **rysunku 5**.

Po zaprogramowaniu mikrokontrolera, zworkę J3 przełączamy w pozycję *WORK* (zwarłe piny 2-3) lub zdejmujemy ją całkowicie. Jeśli zworka została zdjęta, to bufor U3 jest cały czas dołączony do programowanego systemu. Jeśli zworka zwiera piny 2-3, bufor jest przyłączany tylko na czas programowania. Na J2 zakładamy jumper zwierający piny 1-2 (w pozycji *WORK*). Po wykonaniu powyższych czynności, programator powinien być gotowy do pracy. Jeśli tak nie jest, należy wykonać następujące kroki:



Rysunek 9. Numery portów COM w trybie programowania

Urządzenie programujemy plikiem bootloadera (plik *avrboot\_1.hex*), ustawienie bitów konfiguracyjnych jak na **rysunku 4** i **rysunku 5**.

Uruchamiamy program terminala (*HyperTerminal*, *ExTerm*, *Termite* itp.) ustawiamy parametry transmisji 19200, 8, N, 1. Numer portu COM odczytujemy z menedżera urządzeń. Musi się ona zawierać w przedziale 1...4. Jeśli wirtualny port COM interfejsu JTAG otrzymał inny numer, należy go zmienić w menedżerze urządzeń (menu podręczne *Właściwości*, zakładka *Ustawienia portu*, przycisk *Zaawansowane* – **rysunek 6**).

W programatorze ustawiamy zworkę J2 w pozycji *PROG* (uruchomienie bootloadera), po czym dołączamy zasilanie za pomocą złącza JP3 – *JTAG*.

Z terminala wysyłamy znak *S*. W odpowiedzi powinniśmy otrzymać komunikat *AVRBOOT*.

Z menu *Tools* środowiska AvrStudio wybieramy *AVRprog*, a tam wskazujemy na plik *upgrade.ebn*, który po zainstalowaniu AVR Studia standardowo jest umieszczany w folderze *C:\Program Files\Atmel\AVR Tools\JTAGICE\*, po czym naciskamy *Program*.

Po bezbłędnym załadowaniu programu zworkę J2 przełączamy w pozycję *WORK* (zwarłe piny 1-2), natomiast J3 przełączamy zgodnie z wcześniejszym opisem.

**Uwaga!** W trybie *debug* AvrStudio akceptuje numery portów COM w zakresie 1...4! (**rysunek 7**). W wypadku wgrywania programu do systemu docelowego numery portów COM mogą być zawarte w zakresie 1...9 (**rysunek 8**).

**Sławomir Skrzyński, EP**

#### Bibliografia:

Jacek A. Michalski „Interfejs JTAG do procesorów AVR”, *Elektronika Praktyczna* 6/2004, str. 10 (AVT-581)

<http://www.m2uu.com/elektronika:avrjtag>

<http://www.elektroda.pl/rtvforum/topic1463084.html>