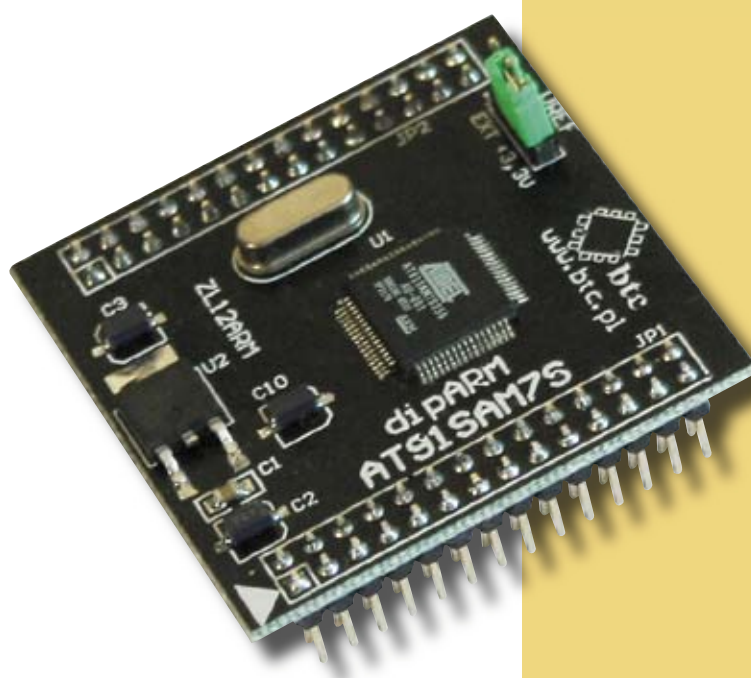


**Moduł DIP z mikrokontrolerem  
AT91SAM7S  
(rdzeń ARM7TDMI-S)**dip  
ARM

*Moduł dipARM-SAM7S opracowano z myślą o ułatwieniu powszechnego stosowania mikrokontrolerów z rodziny AT91SAM7S przez konstruktorów, którzy nie mogą lub nie chcą inwestować w zautomatyzowany montaż elementów o relatywnie „gęstym” rastrze rozmieszczenia wyprowadzeń – 0,5 mm.*

## Podstawowe parametry modułu ZL12ARM

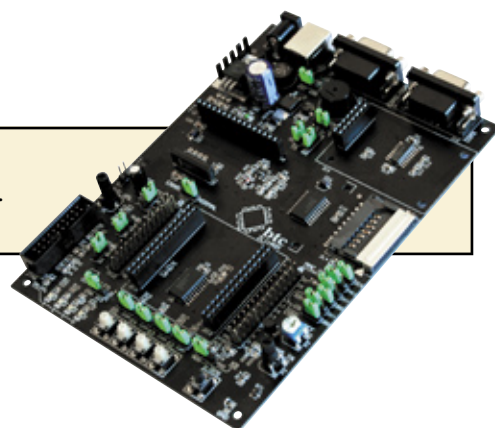
- ▶ mikrokontroler AT91SAM7S256 lub AT91SAM7S64
- ▶ pojemność pamięci Flash: 256 kB lub 64 kB
- ▶ pojemność pamięci SRAM: 64 kB lub 16 kB
- ▶ liczba linii I/O: 32
- ▶ wbudowany interfejs USB 2.0
- ▶ wszystkie linie I/O są przystosowane do współpracy z układami cyfrowymi zasilanymi napięciami: 3,3 lub 5 V
- ▶ możliwość wyboru źródła napięcia odniesienia dla toru analogowego
- ▶ częstotliwość taktowania wewnętrznego: 12 MHz
- ▶ maksymalna częstotliwość taktowania zewnętrznego: 55 MHz
- ▶ zasilanie: 4,4...7 VDC/65 mA (maks.)
- ▶ możliwość programowania ISP (z wykorzystaniem bootloadera SAM-BA i interfejsu DBG)
- ▶ współpraca z interfejsem JTAG (ZL14PRG)
- ▶ przystosowany do współpracy z płytą bazową ZL11ARM



Do modułu dipARM-SAM7S (ZL12ARM) są dostępne bezpłatne biblioteki PCB i SCH do Protela 99SE i DXP. Można je pobrać pod adresem: [http://www.btc.pl/pliki/zl12arm\\_libs.zip](http://www.btc.pl/pliki/zl12arm_libs.zip).

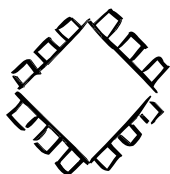


Moduł ZL12ARM współpracuje z płytą bazową ZL11ARM.



## Wyposażenie standardowe

Kod	Opis
ZL12ARM_7S64	dipARM_SAM7S64 – zmontowany i uruchomiony moduł z procesorem AT91SAM7S64.
ZL12ARM_7S256	dipARM_SAM7S256 – zmontowany i uruchomiony moduł z procesorem AT91SAM7S256.



**btc**

BTC Korporacja  
05-120 Legionowo  
ul. Lwowska 5  
tel.: (022) 767-36-20  
faks: (022) 767-36-33  
e-mail: [biuro@kamami.pl](mailto:biuro@kamami.pl)  
<http://www.kamami.pl>

Zastrzegamy prawo do wprowadzania zmian bez uprzedzenia.

Oferowane przez nas płytki drukowane mogą się różnić od prezentowanej w dokumentacji, przy czym zmianom nie ulegają jej właściwości użytkowe.

BTC Korporacja gwarantuje zgodność produktu ze specyfikacją.

BTC Korporacja nie ponosi odpowiedzialności za jakiegokolwiek szkody powstałe bezpośrednio lub pośrednio w wyniku użycia lub nieprawidłowego działania produktu.

BTC Korporacja zastrzega sobie prawo do modyfikacji niniejszej dokumentacji bez uprzedzenia.

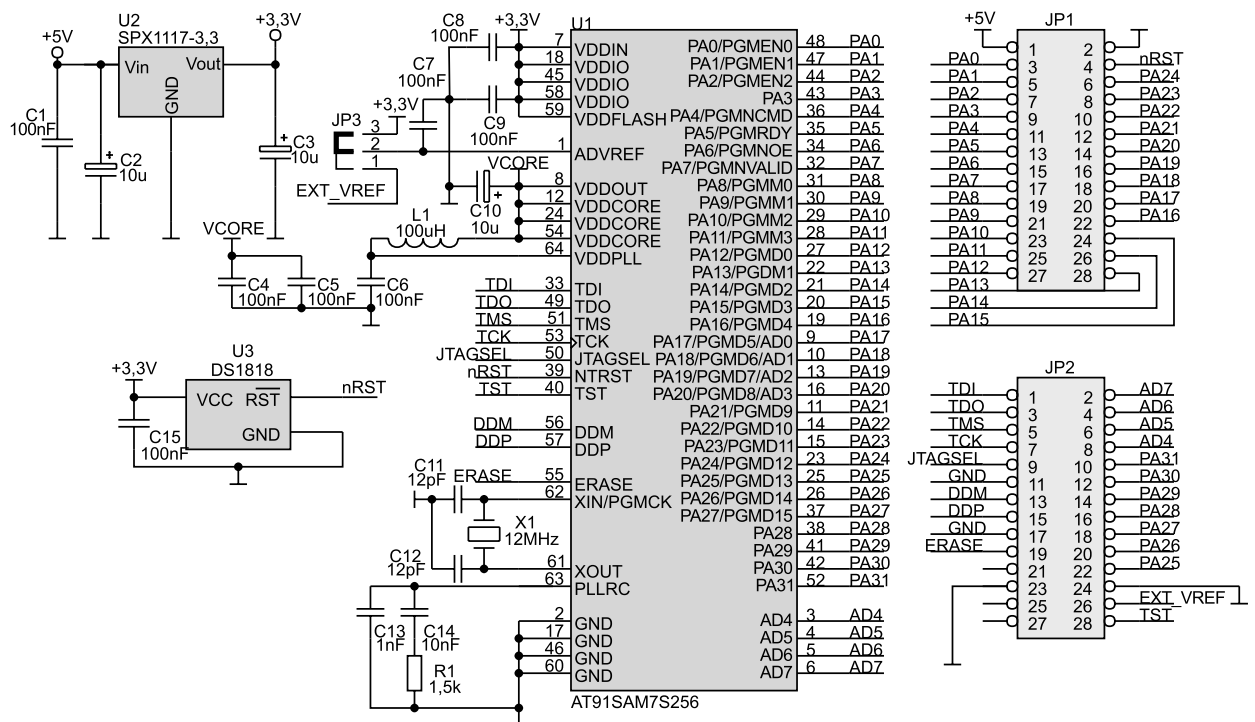
## Budowa modułu

Moduły dipARM-SAM7S (schemat elektryczny pokazano na **rys. 1**) są standardowo wyposażane w mikrokontrolery AT91SAM7S256 lub AT91SAM7S64. Moduł dipARM-SAM7S jest przystosowany do zasilania napięciem 4,4...7 VDC, a wszystkie linie I/O mogą bezpośrednio współpracować z układami cyfrowymi zasilanymi napięciami 3,3...5 VDC.

Moduł wyposażono w liniowy stabilizator napięcia zasilającego oraz rezonator kwarcowy o częstotliwości rezonansowej 12 MHz. Zastosowany rezonator umożliwia programowanie mikrokontrolera w systemie wyłącznie poprzez interfejs DBGU. Programowanie mikrokontrolera poprzez USB wymaga zastosowania kwarcu o częstotliwości rezonansowej 18,432 MHz.

Moduł dipARM-SAM7S wyposażono we własny generator sygnału zerującego (U3 na rys. 1), którego wyjście jest typu OD (*open drain*), w związku z czym wyprowadzony na złącze szpilkowe sygnał *nRST* można wykorzystać do zerowania układów zewnętrznych lub zerować mikrokontroler za pomocą sygnału zewnętrznego (aktywne „0” logiczne).

Zworka JP3 służy do wybrania źródła napięcia odniesienia dla toru analogowego: może to być napięcie 3,3 V służące do zasilania mikrokontrolera lub dowolne źródło zewnętrzne o wartości z przedziału 2,6...3,3 V.



Rys. 1. Schemat elektryczny modułu dipARM-SAM7S (ZL12ARM)

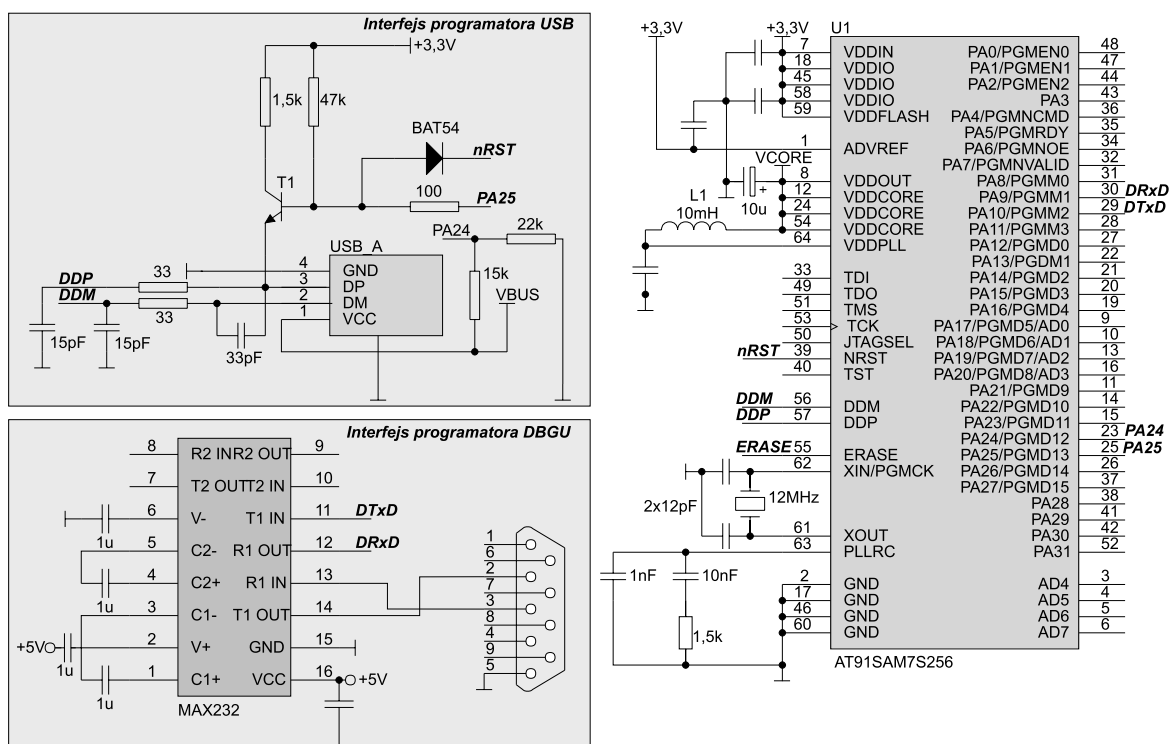
## Programowanie mikrokontrolera

Mikrokontroler zastosowany w module dipARM-SAM7S jest przystosowany do programowania w systemie/aplikacji (ISP/IAP). Do programowania jest niezbędny dostępny bezpłatnie program SAM-BA (*SAM Boot Assistant*) oraz dwukierunkowy konwerter napięciowy zgodny z RS232 dołączony do linii Tx i Rx interfejsu DBGU (*DeBuG Unit*). Schematy interfejsów komunikacyjnych wykorzystywanych do programowania pamięci mikrokontrolerów pokazano na rys. 2.

Mikrokontrolery AT91SAM7S przed pierwszym użyciem muszą zostać zainicjalizowane (do pamięci Flash jest wtedy kopiowany *bootloader*), co wymaga dołączenia przez użytkownika wejścia TST do logicznej „1” i wyzerowania mikrokontrolera. Przejście w tryb uaktywniania *bootloadera* wymaga dołączenia „1” także do linii PGMEN0...2, ale można je zostawić także niepodłączone, mają bowiem wewnętrzne *pull-upy*. Po upływie 10 sekund (minimalny czas programowania każdej strony pamięci Flash wynosi 6 ms) na wejście TST podajemy „0” i zerujemy mikrokontroler. Od tej chwili jest on gotowy do współpracy z programem *SAM-Prog* służącym do programowania pamięci Flash i ładowania programów do pamięci SRAM.

**Uwaga:** producent mikrokontrolerów nie przewidział żadnej sygnalizacji poprawnego przebiegu kopiowania *bootloadera* do pamięci Flash, dlatego nie należy zbytnio skracać czasu przeznaczanego na zapisywanie pamięci Flash!

Zainicjowanie *bootloadera* w pamięci Flash mikrokontrolera jest konieczne za każdym razem, gdy chcemy zaprogramować ją w systemie – wymaga to każdorazowo powtórzenia opisanej powyżej procedury. Każdorazowo także przed kopiowaniem *bootloadera* pamięć Flash mikrokontrolera należy wykasować. W tym celu na wejście ERASE mikrokontrolera należy podać logiczną „1” i wyzerować mikrokontroler. Czas kasowania pamięci wynosi co najmniej 15 ms.



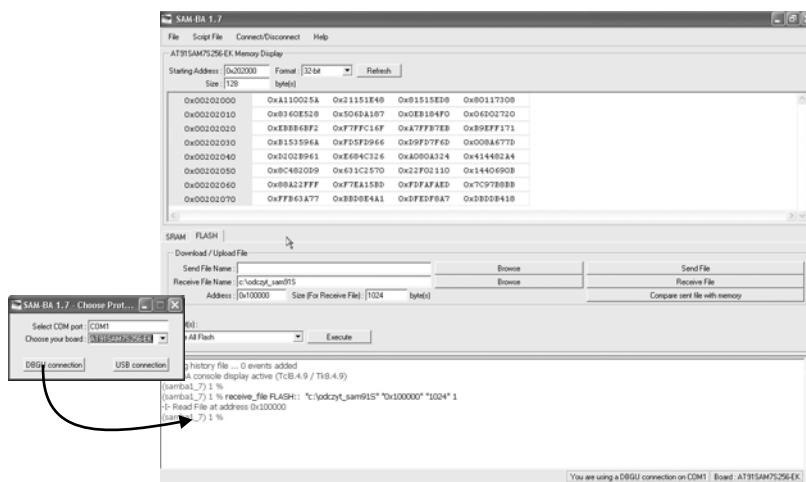
Rys. 2. Schematy interfejsów komunikacyjnych wykorzystywanych do programowania pamięci mikrokontrolerów AT91SAM7S



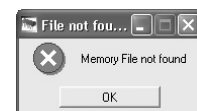
Program SAM-BA jest dostępny bezpłatnie w Internecie pod adresem: <http://www.at91.com>.

## Programowanie pamięci Flash modułu ZL12ARM na płytce bazowej ZL11ARM

1. Za pomocą zworek COM1 (JP10 i JP11) wybieramy interfejs DBG.
2. Zworki TX i RX (JP13, JP12) powinny znajdować się w pozycji *On*.
3. Na liniach mikrokontrolera *PGMEN0...2* należy zapewnić stany logiczne „1” (linie mają wbudowane wewnętrzne *pull-upy*).
4. Do gniazda COM0/DGB (Z11) dołączamy kabel RS232, dołączony drugim końcem do komputera PC (na którym musi być zainstalowany program *SAM-BA*).
5. W gnieździe JP19/21 montujemy moduł dipARM-SAM7S (ZL12ARM) i włączamy zasilanie.
6. Jumper JP24 ustawiamy w pozycję *ERASE* i za pomocą przycisku S1 (*RES*) zerujemy mikrokontroler (minimalny czas kasowania wynosi 15 ms).
7. Następnie zworkę JP24 przestawiamy w pozycję *NORMAL*, a zworkę *TST* w pozycję *1* i zerujemy mikrokontroler.
8. Po upływie co najmniej 10 sekund zworkę *TST* przestawiamy w pozycję *0* i ponownie zerujemy mikrokontroler.
9. Uruchamiamy program *SAM-BA*. Poprawna praca *bootloadera* jest sygnalizowana otwarciem okna programu jak na **rys. A1**, jeżeli komunikacja pomiędzy programem i mikrokontrolerem nie została nawiązana, pojawia się okno jak na **rys. A2**.



**Rys. A1.** Duże okno *SAM-BA* otwiera się po nawiązaniu komunikacji pomiędzy programem i mikrokontrolerem. Jest to objaw poprawnie działającego *bootloadera*.



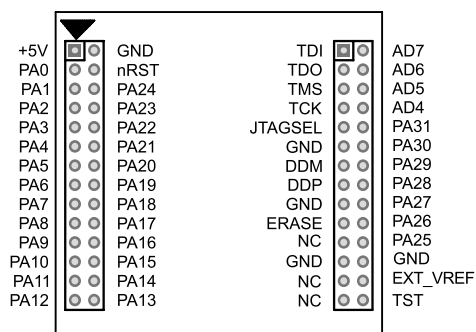
**Rys. A2.** Tak jest sygnalizowany brak komunikacji pomiędzy komputerem i mikrokontrolerem.

**UWAGA**

1. Należy pamiętać, że odtworzenie *bootloadera* w pamięci Flash mikrokontrolera jest konieczne za każdym razem, gdy chcemy zaprogramować ją ISP – wymaga to każdorazowo powtórzenia opisanej powyżej procedury.
2. Każdorazowo także przed kopiowaniem *bootloadera* pamięć Flash mikrokontrolera należy wykasować.

## Wprowadzenia modułu

Na rys. 3 pokazano rozmieszczenie sygnałów na złączach modułu dipARM-SAM7S. Wszystkie linie I/O mogą współpracować bez konieczności wprowadzania jakichkolwiek modyfikacji z układami cyfrowymi zasilanymi napięciem 3,3 lub 5 V.



*Rys. 3. Przypisanie sygnałów do styków modułu dipARM-SAM7S (ZL12ARM)*



Linie I/O modułu dipARM-SAM7S są kompatybilne z logiką 3,3 oraz 5 V.