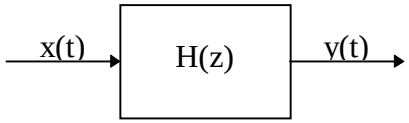
Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Realizacja filtrów cyfrowych z buforowaniem próbek

1. Filtry Cyfrowe

Zadaniem filtracji jest przepuszczanie (tłumienie) składowych sygnału leżących w określonym paśmie częstotliwości. Ogólnie filtr można przedstawić w następujący sposób:



(1.A)

$$Y(z) = H(z) * X(z)$$

(1.B)

$$y(t) = \int_{-\infty}^{+\infty} x(\tau)h(t-\tau)d\tau \quad \text{(splot)}$$

h(t) - jest odpowiedzią filtru na impuls jednostkowy
H(z) - jest transformatą Laplace'a funkcji h(t) **(Dyskretna Transmitancja filtru)**

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Dla układów cyfrowych opis filtru ma postać równania splotu sygnałów cyfrowych:

(1.C)

$$y(n) = \sum_{k=-\infty}^{+\infty} x(k)h(n-k)$$

Warunek stabilności

$$\sum_{k=-\infty}^{+\infty} |h(k)| < \infty$$

lub postać równania różniczkowego:

(1.D)

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Filtry cyfrowe można podzielić na dwa rodzaje:

- Filtry o Nieskończonej Odpowiedzi Impulsowej - **NOI** (filtry rekursywne)
  wymagają mniejszej liczby współczynników,
  ale używane są wyszukane metody projektowania w celu zapewnienia stabilności i dokładności.

- Filtry o Skończonej Odpowiedzi Impulsowej - **SOI**
  bezwarunkowo stabilne, mogą mieć liniową charakterystykę fazową,

Typowy filtr NOI

(1.E)

$$y(n) = \sum_{k=0}^{p} a_k x(n-k) - \sum_{k=1}^{r} b_k y(n-k)$$

Typowy filtr SOI

(1.F)

$$y(n) = \sum_{k=0}^{p} a_k x(n-k)$$

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Synteza filtrów cyfrowych polega na:
1. Określeniu żądanych charakterystyk częstotliwościowych
2. Wyznaczeniu odpowiedniej transmitancji H(z) poprzez aproksymację
3. Wyznaczeniu jednego z równań: (1.E) oraz (1.F)
4. Opracowanie algorytmu obliczającego powyższe równania różniczkowe
5. Zbudowanie i oprogramowanie układu mikroprocesorowego.

Z punktu widzenia cyfrowego przetwarzania sygnałów analogowych jest bardzo ważne , aby dokładnie określić, które składowe sygnału zawierają poszukiwaną informację.
Dla składowej o maksymalnej częstotliwości należy dobrać odpowiednią częstotliwość próbkowania przetwornika A/C.
Aby sygnał wejściowy przetwarzany był prawidłowo należy odpowiednio dobrać:
częstotliwość próbkowania przetwornika A/C f_p:
1.

$$f_p \geq 2*f_{max}^{1} \qquad \textbf{(1.G)}$$

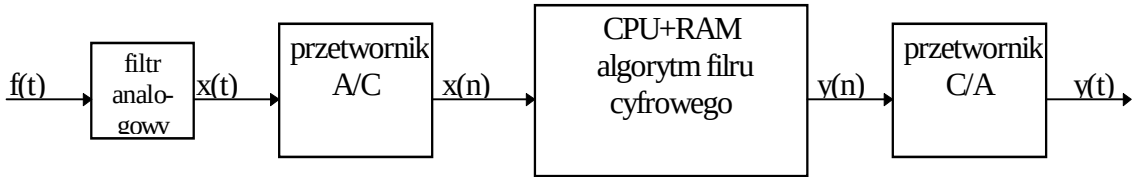
dla poprawnego odtworzenia składowej stałej
2.

$$f_p \geq 4*f_{imax} \qquad \textbf{(1.H)}$$

dla poprawnego odtworzenia i-tej składowej.

[1] wynika z częstotliwości Nyquista.

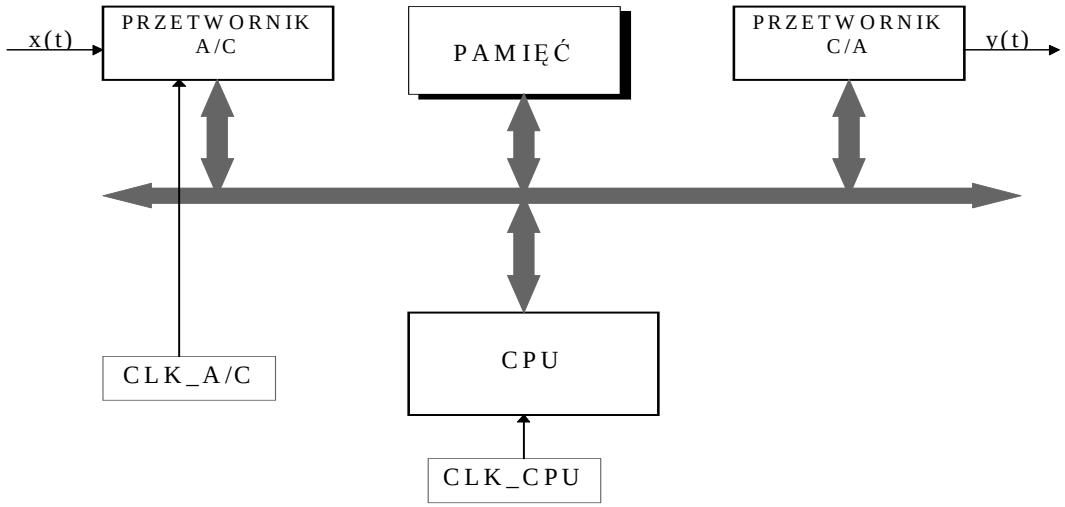
Na Rys. 1-A przedstawiono schemat blokowy układu realizującego filtr cyfrowy. Układ ten zawiera dodatkowo dolnoprzepustowy filtr analogowy, który ogranicza pasmo sygnału wejściowego do ustalonej maksymalnej częstotliwości co zapobiega powstawaniu błędów niekorygowalnych.



Rys. 1-A Schemat blokowy filtru cyfrowego.

Na Rys. 1-B przedstawiono standardowy schemat blokowy systemu mikroprocesorowego realizującego filtr cyfrowy. Głównymi elementami takiego systemu są:

- CPU+pamięć - realizacja algorytm filtru cyfrowego
- przetwornik analogowo-cyfrowy A/C
- przetwornik cyfrowo analogowy C/A.



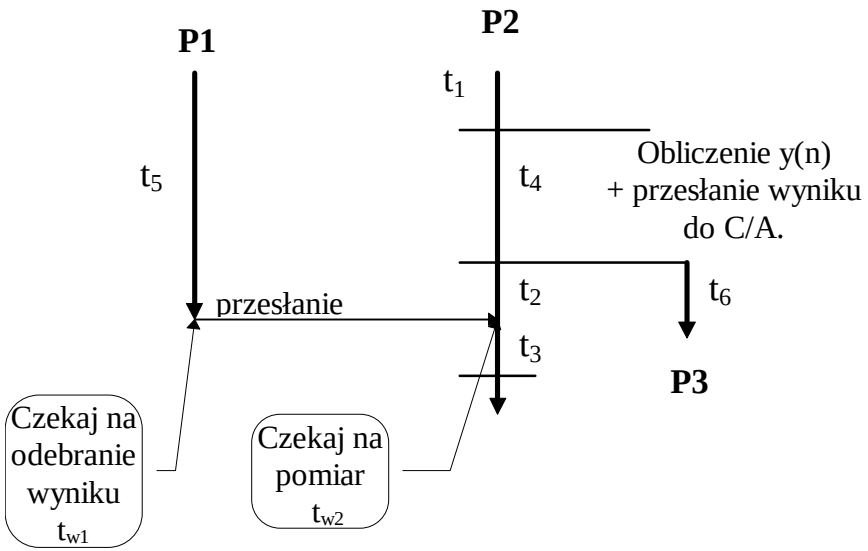
Rys. 1-B Schemat blokowy systemu mikroprocesorowego realizującego filtr cyfrowy

Jak wspomniano wcześniej, maksymalna częstotliwość filtrowanego sygnału x(n) zależy od czasu przetwarzania przetwornika A/C. W rzeczywistości czas wyznaczenia i-tej wartości y(n) jest wydłużony o dodatkowy czas związany z procedurami pomocniczymi (programowanie oraz synchronizacja przetworników).

Dla optymalnego układu, szybkość CPU musi być tak dobrana, aby w czasie pobierania i-tej próbki, mogła być obliczona kolejna wartość y(n).
Rys. 1-C przedstawia przykładowy, jeden z możliwych, przepływ sterowania układu realizującego filtr cyfrowy. Zostały wyróżnione trzy procesy równoległe:

- P1 - proces charakteryzujący przetwarzanie przetwornika A/C
- P2 - proces charakteryzujący program wykonywany przez CPU.
- P3 - proces charakteryzujący przetwarzanie przetwornika C/A

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl



Rys. 1-C Schemat przepływu sterowania przykładowego układu realizującego filtr cyfrowy.

- Czas t_1 reprezentuje inne (oprócz algorytmu filtru) procedury wykonywane przez CPU
- Czasy: t_2 t_3 przedstawiają czas potrzebny na pobranie i zapisanie nowej próbki x(i) oraz uporządkowaniem poprzednich.
- Czas t_4 określa czas obliczenia i-tej próbki i wpisania jej do przetwornika C/A
- Czas t_5 oznacza czas przetwarzania przetwornika A/C
- Czas t_6 wynika z czasu ustalania sygnału wyjściowego przetwornika C/A od momentu wpisania nowej wartości.

Ogólnie czas przetwarzania jednej próbki można wyznaczyć za pomocą wzoru (1.I)

(1.I)

$$T = Max(t_5 , t_1+t_4+t_2) + Max(t_6 , t_2+t_3)$$

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Czas T oznacza również okres pobierania kolejnej próbki i w przypadku filtru cyfrowego, musi mieć wartość stałą. Stąd częstotliwość próbkowania wynosi

(1.J)

$$fp=1/T$$

Zgodnie z zależnościami **(1.G)** i (1.H) można określić maksymalną częstotliwość sygnału przetwarzanego przez filtr

(1.K)

$$f_{max} \leq 1/(4*T)$$

Podczas syntezy filtru cyfrowego występuje sytuacja odwrotna. Na podstawie maksymalnej częstotliwości przetwarzanego sygnału x(n) - f_{max} , należy wyznaczyć okres próbkowania T:

(1.L)

$$T \leq 4/f_{max}$$

Wartość okresu próbkowania T determinuje współczynniki a_k i b_k w zależnościach: (1.E) i (1.F) zgodnie z podstawieniem $z^{-1} = e^{-j\omega T}$.

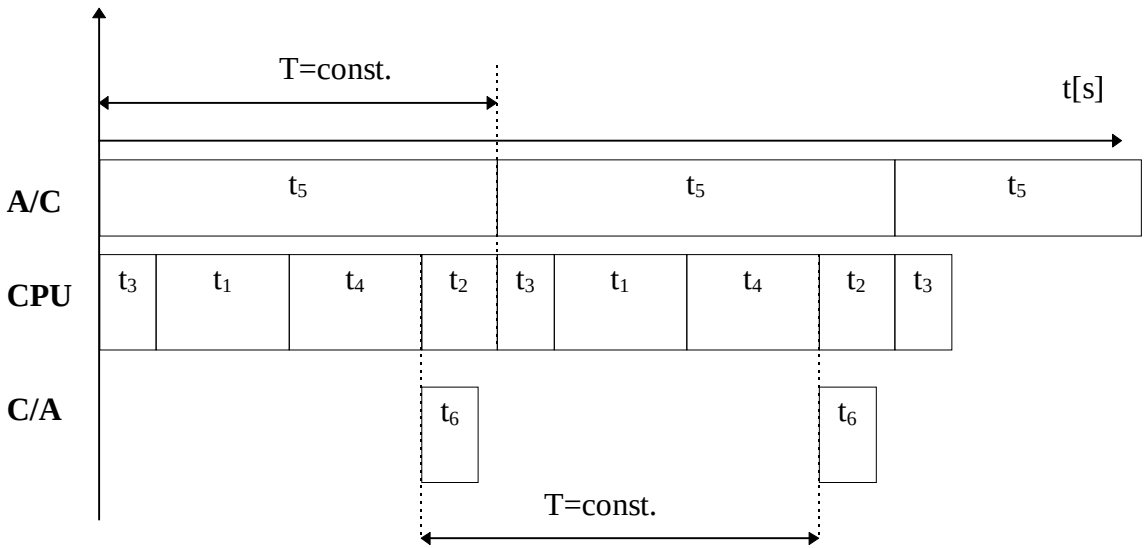
Na Rys. 1-C zaznaczono miejsce pobierania danych z przetwornika A/C. Poprawną współbieżność poszczególnych procesów wymaga odpowiedniej synchronizacji w czasie:

- jeżeli przetwornika A/C wykonał pomiar, a dane nie zostały odebrane, A/C czeka na gotowość procesu P2 - $t_{w1} = t_1+t_4+t_2 - t_5$
- jeżeli pomiar z A/C jest nie gotowy w chwili pobierania danych przez proces P2, CPU wstawia dodatkowe cykle w oczekiwaniu na pomiar - $t_{w2} = t_5 - t1+t_4+t_2$.

Wartości t_{w1}, t_{w2} mogą być miarą stopnia wykorzystania zasobów systemu. W optymalnie zaprojektowanym systemie $t_{w1}=0$ oraz $t_{w2}=0$.

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Rys. 1-D przedstawia graficzny obraz współpracy przetworników oraz CPU w sytuacji optymalnego dobrania poszczególnych elementów.



Rys. 1-D Synchronizacja pracy przetworników oraz CPU.

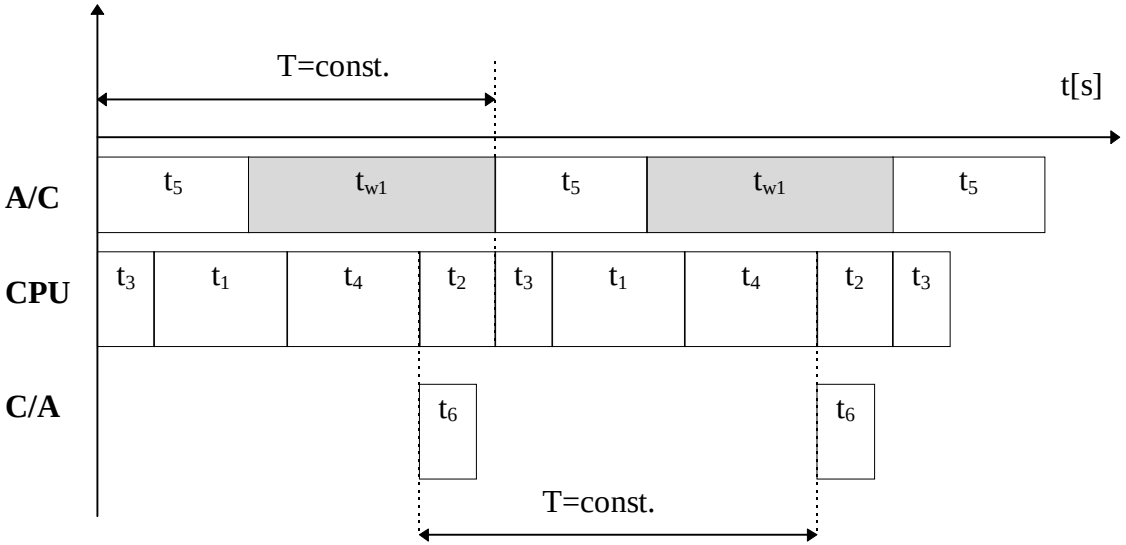
(1.M)

$$T = t_5 = t_1+t_2+t_3+t_4$$

Uzyskanie tak idealnej synchronizacji (dobraniu szybkości przetwarzania A/C i CPU) jak na Rys. 1-D jest bardzo trudne. W praktyce mamy doczynienia z dwoma przypadkami:

1) tw1≠0, tw2=0

Przetwornik A/C jest układem bardzo szybkim w stosunku do CPU i oczekuje na odczytanie wyniku pomiaru t_{w1} - Rys. 1-E



Rys. 1-E Dodatkowy czas oczekiwania t_{w1} przetwornika A/C na odczyt wyniku pomiaru.

W takim przypadku okres próbkowania zależy od szybkości programu wykonywanego przez CPU i wynosi:

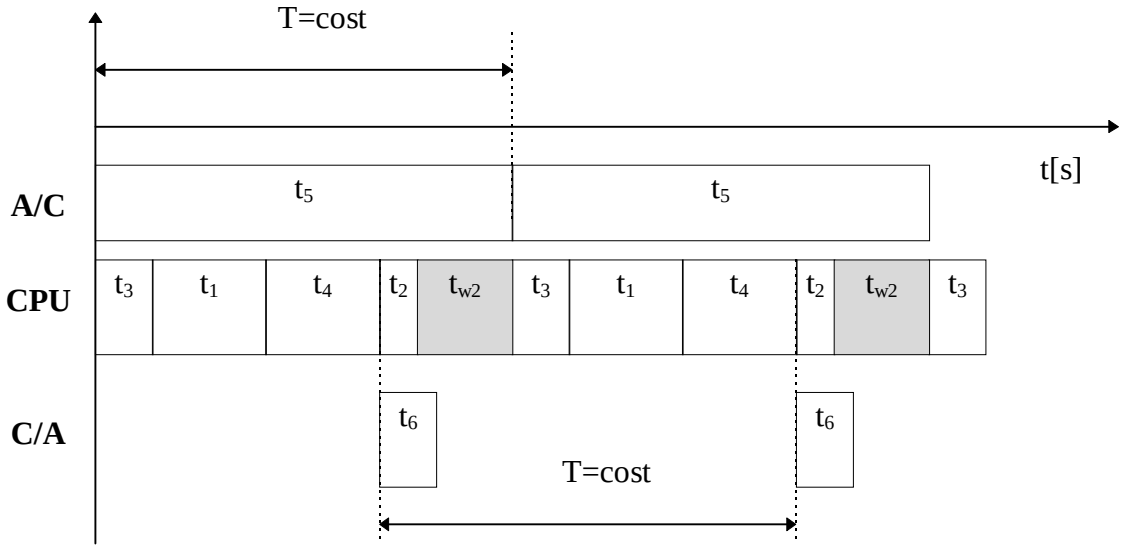
(1.N)

$$T = t_1+t_2+t_3+t_4 = t_5+t_{w1}$$

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

2) tw1=0, tw2≠0

Szybkość przetwarzania CPU dobrana jest z dużym zapasem (dodatkowo mogą być wykonywane inne wątki) w stosunku do A/C i w celu synchronizacji dodawany jest dodatkowy czas oczekiwania t_{w2} - Rys. 1-F .



Rys. 1-F Dodatkowy czas oczekiwania t_{w2} synchronizujący (utrzymujący T=const) proces filtrowania.

Okres próbkowania zależy od szybkości przetwornika A/C i wynosi:

(1.O)

$$T = t_5 = t_1+t_2+t_3+t_4+t_{w2}$$

Można założyć, że suma $t_2+t_3+t_4$ oznaczający cykl obliczenia jednej próbki y(n) jest stały. Zatem

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

(1.P)

$$t_1+t_{w2}=const$$

Zależność (1.P) może byc przydatna podczas syntezy filtru cyfrowego, którego algorytm zakłada, że w pocesie P2 czas t_1 przeznaczony na procedury pomocnicze lub inne wątki jest zmienny (np. zależy od wartości sygnału wejściowego x(t)). Dodanie odpowiedniej nadmiarowości czasowej t_{w2} pozwoli na poprawną pracę filtru.

Powyższe rozważania dotyczyły najprostszego przypadku, kiedy system mikroprocesorowy realizuje tylko algorytm filtrowania, a czasy t_1, t_2, t_3, t_4 są stałe. Zastosowanie buforowania danych przesyłanych pomiędzy A/C, CPU i C/A nie może przyspieszyć przetwarzania. Wynika to bezpośrednio z zależności (1.C), (1.N) i (1.O), a także z zasady działania algorytmu filtru cyfrowego określającego T=const.

Buforowanie przepływu danych pomiędzy systemem (CPU+RAM), a przetwornikami może wykorzystać w następujących przypadkach:

3.
W przypadku filtrów SOI można generować co i-tą wartość y(n) bez zniekształcenia właściwości (transmitancji) filtru cyfrowego, ponieważ y(n) nie zależy od wartości y(n-1), y(n-2), y(n-3), ... (zob. (1.F)).

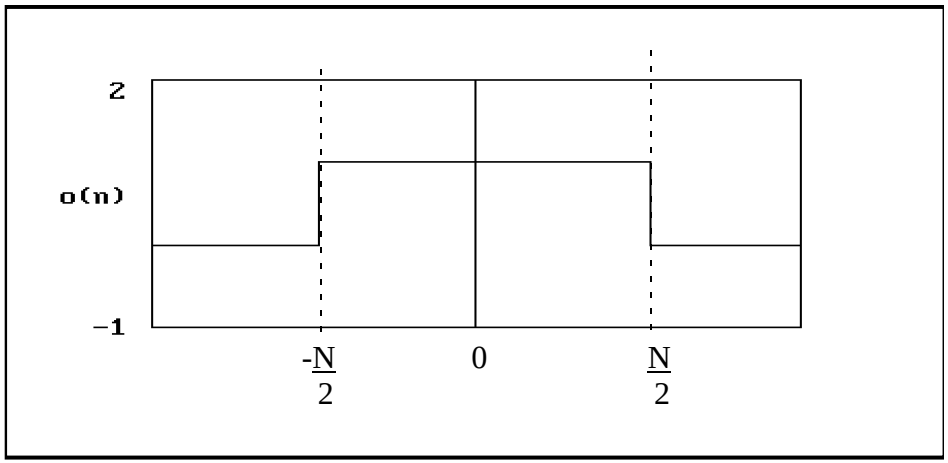
Należy zauważyć fakt, że sygnał y(n) jest opóźniony o jeden okres próbkowania T w stosunku do analogicznego, teoretycznego sygnału analogowego. Opóźnienie to wynika ze skończonego czasu obliczania kolejnej wartości y(n).
Zastosowanie buforowania danych otrzymywanych z A/C będzie dodatkowym opóźnieniem wprowadzający zniekształcenia charakterystyki fazowej. **Opóźnienie to można wykorzystać w syntezie filtru**

2. Działania filtru cyfrowego typu SOI w układzie z buforowaniem danych.

Najbardziej bezpośrednim podejściem do projektowania filtrów SOI jest obcinanie ciągu stanowiącego nieskończoną odpowiedź impulsową. Ograniczenie odbywa się za pomocą „okna czasowego”[2].
Najprostszy filtr SOI ma prostokątne okno pomiarowe przedstawione na Rys. 2-A.

[2] lub „okno pomiarowe”

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl



Rys. 2-A Przykład prostokątnego okna pomiarowego dla.

Poniżej przeprowadzona zostanie analiza tego okna pomiarowego w dziedzinie dyskretnej. Następnie parametry sygnału dyskretnego zostaną powiązane z parametrami sygnału w dziedzinie czasu.

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

2.1 Synteza przykładowego dolnoprzepustowego filtru cyfrowego

Funkcję okna prostokątnego można zapisać za pomocą wyrażenia (1.F), gdzie u(n) oznacza skok jednostkowy zgodnie z wyrażeniem **(1.G)**.

(2.A)

$$o(n) = u(n+N) - u(n-1)$$

(2.B)

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

Z godnie z dyskretnym przekształceniem Fouriera charakterystykę częstotliwościową okna (1.F) opisuje wyrażenie (2.C).

(2.C)

$$O(e^{i\omega}) = \sum_{n=0}^{N-1} e^{-j\omega n} = \frac{1-e^{-j\omega N}}{1-e^{-j\omega}} = e^{-j\omega\frac{(N-1)}{2}} * \frac{\sin\left(\frac{\omega N}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}$$

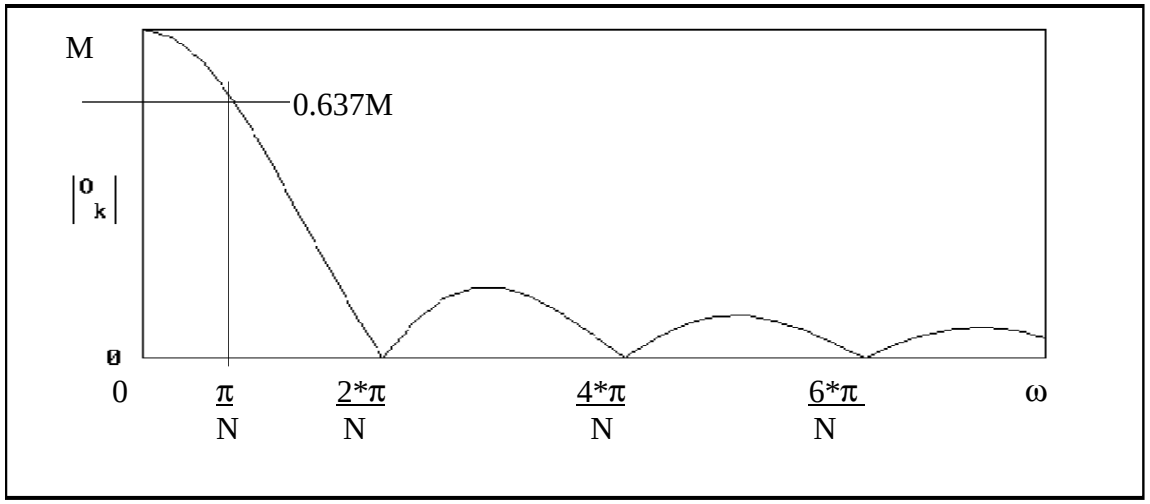
Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Moduł funkcji $O(e^{j\omega})$ został wykreślony na Rys. 2-B. Na rysunku tym łatwo można zauważyć następujące cechy:

- widoczne są: listek główny (od $-2\pi/N$ do $2\pi/N$) oraz listki boczne
- listek główny może stanowić transmitancję filtru dolnoprzepustowego
- przy wzrastającym N szerokość listka głównego zmniejsza się
- przy wzroście N wzrasta amplituda listka głównego, oraz amplitudy listków bocznych w ten sposób, że pole pod każdym z listków jest stałe podczas, gdy szerokość maleje)
- przesunięcie fazowe jest funkcją liniową ω:

(2.D)

$$Q(\omega) = -\omega \cdot \frac{N-1}{2}$$



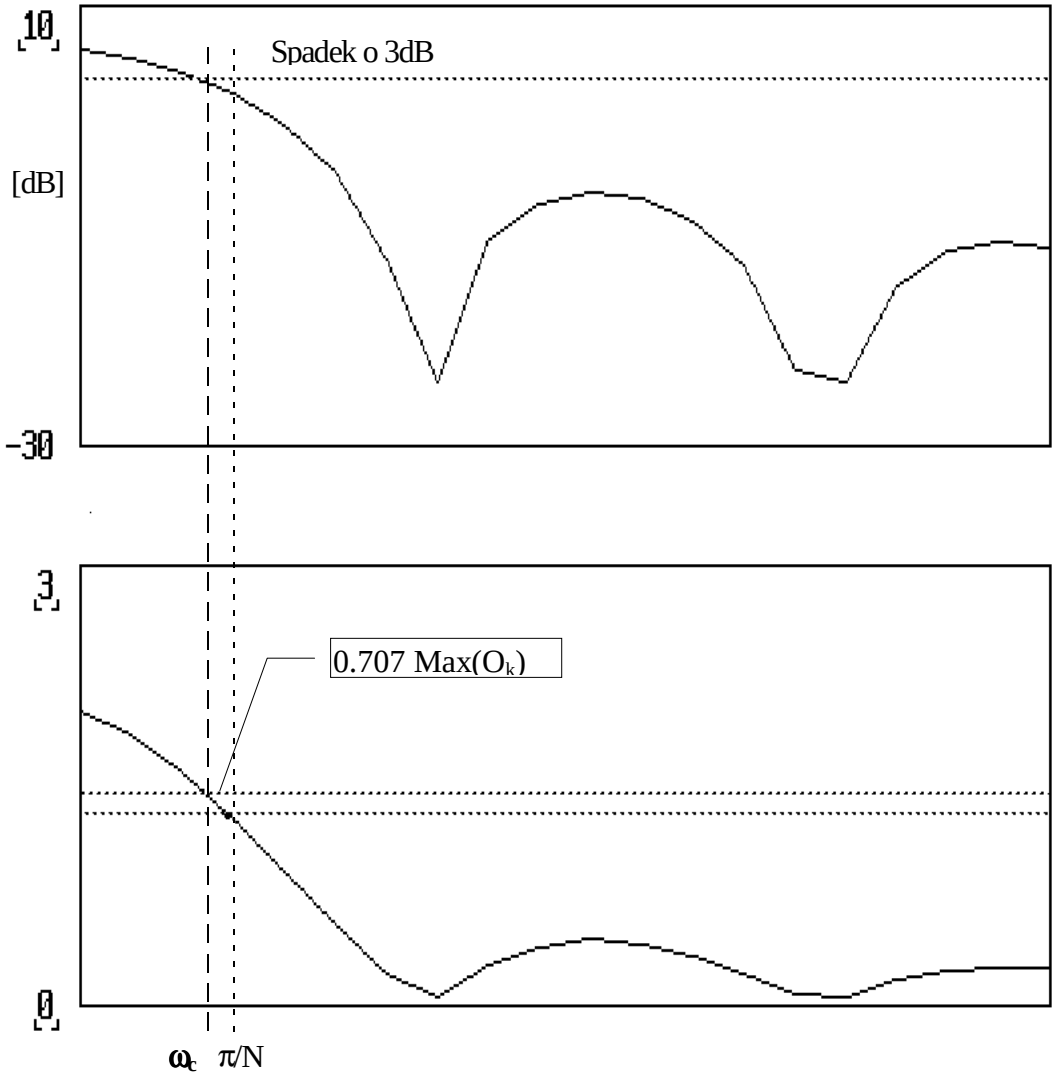
Rys. 2-B Moduł transformaty Fouriera okna prostokątnego dla N-próbek.

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Idealnym filtrem dolnoprzepustowym jest układ, którego charakterystyka częstotliwościowa jest prostokątem. Jednak odpowiedź impulsowa takiego filtru jest nieskończona. Z tego powodu stosuje się charakterystyki rozmyte powstałe poprzez ucięcie odpowiedzi impulsowej np. przy pomocy wyżej zdefiniowane okna pomiarowego.

W celu znalezienia częstotliwości pasma przepuszczania naszego filtru można zbadać logarytmiczną charakterystykę częstotliwościową przedstawioną na Rys. 2-C. Na rysunku tym zaznaczono położenie częstotliwości granicznej ω_c pasma przepustowego wynikającej ze spadku wzmocnienia o 3dB. Tłumienie w paśmie zaporowym określa listek boczny. Minimalne tłumienie wynosi -13dB. Z niewielkim błędem można przyjąć za częstotliwość graniczną punkt π/N. Wzmocnienie dla $\omega = 0$ wynosi N.

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl



Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl

Rys. 2-C Charakterystyki częstotliwościowe z zaznaczonym pasmem przepustowym.

Następnym krokiem analizy jest powiązanie częstotliwości dziedziny dyskretnej ω z częstotliwością sygnału ciągłego za pomocą zależności (2.E):

(2.E)

$$\omega_c = \Omega_c * T_p$$

i wyznaczyć ilość próbek w oknie pomiarowym N:

(2.F)

$$N = \frac{\pi}{W_C \cdot T_P} = \frac{\pi \cdot f_P}{W_C} = \frac{f_P}{2 \cdot f_C}$$

Dobierając parametry f_c oraz f_p należy pamiętać o częstotliwości Nyquista.

Algorytm filtru jest określony poprzez splot sygnału wejściowego x(n) z oknem pomiarowy o(n):

(2.G)

$$y(n) = \frac{1}{N} \cdot \sum_{k=-\infty}^{+\infty} o(k)x(n-k) = \frac{1}{N} \cdot \sum_{k=0}^{N-1} x(n-k)$$

Algorytm taki wymaga jedynie N sumowań. współczynnik 1/N oznacza kompensację wzmocnienia co może być łatwo uzyskane poprzez obcięcie najmniej znaczących bitów wyniku sumowania.
Liczbę sumowań można łatwo zredukować stosując rekursywny zapis algorytmu (2.G) poprzez zastąpienie składników sumy wartością y(n-1). Otrzymany zapis rekursywny przedstawia wyrażenie (2.H).

(2.H)

$$y(n) = \tfrac{1}{N} \cdot (y(n-1) + x(n) - x(n-N))$$

Tak prostą postać algorytmu filtru, jaką przedstawia wyrażenie (2.H), można uzyskać jedynie dla okna prostokątnego. Dla innych kształtów okien (np. okno: Hamminga, Blackmana, itd.) współczynniki a_k są różne od jedności i oprócz N operacji sumowania konieczne jest wykonanie N operacji mnożenia (zob. wyrażenie (1.F)).

W przypadku okna prostokątnego każda kolejna próbka x(n-k) ma zawsze wagę a_k=1, dlatego strumień danych x(n) można buforować w postaci sumy (zob. **Błąd! Nie można odnaleźć źródła odwołania.**]).

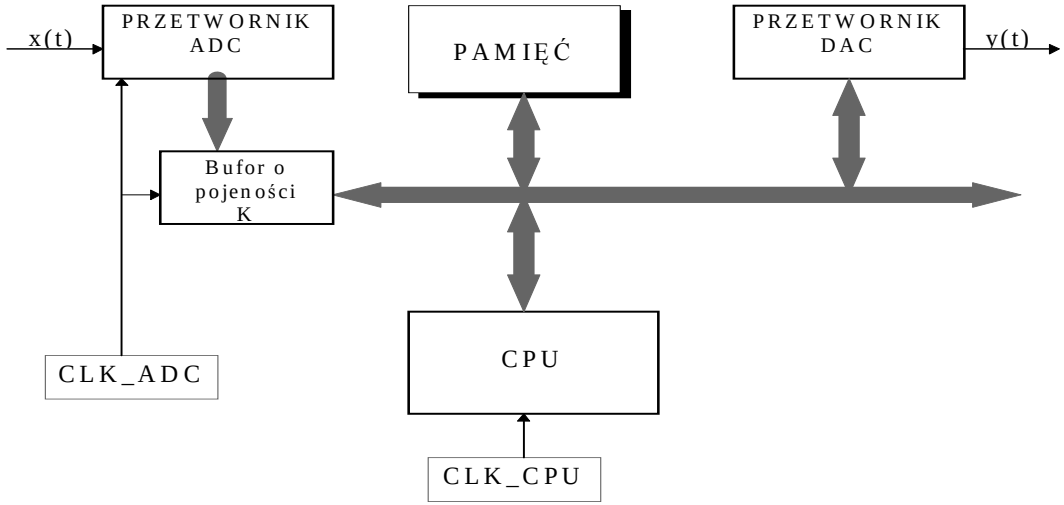
2.2 Zastosowanie buforowania kolejnych próbek pomiarowych

W punkcie 0 przedstawione zostały ogólne zasady budowy i działania systemów mikroprocesorowych realizujących algorytmy filtrów cyfrowych. Z analizy wynika fakt, że bardzo istotnym elementem współpracy CPU i przetworników jest

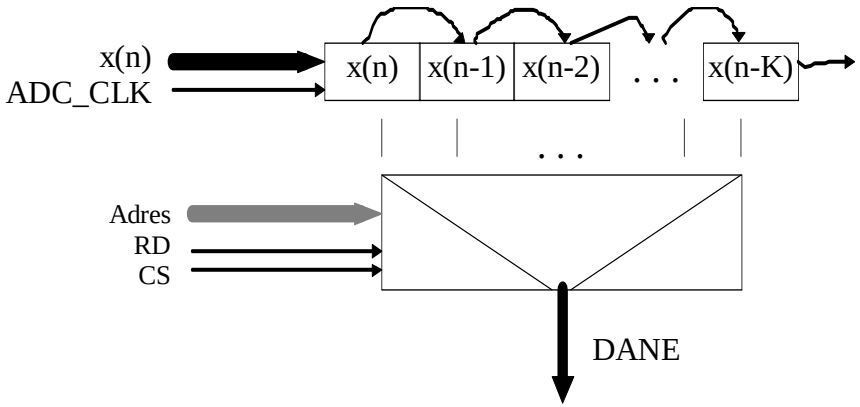
moment pobrania, obliczenia oraz przesłania kolejnych próbek sygnału wejściowego x(n) oraz wyjściowego y(n) - synchronizacja przesyłania danych.

W przypadku filtru SOI, którego przykładową syntezę pokazano w punkcie 2.1, można zastosować buforowanie danych x(n) umożliwiające bardziej efektywne wykorzystanie systemu mikroprocesorowego.

Na Rys. 2-D przedstawiono schemat blokowy z zaznaczonym elementem buforującym wyniki pomiaru przetwornika A/C x(n). Bufor jest układem o wpisie potokowym i odczycie z dowolnej pozycji. Schemat działania tego bufora został przedstawiony na Rys. 2-E. Kolejne próbki wpisywane są do bufora z czętotliwością $1/T_p$ (T_p okres przetwarzania przetwornika A/C), niezależnie od fazy algorytmu (wątku) wykonywanego przez CPU. Dzięki temu sygnał x(t) jest próbkowany ze stałą częstotliwością.



Rys. 2-D Buforowanie danych przesyłanych z przetwornika A/C.

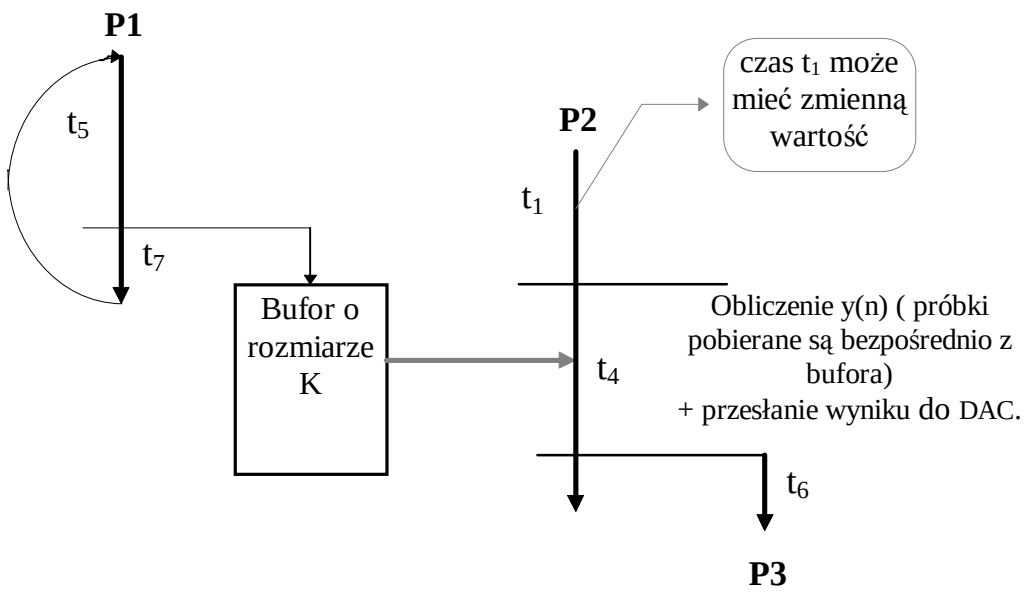


Rys. 2-E Schemat działania układu buforującego.

W takiej organizacji bufora, dane x(n) są automatycznie porządkowane i kompletowane bez udziału CPU. W dowolnej chwili CPU może odczytać kolejne próbki i wyznaczyć wartość y(n) zgodnie z wzorem (1.F). Punkty synchronizacyjne (odczyt i zapis danych) nie są już tak krytyczne jak w układzie z Rys. 1-B, ponieważ o zachowanie włściwości filtru „dba” potok bufora. Praca przetwornika nie jest zakłócana nieodebraniem danych t_{w1}=0.
Ze strony CPU, zniwelowana została konieczność utrzymywania synchronizacji odczytu A/C -t_{w2}=0.
Przepływ sterowania takiego układu został przedstawiony na Rys. 2-F. Proces P1 jest całkowicie niezależny, a czas wykonania procesu P2 wynosi:

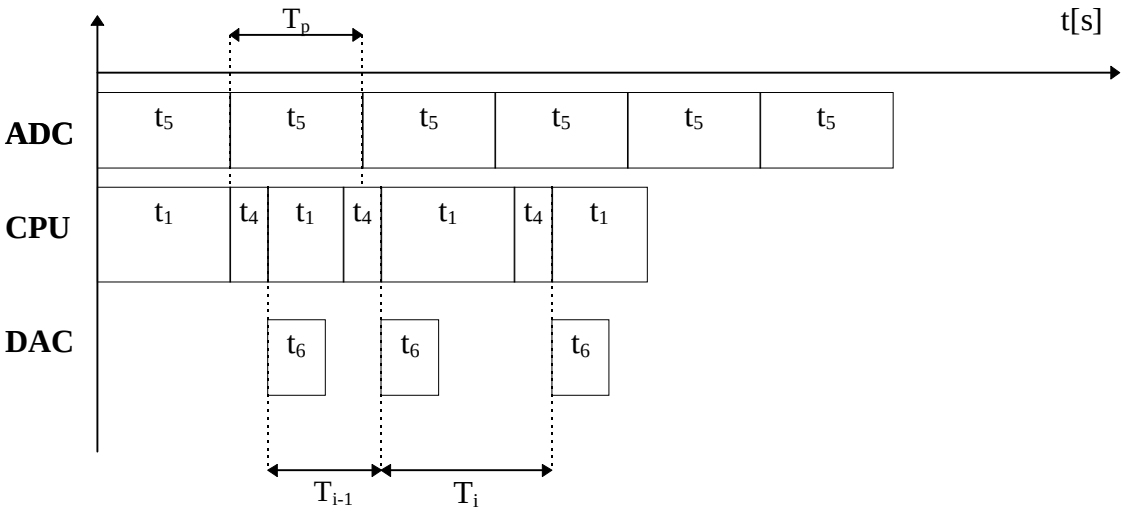
(2.I)

$$T= t_1+t_4$$



Rys. 2-F Schemat przepływu sterowania dla układu zawierającego pamięć buforową.

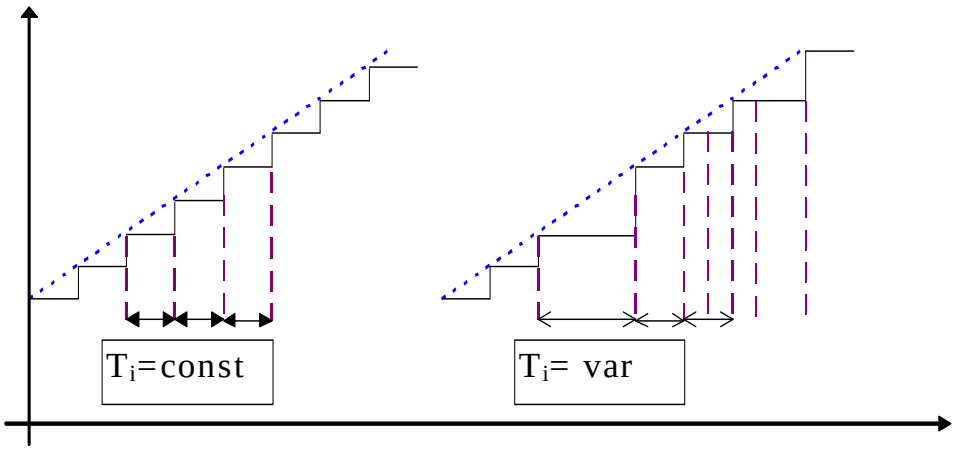
Ze względu na specyfikę algorytmu filtru SOI, nie ma konieczności obliczanie każdej kolejnej próbki y(i). Odpowiada to sytuacji, gdy (zob. Rys. 1-E): $\mathbf{T_i > T_p}$.
Z drgiej strony, gdy: $\mathbf{T_i < T_p}$ kolejne próbki y_i i y_{i+1} mogą mieć tą samą wartość, gdy dane zapisane w buforze nie zostały zmienione w czasie obliczania y_i i y_{i+1}.
Wystąpienie obydwu sytuacji, Rysunek 2-G, nie powoduje znaczącego zniekształcenia y(n).
Jeżeli t_1 będzie czasem wykonania procedur pomocniczych, których czas trwania zależy od zdarzeń oraz bieżących danych wejściowych, czas T_i (zob. (2.I)), będzie różny w kolejnych chwilach i.



Rysunek 2-G Współpraca przetworników A/C i C/A oraz CPU wukładzie z buforowaniem danych.

Zmiany Ti w określonych granicach będą powodować określone zniekształcenia sygnału wyjściowego y(n). Można tak dobrać zakres zmian T_i, aby zniekształcenia były nieistotne dla całego układu. Rysunek 2-H przedstawia przykładowy przebiegi y(n) dla Ti=const oraz Ti=var występujące bezpośrednio na wyjściu przetwornika C/A. Sygnał y(n) jest postacią skwantowanego sygnału y(t).

Andrzej Zeja, U.I . INFO-KAM, andrzej@infokam.kielce.pl



Rysunek 2-H Przebiegi sygnałów wyjściowych z przetwornika C/A dla układu z T_i=const oraz t_i=var.

Jak wcześniej wspomniano, dokładność odtworzenia x(n) zależy od T_p. W układzie z buforowaniem, dodatkowo można dobierać wartość T_i, czyli odstęp pomiędzy kolejnymi próbkami y(n).

3. Literatura

[A] A.V. Oppenheim, R.W.Schafer:*Cyfrowe przetwarzanie sygnałów*. (tłum. z ang.) WKŁ 1979 Warszawa.

[B] A. Wiszniewski: *Algorytmy pomiarów cyfrowych w automatyce elektroenergetycznej.*, WNT 1990 Warszawa.