

*Zastosowania procesorów sygnałowych*

***INTERFEJSY  
PROCESORÓW  
SYGNAŁOWYCH***

Opracowanie: Grzegorz Szwoch

Politechnika Gdańska, Katedra Systemów Multimedialnych

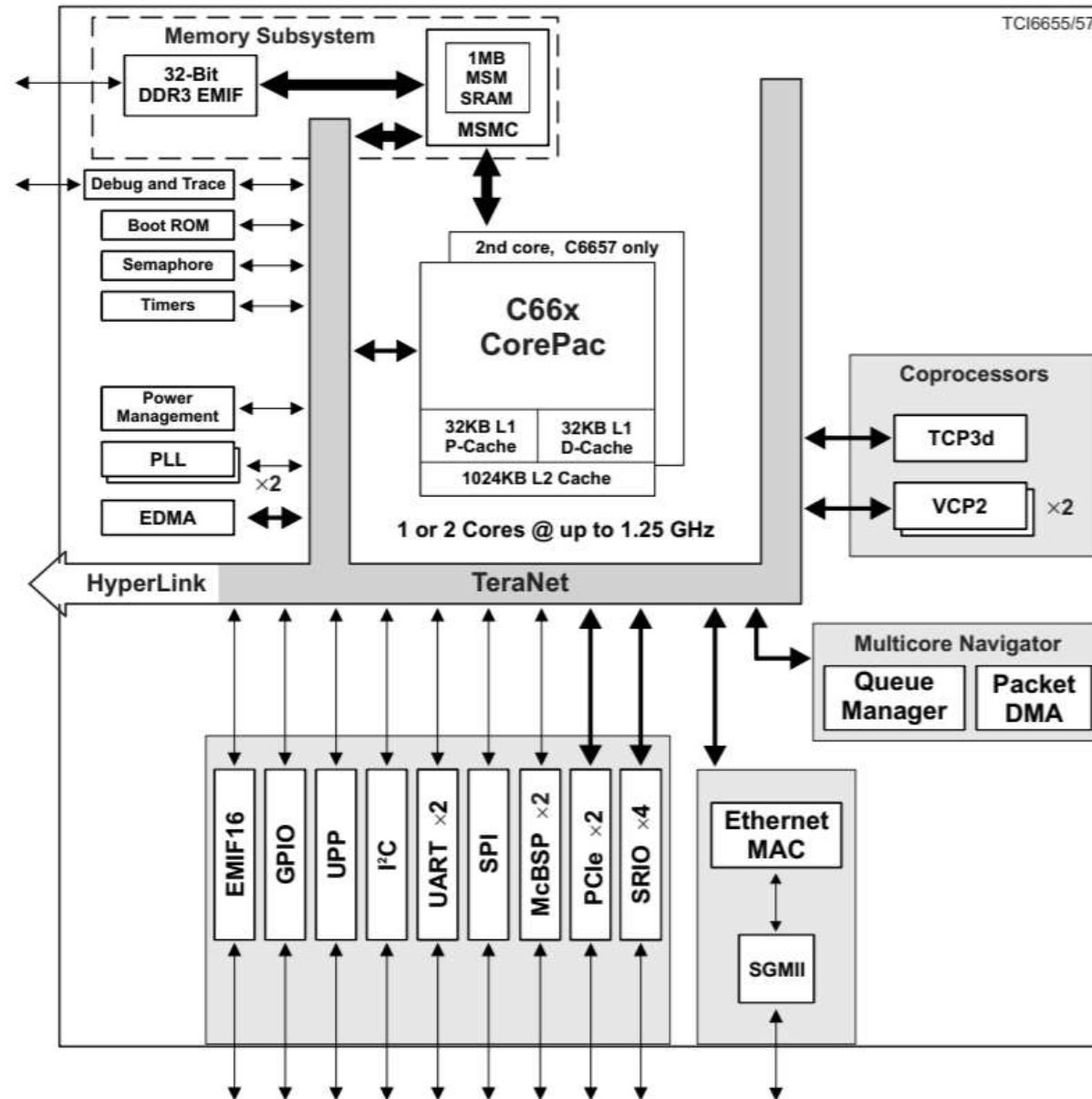
# Interfejsy procesorów sygnałowych

- Procesor sygnałowy przetwarza próbki sygnału. Musimy mieć możliwość odczytu danych z wejścia (np. czujnika) i wysłania wyników na wyjście (np. na wyświetlacz).
- **Interfejsy** (*interface*) wejściowe i wyjściowe umożliwiają pobieranie i przekazywanie sygnałów w różnych formatach, cyfrowych i analogowych.
- Interfejsy są wyprowadzone na stykach (pinach) procesora sygnałowego.
- Liczba i typy interfejsów zależą od modelu procesora.



	Yes	SP14	SP16	TRST	SDA_CLK SPL_CLK	SDA_DI SDA_AK SP12	SDA_DO SP12	SP11	SDA_AK SPL_AK	SDI_DI SDI_AK SP12	SDI_DO SPL_DO	UART_CLK SPL_CLK	UART_AK SPL_AK	Yes
100	SP120	Yes	SP12P	SP12	Yes	SP12	CPass	CPass	SP12_PDI SPL_CLK	CPass	UART_AK SPL_AK	SDI_DO SPL_DO	CPass	
101	SP121	CPass	CPass	SP123	CPass	Yes		SP12_CLK SPL_CLK SP12	CPass	SDI_DO SPL_DO SPL_AK	SDI_DI SPL_DI	SDI_DO SPL_DO	SDI_CLK SPL_CLK	
102	SP122	SP12A	CPass	Yes				Yes	SDI_DO SPL_DO SPL_AK	SDI_DI SPL_DI	SDI_DO SPL_DO	UART_AK	UART_AK	
103	SP123	Yes	Yes							CPass	SDI_DO SPL_DO	UART_AK	UART_AK	
104	SP124	SP12C	SP12									UART_AK	Yes	UART_CLK
105	SP125	CPass	Yes									UART_AK	UART_AK	UART_CLK
106	SP126	CPass										UART_AK	UART_AK	UART_CLK
107	SP127	CPass	Yes									UART_AK	UART_AK	UART_CLK
108	SP128	CPass	Yes	Yes					Yes	CPass	UART_AK	UART_AK	UART_CLK	
109	SP129	CPass	Yes	Yes								UART_AK	UART_AK	UART_CLK
110	SP130	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
111	SP131	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
112	SP132	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
113	SP133	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
114	SP134	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
115	SP135	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
116	SP136	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
117	SP137	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
118	SP138	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
119	SP139	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
120	SP140	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
121	SP141	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
122	SP142	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
123	SP143	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK
124	SP144	CPass	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	UART_AK	UART_AK	UART_CLK

# Interfejsy procesora TI C6657



# Cyfrowe interfejsy szeregowo

- Interfejsy szeregowo umożliwiają podłączanie cyfrowych urządzeń, np. czujników.
- Bity sygnału cyfrowego są przekazywane **szeregowo**, bit po bicie (jeden bit naraz).
- Większość cyfrowych interfejsów procesora sygnałowego jest szeregowych.
- Interfejsy szeregowo mogą być:
  - **synchroniczne**
    - osobne linie danych (DATA) i sygnału zegarowego (CLOCK),
    - impulsy sygnału zegarowego wyznaczają początek każdego bitu;
  - **asynchroniczne**
    - brak sygnału zegarowego, tylko linia danych,
    - odbiorca musi znać prędkość transmisji,
    - odbiorca musi sam wygenerować sygnał zegarowy do synchronizacji.

# GPIO

## GPIO – *General Purpose Input and Output*

- Interfejsy ogólnego przeznaczenia, bez zdefiniowanego formatu przesyłania danych.
- Są to po prostu „wyprowadzone piny”, na które można podawać dowolne sygnały napięcia, reprezentujące sygnały analogowe lub asynchroniczne cyfrowe.
- GPIO dobrze sprawdzają się dla sygnałów dwustanowych „*on-off*”, np.:
  - odczyt stanu przycisku (wciśnięty lub zwolniony),
  - sterowanie diodą LED (zapalenie lub zgaszenie).
- Można za pomocą GPIO przesyłać dowolne sygnały cyfrowe, ale ich generowanie i odczytywanie (tzw. *bit-banging*) spada na programistę i jest mniej wydajne.
- Jeżeli mamy do dyspozycji inne interfejsy cyfrowe, lepiej z nich korzystać.

# GPIO

- Każdy pin na złączu GPIO może być skonfigurowany jako wejście (*input*) lub wyjście (*output*).
- Domyślny stan linii GPIO jest niezdefiniowany, „wiszący” (*floating*). Może samoistnie się zmieniać, a tego nie chcemy.
- Stan domyślny linii jest konfigurowany przez włączenie wewnętrznego rezystora:
  - *pull-up* – „podciągnięcie” linii do domyślnego stanu 1,
  - *pull-down* – „ściągnięcie” linii do domyślnego stanu 0.

# UART (RS-232, RS-485)

## UART – *Universal Asynchronous Receiver/Transmitter*

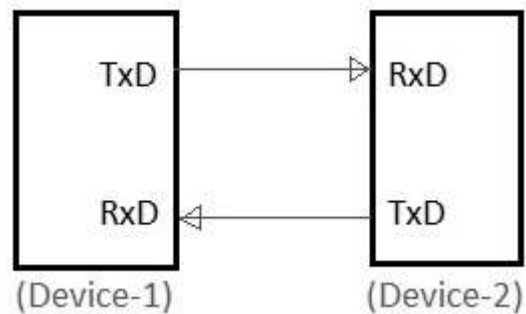
- Cyfrowy interfejs szeregowy, asynchroniczny (brak sygnału zegara).
- Wykorzystywany w standardach transmisji RS-232 i RS-485.
- Stosunkowo niskie prędkości (w praktyce maksymalnie 115200 b/s).
- Transmisja: bit startu, 5-9 bitów danych (zwykle 8), opcjonalny bit parzystości, 1-2 bity stopu.
- Odbiorca musi znać format danych. Przykład specyfikacji (standardowy tryb): **9600/8N1** – prędkość 9600 bit/s, 8 bitów danych, brak bitu parzystości, 1 bit stopu.
- Stosowany głównie do komunikacji ze starszymi urządzeniami. Rzadko stosowany we współczesnych urządzeniach.
- Przesyłane dane są „surowe”, nie są opakowane w żadne struktury.
- Wymiana danych wymaga często stosowania specjalnych protokołów, np. MODBUS.

# UART (RS232, RS485)

Linie przesyłowe:

- Tx (*transmit*) – wysyłanie danych,
- Rx (*receive*) – odbiór danych,
- linia zasilania i masy (GND).

UWAGA: złącza urządzeń łączymy „na krzyż”: Rx1 – Tx2, Rx2 – Tx1





# I<sup>2</sup>C

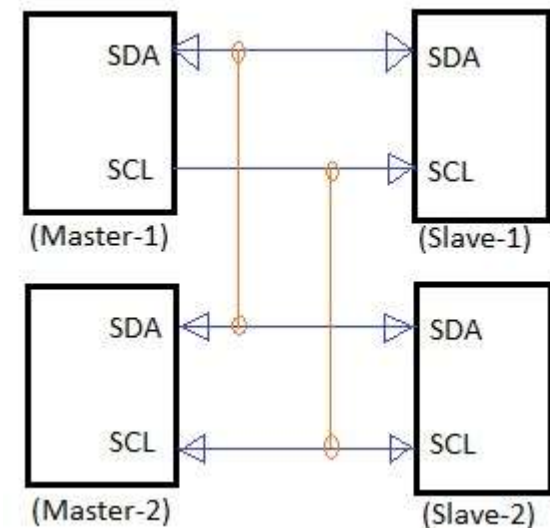
## I<sup>2</sup>C – *Inter-Integrated-Circuit*

- Najczęściej stosowany interfejs cyfrowych czujników.
- Cyfrowy interfejs szeregowy, synchroniczny. Linia danych (SDA) i zegara (SCL).
- Procesor pełni rolę zarządcy komunikacji (*Master*).
- Możliwość podłączenia wielu urządzeń zewnętrznych (*Slave*).
- Każde urządzenie podrzędne musi mieć unikalny adres.
- Typowe maksymalne prędkości: 100 kbit/s (*Standard*), 400 kbit/s (*Fast*), rozszerzenia do 3,4 Mbit/s.
- Prędkość transmisji zależy też od długości i jakości połączeń oraz od liczby podłączonych urządzeń.
- Wykorzystujemy ten interfejs do czujników, które mają umiarkowaną częstotliwość odczytu i objętość danych, np. cyfrowe czujniki temperatury powietrza.

# I<sup>2</sup>C

Uproszczony przebieg transmisji.

- Master przełącza stan linii SDA i SCL (Start), wysyła adres docelowy do wszystkich podłączonych urządzeń Slave.
- Slave o podanym adresie odpowiada bitem ACK, przełącza stan linii SDA.
- Następuje przesłanie danych (Master czyta lub wysyła) przez SDA.
- Odbiorca potwierdza bitem ACK.
- Master kończy transmisję (Stop) przełączając stan SCL i SDA.



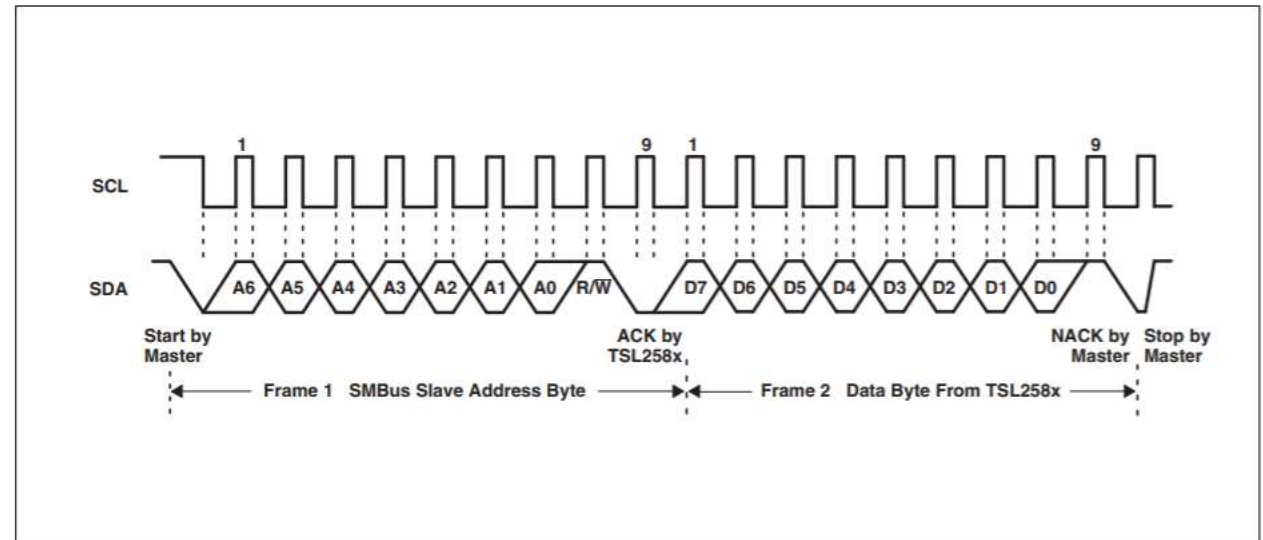
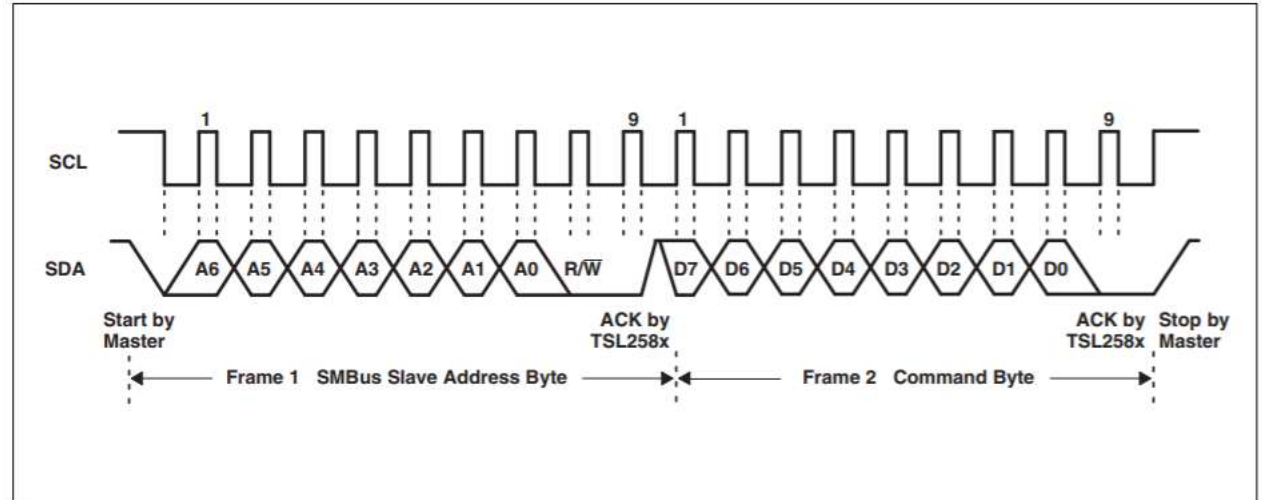
## I<sup>2</sup>C - przykład transmisji

Master wysyła komendę odczytu danych – adres urządzenia oraz kod komendy (np. „podaj temperaturę”).

ACK – potwierdzenie odbioru.

Master odczytuje żądane dane z urządzenia o podanym adresie.

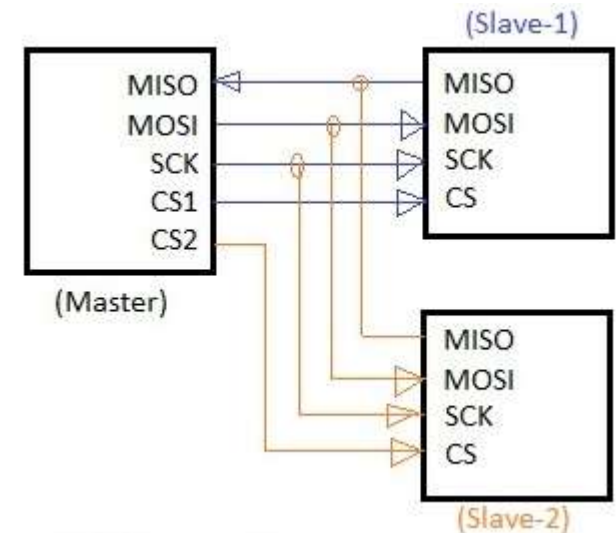
NACK – potwierdzenie odebrania danych przez Mastera.



# SPI

## SPI – *Serial Peripheral Interface*

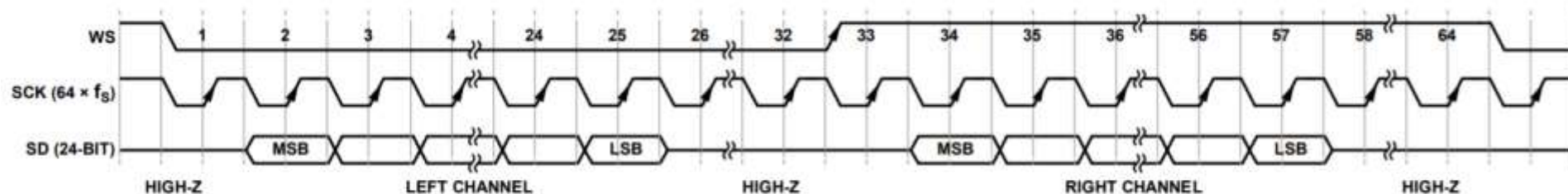
- Cyfrowy interfejs szeregowy, synchroniczny.
- Do jednego Mastera może być podłączonych kilka urządzeń, ale każde wymaga osobnej linii *Chip Select* (CS).
- 4 linie: MOSI (*Master Out, Slave In*), MISO (*Master In, Slave Out*), CLK (sygnał zegarowy), CS (wybór urządzenia).
- Znacznie większe prędkości niż w I<sup>2</sup>C, rzędu 8 Mbit/s.
- W pełni dwustronna komunikacja (*full duplex*).
- Stosowany w przypadkach, gdy konieczna jest duża prędkość transmisji i duża przepływność bitowa, np. interfejsy kart pamięci, wyświetlacze OLED.



# I<sup>2</sup>S

I<sup>2</sup>S – *Inter-IC Sound* (nie mylić z I<sup>2</sup>C)

- Cyfrowy, synchroniczny interfejs szeregowy do przesyłania próbek cyfrowego dźwięku (przetworniki A/C, C/A, cyfrowe mikrofony, np. MEMS).
- Linie:
  - SCK – sygnał zegarowy, impuls dla każdego bitu,
  - WS lub FS – sygnał zegarowy, zmiana stanu dla początku nowej próbki dźwięku (0/1 dla kanału L/R),
  - SD – linie danych, bity cyfrowego dźwięku
  - MC (*master clock*) – opcjonalny zegar synchronizujący.



# USB

## USB – *Universal Serial Bus*

- Cyfrowy interfejs szeregowy, asynchroniczny.
- Umożliwia komunikowanie się z urządzeniami USB:
  - procesor jako *USB Slave* – odczytuje dane z zewnętrznego urządzenia,
  - procesor jako *USB Master* – udostępnianie wyników przetwarzania innym urządzeniom.
- Rzadko stosowane w procesorach sygnałowych do przesyłania sygnałów, częściej jako „tryb diagnostyczny”, np. zewnętrzny komputer pobiera wyniki analizy.
- USB jest też używane do debugowania programów na procesorach sygnałowych (tryb *debug probe*).
- Prędkość zależna od trybu (1.0: 1,5 Mb/s; 1.1: 12 Mb/s, 2.0: 480 Mb/s).

# Interfejsy sieciowe

## *Ethernet, EMAC (Ethernet Media Access Controller)*

- Umożliwia funkcjonowanie procesora sygnałowego jako urządzenia sieciowego.
- Wymaga implementacji protokołów sieciowych za pomocą oprogramowania (np. TI NDK – *Network Development Kit*).
- Prędkość transmisji zależna od trybu. GbE (*Gigabit Ethernet*) – do 1 Gb/s.
- Stosowany głównie w nowszych procesorach zmiennoprzecinkowych.
- Interfejs sieciowy *Ethernet* może być stosowany do przesyłania dowolnych sygnałów cyfrowych pomiędzy dwoma elementami systemu, np. między zewnętrznym przetwornikiem A/C a procesorem sygnałowym. Wygodny interfejs o dużej przepustowości.

## Interfejsy równoległe

- W przeciwieństwie do interfejsów szeregowych, w interfejsach równoległych (*parallel*) wszystkie bity słowa pojawiają się równocześnie na wejściu/wyjściu.
- Wymaga osobnej linii danych dla każdego bitu. Np. przesyłanie słów 16-bitowych wymaga 16 osobnych linii danych.
- Dość rzadko spotykane w procesorach sygnałowych.
- Większa prędkość niż dla interfejsów szeregowych.
- Przydatne w komunikacji z urządzeniami wymagającymi dużej przepływności danych, np. przetworników analogowo-cyfrowych.
- Przykład: UPP (*Universal Parallel Port*) w TI C6657, od 8 do 16 bitów.



## Interfejsy analogowe

- Wprowadzenie sygnałów analogowych do procesora sygnałowego wymaga zastosowania przetwornika analogowo-cyfrowego (ADC – *Analog to Digital Converter*).
- Przetworniki ADC mogą być:
  - wbudowane w procesor (np. SAR – *Successive Approximation Register*),
  - zewnętrzne – komunikujące się przez SPI, UPP, *Ethernet*, itp.
- Parametry:
  - prędkość (próbek/s, sa/s) – definiuje częstotliwość próbkowania,
  - rozdzielczość bitowa (np. 8, 12, 16 bit).
- Np. przetwornik SAR w C5535: 10 bit, do 64 ksa/s (64 kHz).
- Interfejs analogowy pozwala podłączyć dowolne urządzenie analogowe, np. analogowy czujnik temperatury lub analogowy akcelerometr.

## Interfejsy pamięci zewnętrznej

Komunikacja procesora sygnałowego z zewnętrznymi modułami pamięci:

- interfejs DDR3 – komunikacja z pamięcią RAM w tym standardzie,
- interfejs eMMC/SD – zapis/odczyt na kartach pamięci *flash*,
- EMIF (*External Memory Interface*) – zunifikowany interfejs do komunikacji z pamięcią zewnętrzną różnego typu (flash, SRAM, SDRAM).

## Inne interfejsy

Przykłady mniej standardowych interfejsów wykorzystywanych w procesorach sygnałowych.

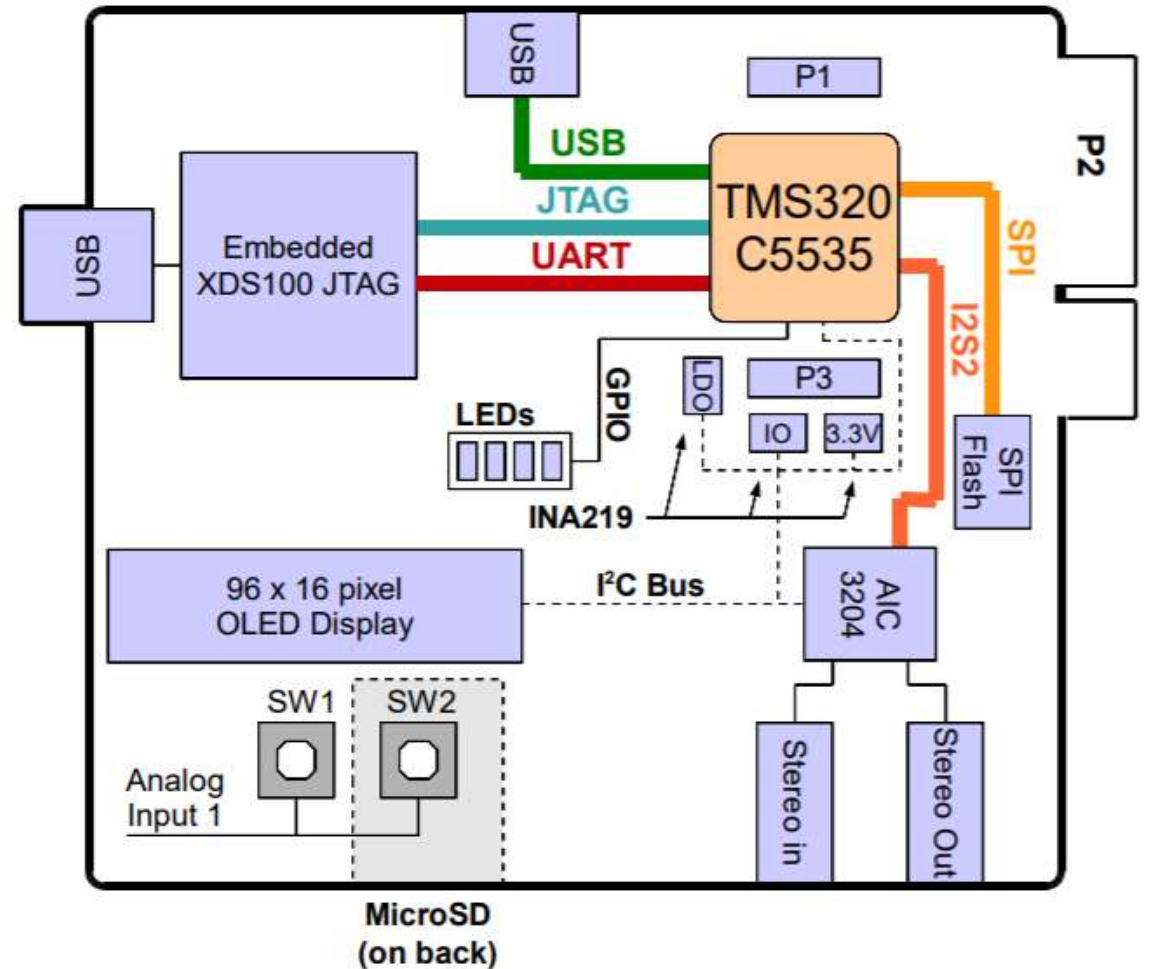
- *McBSP (Multichannel Buffered Serial Port)* – interfejs Texas Instruments, wielokanałowy interfejs szeregowy, głównie do przesyłania dźwięku cyfrowego.
- *PCI Express (PCIe)* – do komunikacji z urządzeniami pracującymi w tym standardzie, duża prędkość transmisji.
- *HyperLink* – interfejs pozwalający na łączenie procesorów sygnałowych firmy Texas Instruments w układ wieloprocessorowy i wymianę danych.
- *SRIO (Serial RapidIO)* – interfejs szeregowy wysokich prędkości.

# Przykład wykorzystania interfejsów

Procesor C5535 i płytki uruchomieniowa *eZdsp5535* (z laboratorium ZPS)



TMDX5535EZDSP USB Stick Development Kit



## Moduł uruchomieniowy eZdsp5535

Interfejsy i zewnętrzne komponenty dostępne w układzie *eZdsp5535*:

- pamięć *flash* 8 MB (interfejs SPI),
- wyświetlacz monochromatyczny OLED 96×16 pikseli (I<sup>2</sup>C),
- 5 diod LED (GPIO),
- 2 przyciski sterujące SW (GPIO, wejścia analogowe przez przetwornik SAR),
- przetworniki A/C i C/A – kodek dźwięku AIC 3204 (I<sup>2</sup>S, 4 linie),
- interfejs USB 2.0 (wymiana danych z komputerem),
- interfejs do debugowania programów (JTAG, UART),
- próbniki do pomiaru napięcia (I<sup>2</sup>C),
- wyprowadzenie interfejsów na zewnętrzne złącza na płytce (P1, P2, P3).