



Politechnika Poznańska
Wydział Elektroniki i Telekomunikacji

PRACA MAGISTERSKA

Moduł sterownika kart pamięci SD/MMC dla układów FPGA

Autor:

Marcin Brach

Promotor:

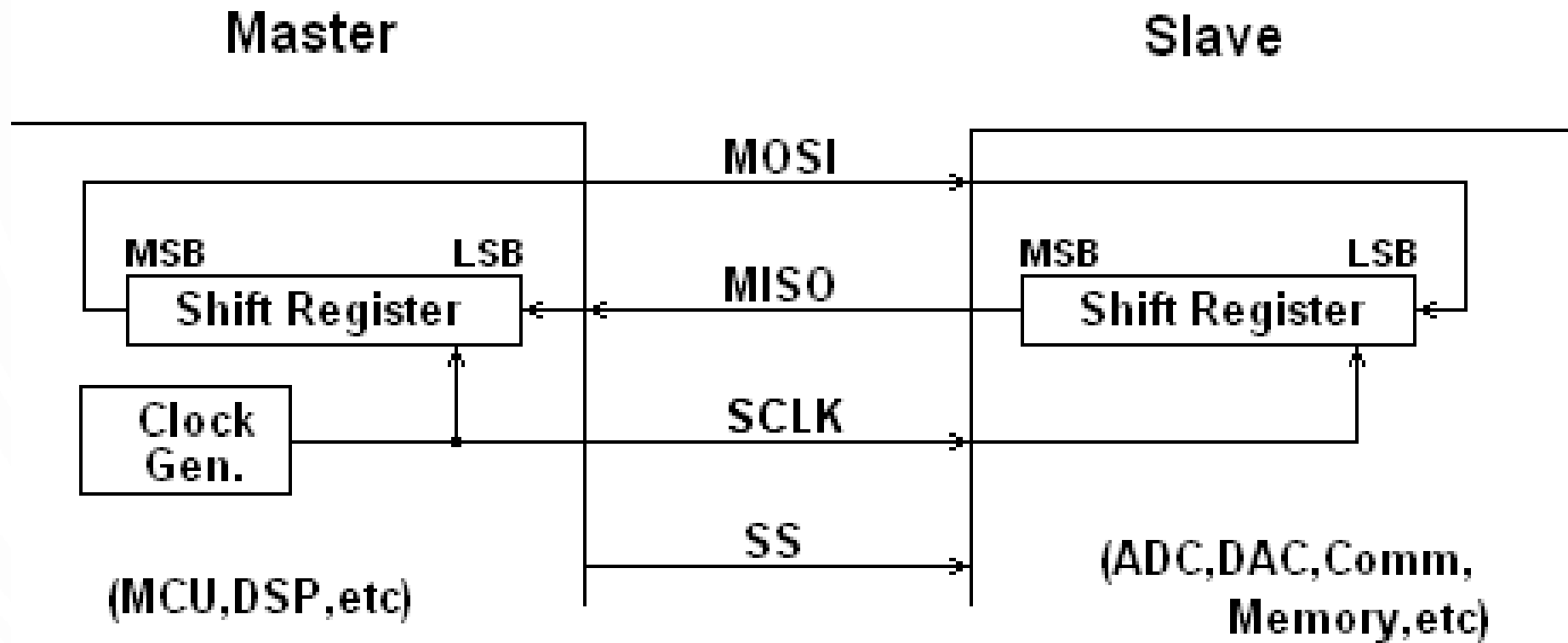
dr inż. Olgierd Stankiewicz

CEL PRACY

Zaprojektowanie oraz implementacja modułu sterownika obsługi kart pamięci w języku Verilog

- zapoznanie się z protokołami komunikacyjnymi oraz standardami kart pamięci
- implementacja wybranego interfejsu oraz modułu sterownika dla układu FPGA
- symulacja projektowanych modułów
- opracowanie sposobu testowania z użyciem płytki laboratoryjnej
- wykonanie testów funkcjonalnych oraz pomiaru prędkości dla wybranych kart pamięci

PROTOKÓŁ SPI



Sposób łączenia układów poprzez interfejs SPI.

(Źródło: http://elm-chan.org/docs/spi_e.html)

TRYBY PRACY INTERFEJSU SPI

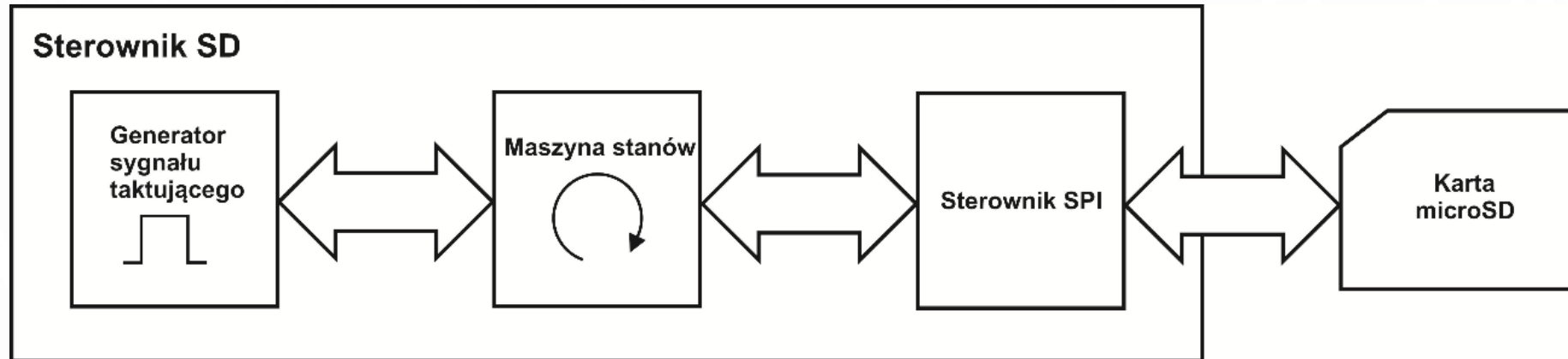
Tryby pracy SPI	Przebiegi czasowe
Tryb 0 Zbocze narastające Zatrzask i przesunięcie (CPHA=0, CPOL=0)	
Tryb 1 Zbocze narastające Przesunięcie i zatrzask (CPHA=1, CPOL=0)	
Tryb 2 Zbocze opadające Zatrzask i przesunięcie (CPHA=0, CPOL=1)	
Tryb 3 Zbocze opadające Przesunięcie i zatrzask (CPHA=1, CPOL=1)	

Cztery tryby pracy interfejsu SPI. (Źródło: http://elm-chan.org/docs/spi_e.html)

CPHA - Przesunięcie w fazie odbierania/wysyłania danych

CPOL - Polaryzacja sygnału zegarowego

PROJEKTOWANY STEROWNIK SD



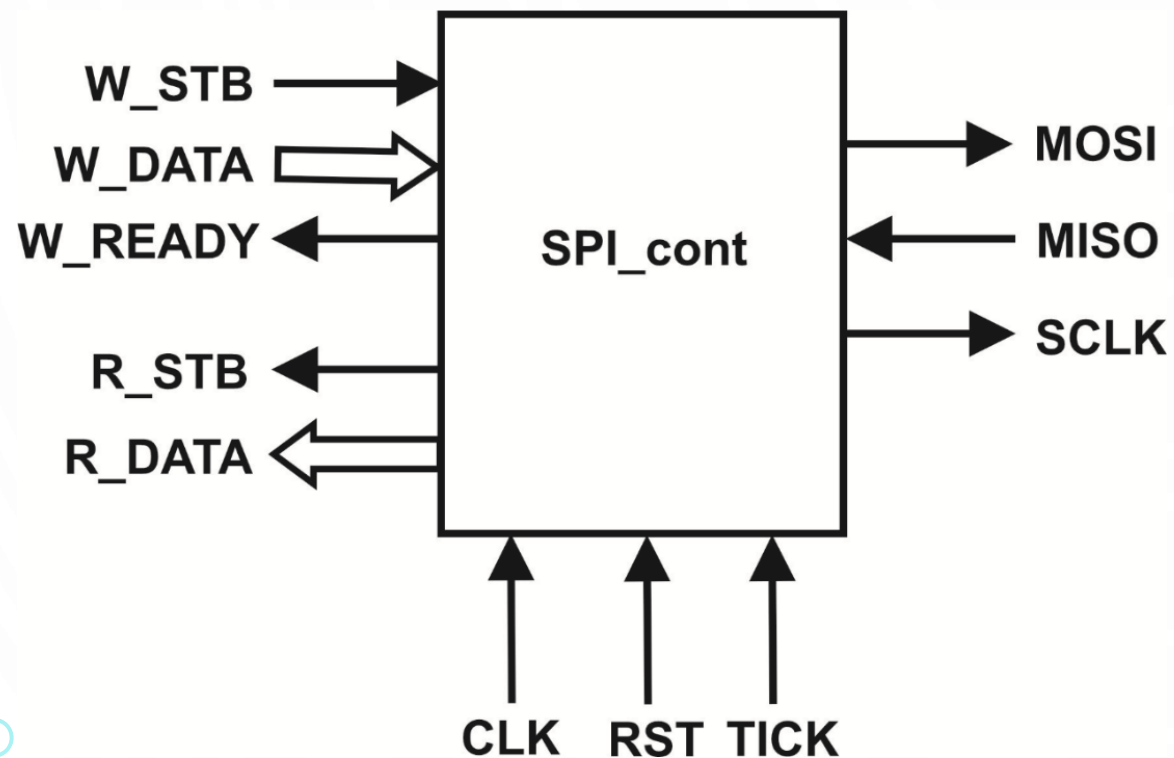
Ogólny schemat blokowy sterownika SD

Generator sygnału taktującego – generuje sygnał zegarowy na wyjście SCLK podczas inicjalizacji karty oraz podczas transmisji danych

Maszyna stanów – to automat który steruje sygnałami z interfejsu SPI oraz wysyła odpowiednie komendy do karty pamięci

Sterownik SPI – odpowiada za transmisję danych przez interfejs SPI

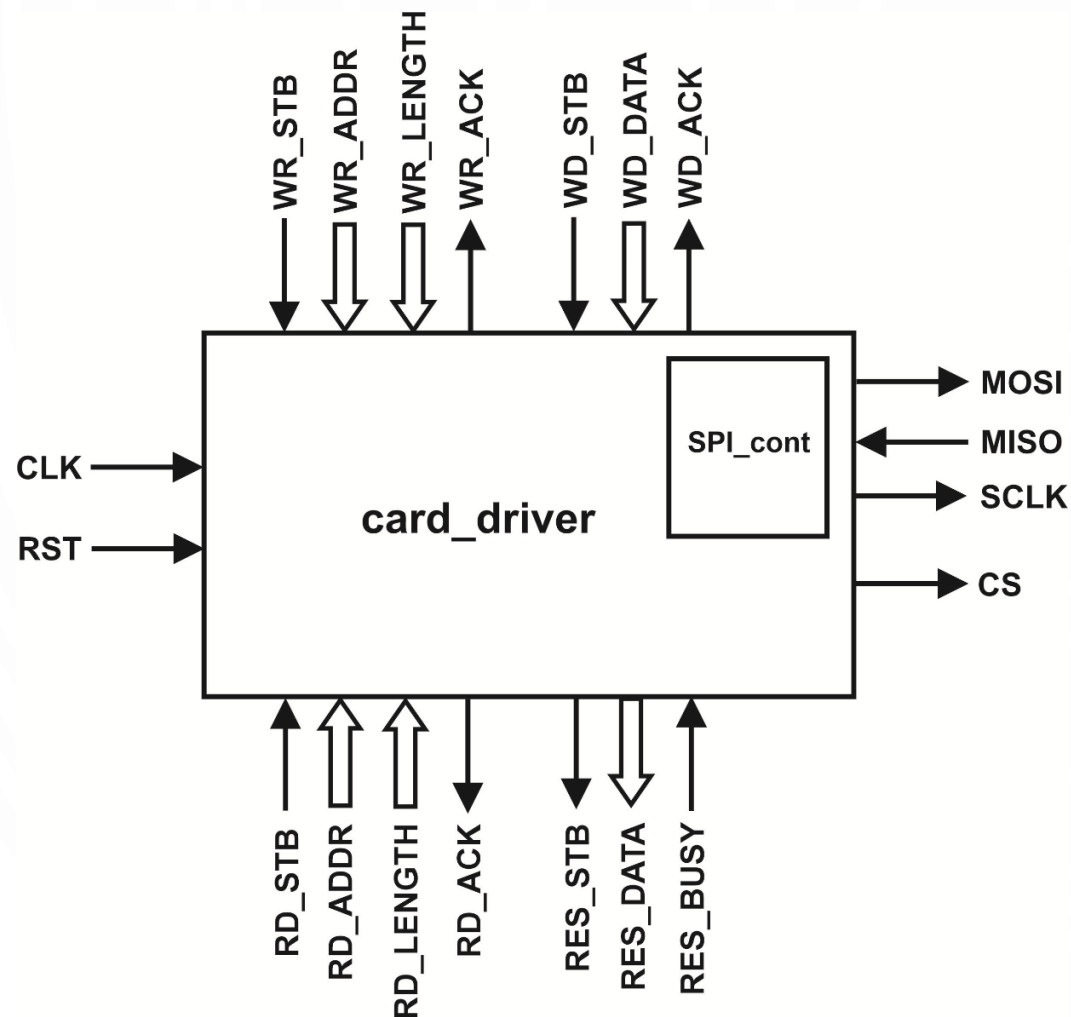
MODUŁ STEROWNIKA SPI



Schemat funkcjonalny modułu sterownika SPI

Nazwa sygnału:	Opis:
CLK	Wejście zegarowe
RST	Wejście resetujące układ (Reset)
TICK	Wejście wyzwalające działanie modułu
W_STB	Wejście strobulujące zapis danych
W_DATA	8-bitowa magistrala danych zapisywanych
W_READY	Wyjście potwierdzające przesłanie danych
R_STB	Wyjście strobulujące odczyt
R_DATA	8-bitowa magistrala danych odczytywanych
MOSI	Wyjście danych 1-bitowych interfejsu SPI
MISO	Wejście danych 1-bitowych interfejsu SPI
SCLK	Wyjście zegarowe interfejsu SPI

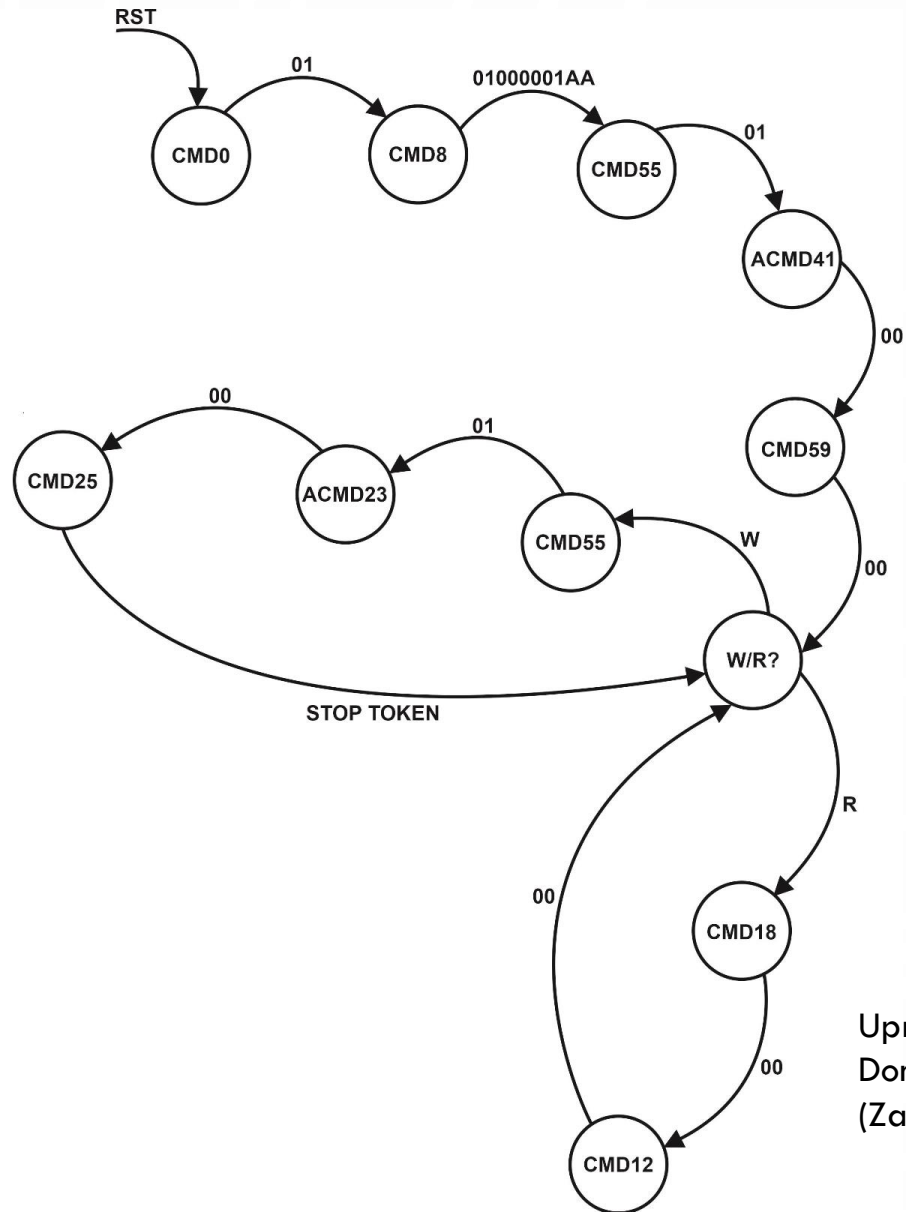
MODUŁ STEROWNIKA KARTY PAMIĘCI SD



Nazwa sygnału:	Opis:
CLK	Sygnal zegarowy
RST	Reset
WR_STB	Wejście strobujące adres do zapisu danych
WR_ADDR	32-bitowe wejście adresowe do zapisu danych
WR_LENGTH	23-bitowe wejście adresowe określające liczbę bloków do zapisu
WR_ACK	Wyjście potwierdzające zapis
WD_STB	Wejście strobujące zapis danych
WD_DATA	8-bitowe wejście danych do zapisu
WD_ACK	Wyjście potwierdzające zapis danych
RD_STB	Wejście strobujące adres do odczytu danych
RD_ADDR	32-bitowe wejście adresowe do odczytu danych
RD_LENGTH	23-bitowe wejście adresowe określające liczbę bloków do odczytu
RD_ACK	Wyjście potwierdzające odczyt
RES_STB	Wyjście strobujące odczyt danych
RES_DATA	8-bitowe wyjście dla odczytanych danych
RES_BUSY	Wejście informujące o zajętości magistrali
MOSI	Wyjście danych
MISO	Wejście danych
SCLK	Wyjście zegarowe
CS	Sygnal wyjściowy Chip Select

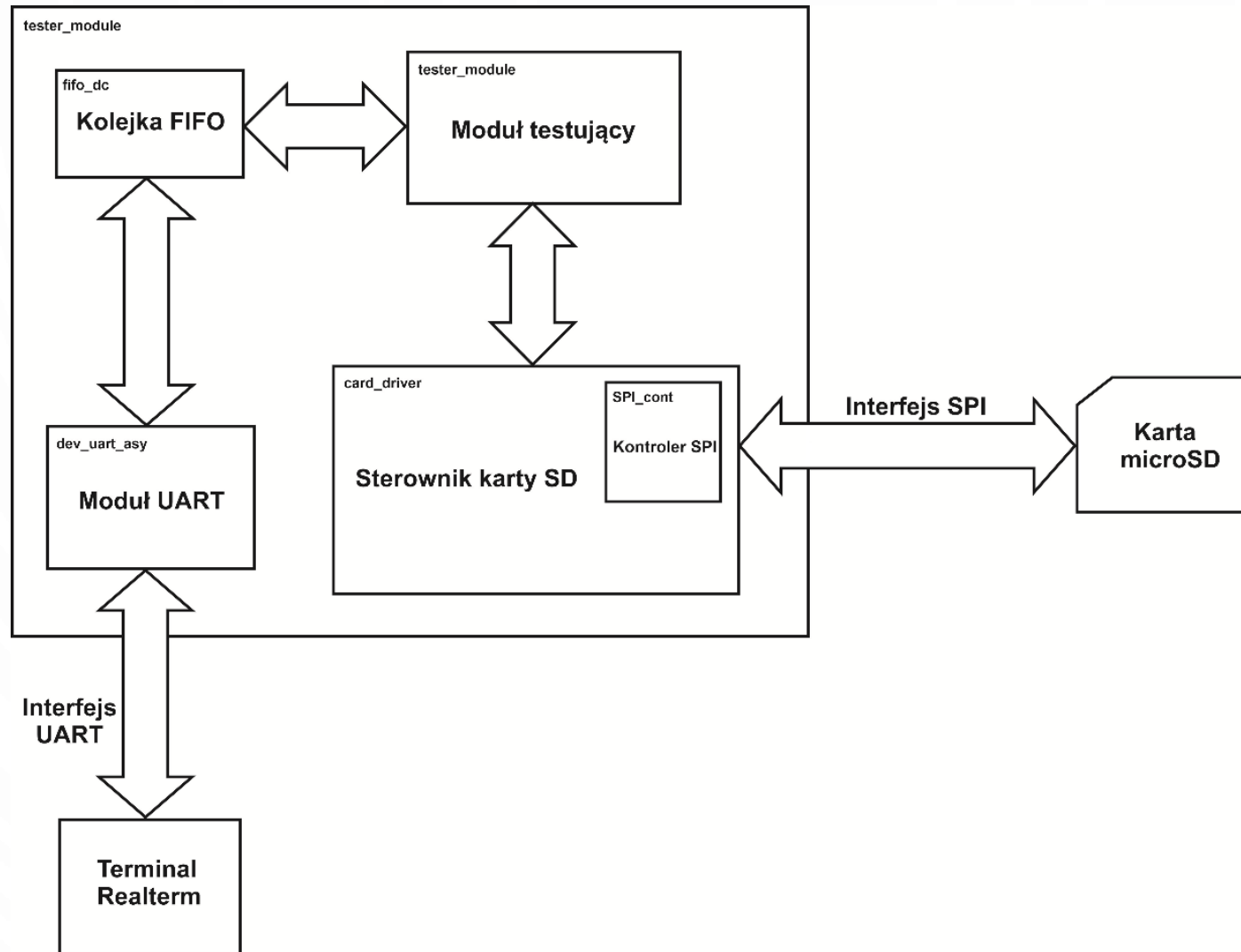
Schemat funkcjonalny projektowanego sterownika karty pamięci SD

GRAF AUTOMATU STEROWNIKA KART PAMIĘCI SD



Uproszczony (konceptyjny) graf automatu sterownika kart pamięci SD. Domyślnie automat pozostaje w aktualnym (dotychczasowym) stanie (Zaimplementowany automat posiada około 200 stanów)

MODUŁ TESTUJĄCY

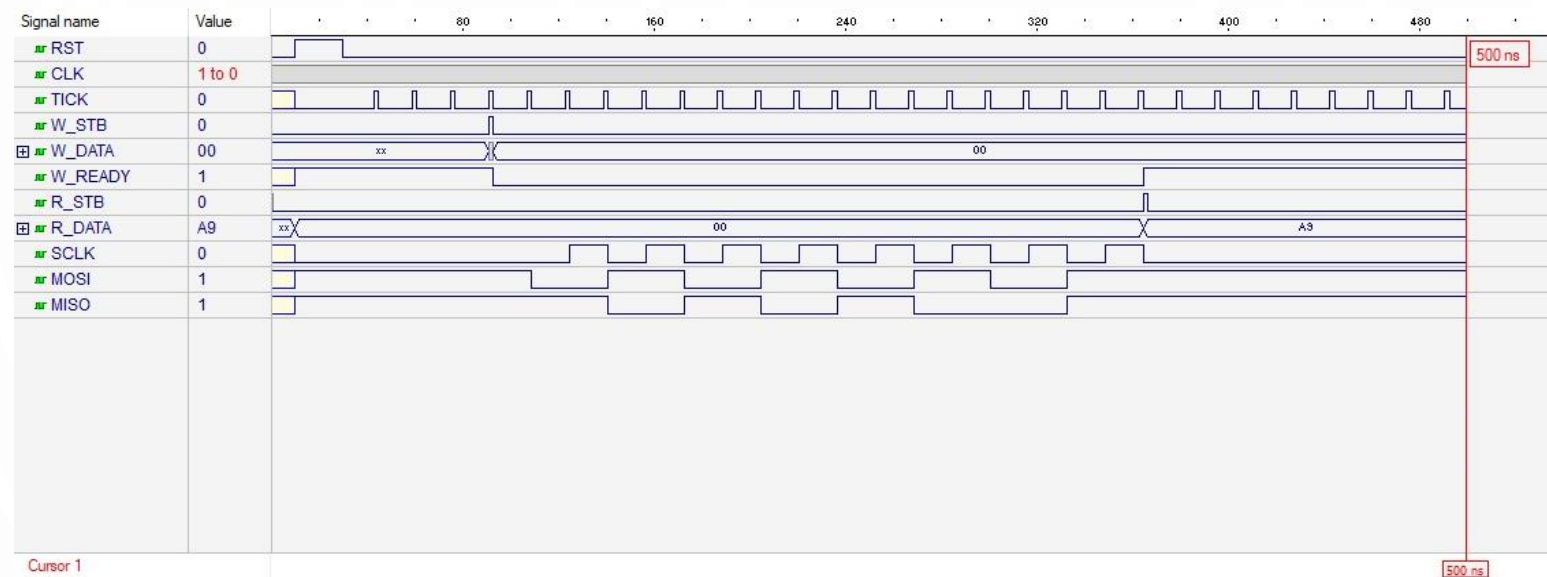


Schemat blokowy testowanego modułu sterownika

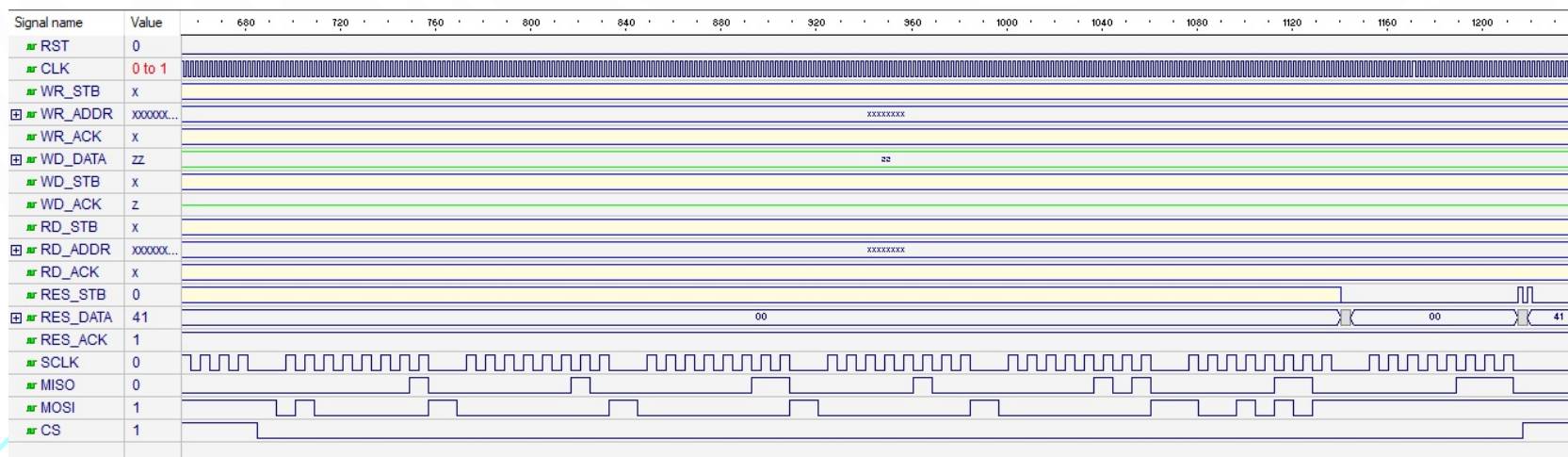
IMPLEMENTACJA

- wykonana na podstawie zaprojektowanych modułów
- przeprowadzenie symulacji przy pomocy oprogramowania Active HDL 10.5
- uruchomienie syntezy, zaprogramowanie płytki laboratoryjnej oraz przeprowadzenie testów z użyciem oprogramowania Lattice Diamond 3.10

WYNIKI SYMULACJI

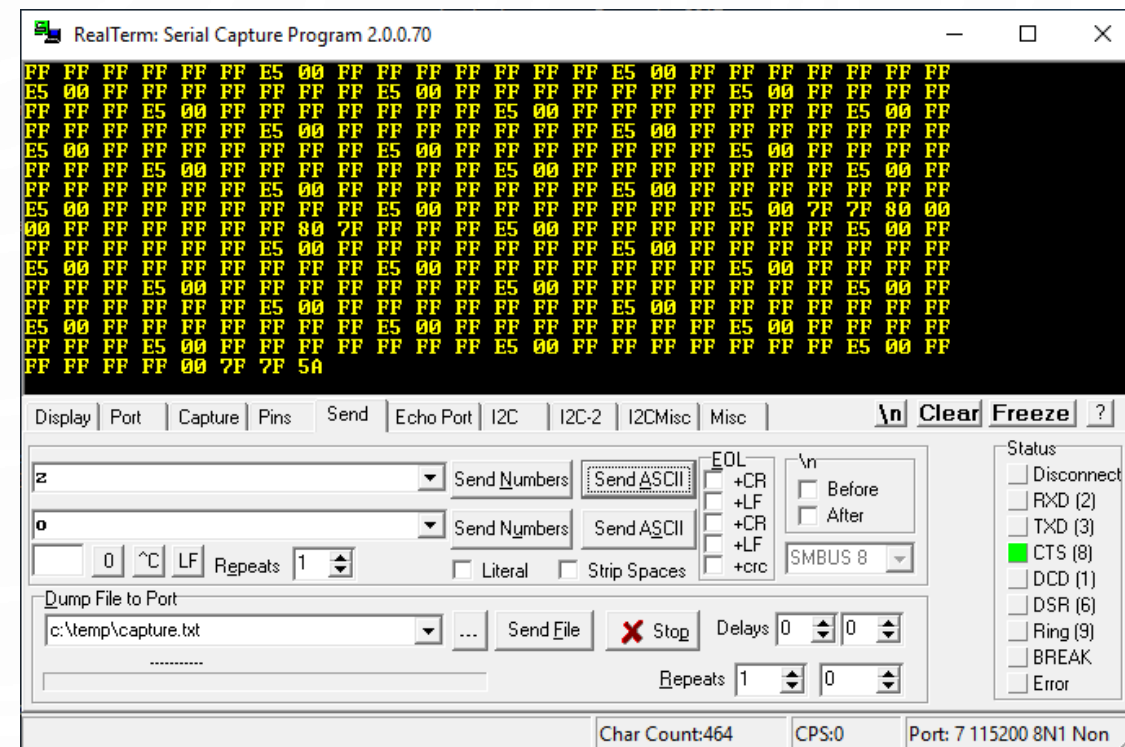
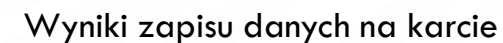


Wyniki symulacji sterownika SPI

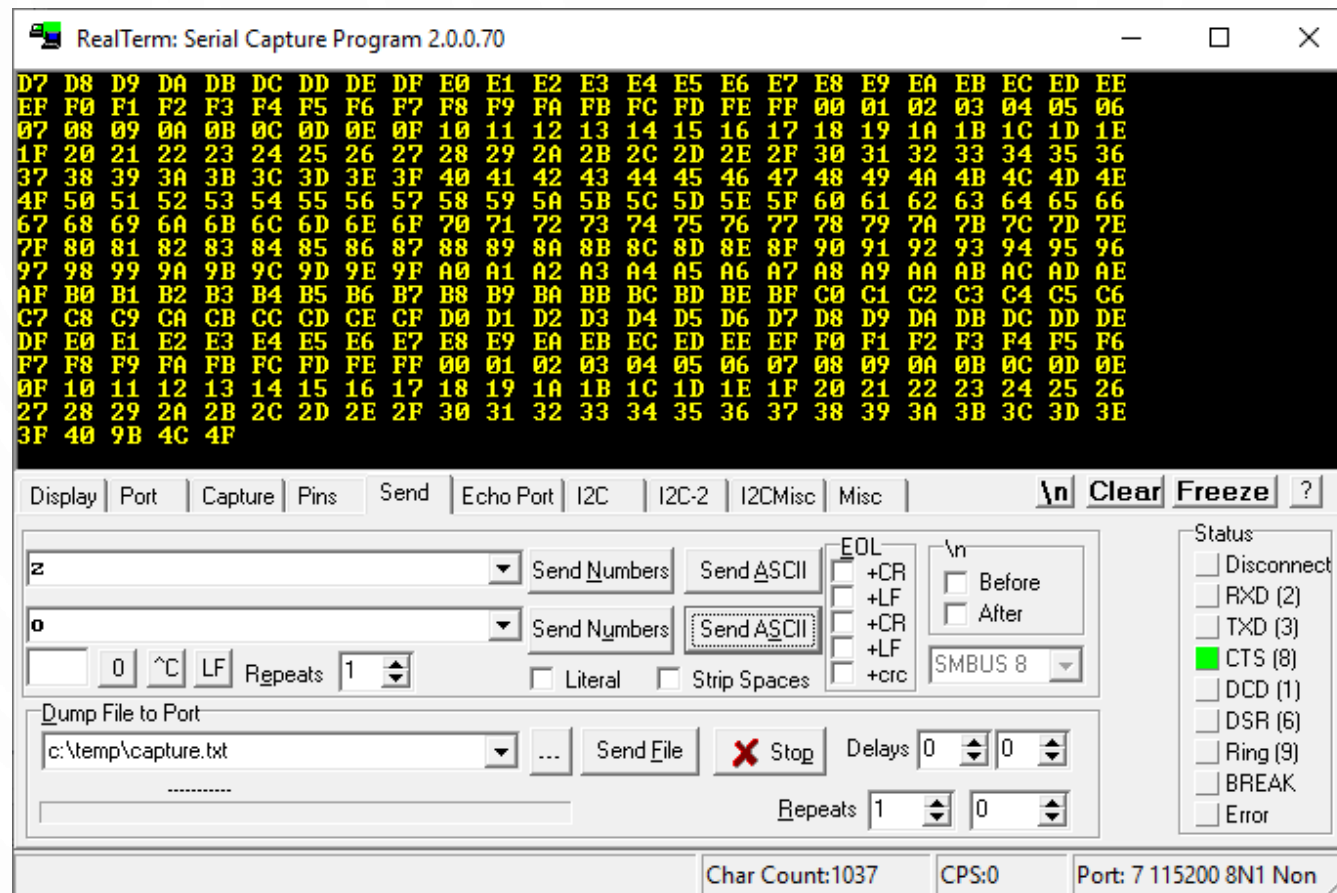


Wyniki symulacji przestania komendy CMD0 i odbioru przykładowych danych

Wyniki procesu inicjalizacji karty pamięci



TESTOWANIE W UKŁADZIE FPGA



Wyniki odczytu danych z karty

SPOSÓB POMIARU PRĘDKOŚCI

- pomiar wyemitowanych znaków sterujących „w” lub „r” wysłanych po ukończeniu każdej operacji wieloblokowej.
- czas pomiaru 10 sekund
- liczba bloków w trybie wieloblokowym: 1, 2, 5, 10, 100, 1000, 10000
- prędkość wyznaczana ze wzoru:

$$V = \frac{\text{liczba odczytanego znaku} * \text{multiburst} * \text{długość bloku (512)}}{10}$$

- wykorzystanie terminalu Realterm

SPOSÓB POMIARU PRĘDKOŚCI

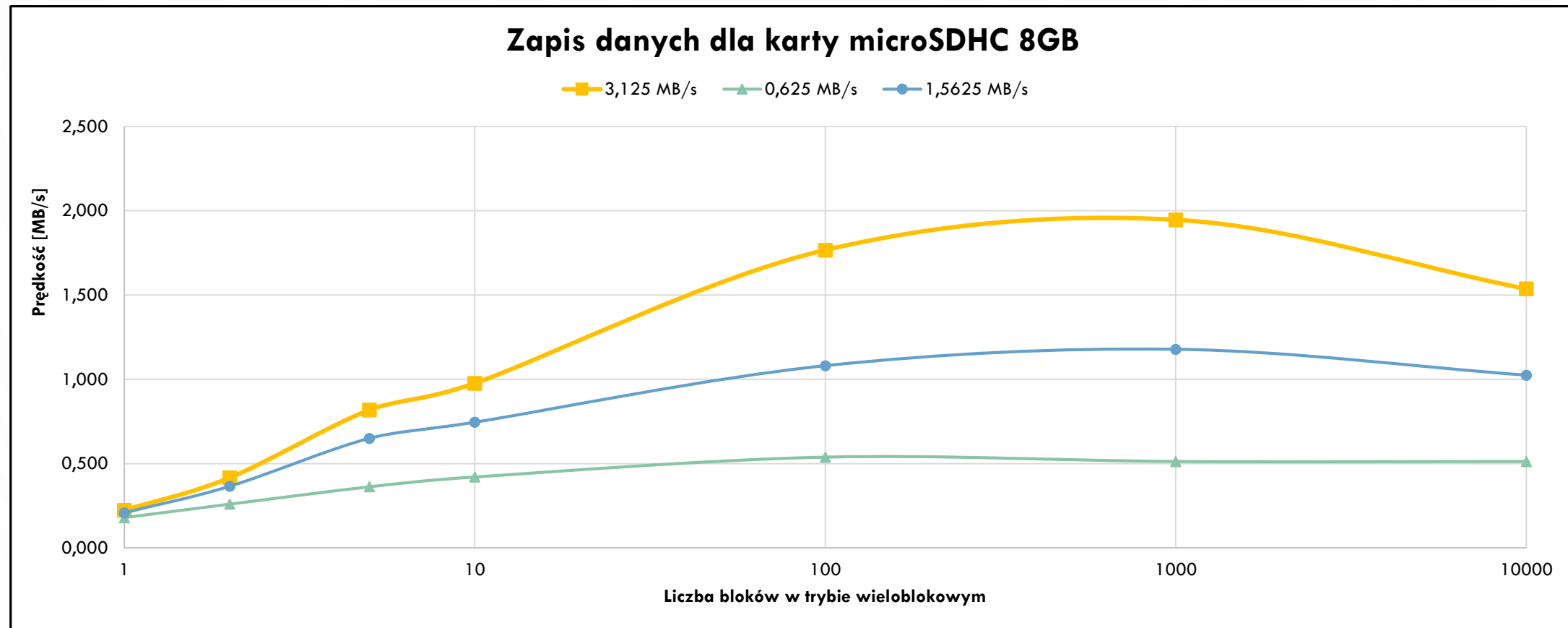
a) Pomiar wykonany dla karty pamięci SANDISK microSDHC 8GB dla prędkości:

$$V_{max} = 50\,000\,000 \text{ [Hz]} / 5[\text{dzielnik}] / 8 \text{ [takt/bajt]} / 2 \text{ [1/takt]} = 0,625 \text{ [MB/s]}$$

$$V_{max} = 50\,000\,000 \text{ [Hz]} / 2[\text{dzielnik}] / 8 \text{ [takt/bajt]} / 2 \text{ [1/takt]} = 1,562 \text{ [MB/s]}$$

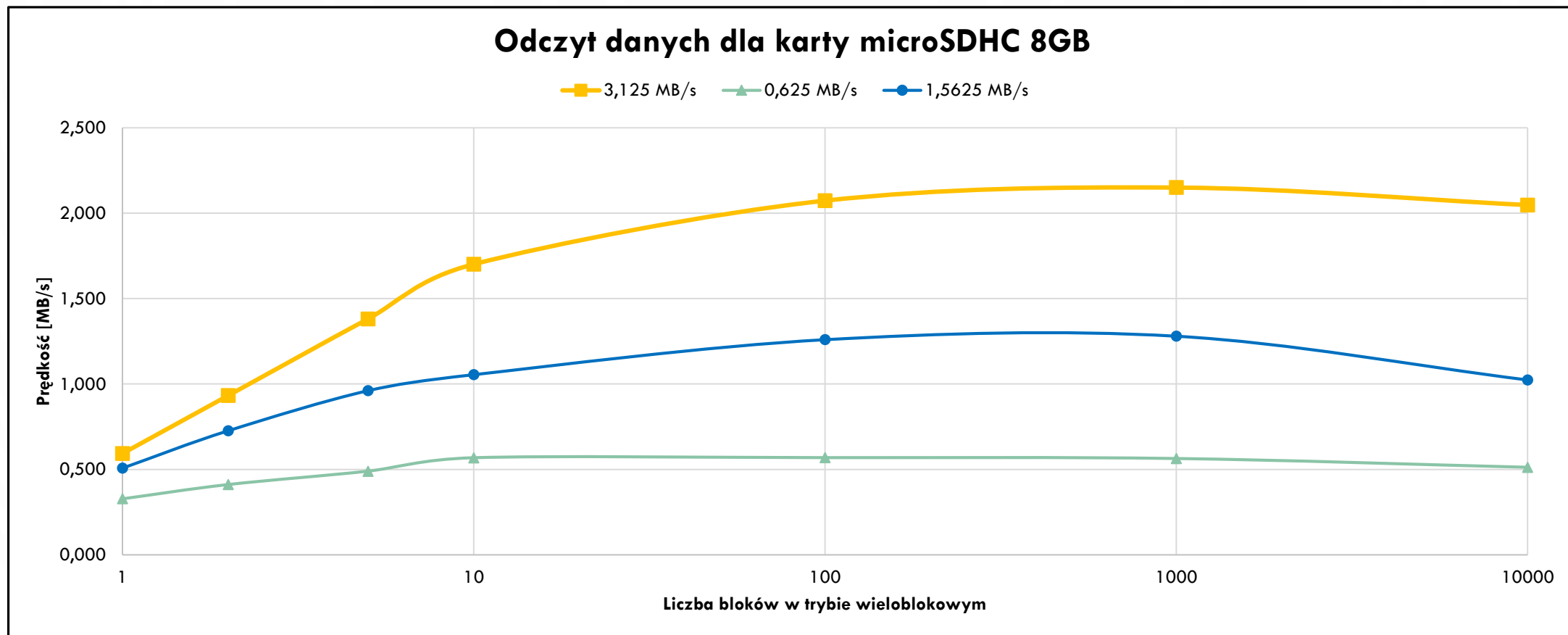
$$V_{max} = 50\,000\,000 \text{ [Hz]} / 1[\text{dzielnik}] / 8 \text{ [takt/bajt]} / 2 \text{ [1/takt]} = 3,125 \text{ [MB/s]}$$

WYNIKI POMIARÓW DLA KARTY PAMIĘCI MICROSDHC 8GB



Porównanie wyników dla zapisu danych na karcie pamięci microSDHC 8GB dla różnych Vmax

WYNIKI POMIARÓW DLA KARTY PAMIĘCI MICROSDHC 8GB



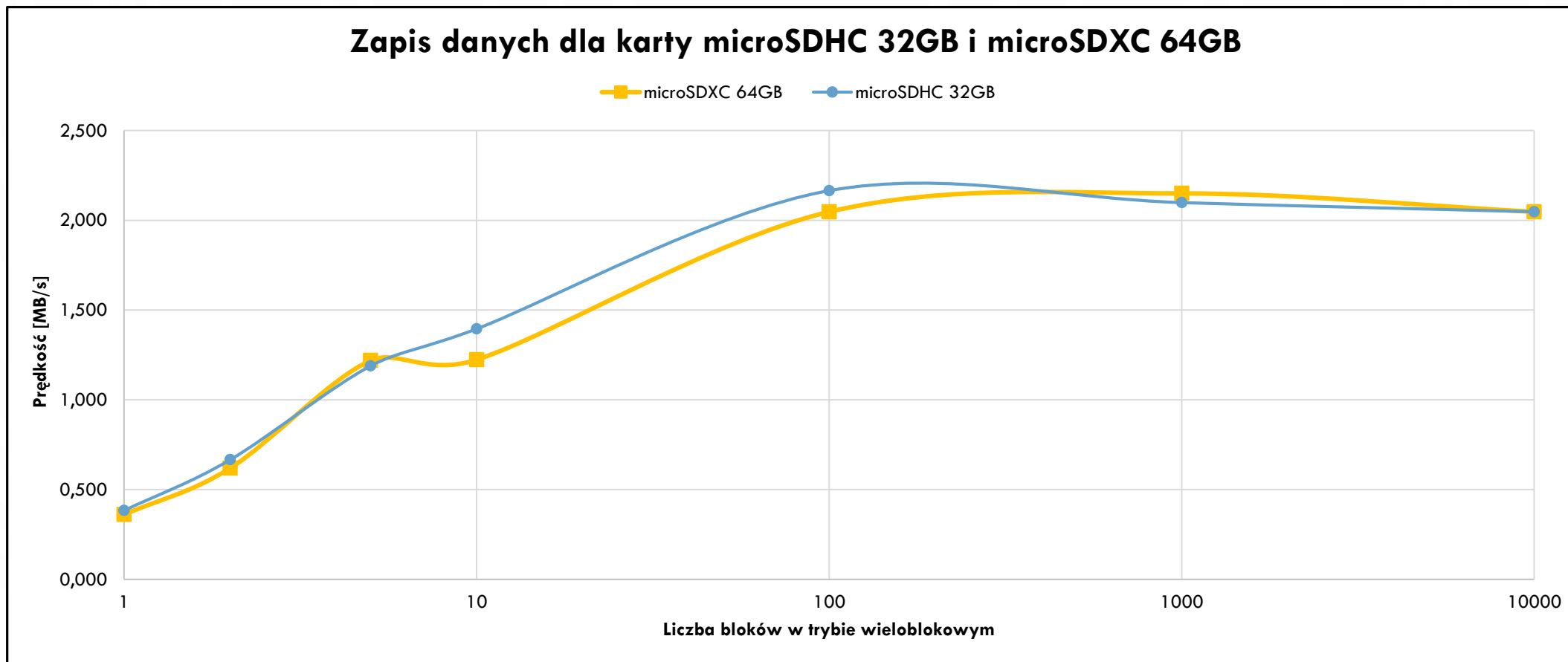
Porównanie wyników dla odczytu danych na karcie pamięci microSDHC 8GB dla różnych Vmax

SPOSÓB POMIARU PRĘDKOŚCI

b) Pomiar wykonany dla karty pamięci SANDISK ULTRA microSDHC 32GB oraz dla karty SANDISK ULTRA ANDROID microSDXC 64GB dla prędkości:

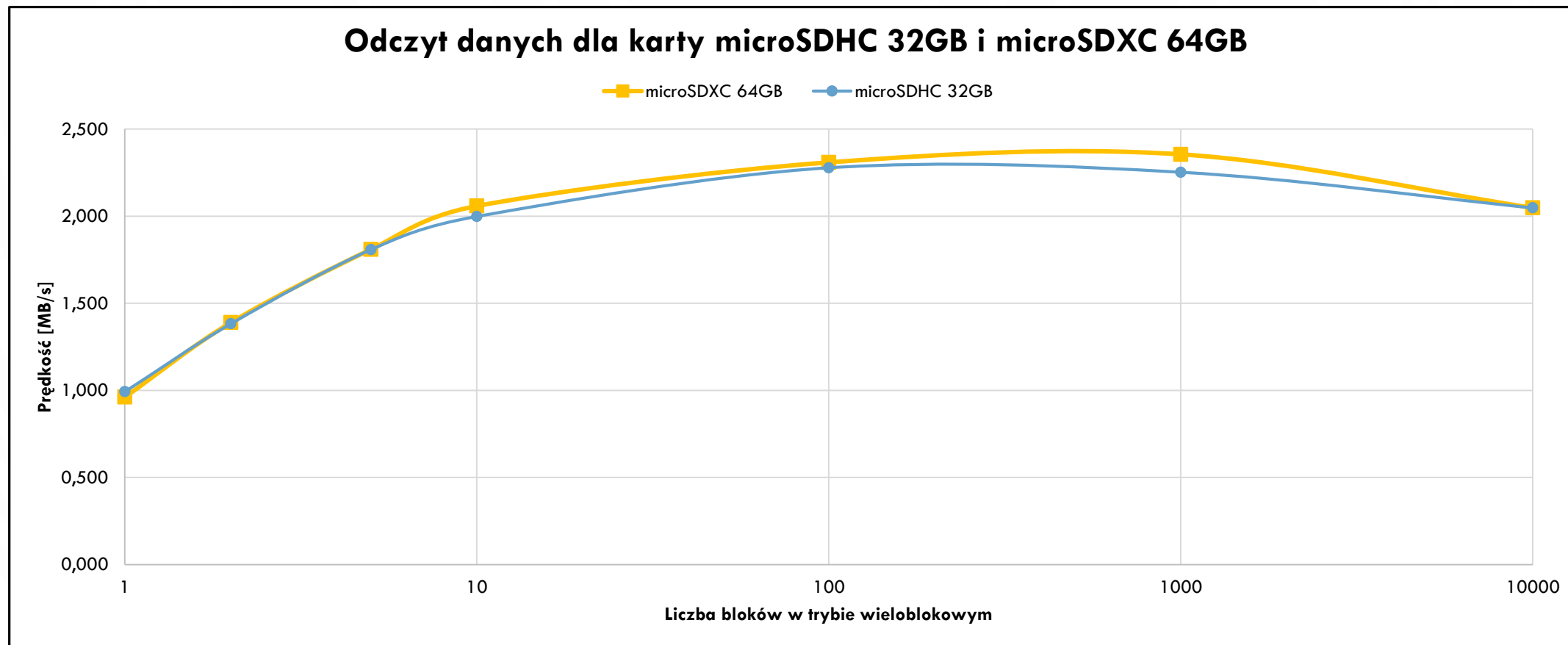
$$V_{max} = 50\,000\,000 \text{ [Hz]} / 1[\text{dzielnik}] / 8 [\text{takt/bajt}] / 2 [1/\text{takt}] = 3,125 \text{ [MB/s]}$$

WYNIKI POMIARÓW DLA KARTY PAMIĘCI MICROSDHC 32GB ORAZ MICROSDXC 64GB



Zapis danych dla karty microSDHC 32GB i microSDXC 64GB dla $V_{max} = 3,125 \text{ MB/s}$

WYNIKI POMIARÓW DLA KARTY PAMIĘCI MICROSDHC 32GB ORAZ MICROSDXC 64GB



Odczyt danych dla karty microSDHC 32GB i microSDXC 64GB dla $V_{max} = 3,125 \text{ MB/s}$

WYNIKI SYNTEZY

Wyniki dla całego systemu testowego:

- Kontroler SPI
- Sterownik karty SD
- Moduł testujący
- Kolejka FIFO
- Moduł UART

Liczba rejestrów	331/4665 (7%)
Liczba komórek LUT-4	822/4320 (19%)
Liczba komórek SLICE	413/2160 (19%)

Dopuszczalny zegar: 101 MHz

Design Summary

Number of registers: 331 out of 4665 (7%)
PFU registers: 327 out of 4320 (8%)
PIO registers: 4 out of 345 (1%)
Number of SLICES: 413 out of 2160 (19%)
SLICES as Logic/ROM: 413 out of 2160 (19%)
SLICES as RAM: 0 out of 1620 (0%)
SLICES as Carry: 96 out of 2160 (4%)
Number of LUT4s: 822 out of 4320 (19%)
Number used as logic LUTs: 630
Number used as distributed RAM: 0
Number used as ripple logic: 192
Number used as shift registers: 0
Number of PIO sites used: 15 + 4(JTAG) out of 115 (17%)
Number of block RAMs: 1 out of 10 (10%)
Number of GSRs: 1 out of 1 (100%)

WYNIKI SYNTEZY

Wyniki tylko dla zaprojektowanych modułów:

- Kontroler SPI
- Sterownik karty SD

Liczba rejestrów	171/5157 (3%)
Liczba komórek LUT-4	718/4320 (17%)
Liczba bloków SLICE	360/2160 (17%)

Dopuszczalny zegar: 107,6 MHz

Design Summary

Number of registers: 171 out of 5157 (3%)

PFU registers: 156 out of 4320 (4%)

PIO registers: 15 out of 837 (2%)

Number of SLICEs: 360 out of 2160 (17%)

SLICEs as Logic/ROM: 360 out of 2160 (17%)

SLICEs as RAM: 0 out of 1620 (0%)

SLICEs as Carry: 39 out of 2160 (2%)

Number of LUT4s: 718 out of 4320 (17%)

Number used as logic LUTs: 640

Number used as distributed RAM: 0

Number used as ripple logic: 78

Number used as shift registers: 0

Number of PIO sites used: 143 + 4(JTAG) out of 279 (53%)

Number of block RAMs: 0 out of 10 (0%)

Number of GSRs: 1 out of 1 (100%)

PODSUMOWANIE

- cel pracy został spełniony
- zapoznano się z dokumentacją interfejsu SPI
- zapoznano się parametrami i działaniem kart pamięci SD/MMC
- przed uruchomieniem w układzie została wykonana symulacja
- zostało stworzone dedykowane środowisko testowe
- zapis i odczyt wieloblokowy
- zastosowanie hierarchii modułów
- wraz ze wzrostem liczby bloków w trybie wieloblokowym rośnie prędkość zapisu lub odczytu
- odczyt jest szybszy niż zapis
- prędkość karty pamięci SD zależy od prędkości taktowania



DZIĘKUJĘ ZA UWAGĘ