

Agnieszka Dąbrowska^{*, **}, Kazimierz Wiatr^{*, **}

Implementacja procesu kwantyzacji w strukturach FPGA dla potrzeb kompresji obrazu

1. Wprowadzenie

Kwantyzacja jest procesem, dzięki któremu można zmniejszyć liczbę informacji reprezentujących obraz, bez znaczącej utraty jakości kompresowanego obrazu. Niestety jest to stratna metoda kompresji, a co za tym idzie, odtworzony obraz może nie być identyczny z oryginałem. Proces kwantyzacji pozwala na redukcję liczby bitów potrzebnych do reprezentacji wartości pikseli kompresowanego obrazu.

Ze względu na rodzaj danych wejściowych i wyjściowych wyróżnia się:

- kwantyzację skalarną,
- kwantyzację wektorową.

Kwantyzacja skalarna jest wykorzystywana w popularnych standardach kompresji obrazu, do których zaliczyć można między innymi: JPEG, H.261, H.263, MPEG-1, MPEG-2.

W powyższych standardach kwantyzacji poddawane są współczynniki otrzymane z dwuwymiarowej dyskretnej transformacji kosinusowej. Każdy współczynnik otrzymany z 2D-DCT jest dzielony przez osobny współczynnik kwantyzacji. Tak otrzymana wartość jest zaokrąglana do najbliższej liczby całkowitej.

Proces kwantyzacji charakteryzuje się tym, że z większą dokładnością odwzorowywane są współczynniki DCT reprezentujące niższe częstotliwości, a współczynniki reprezentujące wyższe częstotliwości są odwzorowywane z mniejszą dokładnością. Taki sposób odwzorowania jest zgodny z charakterystyką czułości ludzkiego oka bardziej wyczulonego na składowe niższych częstotliwości.

2. Kwantyzacja w MPEG-2

W MPEG-2 wykorzystywana jest tzw. zmienna kwantyzacja skalarna [1, 2, 3], która opiera się na skalowaniu za pomocą parametru *quantizer_scale*, który może być określany

* Katedra Elektroniki, Akademia Górniczo-Hutnicza, Kraków

** ACK CYFRONET AGH, Kraków

indywidualnie dla każdego obrazu, warstwy czy nawet makrobloku. Taka kwantyzacja pozwala na osiągnięcie większego stopnia kompresji, jak również pomaga utrzymać stałą prędkość bitową zarówno kodera, jak i dekodera. Parametr *quantizer_scale* jest określany na podstawie dwóch parametrów transmitowanych w nagłówku obrazu, warstwy lub makrobloku:

- 1) *q_scale_type*,
- 2) *quantizer_scale_code*.

Jeżeli *q_scale_type* = 0, to wtedy parametr *quantizer_scale* jest określany na podstawie wyrażenia

$$\text{quantizer_scale} = 2 \times \text{quantizer_scale_code} \quad (1)$$

W przypadku gdy parametr *q_scale_type* = 1, *quantizer_scale* jest określany na podstawie tabeli 1.

Tabela 1
Relacja pomiędzy parametrem *quantizer_scale* a parametrem *quantizer_scale_code*

<i>quantizer_scale_code</i>	<i>quantizer_scale</i>	<i>quantizer_scale_code</i>	<i>quantizer_scale</i>
0	Zarezerwowane	16	24
1	1	17	28
2	2	18	32
3	3	19	36
4	4	20	40
5	5	21	44
6	6	22	48
7	7	23	52
8	8	24	56
9	10	25	64
10	12	26	72
11	14	27	80
12	16	28	88
13	18	29	96
14	20	30	104
15	22	31	112

W standardzie MPEG-2 rozróżniane są dwa tryby kodowania:

- 1) kodowanie wewnątrzobrazowe,
- 2) kodowanie międzyobrazowe.

W obu przypadkach definicja procesu kwantyzacji jest inna. W trybie międzyobrazowym współczynniki dyskretnej transformaty kosinusowej [4] są poddane kwantyzacji opisanej wyrażeniem

$$NonIntraq_{ij} = \frac{32 \times y_{ij}}{2 \times quantizer_scale \times NonIntraQ_{ij}} \quad (2)$$

gdzie:

- $NonIntraq_{ij}$ – współczynniki DCT po kwantyzacji w trybie międzyobrazowym,
- $NonIntraQ_{ij}$ – elementy macierzy kwantyzacji trybu międzyobrazowego,
- y_{ij} – współczynniki DCT,
- $quantizer_scale$ – parametr skalowania.

Dla trybu wewnątrzobrazowego proces kwantyzacji współczynników AC dyskretnej transformaty kosinusowej jest opisany wyrażeniem

$$Intraq_{ij} = \frac{32 \times y_{ij}}{2 \times quantizer_scale \times IntraQ_{ij}} \quad \text{dla } i, j \neq 0 \quad (3)$$

gdzie:

- $Intraq_{ij}$ – współczynniki DCT po kwantyzacji w trybie wewnątrzobrazowym,
- $IntraQ_{ij}$ – elementy macierzy kwantyzacji trybu wewnątrzobrazowego.

Współczynnik DC jest to współczynnik y_{00} otrzymany w wyniku dwuwymiarowej dyskretnej transformacji kosinusowej. Krok kwantyzacji w przypadku współczynników DC otrzymanych z 2D-DCT [5] w trybie wewnątrzobrazowym jest uzależniony od precyzji. Precyzja DC oznacza liczbę bitów użytych do reprezentacji współczynników DC po procesie kwantyzacji.

W standardzie MPEG-2 dopuszczone są trzy rodzaje precyzji:

- 1) 8-bitowa,
- 2) 9-bitowa,
- 3) 10-bitowa.

Współczynniki DC w trybie wewnątrzobrazowym są poddane procesowi kwantyzacji zgodnie z zależnością

$$Intraq_{00} = \frac{y_{00}}{k} \quad (4)$$

gdzie:

- y_{00} – współczynnik DC transformacji 2D-DCT,
- $Intraq_{00}$ – współczynnik DC 2D-DCT po kwantyzacji w trybie wewnątrzobrazowym,
- $k = 8$ dla 8-bitowej precyzji,
- $k = 4$ dla 9-bitowej precyzji,
- $k = 2$ dla 10-bitowej precyzji.

W standardzie MPEG-2 są określone typowe macierze kwantyzacji dla współczynników transformacji, w zależności od trybu kodowania, uwzględniające własności systemu wizyjnego człowieka. Macierze te obowiązują zarówno dla współczynników DCT luminancji, jak i współczynników DCT chrominancji. Wyrażenie (5) przedstawia macierz kwantyzacji dla trybu wewnątrzobrazowego, a wyrażenie (6) macierz trybu międzyobrazowego.

$$\begin{bmatrix} 8 & 16 & 19 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 49 & 34 & 37 \\ 19 & 22 & 26 & 27 & 29 & 34 & 34 & 38 \\ 22 & 22 & 26 & 27 & 29 & 34 & 37 & 40 \\ 22 & 26 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 58 \\ 26 & 27 & 29 & 34 & 38 & 46 & 56 & 69 \\ 27 & 29 & 35 & 38 & 46 & 56 & 69 & 83 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix} \quad (6)$$

3. Implementacja

Proces kwantyzacji został zaimplementowany w układzie FPGA [8, 9] przy następujących założeniach dotyczących parametrów q_scale_type , $quantizer_scale_code$ oraz $quantizer_scale$:

- Jeżeli $q_scale_type = 0$ oraz $quantizer_scale_code = 8$, to zgodnie z zależnością (1) $quantizer_scale = 16$.
- Jeżeli $q_scale_type = 1$ oraz $quantizer_scale_code = 12$, to zgodnie z tabelą 1 $quantizer_scale = 16$.

Przy powyższych założeniach wyrażenia (2) oraz (3) można uprościć odpowiednio do postaci (7) oraz (8):

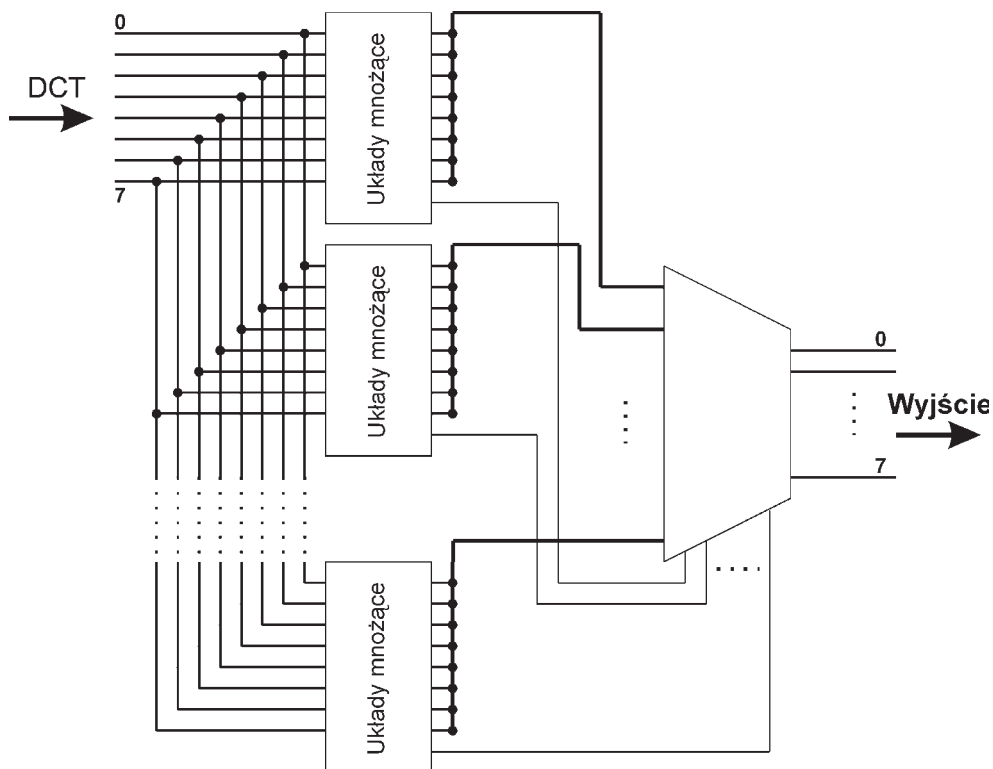
$$NonIntraq_{ij} = \frac{y_{ij}}{NonIntraQ_{ij}} \quad (7)$$

$$Intraq_{ij} = \frac{y_{ij}}{IntraQ_{ij}} \quad \text{dla } i, j \neq 0 \quad (8)$$

Parametry q_scale_type , $quantizer_scale_code$, jak również $quantizer_scale$, nie mają wpływu na sposób kwantowania współczynnika DC w trybie wewnątrzobrazowym. Jedy-
nym parametrem warunkującym w tym trybie kwantyzację DC jest precyzja reprezentacji
skwantowanego współczynnika. Dla potrzeb implementacji założono, że parametr k wyni-
kający z precyzji będzie równy 8. Jest to równoznaczne z przyjęciem 8-bitowej repre-
zentacji skwantowanego współczynnika y_{00} otrzymanego z dwuwymiarowej dyskretnej trans-
formacji kosinusowej. Wynikiem powyższego założenia jest przekształcenie wyrażenia (4)
do wyrażenia

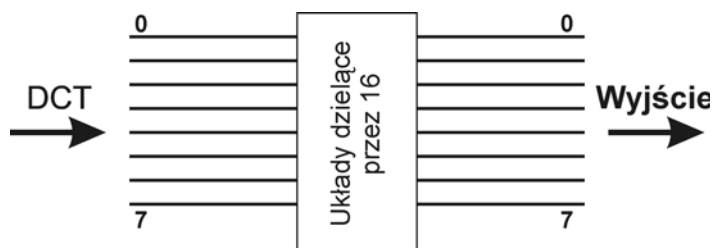
$$Intraq_{00} = \frac{y_{00}}{8} \quad (9)$$

Schematy ideowe zaimplementowanych kwantyzatorów trybu wewnątrzobrazowego
i trybu międzyobrazowego zostały przedstawione na rysunkach 1 oraz 2.



Rys. 1. Schemat ideowy kwantyzatora dla trybu wewnątrzobrazowego

Do realizacji operacji dzielenia zostały wykorzystane układy mnożące bezmnożne MM [6, 7] opierające się na algorytmach CSD, SS oraz CSD-SS.



Rys. 2. Schemat ideowy kwantyzatora dla trybu międzyobrazowego

W proponowanym rozwiązaniu kwantyzacji poddawane jest jednocześnie 8 współczynników 2D-DCT. Konsekwencją tego będzie pojawianie się na wyjściu kwantyzatora, co cykl zegara, 8 skwantowanych współczynników DCT.

Implementacja kwantyzatora w układzie XCV200BG532(-6) firmy Xilinx zajmuje 1735 bloków SLICE (73% zasobów układu). Na taką zajętość układu składa się 649 wykorzystanych przerzutników Flip-Flop oraz 3003 bloków LUT (63% dostępnych bloków LUT w układzie), z czego 2664 bloki zajmuje logika, natomiast 339 jest wykorzystanych jako zasoby połączeniowe. Maksymalna osiągnięta częstotliwość dla tej implementacji wynosi 94,616MHz.

Jeśli proponowane rozwiązanie byłoby zaimplementowane w układzie firmy Xilinx XCV2P125FF1704, to zostałyby wykorzystanych 1755 bloków SLICE, co stanowi 3% zasobów danego układu. Na takie wykorzystanie składa się 648 przerzutników typu Flip-Flop oraz 3042 bloków LUT (2% dostępnych bloków układu), z czego 2665 bloków wykorzystywałaby logika, a 377 byłoby wykorzystanych jako zasoby połączeniowe. Maksymalna osiągnięta częstotliwość dla tej implementacji wynosiłaby 232,829 MHz

4. Podsumowanie

Zaletą zaimplementowanego kwantyzatora jest możliwość jednoczesnego kwantowania 8 współczynników otrzymanych z 2-wymiarowej dyskretnej transformacji kosinusowej DCT. Na wyjściu wyniki pojawiają się po każdym cyklu zegara, zatem przetworzenie wszystkich 64 współczynników otrzymanych z transformacji 2D-DCT bloku o rozmiarach 8×8 pikseli zajmuje 8 cykli zegara od momentu pojawienia się na wyjściu pierwszych ośmiu wartości. Niestety, konsekwencją takiego rozwiązania jest większe wykorzystanie zasobów układu niż dla rozwiązania, w którym przetwarzany jest tylko pojedynczy współczynnik otrzymany z 2D-DCT.

Implementacja kwantyzatora zajmuje 39 422 bramki, co przekłada się na zajętość zasobów układu FPGA, odpowiednio: dla XCV200BG352 jest to 73% wszystkich zasobów, a dla XCV2P125FF1704 jest to 2%. Pozostałe parametry implementacji zostały przedstawione w tabeli 2.

Tabela 2
Parametry implementacji kwantyzatora w układzie XCV200BG352 oraz XCV2P125FF1704

Układ	Kwantyzacja	
	XCV200BG352	XCV2P125FF1704
SLICE	1729 (73%)	1755 (3%)
LUT	3003 (63%)	3042 (2%)
Częstotliwość [MHz]	94,616 (10,569 ns)	232,829 (4,295 ns)

Dalsze plany dotyczące implementacji procesu kwantyzacji dotyczą uwzględnienia zmienności parametrów q_scale_type , $quantizer_scale_code$, a co za tym idzie, również $quantizer_scale$ oraz precyzji odwzorowania skwantowanych współczynników DC.

Literatura

- [1] Skarbek W.: *Multimedia. Algorytmy i standardy kompresji*. Warszawa, Akademicka Oficyna Wydawnicza PLJ 1998
- [2] Haskel B.G., Puri A., Netravali A.N.: *Digital Video: An introduction to MPEG-2*. New York, Chapman and Hall 1997
- [3] Latha Pillai: *Quantization*. www.xilinx.com/bvdocs/appnotes/xapp615.pdf, 2003
- [4] Dąbrowska A., Wiatr K.: *Fast-DCT Modified Algorithms Implemented in FPGA Chips for Real-Time Image Compression*. Proc. of IFAC Workshop Programmable Devices and Systems, Silesian University, Cracow 2004
- [5] Dąbrowska A., Wiatr K.: *Implementacja algorytmów transformacji FDCT w strukturach FPGA*. Mat. VII Konferencji Reprogramowalne Układy Cyfrowe, Politechnika Szczecińska, Szczecin 2005
- [6] Wiatr K.: *Sprzętowe implementacje algorytmów przetwarzania obrazów w systemach wizyjnych czasu rzeczywistego*. Kraków, UWND AGH 2002
- [7] Wiatr K., Jamro E.: *Implementation of Multipliers in FPGA Structure*. Proc. of the IEEE Int. Symp. on Quality Electronic Design, Los Alamitos CA, IEEE Computer Society, 2001, pp. 415–420
- [8] www.xilinx.com
- [9] Pasierbiński J., Zbysiński P.: *Układy programowalne w praktyce*. Wyd. 2, Warszawa, Wydawnictwa Komunikacji i Łączności 2002

