



Pierwsze kroki z FPGA (7)

Termometr z analogowym sensorem i wyświetlaczem 7-segmentowym

Prezentujemy kolejną kompletną aplikację MAXimatora, w której spełnia on (a dokładniej – zastosowany w MAXimatorze układ FPGA) cyfrowego termometru, wyświetlającego temperaturę otoczenia zmierzoną za pomocą sensora STLM20 na 4-cyfrowym, multipleksowanym wyświetlaczu LED. W projekcie zastosowano m.in. wbudowany w FPGA MAX10 przetwornik ADC, który służy do konwersji sygnału analogowego do postaci cyfrowej.

Działanie projektu zaimplementowanego w FPGA najlepiej wyjaśni schemat blokowy pokazany na **rysunku 1**. Na wejście jednego z kanałów ADC wbudowanego w FPGA jest podawany sygnał napięciowy z wyjścia sensora STLM20. Przetwornik przetwarza napięcie na 12-bitową wartość cyfrową, następnie za pomocą techniki tablicowania (*Lookup Table*) odczytywana wartość jest konwertowana na wartość wyrażoną w stopniach Celsjusza i wyświetlana na multipleksowanym wyświetlaczu 7-segmentowym. Wyświetlacz LED oraz analogowy sensor temperatury znajdują się na ekspanderze (shieldzie – **fotografia 2**), który wraz z programatorem JTAG jest jednym z elementów wyposażenia promocyjnej wersji MAXimatora.

W **tabeli 1** zestawiono wyprowadzenia FPGA z zestawu MAXimator, które zostały wykorzystane w prezentowanym projekcie.

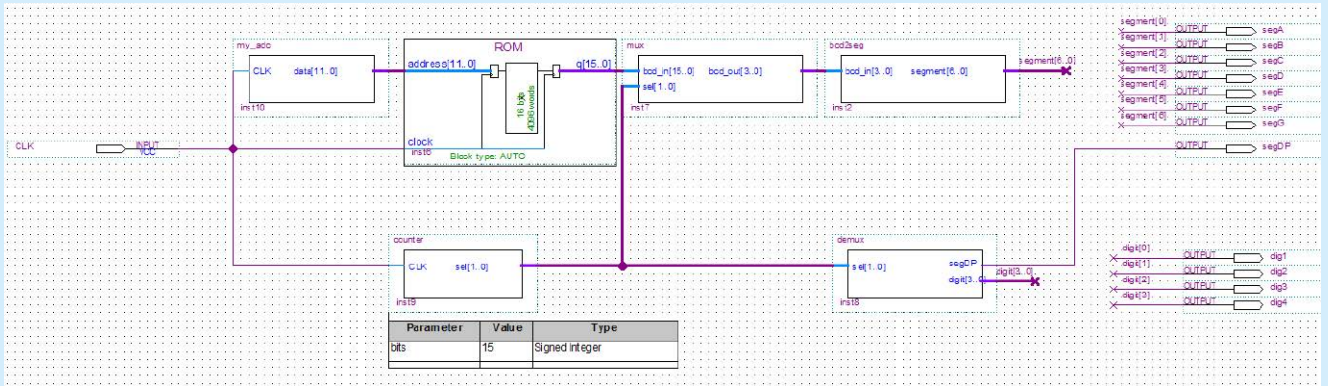
Blok *my_adc* przekazuje cyfrową wartość odpowiadającą zmierzonej temperaturze do bloku *ROM*, który przelicza ją na wartość w skali Celsjusza zapisaną w kodzie BCD. Moduł *mux* wybiera pojedynczą cyfrę z kodu BCD przekazuje ją do bloku *bcd2seg*, który zapisuje cyfrę w notacji 7-segmentowej. Moduł *counter* opóźnia multipleksowanie oraz służy do wybierania liczb do wyświetlenia. Moduł *demux*, na podstawie wartości otrzymanej z bloku *counter* wybiera odpowiedni wyświetlacz, na którym wyświetlona zostaje cyfra.

Więcej informacji:

Kompletny projekt dla programu Quartus Prime Lite wraz z plikami źródłowymi jest dostępny do pobrania na stronie www.maximator-fpga.org

Tabela 1. Wyprowadzenia FPGA (MAX10), które wykorzystano w projekcie prezentowanym w artykule

Nazwa	Kierunek	Funkcja
ADC1_15	Wejście	Wejście sygnału analogowego, podłączone do czujnika temperatury. Wejście jest ustawiane w bloku <i>my_adc</i>
CLK0p	Wejście	Sygnał zegarowy 10 MHz
Wyprowadzenia: L16, J15, J16, H15, H16, G15, G16, F16	Wyjście	Piny podłączone do wyświetlacza 7-segmentowego odpowiadające odpowiednio za segmenty: A, B, C, D, E, F, G, DP
Wyprowadzenia: E15, E16, D15, D16	Wyjście	Piny podłączone do wyświetlacza 7-segmentowego odpowiadające za wybranie aktywnej cyfry (jednej z czterech)



Rysunek 1. Schemat blokowy projektu zaimplementowanego w FPGA

Tabela 2. Wejścia i wyjścia modułu my_adc			
Nazwa	Kierunek	Ilość bitów	Funkcja
CLK	Wejście	1	Sygnal zegarowy
data	Wyjście	12	Przetworzona wartość cyfrowa

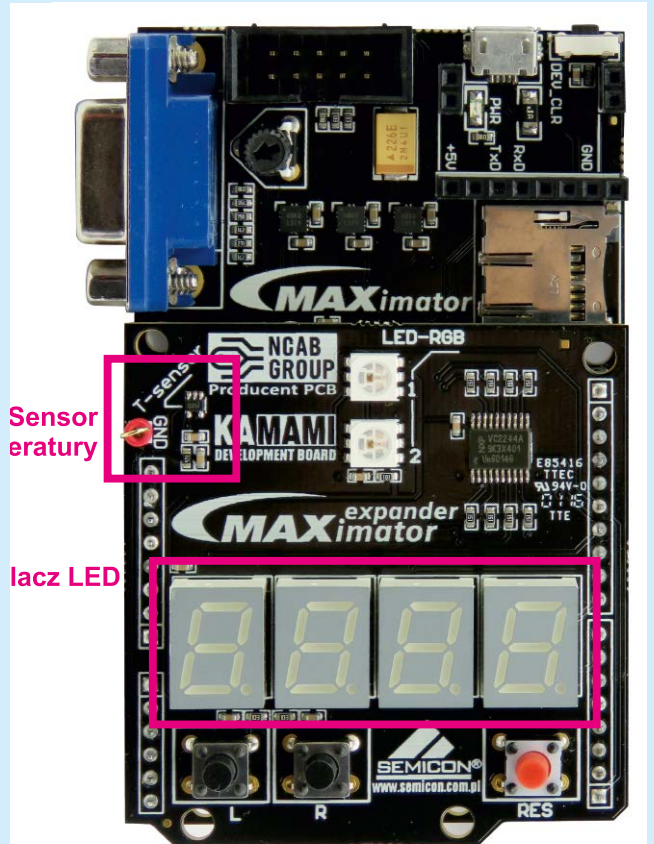
Tabela 3. Wejścia i wyjścia modułu ROM			
Nazwa	Kierunek	Ilość bitów	Funkcja
address	Wejście	12	Cyfrowa wartość temperatury
Q	Wyjście	16	Wartość temperatury w skali Celsjusza, zapisana w kodzie BCD

Tabela 4. Wejścia i wyjścia modułu ROM			
Nazwa	Kierunek	Ilość bitów	Funkcja
bcd_in	Wejście	16	Temperatura zapisana w kodzie BCD
Sel	Wejście	2	Decyduje o wyborze fragmentu z wejściowego kodu BCD
bcd_out	Wyjście	4	Pojedyncza cyfra zapisana w kodzie BCD

Funkcje poszczególnych bloków i ich sygnały są następujące:
MY_ADC. Moduł przetwarza wartość analogową z czujnika temperatury na wartość cyfrową. Wykorzystano w nim komponent ADC wygenerowany przy pomocy narzędzia Qsys programu Quartus Prime Lite. W module można ustawić, który kanał ma być wejściem analogowym przetwornika. W projekcie jest to kanał nr 15 – ADC1_15, co wynika ze sposobu podłączenia sensora STLM20 do wyprowadzeń FPGA. Wejścia i wyjścia tego modułu opisano w tabeli 2.

ROM. Jest to moduł pamięci ROM, zainicjalizowanej ręcznie utworzonym plikiem LUT.mif. Moduł odpowiada za przetworzenie wartości otrzymanej z czujnika temperatury na wartość w skali Celsjusza w kodzie BCD. Przeliczenie jest zrealizowane za pomocą techniki tablicowania (Lookup Table). Moduł został wygenerowany przy pomocy IP Core'a ROM: 1-PORT dostępnego w programie Quartus Prime Lite. Ze względu na znaczne wahania wartości odczytanej z czujnika temperatury w pliku LUT.mif nie została

Tabela 5. Wejścia i wyjścia modułu bcd2seg			
Nazwa	Kierunek	Ilość bitów	Funkcja
bcd_in	Wejście	4	Cyfra w kodzie BCD
segment	Wyjście	7	Cyfra w zapisie 7-segmentowym



Fotografia 2. Maximator z zainstalowaną płytą rozszerzenia2

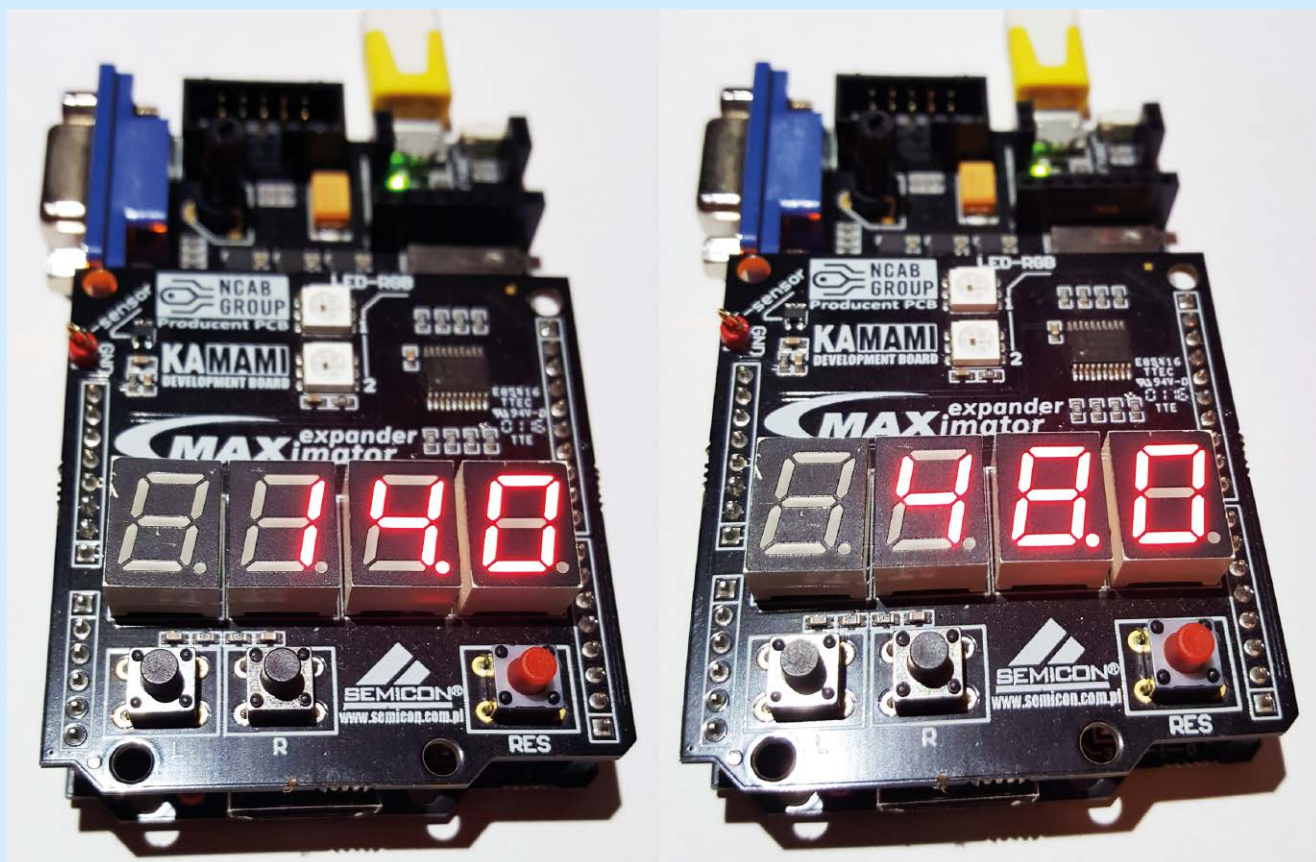
zamieszczona część dziesiąta wartości przedstawiającej temperaturę (dla każdej wartości część dziesiąta jest równa zero). Wejścia i wyjścia modułu ROM opisano w tabeli 3.

MUX. Jest to multiplexer, który wybiera z kodu BCD pojedynczą cyfrę do wyświetlenia. O wyborze cyfry decyduje wejściowy sygnał sel. Wejścia i wyjścia tego modułu opisano w tabeli 4.

BCD2SEG. Moduł otrzymaną cyfrę w kodzie BCD zapisuje w notacji 7-segmentowej. Wejścia i wyjścia tego modułu opisano w tabeli 5.

COUNTER. LICZNIK N-BITOWY. Ilość bitów ustawia się w parametrze bits. Funkcją licznika jest opóźnienie multipleksowania oraz wybór cyfry do wyświetlenia. Wejścia i wyjścia tego modułu opisano w tabeli 6.

Tabela 6. Wejścia i wyjścia modułu counter			
Nazwa	Kierunek	Ilość bitów	Funkcja
CLK	Wejście	1	Sygnal zegarowy
Sel	Wyjście	2	Sygnal sterujący wyborem cyfry



Fotografia 3. Wyniki kilku przykładowych pomiarów temperatury

Tabela 7. Wejścia i wyjścia modułu demux

Nazwa	Kierunek	Ilość bitów	Funkcja
sel	Wejście	2	Sygnal sterujący wyborem cyfry
digit	Wyjście	4	Sygnal wybierający cyfrę do wyświetlenia
segDP	Wyjście	1	Segment wyświetlacza odpowiedzialny za kropkę

DEMUX. Demultiplexer wybiera cyfrę do wyświetlenia, dodatkowo włącza kropkę oddzielającą część dziesiętną wartości temperatury. Wejścia i wyjścia tego modułu opisano w tabeli 7.

Prezentowany projekt jest dość prosty do wykonania, różni się od typowych implementacji realizowanych w FPGA przede wszystkim faktem wykorzystania wbudowanego w układ przetwornika A/C (wyposażenie rzadko spotykane w FPGA). Na fotografii 3 przedstawiono wyniki kilku przykładowych pomiarów temperatury wykonanych za pomocą MAXimatora i zastosowanego w shieldzie sensora temperatury STLM20.

Przykład został zaimplementowany w zestawie MAXimator za pomocą bezpłatnego narzędzia Quartus Prime Lite (do pobrania na stronie www.altera.com), komplet plików jest dostępny do pobrania na stronie maximator-fpga.org.

Przemysław Sala

<http://m.ep.com.pl>
**Najlepszy Mobilny
 Adres w Sieci**

