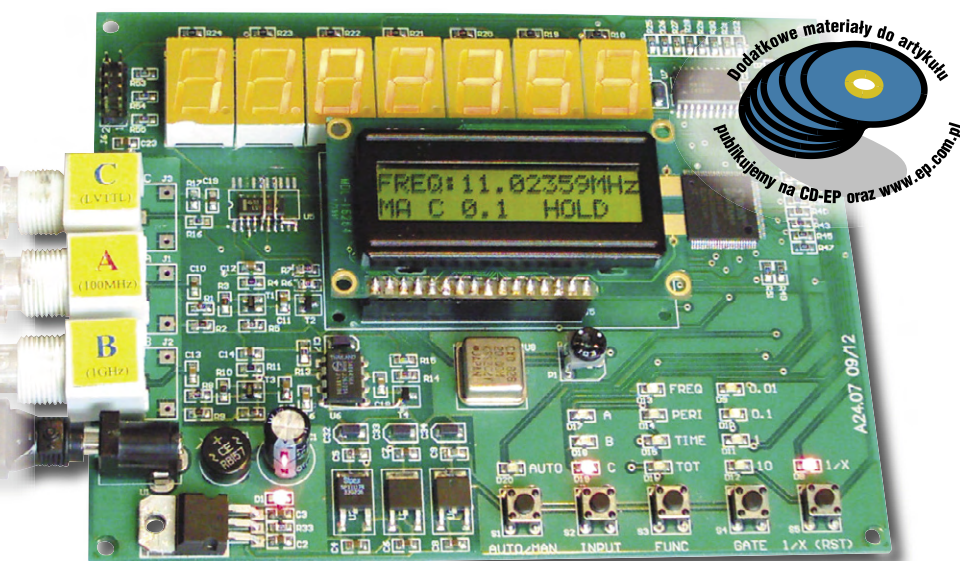


Uniwersalny miernik: częstotliwości, czasu, okresu na FPGA, część 2

AVT-5115

Jednym z przyrządów bardzo często wykorzystywanych w pracowni elektronika jest miernik częstotliwości. Tego typu przyrządy pomiarowe noszą czasem nazwę liczników uniwersalnych i oprócz pomiaru częstotliwości oferują również pomiar innych parametrów takich jak okres, czas trwania impulsu czy też całkowita liczba impulsów w pewnej grupie. W artykule przedstawiono opis budowy takiego licznika do którego konstrukcji wykorzystano układy programowalne FPGA oraz język opisu sprzętu Verilog.



Budowa licznika uniwersalnego

Uproszczony schemat blokowy części cyfrowej urządzenia, zawartej w układzie programowalnym, przedstawiono na rys. 7. Układ złożony z multiplekserów *MUX1*, *MUX2* i *MUX3* oraz przerzutników *FF1* i *FF2* odpowiada za przełączanie źródła sygnału mierzzonego (jedno z trzech wejść A, B, C) oraz odpowiednią konfigurację sygnałów z generatora podstawy czasu i sygnałów wejściowych, tak aby możliwy był pomiar częstotliwości, okresu i czasu trwania impulsu (por. rys. 1, 2, 3).

Generator podstawy czasu jest w rzeczywistości zbiorem dzielników częstotliwości, które zewnętrzny sygnał taktujący z generatora kwarcowego poddają podziałowi przez odpowiednie współczynniki. Generator podstawy czasu dostarcza impulsów o czasie trwania 10 ms, 64 ms, 100 ms, 640 ms, 1 s i 10 s, które w przypadku funkcji pomiaru częstotliwości, poprzez multiplekser *MUX3* (wybór impulsu o określonym czasie trwania) sterują bramką główną *G1* i *G2*. Impulsy o czasie trwania 64 ms i 640 ms

wykorzystywane są w przypadku dokonywania pomiarów częstotliwości sygnału z wejścia B, do którego dołączony jest zewnętrzny prescaler rozszerzający zakres pomiarowy miernika, o współczynniku podziału 1:64. Do celów pomiaru okresu i czasu trwania impulsu generator podstawy czasu wytwarza przebiegi o częstotliwości 1 kHz, 100 kHz, 1 MHz, 10 MHz oraz 100 MHz. Jeden z tych przebiegów jest wybierany (na podstawie ustawień układu sterownia i układu automatycznej zmiany zakresu) w multiplekserze *MUX4* i dostarczany do bramki głównej. Częstotliwość 100 MHz jest wytwarzana dzięki wykorzystaniu syntezerza częstotliwości dostępnego w ramach bloku DCM wybranego układu FPGA.

Ważną rolę spełnia moduł synchronizatora wykorzystywany podczas pomiarów okresu oraz czasu trwania impulsu. Zadaniem tego modułu jest wychwycenie z przebiegu wejściowego pojedynczego okresu lub impulsu sygnału wejściowego i umożliwienie pomiaru jego czasu trwania. W odpowiedzi na całkowicie asynchroniczny z mierzonym przebiegiem wejścio-

PODSTAWOWE PARAMETRY

- pomiar częstotliwości, okresu, czasu trwania impulsu (z funkcją stopera), całkowitej liczby impulsów,
- automatyczna zmiana zakresu,
- pomiary w trybie licznika konwencjonalnego i odwrotnego,
- podwójny wyświetlacz 7-cyfrowy, 7-segmentowy LED i/lub alfanumeryczny LCD 2x16 znaków,
- maksymalna rozdzielczość pomiaru częstotliwości: 10^{-6} Hz, czasu: 10 ns,
- trzy wejścia pomiarowe:
 - wejście A: zakres częstotliwości: 30 Hz...100 MHz, czułość $S < 75$ mV ($f_{in} = 10$ MHz), impedancja wejściowa $Z_{in} > 1,3$ M Ω ($f = 1$ kHz),
 - wejście B: zakres częstotliwości: 70 MHz – 1 GHz, czułość (bez wtórnika T4) $S \approx 10$ mV, impedancja wejściowa $Z_{in} = 50$ Ω ,
 - wejście C: wejście cyfrowe LVTTTL (z tolerancją 5 V), zakres częstotliwości od 0 do ok. 150 MHz
- zasilanie minimum 6 VAC lub 7,5 VDC

wym sygnał sterujący (uzbrajający) z układu sterowania, synchronizator odblokowuje bramkę G3 w momencie gdy wejściowy sygnał mierzony przyjmuje poziom niski, tak aby możliwy był pomiar czasu trwania pełnego jednego okresu

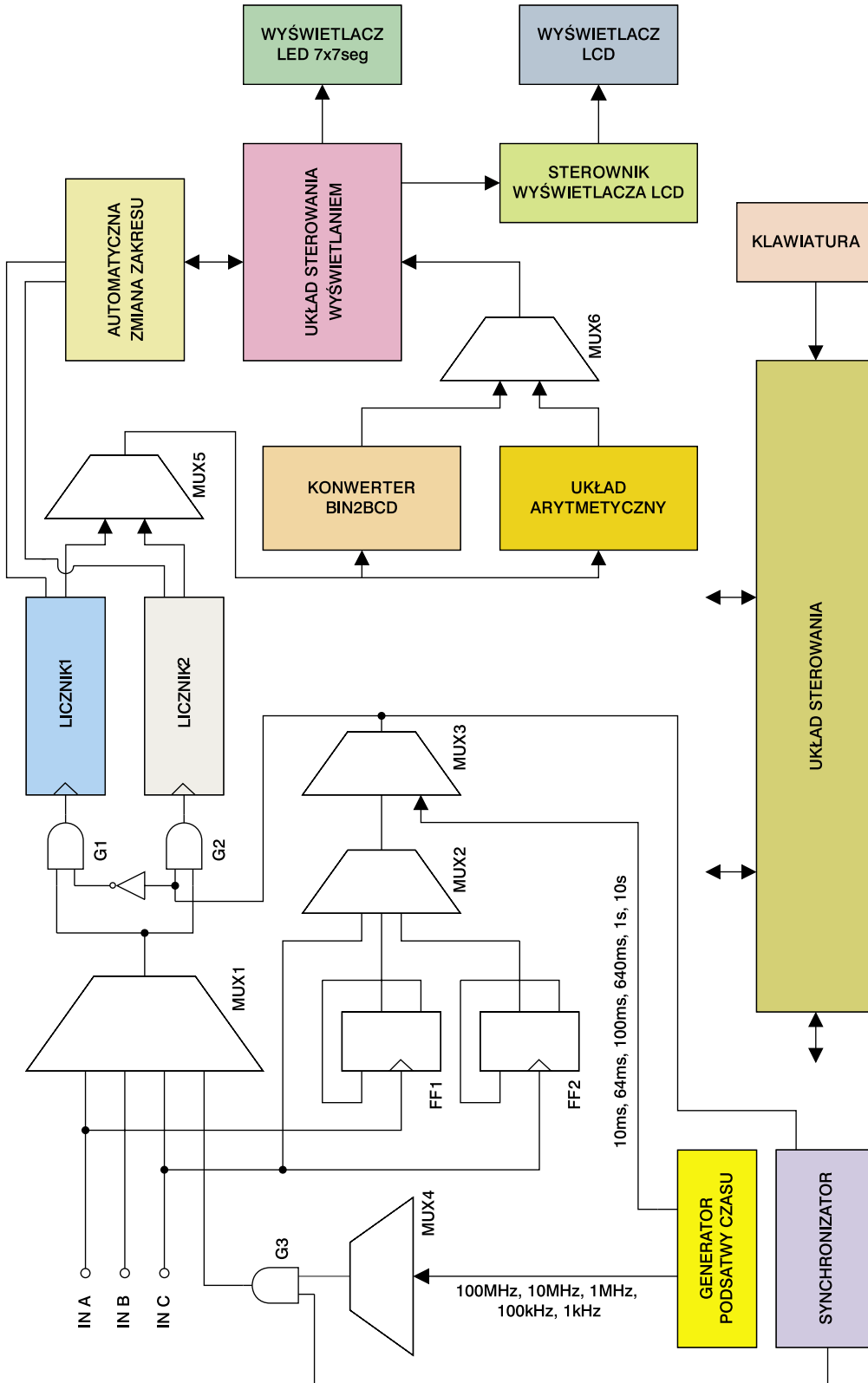
lub impulsu tego sygnału. Moduł synchronizatora został zrealizowany jako układ asynchroniczny. Realizacja układów asynchronicznych w strukturach FPGA jest szczególnie wymagająca ze względu na to, że zarówno narzędzia syntezy jak

i architektura komercyjnych układów FPGA jest zorientowana na projektowanie i implementację układów synchronicznych.

W prezentowanym urządzeniu, do celów zliczania impulsów zastosowano dwa 24-bitowe liczniki synchroniczne pracujące w naturalnym kodzie binarnym. W czasie wykonywania większości pomiarów wykorzystywany jest tylko jeden licznik. Drugi licznik został wprowadzony w celu szybszego uzyskania wyniku pomiaru częstotliwości dla czasów otwarcia bramki równych 1 sekunda i 10 sekund. W takim przypadku obydwa liczniki pracują naprzemiennie w zależności od polaryzacji sygnału z generatora podstawy czasu (naprzemiennie otwierane są bramki G1 i G2). Dzięki temu gotowy wynik pomiaru uzyskuje się co każdą 1 sekundę lub każde 10 sekund – zależnie od wybranego zakresu pomiarowego, nie zaś co 2 sekundy lub 20 sekund, jak miało by to miejsce w przypadku zastosowania tylko jednego licznika.

Informacja o ilości zliczonych impulsów w licznikach, poprzez multiplexer MUX5, trafia albo do modułu konwertera przekształcającego naturalny kod binarny do kodu BCD, albo do układu arytmetycznego – zależnie od trybu pracy miernika. W trybie pracy urządzenia jako licznika konwencjonalnego (bezpośredni pomiar częstotliwości) wykorzystywany jest konwerter kodu. W przypadku pracy miernika w trybie licznika odwrotnego używany jest układ arytmetyczny obliczający wartość odwrotną do wartości stanu licznika 1. Wynik przetwarzania tego modułu (wartość odwrotna) jest reprezentowany również w kodzie BCD.

Informacja z konwertera kodu lub układu arytmetycznego trafia następnie do modułu sterowania wy-



Rys. 7. Uproszczony schemat blokowy części cyfrowej licznika uniwersalnego

świeceniem. Moduł ten odpowiada za prawidłowe wyświetlanie jednostek mierzonych wielkości (zapalenie diody LED reprezentującej daną jednostkę), odpowiednie sterowanie położeniem kropki dziesiętnej na wyświetlaczu, wygaszanie nieznaczących zer (z lewej strony) na wyświetlaczu oraz realizację funkcji zatrzymania na wyświetlaczu bieżącego wyniku pomiaru (HOLD). Moduł jest również wyposażony w multipleksowy sterownik 7-segmentowego wyświetlacza LED.

Moduł sterowania wyświetlaniem dostarcza również informację dla modułu obsługującego alfanumeryczny wyświetlacz LCD, który umożliwia alternatywne korzystanie z wyświetlacza LED i/lub LCD.

Dzięki układowi automatycznej zmiany zakresu pomiarowego możliwe jest uzyskanie optymalnej, w danych warunkach, rozdzielczości pomiaru. Układ ten na podstawie informacji o wystąpieniu przepełnienia liczników oraz informacji z układu sterowania wyświetlaniem (ilość nieznaczących zer w wyniku pomiaru), we współpracy z nadrzędnym układem sterowania, automatycznie dobiera nastawy generatora podstawy czasu.

Zasadniczą rolę spełnia układ sterowania, który koordynuje pracę innych modułów, obsługując również interfejs użytkownika (klawiaturowy wybór funkcji, nastawy podstawy czasu, itp.).

W dalszej części tekstu przedstawione zostaną bardziej szczegółowo wybrane moduły urządzenia wraz z opisującym je kodem w języku Verilog, który można potraktować jako rodzaj wirtualnych komponentów. Komponenty te mogą być później dowolnie wykorzystane w innych projektach i aplikacjach.

Układ arytmetyczny

Układ ten, którego zadaniem jest obliczenie wartości odwrotnej ($f(x)=1/x$), składa się z dwóch modułów: modułu realizującego dzielenie całkowite z resztą oraz nadrzędnego modułu wyliczania wartości odwrotnej, wykorzystujący moduł dzielenia całkowitego.

Działanie modułu dzielenia całkowitego opiera się na restytucyjnej (*restoring*) metodzie dzielenia. Metoda ta polega na iteracyjnym przesuwaniu w lewo wszystkich bitów słowa złożonego z dzielnej

i reszty częściowej, odejmowaniu dzielnika i badaniu znaku różnicy. Jeżeli znak ten jest ujemny oznacza to, że trzeba powrócić do poprzedniej wartości reszty częściowej (restytuować ją) lub, jeśli nie, to jako bieżącą wartość reszty częściowej przyjmuje się wynik odejmowania. Bardziej szczegółowo metoda ta została pokazana na przykładzie czterobitowego dzielenia liczby 7

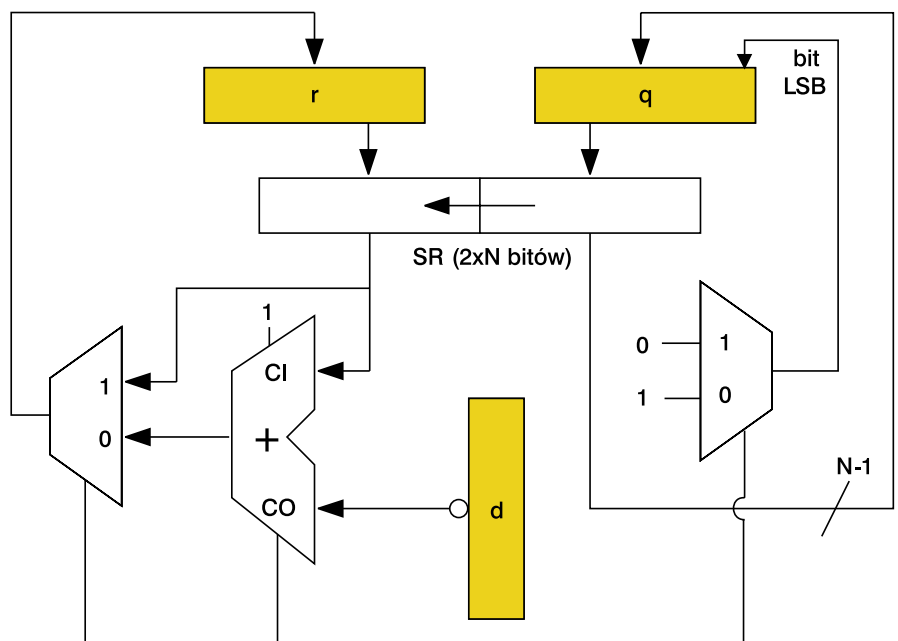
przez 3 (rys. 8). W przykładzie tym najpierw zestawia się słowo o podwójnej ilości bitów w stosunku do dzielnej, inicjując go wartością 0 dla starszej części słowa (HW) i wartością dzielnej dla młodszej części słowa (LW). Następnie słowo to jest przesuwane o jeden bit w lewo. W kolejnym kroku wykonuje się odejmowanie dzielnika od starszej części tego słowa. Jeżeli wynik odejmowania (zapisany w kodzie U2) jest ujemny wówczas odtwarzana jest (albo inaczej: pozostawiana jest bez zmian) poprzednia wartość starszej części słowa HW (wartość po przesunięciu całego słowa o jeden bit w lewo). Jednocześnie bieżący bit ilorazu częściowego jest

HW	LW	
0000	0111	
0000	111	przesunięcie w lewo
-0011		odjęcie dzielnika
(1)1101	1110	wynik ujemny, bit ilorazu równy 0
0000	1110	restytucja HW
0001	110	przesunięcie w lewo
-0011		odjęcie dzielnika
(1)1110	1100	wynik ujemny, bit ilorazu równy 0
0001	1100	restytucja HW
0011	100	przesunięcie w lewo
-0011		odjęcie dzielnika
(0)0000	1001	wynik nieujemny, bit ilorazu równy 1
0001	001	przesunięcie w lewo
-0011		odjęcie dzielnika
(1)1110	0010	wynik ujemny, bit ilorazu równy 0
0001	0010	restytucja HW

Rys. 8. Przykład dzielenia dwóch liczb binarnych metodą restytucyjną

równy 0. Bit ten zapisywany jest na najmłodszej pozycji półsłowa LW. Jeżeli wynik odejmowania jest dodatni wówczas do HW zapisuje się ten wynik, a bieżący bit ilorazu częściowego przyjmuje wartość 1. Opisane powyżej kroki powtarza się tyle razy, ile bitów liczy słowo dzielnej. Po wykonaniu ostatniej iteracji w półsłowie LW zawarty jest wynik dzielenia (iloraz), a w półsłowie HW reszta z tego dzielenia. W przykładzie z rys. 8 po czterech iteracjach otrzymano wynik dzielenia równy 2 (LW) i resztę 1 (HW).

Klasyczny sposób realizacji sprzętowej tego algorytmu wymaga, w najgorszym przypadku, 2N taktów zegara, gdzie N oznacza wy-



Rys. 9. Architektura układu wykonującego operację dzielenia całkowitego

maganą liczbę bitów słowa danych wejściowych (operacja restytucji HW zajmuje dodatkowy takt zegara). Jednak architekturę układu realizującego dzielenie można tak zmodyfikować, by niezależnie od wartości danych wejściowych czas wykonania operacji dzielenia był zawsze równy N taktów zegara. Schemat blokowy takiej architektury układu dzielącego przedstawiono na **rys. 9**. Bloki oznaczone na rysunku jako r i q to rejestry przechowujące odpowiednio resztę z dzielenia i iloraz (odpowiedniki półsłów HW i LW). Blok oznaczony jako d to rejestr statyczny zawierający wartość dzielnika. Wszystkie pozostałe bloki są układami kombinacyjnymi. Blok SR realizujący kombinacyjną funkcję przesunięcia o jeden bit w lewo słowa złożonego z zawartości rejestrów r i q . Funkcja odejmowania realizowana jest w układzie sumatora z ustawionym bitem wejścia przeniesienia i zanegowanym słowem reprezentującym odjemnik (zgodnie z równaniem: $A-B=A+B'+1$). Wyjście przeniesienia sumatora steruje wejściami adresowymi dwóch multiplexerów. Jednego odpowiedzialnego za zapis informacji do rejestru r i drugiego sterującego zapisem najmłodszego bitu ilorazu częściowego w rejestrze q . Wyjście przeniesienia $CO=1$ oznacza, że wynik odejmowania jest ujemny i wówczas do rejestru q jest zapisywana starsza część słowa złożonego z wartości rejestrów q i r – przesuniętego o jeden bit w lewo w układzie kombinacyjnym SR. Dla $CO=0$ do rejestru q jest zapisywa-

List. 1. Opis HDL modułu dzielenia całkowitego

```

module divNN(clk,start,divident,divisor,quotient,remainder,done);
parameter N=28; //liczba bitów dla danych wejściowych i wyniku
parameter BIT_CNT_WIDTH=5; //minimalna liczba bitów do zapisania liczby N
input clk,start;
input [N-1:0] divident,divisor;
output [N-1:0] quotient,remainder;
output reg done;

//licznik bitów
reg [BIT_CNT_WIDTH-1:0] bcnt;
wire [2*N-1:0] sreg;
wire [N:0] hw;
reg [N-1:0] rem;
reg [N-1:0] q;

//przesunięcie o jeden bit w lewo słowa złożonego
//z reszty częściowej (rem) i ilorazu częściowego (q)
assign sreg={rem,q}<<1;
//od reszty częściowej (starsza część sreg) odejmujemy dzielnik (divisor)
//hw ma szerokość o jeden bit większą na przechowanie znaku odejmowania
assign hw=sreg[2*N-1:N]-divisor;

always @(posedge clk)
begin
if(start) //jeżeli start=1 inicjowanie wartości początkowych
begin q<=divident; rem<=0; bcnt<=N; done<=0;end
else
begin
if(bcnt!=0) //jeżeli poniższych operacji nie wykonano jeszcze N razy
begin
if(hw[N]) //badanie wyniku operacji odejmowania (najstarszego bitu hw)
begin
//jeżeli wynik ujemny zapamiętanie (przywrócenie) wartości reszty częściowej
rem<=sreg[2*N-1:N]; q<=sreg[N-1:0];
end else
begin
//jeżeli wynik dodatni reszta częściowa otrzymuje wartość wyniku odejmowania
//bieżący bit ilorazu częściowego równy jeden
rem<=hw[N-1:0]; q<={sreg[N-1:1],1'b1};
end
bcnt<=bcnt-1; //zmniejszenie licznika bitów o jeden
end
else done<=1'b1; //koniec sygnalizowany ustawieniem rejestru done
end
end
assign quotient=q; //iloraz
assign remainder=rem; //reszta z dzielenia
endmodule
    
```

na wartość wyniku odejmowania. Dzięki takiej architekturze uzyskuje się jeden bit ilorazu częściowego w jednym takcie zegara. Wykonanie dzielenia dla danych wejściowych o długości słowa równego N bitów wymaga więc N taktów zegara.

Implementację architektury z **rys. 9** w postaci modułu opisanego w języku Verilog przedstawiono na **list. 1**. Moduł jest w pełni sparametryzowany, co oznacza, że poprzez zmianę wartości parametru N (i powiązanego z tą wartoś-

cią parametru BIT_CNT_WIDTH) można uzyskać realizację operacji dzielenia całkowitego dla danych o długości N bitów ($N>1$). Rozpoczęcie operacji dzielenia wymaga ustawienia przez jeden takt zegara sygnału $start$. Zakończenie dzielenia sygnalizowane jest ustawieniem sygnału $done$. Sygnał ten jest zerowany przy następnym aktywnym poziomie sygnału na wejściu $start$.

Zbigniew Hajduk
zhajduk@prz-rzeszow.pl

R	E	K	L	A	M	A
<h1 style="color: red; margin: 0;">WYKRYWACZE METALI</h1>						
 <p>CS150 Dyskryminator audio VU meter Wodoszczelna sonda (20 cm)</p>	<p>Cena: 390 zł</p>			 <p>CS10MD Wykrywacz "ręczny" Idealny dla policjantów i ochroniarzy.</p>	<p>Cena: 190 zł</p>	
Zamówienia przyjmuje Dział Handlowy AVT 01-939 Warszawa, ul. Burleska 9, tel. 022 568 99 50, fax 022 568 99 55, e-mail: handlowy@avt.pl, www.sklep.avt.pl						