Użycie złącza JTAG w systemach mikroprocesorowych do testowania integralności połączeń systemu oraz oprogramowania zainstalowanego w pamięciach stałych.

JTAG – Joint Test Action Group

mgr inż. Tadeusz Andrzejewski

Plan prezentacji

1. Normy określające złącze JTAG,
2. Czynniki wymuszające testowanie układów,
3. Omówienie złącza JTAG,
4. Przedstawienie koncepcji testu oprogramowania umieszczonego w pamięciach stałych.

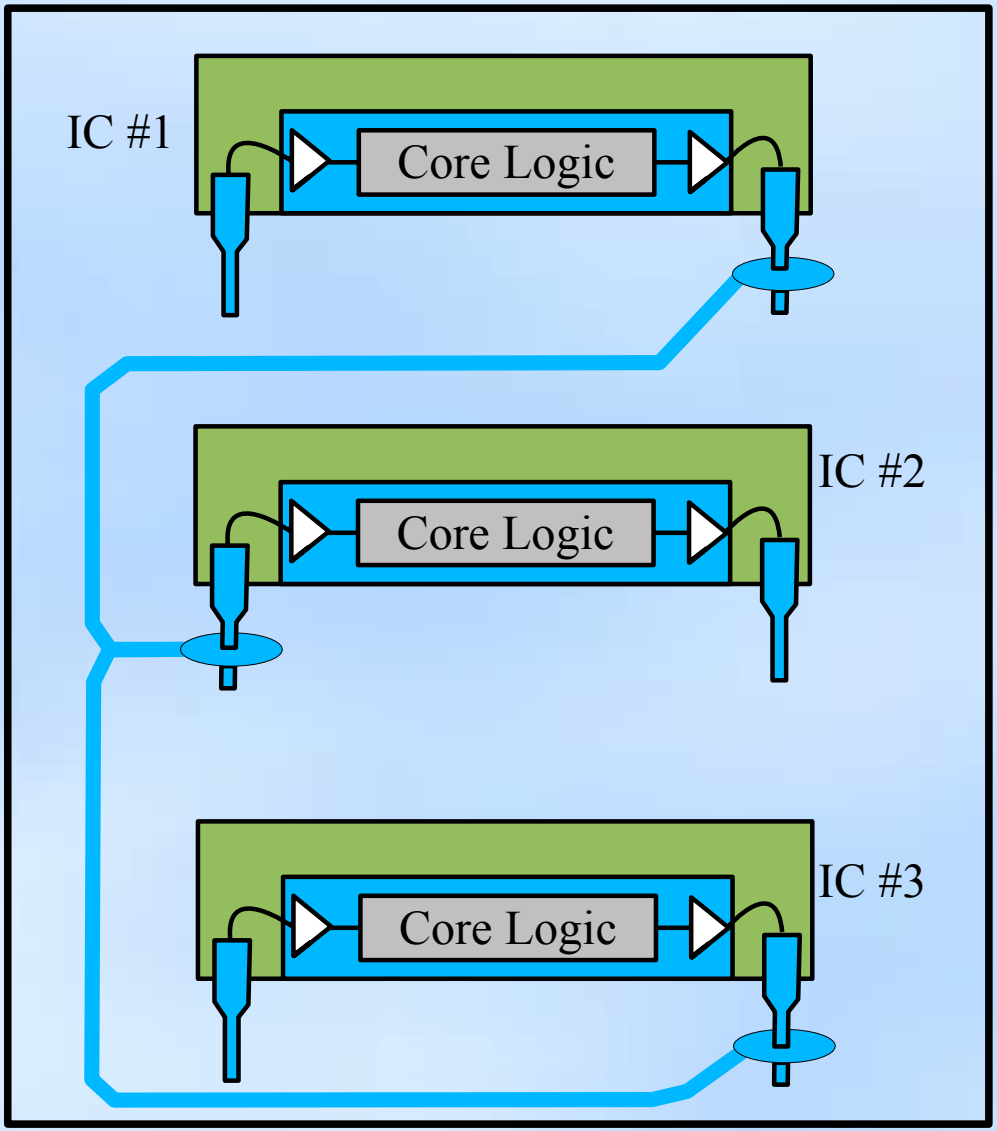
Normy określające JTAG

1. IEEE 1149.1 - 2001 Standard Test Access Port and Boundary-Scan Architecture
2. IEEE 1149.4 - 1999 Standard for a Mixed-Signal Test Bus
3. IEEE 1149.6 - 2003 Standard for Boundary-Scan Testing of Advanced Digital Networks

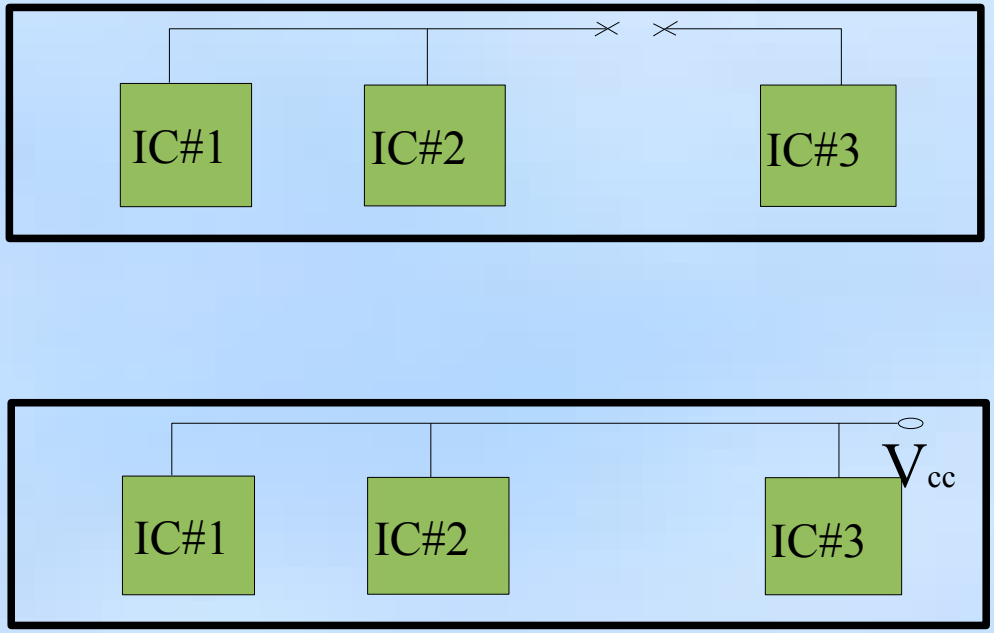
Czynniki wymuszające implementację testów wbudowanych

1. Potrzeba usunięcia wszystkich błędów występujących w układzie,
2. Obniżenie kosztów wprowadzenia produktu na rynek,
3. Trudności w serwisowaniu urządzeń pozbawionych testów wewnętrznych.

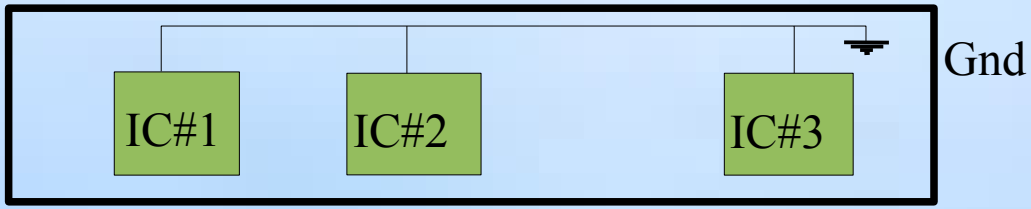
Przykład połączenia: wyjście układu steruje wejścia dwóch układów

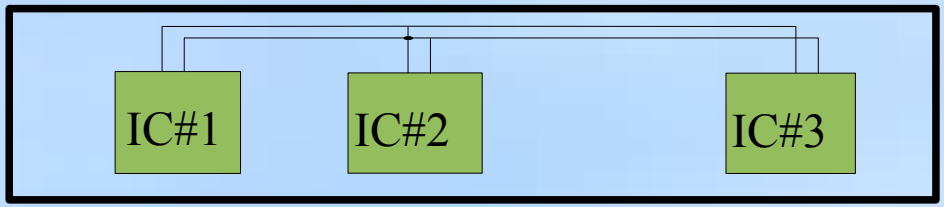


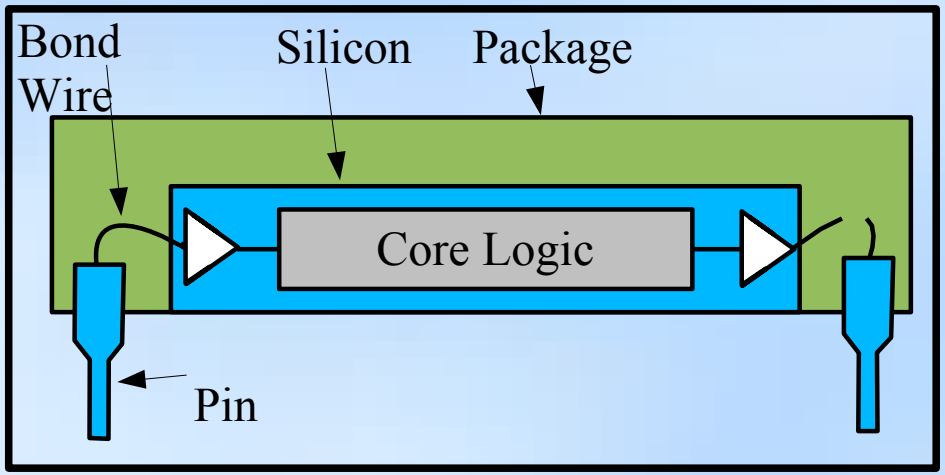
Przykłady uszkodzonych połączeń



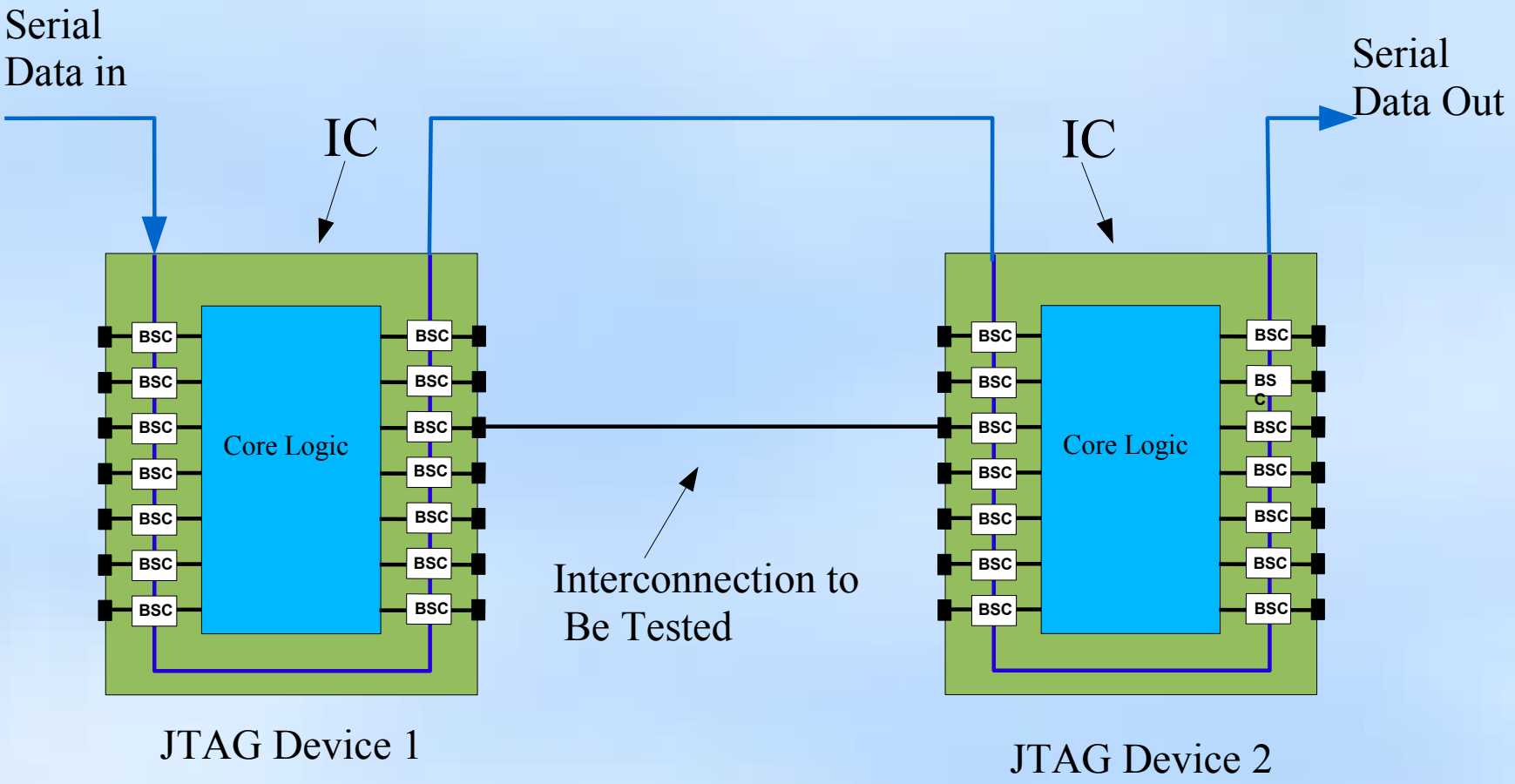
Przykłady uszkodzonych połączeń





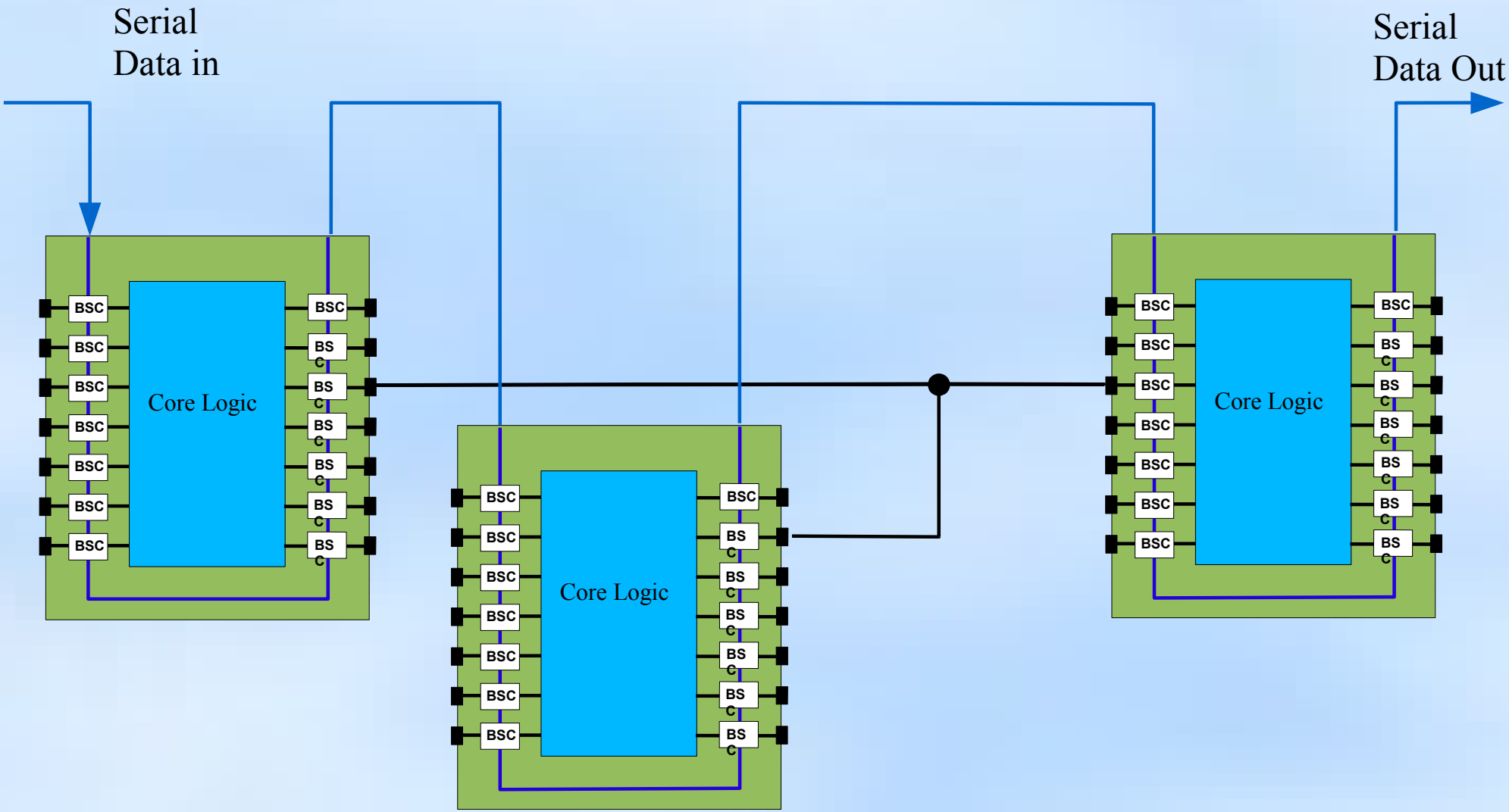


Idea testu połączenia

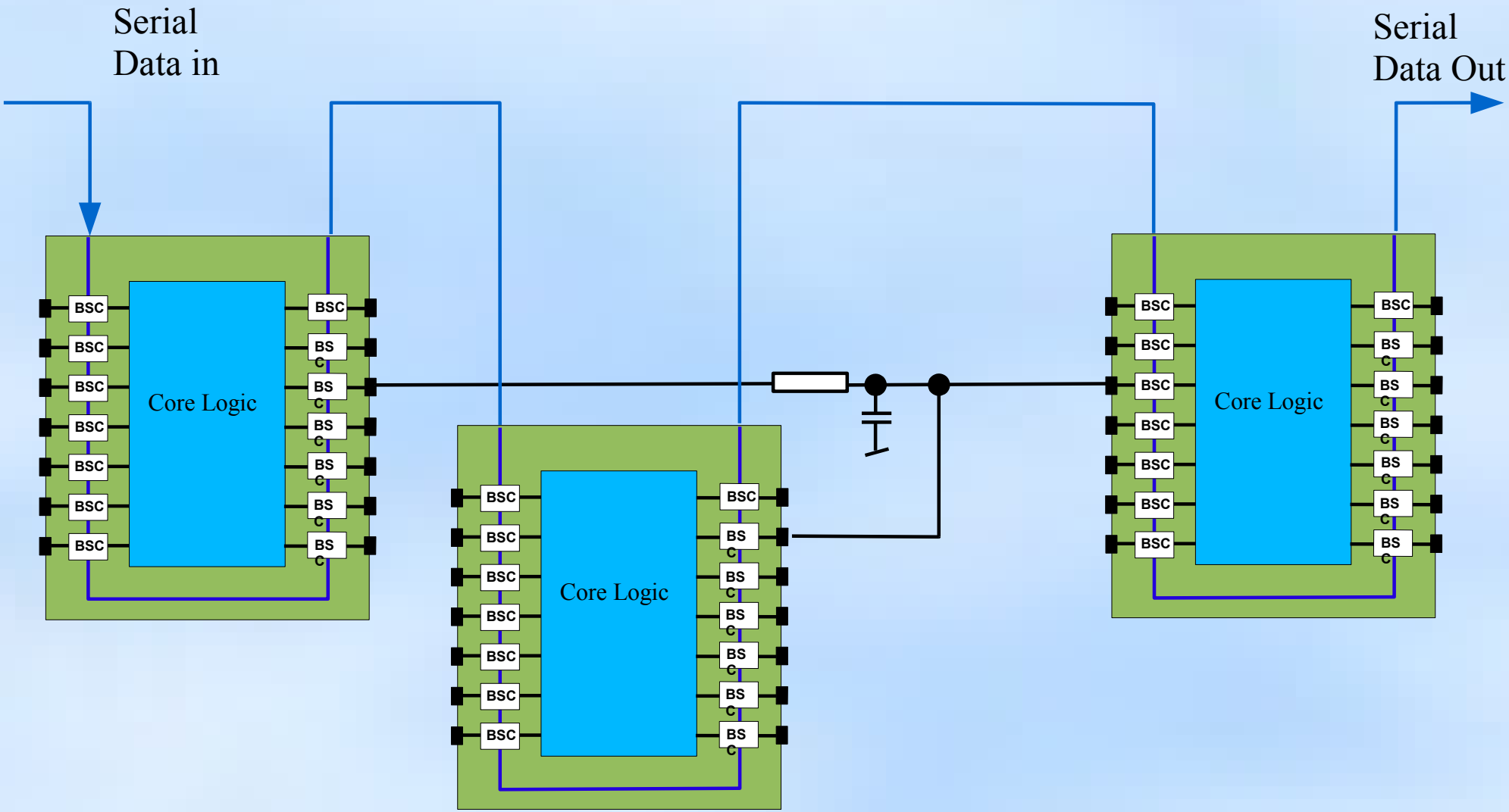


BSC – Boundary Scan Cell

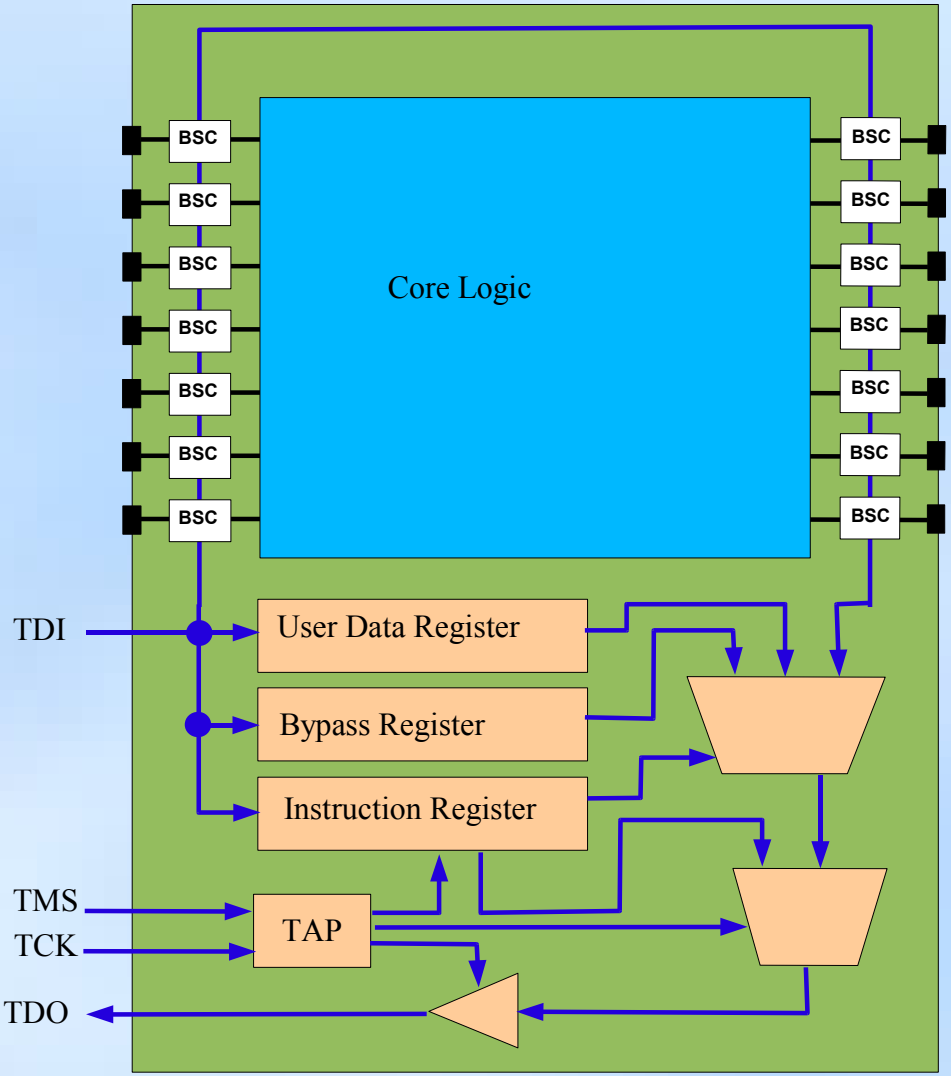
Idea testu połączenia



Idea testu połączenia

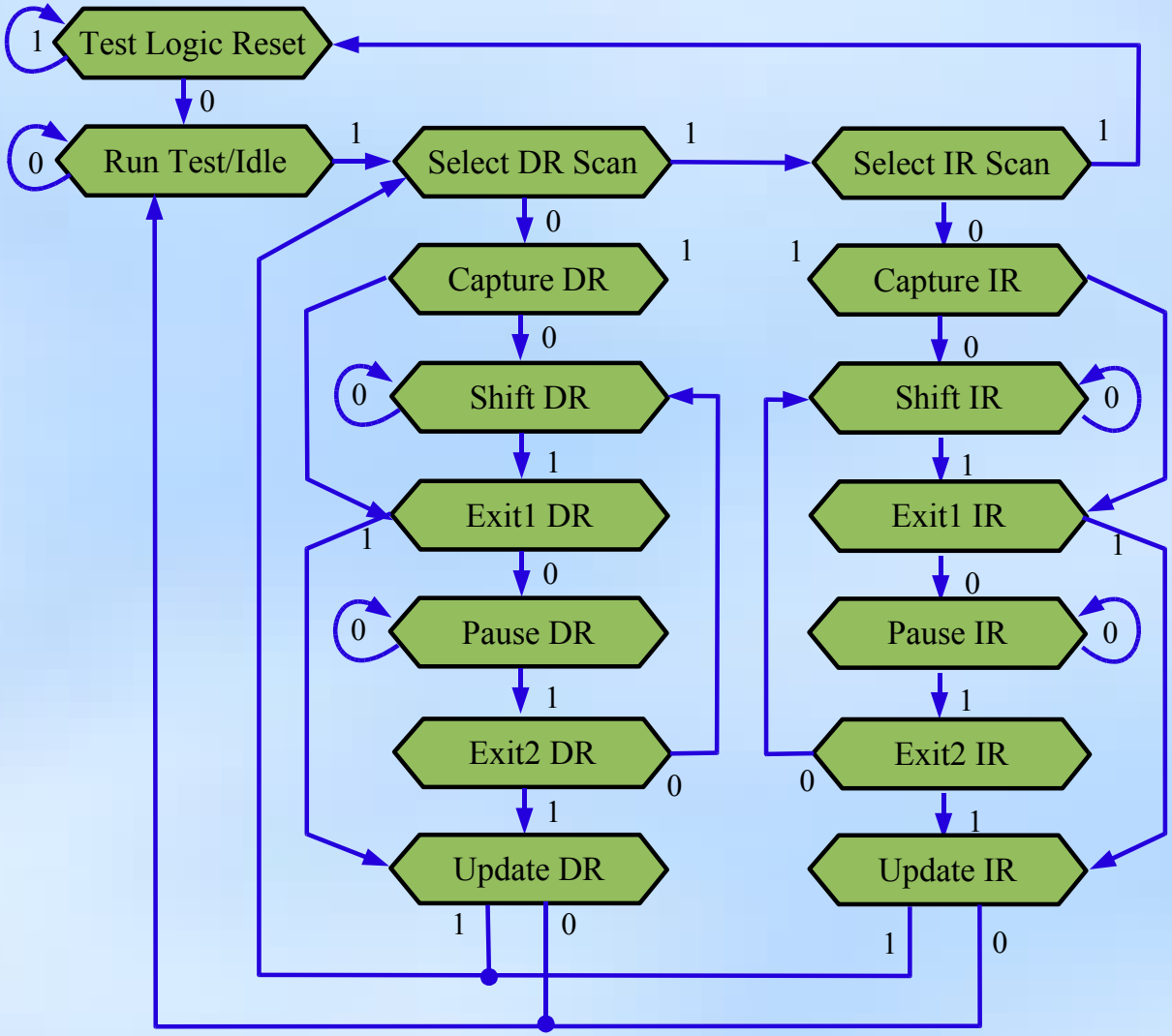


Architektura układu wyposażonego w JTAG



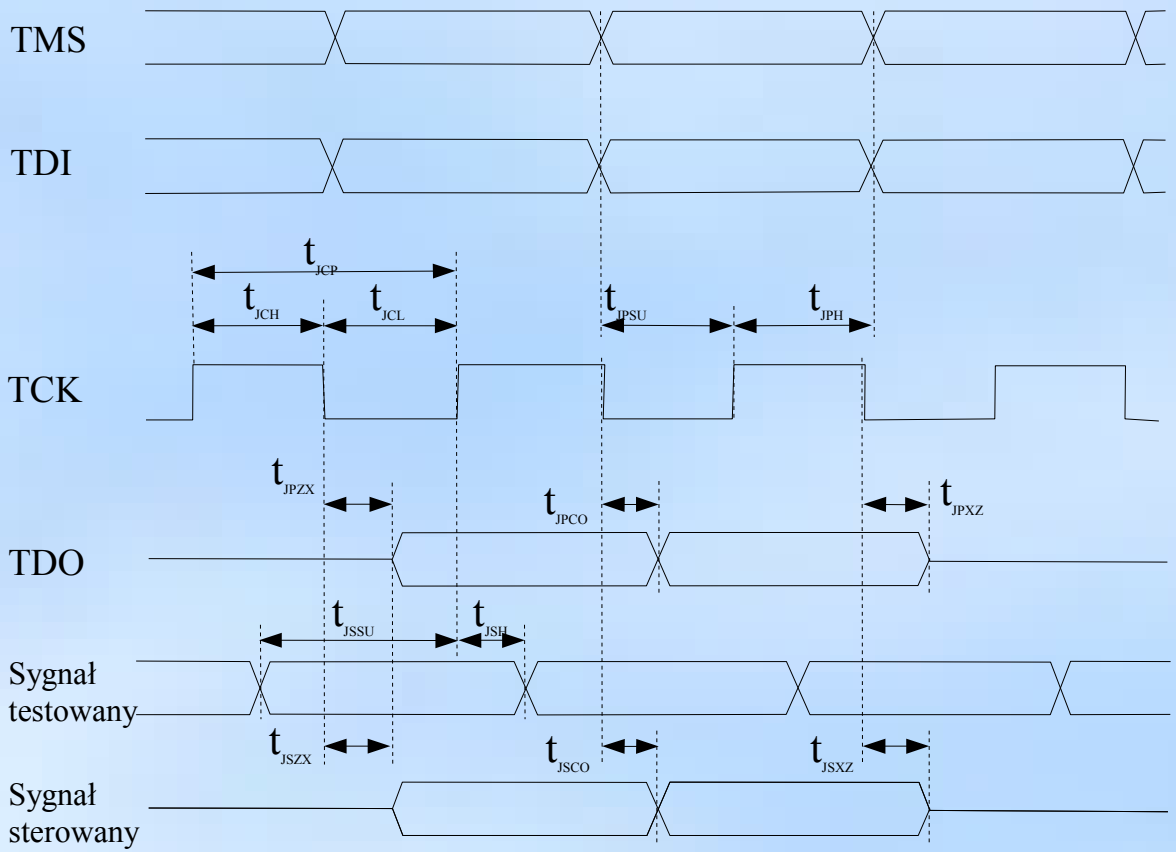
TDI – test data input
TMS – test mode select
TCK – test clock input
TDO – test data output
TAP – test acces port

Stany wewnętrzne układu logicznego TAP

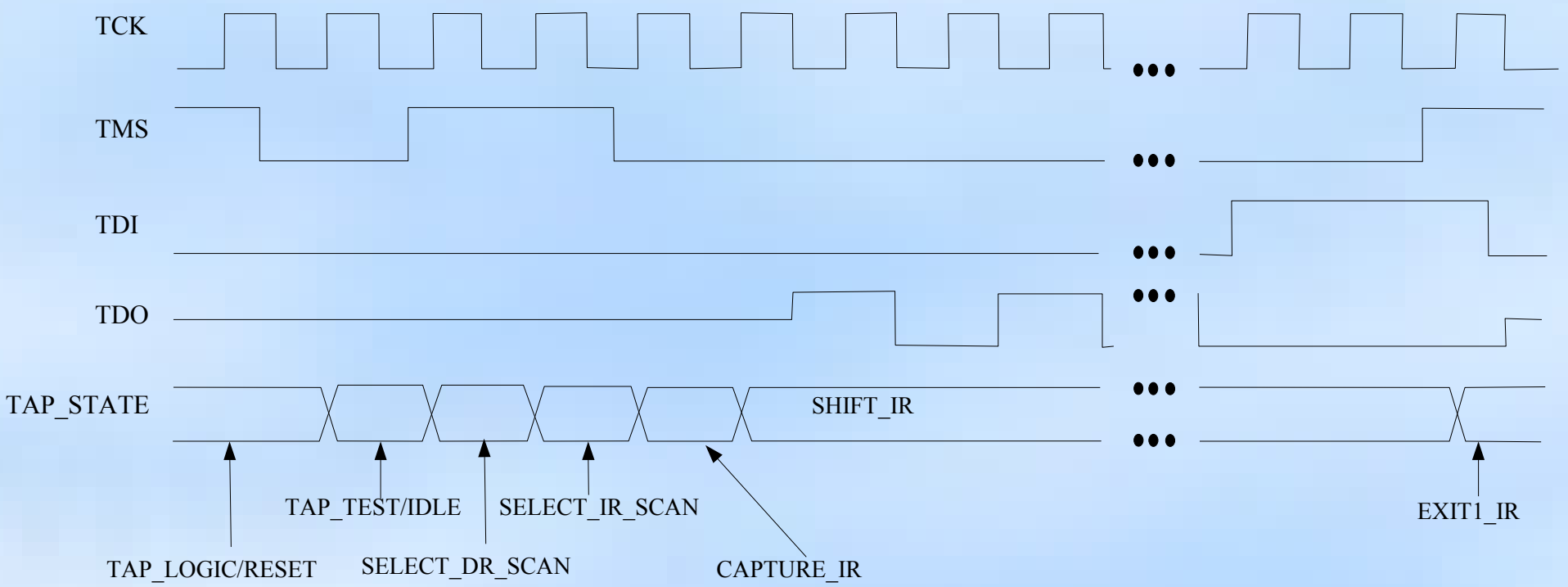


DR – data register
IR – instruction register

Zależności czasowe

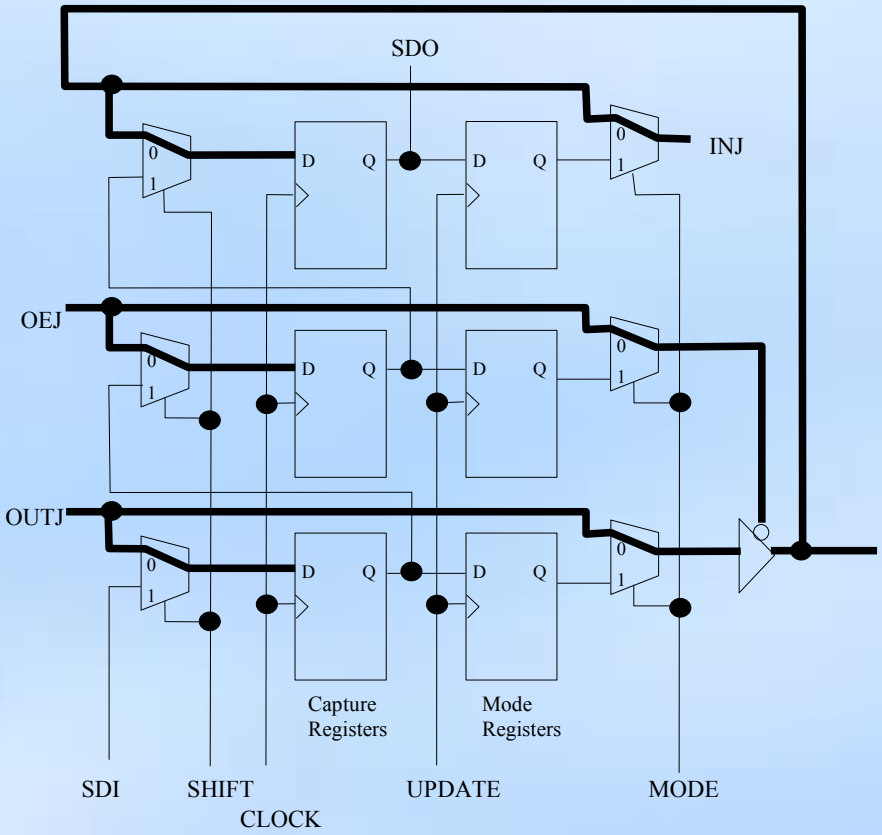


Zależności czasowe wybór IR



TAP – test acces port

Schemat wyjścia układu dla fazy Capture rozkazu SAMPLE/PRELOAD



Wymagane rozkazy złącza JTAG

- BYPASS – układ zachowuje swoje funkcje. Wejście TDI jest połączone z wyjściem TDO przez wewnętrzny jednobitowy rejestr. Dane z wejścia są przekazywane na wyjście bez zmian,
- SAMPLE/PRELOAD – układ zachowuje swoje funkcje. Rejestr BR (Boundary-Scan Register) jest podłączony między TDI i TDO. BR jest dostępny przez operację skanowania rejestru,
- EXTEST – rejestr BR jest podłączony między TDI i TDO. BR jest dostępny przez operację skanowania rejestru.

- INTEST – rejestr BR jest podłączony między TDI i TDO. Wewnętrzna logika układu może być wysterowywana i odczytywana,
- RUNBIST – układ jest wprowadzany w tryb wewnętrznego autotestu. Między TDI i TDO jest umieszczany rejestr zdefiniowany przez producenta układu. Stany wewnętrzne układu nie kolidują z układami sąsiednimi,
- CLAMP – ustawia wyjścia układu stosownie do zawartości rejestru BR i podłącza Bypass Register między TDI i TDO. Przed wykonaniem tej instrukcji należy ustawić właściwą zawartość rejestru BR instrukcją SAMPLE/PRELOAD,

- HIGHZ – wszystkie wyjścia i wejścia układu ustawia w stan wysokiej impedancji i podłącza Bypass Register między TDI i TDO,
- IDCODE – układ zachowuje swoją funkcjonalność. Między TDI i TDO umieszczany jest 32 bitowy rejestr zawierający informację o producencie, typie układu, wersji,
- USERCODE – układ zachowuje swoją funkcjonalność. Między TDI i TDO podłączany jest rejestr danych zdefiniowany przez producenta układu.

Minimalna ilość testów:

$NT = \log NET / \log 2$

Jeśli weźmiemy pod uwagę zwarcia do 1 i do 0 ilość testów wynosi:

$IT = NT + 2$

W przypadku konieczności testowania sieci logicznych ilość testów rośnie i zaczyna zależeć od stopnia skomplikowania testowanego układu.

Przykład testu sieci

Wektor	N1	N2	N3	N4	N5	N6	N7	N8
1	0	0	0	0	1	1	1	1
2	0	0	1	1	0	0	1	1
3	0	1	0	1	0	1	0	1

Wektory dla poszczególnych sieci:

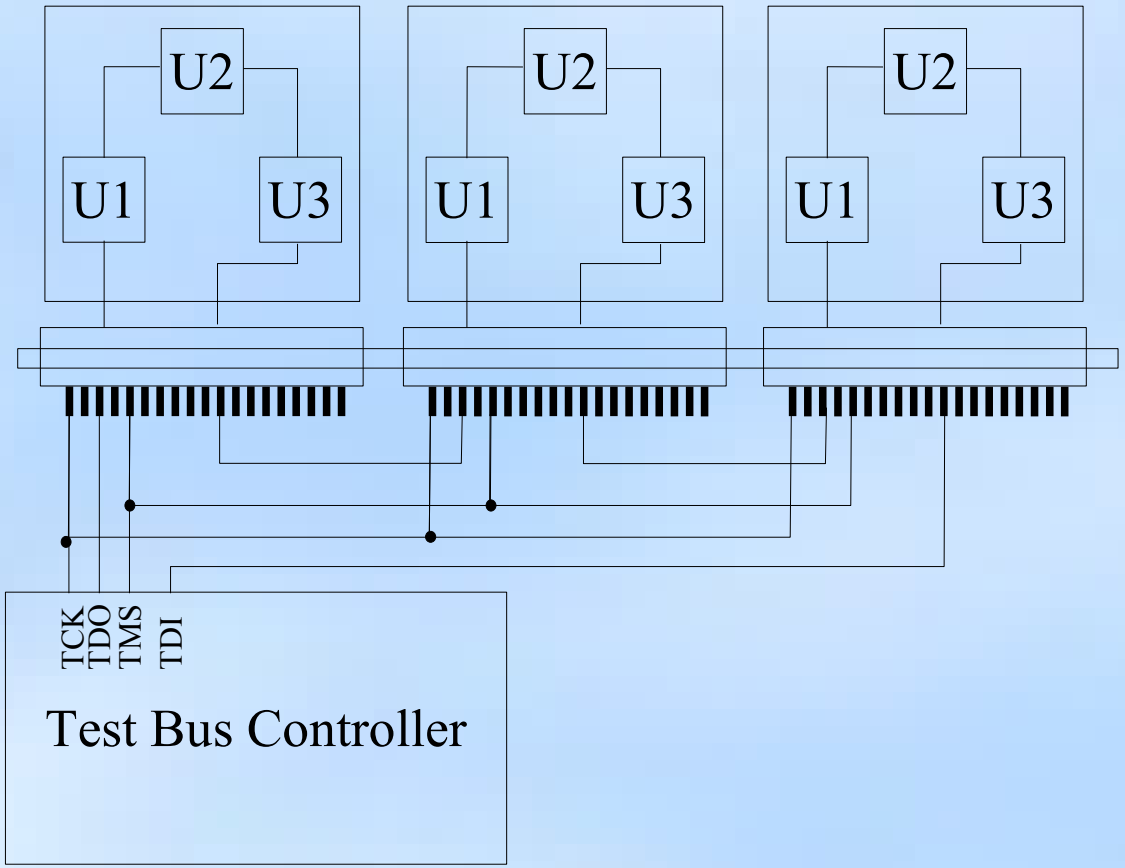
N1 000 N3 010 N5 100 N7 110
N2 001 N4 011 N6 101 N8 111

Dodatkowe funkcje JTAG

- funkcje debuggera dla procesorów,
- testowanie układów wewnętrznych,
- analizatory stanów logicznych,
- programowanie pamięci wewnętrznej urządzeń,
- testowanie układów analogowych.

Architektura połączeń

Układ połączeń w obrębie urządzenia, typ połączenia: koło



Urządzenia z testerem wbudowanym

Testery wbudowane umożliwiają przeprowadzenie testów przed każdym uruchomieniem urządzenia.

Zalety:

- nie zostanie uruchomione urządzenie niesprawne,
- pomoc w serwisowaniu.

Wady:

- koszty związane z dodatkowym sprzętem,
- długi czas uruchamiania urządzenia

Testowanie oprogramowania zapisanego w pamięciach stałych.

- Oprogramowanie testujące może wygenerować na pinach układu procesora cykle podobne do cykli dostępu do pamięci,
- Tak uzyskane dane wyczytane z pamięci stałych mogą być weryfikowane w zewnętrznym kontrolerze,
- Zaletą takiego rozwiązania jest niezależność testu od oprogramowania umieszczonego w badanym systemie.