SUB-PIXEL ACCURATE H.264 MOTION ESTIMATION HARDWARE DESIGN

by

SERKAN ÖKTEM

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
Spring 2007

SUB-PIXEL ACCURATE H.264 MOTION ESTIMATION
HARDWARE DESIGN

APPROVED BY:

Assist. Prof. Dr. İlker Hamzaoğlu …………………………….

(Thesis Supervisor)

Assist. Prof. Dr. Ahmet Onat …………………………….

Assist. Prof. Dr. Cem Öztürk …………………………….

Assist. Prof. Dr. Hakan Erdogan …………………………….

Assist. Prof. Dr. Müjdat Çetin …………………………….

DATE OF APPROVAL: …………………………….

SUB-PIXEL ACCURATE H.264 MOTION ESTIMATION
HARDWARE DESIGN

Serkan Oktem

EECS, Master Thesis, 2007

Thesis Supervisor: Assist. Prof. Dr. İlker Hamzaoğlu

ABSTRACT

The new international standard for video compression named H.264 / MPEG-4 Part 10 offers significantly better video compression efficiency than previous international standards. Integer-pixel motion estimation is the most compute-intensive part of an H.264 video encoder. In order to increase the performance of integer-pixel motion estimation, sub-pixel (half-pixel and quarter-pixel) accurate variable block size motion estimation is performed. In this thesis, we developped an efficient hardware architecture for real-time implementation of sub-pixel accurate variable block size ME for H.264 video coding standard. We have considered several alternative designs and decided on this architecture based on a cost/performance analysis. The proposed hardware includes novel half-pixel and quarter-pixel interpolation and search hardwares designed for each block size. In the proposed hardware, half-pixel interpolation hardwares are shared by half-pixel search hardwares for reducing area. The proposed design performs quarter-pixel interpolation dynamically for reducing the amount of computation performed for quarter-pixel interpolation and therefore reducing the power consumption. This hardware is designed to be used as part of a complete H.264 video coding system for portable applications. The proposed hardware architecture is implemented in Verilog HDL. The Verilog RTL code is verified to work at 60 MHz in a Xilinx Virtex II FPGA. The FPGA implementation can process 34 VGA frames (640x480) per second.

H.264 ALT-PİKSEL HAREKET TAHMİNİ DONANIM TASARIMI

Serkan Öktem

EECS, Yüksek Lisans Tezi, 2007

Tez Danışmanı: Yard. Doç. Dr. İlker Hamzaoğlu

ÖZET

Yakın tarihte geliştirilmiş uluslararası bir standart olan H.264 / MPEG4 Part 10, kendinden önceki standartlara göre belirgin şekilde daha iyi sıkıştırma verimi sunmaktadır. Tamsayı-piksel hareket tahmini, H.264 video kodlayıcının en fazla işlemsel yoğunluk gerektiren bölümüdür. Tamsayı-piksel hareket tahminin performansını artırmak için, alt-piksel (yarım-piksel ve çeyrek-piksel) değişken blok boyutlu hareket tahmini yapılmaktadır. Bu tez çalışmasında, H.264 video kodlama standardında kullanılan, alt-piksel değişken blok boyutlu hareket tahminini gerçek-zamanlı gerçekleştiren donanım mimarisi geliştirilmiştir. Çeşitli alternatif tasarımları göz önünde bulundurduk ve maliyet/performans analizine göre bu tasarımda karar kıldık. Bu donanım mimarisinde her blok boyutu için özgün yarım-piksel ve çeyrek-piksel interpolasyon ve arama donanımları kullanılmaktadır. Bu donanım mimarisinde yarım-piksel interpolasyon donanımları yarım-piksel arama donanımları tarafından ortak kullanılarak alandan kazanılmaktadır. Bu donanım mimarisinde çeyrek-piksel interpolasyonu dinamik olarak gerçekleştirilerek, çeyrek-piksel interpolasyonundaki hesaplamaların azaltılması ve dolayısı ile güç tüketiminin azaltılması sağlanıyor. Bu donanım taşınabilir uygulamalar için kullanılacak bir H.264 video kodlama sisteminin bir parçası olarak kullanılmak üzere tasarlandı. Tasarlanan donanım mimarisi Verilog HDL dili kullanılarak gerçekleştirildi. Verilog RTL kodu bir Xilinx Virtex II FPGA'de 60 MHz'de çalışacak şekilde gerçeklendi. FPGA gerçekleştirmesi saniyede 34 VGA çerçevesi işleyebilmektedir.

To My Grandmother and Grandfather,
To my beloved family…

ACKNOWLEDGEMENTS

I would like to express my deep appreciation to my supervisor, Assist. Prof. Dr. Ilker Hamzaoglu, for his skills, enthusiasm, unconditional support, guidance and patience during the process of this thesis. I appreciate very much for his suggestions, detailed reviews and invaluable advices. I feel myself privileged as his student.

Thanks also to my partner in H.264 research project, Sinan Yalcin, for valuable discussions and feedback throughout the project.

Special thanks to my family including my grandparents for their constant support and encouragement for going through my tough periods with me.

My warmest thanks go to Esra Sahin, for her friendship.

Finally, my acknowledgements go to Sabanci University and TUBITAK for supporting H.264 video encoder hardware design research project.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

ASIC	Application Specific Integrated Circuit
CIF	Common Intermediate Format
CPU	Central Processing Unit
DFF	D Flip Flop
DVD	Digital Versatile Disc
FPGA	Field Programmable Gate Array
HDL	Hardware Description Language
ISO/IEC	International Standards Organization, International Electrotechnical Commission
ITU-T	International Telecommunications Union, Telecommunications Standardization Sector
MB	Macroblock
MPEG	Motion Picture Experts Group
NAL	Network Abstraction Layer
PVT	Process Voltage Temperature
RAM	Random Access Memory
SAD	Sum of Absolute Difference
VCL	Video Coding Layer
VGA	Video Graphics Array

CHAPTER 1

INTRODUCTION

1.1 Motivation

Video compression systems are used in many commercial products, from consumer electronic devices such as digital camcorders, cellular phones to video teleconferencing systems. These applications make the video compression hardware devices an inevitable part of many commercial products. To improve the performance of the existing applications and to enable the applicability of video compression to new real-time applications, recently, a new international standard for video compression has been developed. This new standard, offering significantly better video compression efficiency than previous video compression standards, has been developed with the collaboration of ITU and ISO standardization organizations. Hence it is called with two different names, H.264 and MPEG4 Part 10 [1].

H.264 video coding standard has a much higher coding efficiency potential (capable of saving up to %50 bit rate at the same level of video quality) than the previous standards. Due to its high coding efficiency and due to its flexibility and robustness to different communication environments, in the near future, H.264 is expected to be widely used in many applications such as digital TV, DVD, video transmisson in wireless networks, and video conferencing over the internet.

The H.264 standard includes a Video Coding Layer (VCL), which efficiently represents the video content, and a Network Abstraction Layer (NAL), which formats the VCL representation of the video and provides header information in a manner suitable for transportation by particular transport layers or storage media [2].

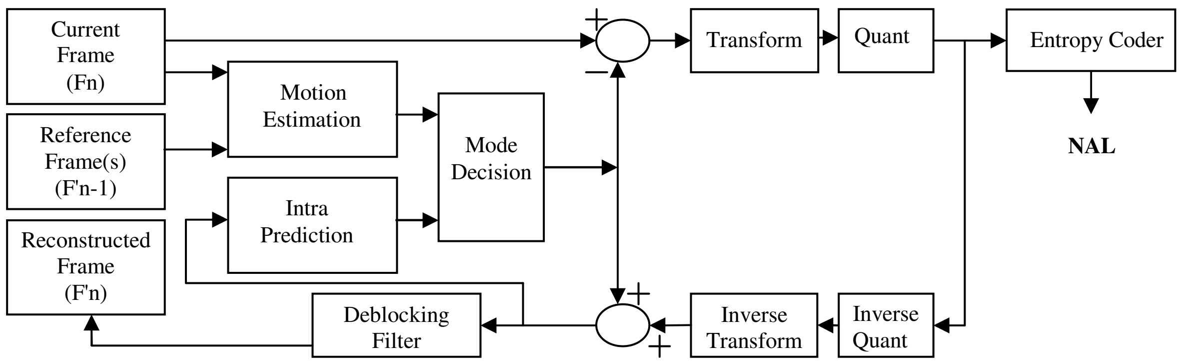


Figure 1.1 H.264 Encoder Block Diagram

The top-level block diagram of an H.264 Encoder is shown in Figure 1.1. As shown in Figure 1.1, an H.264 encoder has a forward path and a reconstruction path. The forward path is used to encode a video frame by using intra or inter predictions and to create the bit stream. The reconstruction path is used to decode the encoded frame and to reconstruct the decoded frame. Since a decoder never gets original images, but rather works on the decoded frames, reconstruction path in the encoder ensures that both encoder and decoder use identical reference frames for intra and inter prediction. This avoids possible encoder – decoder mismatches [2, 3, 4].

Forward path starts with partitioning the input frame into MBs. Each MB is encoded in intra or inter mode depending on the mode decision. In both intra and inter modes, the current MB is predicted from the reconstructed frame. Intra mode generates the predicted MB based on spatial redundancy, whereas inter mode, generates the predicted MB based on temporal redundancy. Mode decision compares the required amount of bits to encode a MB and the quality of the decoded MB for both of these modes and chooses the mode with better quality and bit-rate performance. In either case, intra or inter mode, the predicted MB is subtracted from the current MB to generate the residual MB. Residual MB is transformed using 4x4 and 2x2 integer transforms. Transformed residual data is quantized and quantized transform coefficients are re-ordered in a zig-zag scan order. The reordered quantized transform coefficients are entropy encoded. The entropy-encoded coefficients together with header information, such as MB prediction mode and quantization step size, form the compressed bit stream. The compressed bit stream is passed to network abstraction layer (NAL) for storage or transmission.

Reconstruction path begins with inverse quantization and inverse transform operations. The quantized transform coefficients are inverse quantized and inverse transformed to generate the reconstructed residual data. Since quantization is a lossy process, inverse quantized and inverse transformed coefficients are not identical to the original residual data. The reconstructed residual data are added to the predicted pixels in order to create the reconstructed frame. A deblocking filter is applied to reduce the effects of blocking artifacts in the reconstructed frame.

As illustrated in Figure 1.2, motion estimation is the process of searching a search window in a reference frame to determine the best match for a block in a current frame based on a search criterion such as minimum sum of absolute difference (SAD) [5]. The location of a block in a frame is given using the (x,y) coordinates of top-left corner of the block. The search window in the reference frame is the [-p,p] size region around the location of the current block in the current frame. The SAD value for a current block in the current frame and a candidate block in the reference frame is calculated by accumulating the absolute differences of corresponding pixels in the two blocks as shown in the following formula:

$$\text{SAD}_{B_{mxn}}(\mathbf{d}) = \sum_{x=1,y=1}^{m,n} |c(x,y) - r(x+d_x, y+d_y)|$$

where B_{mxn} is a block of size mxn, d=*(dx, dy)* is the motion vector (MV), *c* and *r* are current and reference frames respectively. Since a motion vector expresses the relative motion of the current block in the reference frame, motion vectors are specified in relative coordinates. If the location of the best matching block in the reference frame is (x+u, y+v), then the motion vector is expressed as (u,v).

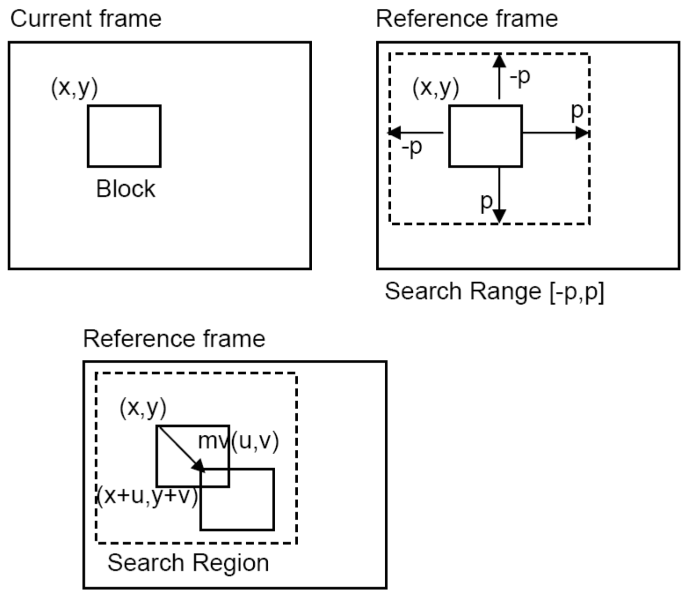


Figure 1.2 Motion estimation process

The motion estimation and motion compensation flows in a video encoder and a video decoder are shown in Figure 1.3 [5]. Motion estimation is performed on the luminance (Y) component of a YUV image and the resulting motion vectors are also used for the chrominance (U and V) components. After the motion vector for a block is determined, the residual block (the difference between the current block and the reference block pointed by the motion vector) is calculated by the motion compensation module. The motion vector and the residual block (I(x,y,t) – I(x-u, y-v, t-1)) are coded in the encoder and transmitted. This process is repeated for all the blocks in the current frame. In the decoder, the motion vector and the residual block are decoded. Then, the reference block in the reference frame pointed by the motion vector (I'(x-u, y-v, t-1)) is determined by the motion compensation module, and it is added to residual block. The resulting reconstructed block is stored in the frame memory and it is used for motion compensation for the next frame. This reconstruction is also done in the encoder in order to ensure that encoder and decoder use identical reference frames for motion compensation [6].

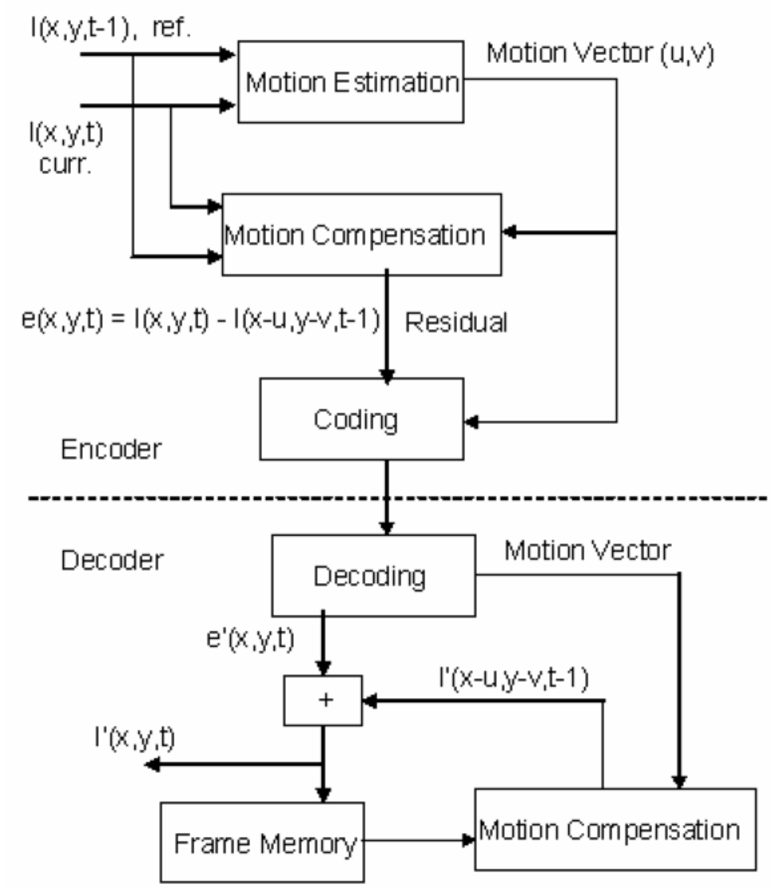


Figure 1.3 Motion Estimation and Motion Compensation Flow

Motion estimation is the most computationally expensive part of the video encoders. In order to increase the performance of integer-pixel motion estimation, sub-pixel (half-pixel and quarter-pixel) accurate variable block size motion estimation is performed [7, 8, 9]. However, the amount of computation required by sub-pixel motion estimation is even more than the amount required by integer-pixel motion estimation. It is shown that sub-pixel motion estimation accounts for about 68% of CPU usage of a H.264 video encoder in fast motion estimation mode which is already 65% faster than full search mode [10]. Therefore, the coding gain obtained by sub-pixel motion estimation comes with an increase in encoding complexity which makes it an exciting challenge to have a real-time implementation of sub-pixel accurate variable block size motion estimation for H.264 video coding.

In this thesis, we present a half-pixel accurate ME hardware for 4x4 block size in Chapter 2 and a quarter-pixel accurate variable block size ME hardware in Chapter 3. We integrated this quarter pixel accurate ME hardware with the half pixel accurate variable block size ME hardware presented in [11] to implement a sub-pixel accurate variable block size ME hardware for H.264 video coding. This hardware is designed to be used as part of a complete H.264 video coding system for portable applications together with the integer-pixel motion estimation hardware presented in [12]. The proposed hardware architecture is implemented in Verilog HDL. The Verilog RTL code is verified to work at 60 MHz in a Xilinx Virtex II FPGA. The FPGA implementation can process 34 VGA frames (640x480) per second.

Several hardware architectures for real-time implementation of sub-pixel accurate variable block size ME for H.264 video coding are presented in the literature [13, 14]. The hardware architecture presented in [13] uses less hardware than our hardware design and has lower performance than our hardware design. The hardware architecture presented in [14] achieves higher performance than our hardware design at the expense of a much higher hardware cost. It uses much more hardware in order to process 30 HDTV frames (1280x720) per second. Our hardware design is a more cost-effective solution for portable applications.

1.2 Thesis Organization

The rest of the thesis is organized as follows.

Chapter 2 explains half-pixel accurate 4x4 block size H.264 motion estimation hardware. First, it introduces half-pixel accurate motion estimation algorithm used in H.264 video coding standard. Then, it describes the designed hardware in detail and presents the implementation results.

Chapter 3 explains quarter-pixel accurate variable block size H.264 motion estimation hardware. First, it introduces quarter-pixel accurate motion estimation algorithm used in

H.264 video coding standard. Then, it describes the designed hardware in detail and presents the implementation results.

Chapter 4 explains sub-pixel accurate variable block size H.264 motion estimation hardware in detail and presents the implementation results.

Chapter 5 presents the conclusions and the future work.

CHAPTER 2

HALF-PIXEL ACCURATE H.264 MOTION ESTIMATION HARDWARE DESIGN

2.1 Overview of Half-Pixel Accurate Motion Estimation Algorithm

The search locations for half-pixel accurate motion estimation are shown in Figure 2.1. First, integer-pixel motion estimation is performed at the integer-pixel search locations and best integer-pixel motion vector (MV) is determined based on a performance metric, e.g. minimum Sum of Absolute Difference (SAD). Then, half-pixel motion estimation is performed at the half-pixel search locations around the best integer-pixel MV with a search range of [-1, 1], and the integer-pixel MV is refined by the best half-pixel accurate MV.

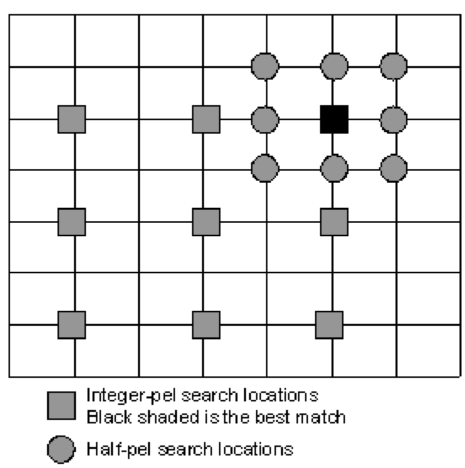


Figure 2.1 Half-Pixel Search Locations

Before searching for the best half-pixel accurate MV, half pixels in the half-pixel search window are interpolated from neighboring pixels using a 6-tap FIR filter with weights (1/32, -5/32, 5/8, 5/8, -5/32, 1/32). First, the half pixels that are adjacent to two integer pixels are interpolated from 6 integer pixels. Then, the remaining half pixels are interpolated from 6 horizontal or 6 vertical half pixels. A half-pixel interpolation example is shown in Figure 2.2. First, the half pixels a, b, c, d, e, f are interpolated from 6 corresponding horizontal integer pixels. For example, half pixel c is interpolated from the 6 horizontal integer pixels A, B, C, D, E, F (c = round ((A-5B+20C+20D-5E+F) / 32)). Then, the half pixels g, h, i, j, k, m are interpolated from 6 corresponding vertical integer pixels. For example, half pixel i is interpolated from the 6 vertical integer pixels M, N, C, I, O, P. Finally, half-pixel n can be interpolated from either horizontal half pixels g, h, i, j, k, m or vertical half-pixels a, b, c, d, e, f.

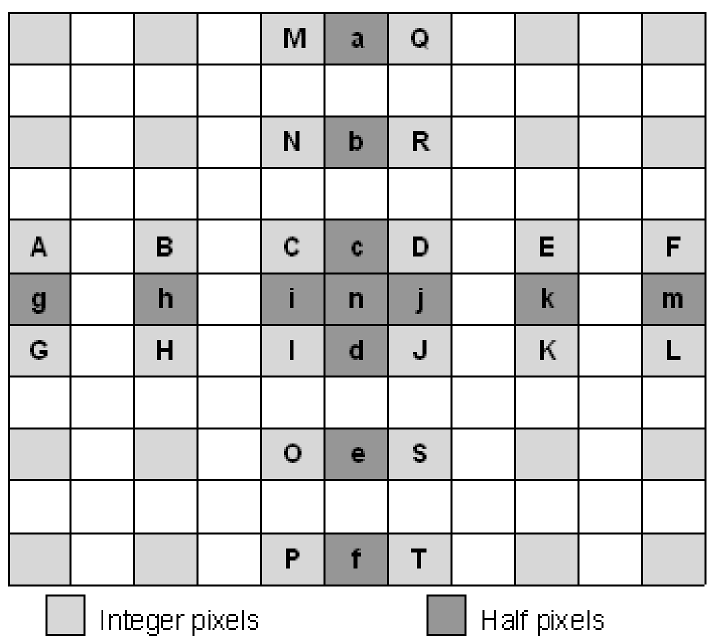


Figure 2.2 Half-Pixel Interpolation Example

2.2 Proposed Hardware Architecture

The proposed half-pixel accurate 4x4 block size motion estimation hardware is shown in Figure 2.3. The hardware is composed of two parts; the upper part performs half-pixel interpolation and the lower part performs half-pixel search. First, half-pixel interpolation hardware calculates the half pixels in the half-pixel search window of a 4x4 block. Then, half-pixel search hardware searches the half-pixel search locations and determines the best half-pixel accurate MV. Half-pixel accurate motion estimation for a 4x4 block takes 54 clock cycles; half-pixel interpolation takes 13 clock cycles and half-pixel search takes 41 clock cycles. Half-pixel accurate 4x4 block size motion estimation for a MB, therefore, takes 16*54=864 clock cycles.

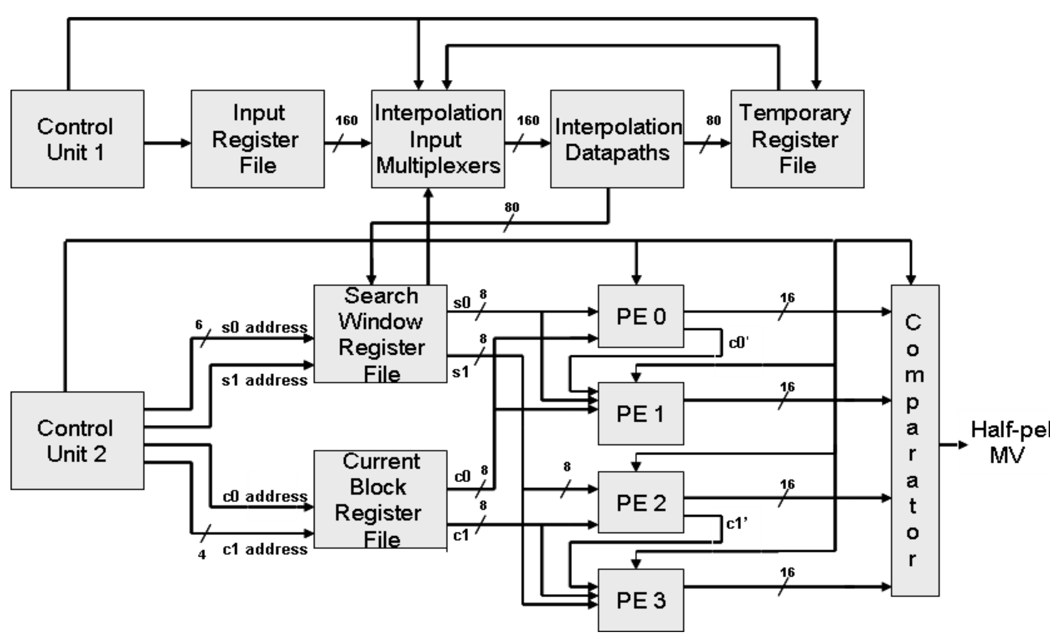


Figure 2.3 Proposed Half Pixel Accurate Motion Estimation Hardware

2.2.1 Half–Pixel Interpolation Hardware

The proposed half-pixel interpolation flow for a 4x4 block is shown in Figure 2.4. The light gray rectangles denote integer pixels (e.g. 0-0) and dark gray rectangles denote half pixels (e.g. A00, B00, C00). The integer-pixel MV for this 4x4 block points to the integer pixel 3-3. The half-pixel search locations around this integer pixel (C00, A03, C01, B01, C11, A13, C10, B00) have to be searched to determine the best half-pixel accurate MV. Therefore, the top-left corner of the half-pixel search window is C00 and the bottom-right corner is C44. There are 9x9 (integer and half pixels) – 4x4 (integer pixels) = 65 half pixels in the half-pixel search window and 100 integer pixels (0-0 to 9-9) are required to calculate these half pixels. The integer pixels are stored in the input register file and the half pixels in the half-pixel search window are stored in the search window register file.

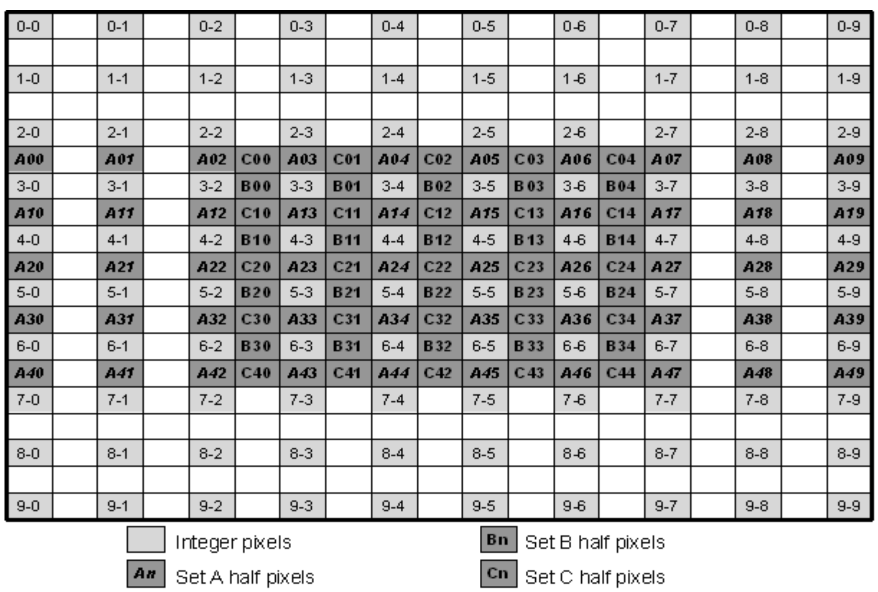


Figure 2.4 Half-Pixel Interpolation Flow

The half pixels are grouped according to their calculation order; first set A half pixels, then set B half pixels and finally set C half pixels are calculated. Set A half pixels are interpolated from 6 corresponding vertical integer pixels. For example, A00 is interpolated from integer pixels 0-0, 1-0, 2-0, 3-0, 4-0, 5-0 and A10 is interpolated from integer pixels 1-0, 2-0, 3-0, 4-0, 5-0, 6-0. Therefore, the first column of set A half pixels (A00 to A40) are calculated using the first column of integer pixels (0-0 to 9-0). The remaining set A half pixels are calculated similarly.

Set B half pixels are interpolated from 6 corresponding horizontal integer pixels. For example, B00 is interpolated from integer pixels 3-0, 3-1, 3-2, 3-3, 3-4, 3-5 and B01 is interpolated from integer pixels 3-1, 3-2, 3-3, 3-4, 3-5, 3-6. Therefore, the first row of set B half pixels (B00 to B04) are calculated using the fourth row of integer pixels (3-0 to 3-9). The remaining set B half pixels are calculated similarly.

Finally, set C half pixels are interpolated from 6 corresponding horizontal set A half pixels. Therefore, in addition to the set A half pixels that are in the search window, the set A half pixels that are required to calculate set C half pixels are also calculated and stored in temporary register file. For example, C00 is interpolated from set A half pixels A00, A01, A02, A03, A04, A05 and C01 is interpolated from set A half pixels A01, A02, A03, A04, A05, A06. Therefore, the first row of set C half pixels (C00 to C04) are calculated using the first row of set A half pixels (A00 to A09). The remaining set C half pixels are calculated similarly.

The proposed half-pixel interpolation datapath is shown in Figure 2.5. The datapath implements the 6-tap FIR filter round ((A-5B+20C+20D-5E+F) / 32). It takes 6 input pixels and calculates the corresponding half pixel. The datapath is pipelined into 2 stages using pipeline registers (P Reg) to increase the clock frequency and interpolation throughput. The multiplications with coefficients 5 and 20 are implemented with shifters and adders instead of multipliers to reduce area. For example, 5X is calculated by 2-bit left shift (4X) and addition (4X+X). The output of the datapath is clipped to range [0-255].

Since one set A half pixel is interpolated from 6 integer pixels, if we use 1 half-pixel datapath, 5 set A half pixels will be interpolated in 5 clock cycles by accessing 30 integer pixels. However, since one column of set A half pixels (5 pixels) can be calculated using one column of integer pixels (10 pixels), if we use 5 half-pixel interpolation datapaths, 5 set A half pixels can be interpolated in 1 clock cycle by accessing 10 integer pixels. This reduces the number of input register file accesses by 3 and the number of clock cycles by 5. We used 10 half-pixel interpolation datapaths to further reduce the clock cycle count.

Therefore, two columns of set A half pixels (10 pixels) are calculated in 1 clock cycle by accessing 20 integer pixels.

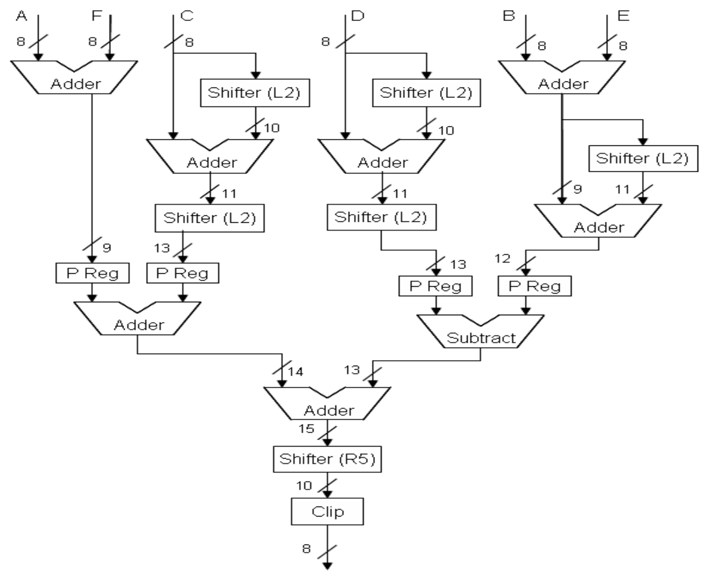


Figure 2.5 Half-Pixel Interpolation Datapath

Similarly, two rows of set B half pixels (10 pixels) are calculated in 1 clock cycle by accessing 20 integer pixels. However, since set C half pixels are interpolated from set A half pixels and accessing two rows of set A half pixels in 1 clock cycle increases the complexity of the register files, one row of set C half pixels (5 pixels) are calculated in 1 clock cycle by accessing one row of set A half pixels (10 pixels).

Since the half-pixel interpolation datapaths access 20 integer pixels in 1 clock cycle, the input register file should have 20 read ports. In order to reduce the area and the read access time of the input register file, instead of using generic read ports, we implemented 20 special purpose read ports. Each special purpose read port only multiplexes the integer pixels that will be read from that read port. Since the half-pixel interpolation datapaths calculate 10 half pixels in 1 clock cycle, the search window register file should have 10 write ports. Similarly, in order to reduce the area and the write access time of the search window register file, instead of using generic write ports, we implemented 10 special purpose write ports. Each special purpose write port only involves the half pixel locations that will be written from that write port.

2.2.1 Half–Pixel Search Hardware

The proposed half-pixel search flow for a 4x4 block is shown in Figure 2.6. c0-c15 are the current block pixels and s0-s80 are the search window pixels. s0, s1, s2, s9, s10, s11, s18, s19, s20 correspond to C00, A03, C01, B00, 3-3, B01, C10, A13, C11 in Figure 2.4. The SAD value for a search location is calculated by a processing element in 16 clock cycles. Since there are 8 half-pixel search locations, half-pixel search would take 8*16=128 clock cycles using one PE. We used 4 PEs in order to perform the half-pixel search operation faster. Each PE calculates the SAD for two half-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best half-pixel accurate MV.

The half-pixel search locations are allocated to PEs as follows. First, 4 search locations that use the set C half pixels are searched. PE0 calculates the SAD for the search location C00, PE1 calculates the SAD for the search location C01, PE2 calculates the SAD for the search location C10 and PE3 calculates the SAD for the search location C11. Then, PE0 and PE1 searches the two set B search locations (B00, B01) and PE2 and PE3 searches the two set A search locations (A03, A13).

A novel schedule is used for the calculations done by each PE in order to reduce the number of current block and search window register ports and number of accesses to these registers. Because of the proposed schedule, PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0’). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1’). In addition, PE0 and PE1 can use the same search window pixel in the same clock cycle. Similarly, PE1 and PE2 can use the same search window pixel in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

Because of the proposed allocation of half-pixel search locations to PEs and the proposed schedule, both the search window register file and the current block register file have only two 8-bit read ports; s0, s1 and c0, c1 respectively. PE0 and PE1 use s0 and c0 ports, PE2 and PE3 use s1 and c1 ports.

clock cycle	c0	c1	s0	s1	PE0	PE1	PE2	PE3
1	0	0	0	18	c0 s0		c0 s18	
2	1	1	2	20	c1 s2	c0 s2	c1 s20	c0 20
3	2	2	4	22	c2 s4	c1 s4	c2 s22	c1 s22
4	3	3	6	24	c3 s6	c2 s6	c3 s24	c2 s24
5	4	4	18	36	c4 s18		c4 s36	
6	5	5	20	38	c5 s20	c4 s20	c5 s38	c4 c38
7	6	6	22	40	c6 s22	c5 s22	c6 s40	c5 s40
8	7	7	24	42	c7 s24	c6 s24	c7 s42	c6 s42
9	8	8	36	54	c8 s36		c8 s54	
10	9	9	38	56	c9 s38	c8 s38	c9 s56	c8 s56
11	10	10	40	58	c10 s40	c9 s40	c10 s58	c9 s58
12	11	11	42	60	c11 s42	c10 s42	c11 s60	c10 s60
13	12	12	54	72	c12 s54		c12 s72	
14	13	13	56	74	c13 s56	c12 s56	c13 s74	c12 s74
15	14	14	58	76	c14 s58	c13 s58	c14 s76	c13 s76
16	15	15	60	78	c15 s60	c14 s60	c15 s78	c14 s78
17	3	15	8	80		c3 s8		c15 s80
18	7	3	26	26		c7 s26		c3 s26
19	11	7	44	44		c11 s44		c7 s44
20	15	11	62	62		c15 s62		c11 s62
21	0	0	9	1	c0 s9		c0 s1	
22	1	4	11	19	c1 s11	c0 s11	c4 s19	c0 s19
23	2	8	13	37	c2 s13	c1 s13	c8 s37	c4 s37
24	3	12	15	55	c3 s15	c2 s15	c12 s55	c8 s55
25	4	1	27	3	c4 s27		c1 s3	
26	5	5	29	21	c5 s29	c4 s29	c5 s21	c1 s21
27	6	9	31	39	c6 s31	c5 s31	c9 s39	c5 s39
28	7	13	33	57	c7 s33	c6 s33	c13 s57	c9 s57
29	8	2	45	5	c8 s45		c2 s5	
30	9	6	47	23	c9 s47	c8 s47	c6 s23	c2 s23
31	10	10	49	41	c10 s49	c9 s49	c10 s41	c6 s41
32	11	14	51	59	c11 s51	c10 s51	c14 s59	c10 s59
33	12	3	63	7	c12 s63		c3 s7	
34	13	7	65	25	c13 s65	c12 s65	c7 s25	c3 s25
35	14	11	67	43	c14 s67	c13 s67	c11 s43	c7 s43
36	15	15	69	61	c15 s69	c14 s69	c15 s61	c11 s61
37	3	12	17	73		c3 s17		c12 s73
38	7	13	35	75		c7 s35		c13 s75
39	11	14	53	77		c11 s53		c14 s77
40	15	15	71	79		c15 s71		c15 s79

Figure 2.6 Half-Pixel Search Flow

2.2.3 Implementation Results

The proposed architecture is implemented in Verilog HDL. The implementation is verified with RTL simulations using Mentor Graphics ModelSim SE. The Verilog RTL is then synthesized to a 2V8000ff1152 Xilinx Virtex II FPGA with speed grade 5 using Mentor Graphics Leonardo Spectrum. The resulting netlist is placed and routed to the same FPGA using Xilinx ISE Series 7.1.

The FPGA implementation is verified to work at 90 MHz under worst-case PVT conditions with post place and route simulations. The FPGA implementation can process an HDTV frame in 34.5 msec. (3600 MB * 864 clock cycles per MB * 11.09 ns clock cycle = 34.5 msec). Therefore, it can process 1000/34.5 = 29 HDTV frames (1280x720) per second.

The FPGA implementation uses the following FPGA resources; 3225 Function Generators, 1613 CLB Slices, 2202 DFFs, i.e. %3.46 of Function Generators, %3.46 of CLB Slices, %2.3 of DFFs.

CHAPTER 3

QUARTER-PIXEL ACCURATE H.264 MOTION ESTIMATION HARDWARE DESIGN

3.1 Overview of Quarter-Pixel Accurate Motion Estimation Algorithm

The search locations for half-pixel and quarter-pixel accurate motion estimation are shown in Figure 3.1. First, integer-pixel motion estimation is performed at the integer-pixel search locations and the best integer-pixel motion vector (MV) is determined based on a performance metric, e.g. minimum Sum of Absolute Difference (SAD). Then, half-pixel motion estimation is performed at the half-pixel search locations around the location pointed by the best integer-pixel MV with a search range of [-1, 1], and the integer-pixel MV is refined by the best half-pixel accurate MV.

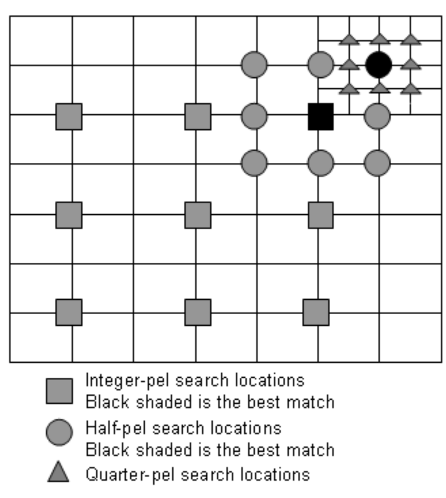


Figure 3.1 Half-Pixel and Quarter-Pixel Search Locations

Finally, quarter-pixel motion estimation is performed at the quarter-pixel search locations around the location pointed by the best half-pixel MV with a search range of [-1, 1], and the half-pixel MV is refined by the best quarter-pixel accurate MV.

Before searching for the best half-pixel accurate MV, half pixels in the half-pixel search window are interpolated from neighboring pixels using a 6-tap FIR filter with weights (1/32, -5/32, 5/8, 5/8, -5/32, 1/32). First, the half pixels that are adjacent to two integer pixels are interpolated from 6 integer pixels. Then, the remaining half pixels are interpolated from 6 horizontal or 6 vertical half pixels. A half-pixel interpolation example is shown in Figure 3.2. First, the half pixels a, b, c, d, e, f are interpolated from 6 corresponding horizontal integer pixels. For example, half pixel c is interpolated from the 6 horizontal integer pixels A, B, C, D, E, F (c = round ((A-5B+20C+20D-5E+F) / 32)). Then, the half pixels g, h, i, j, k, m are interpolated from 6 corresponding vertical integer pixels. For example, half pixel i is interpolated from the 6 vertical integer pixels M, N, C, I, O, P. Finally, half-pixel n can be interpolated from either horizontal half pixels g, h, i, j, k, m or vertical half-pixels a, b, c, d, e, f.

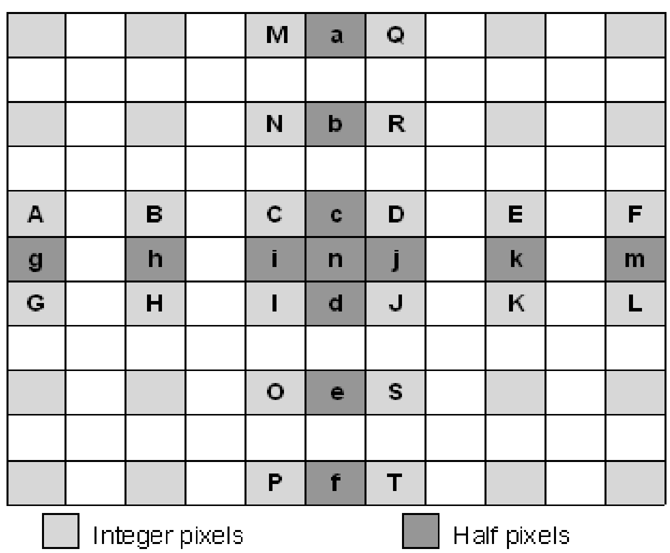


Figure 3.2 Half-Pixel Interpolation Example

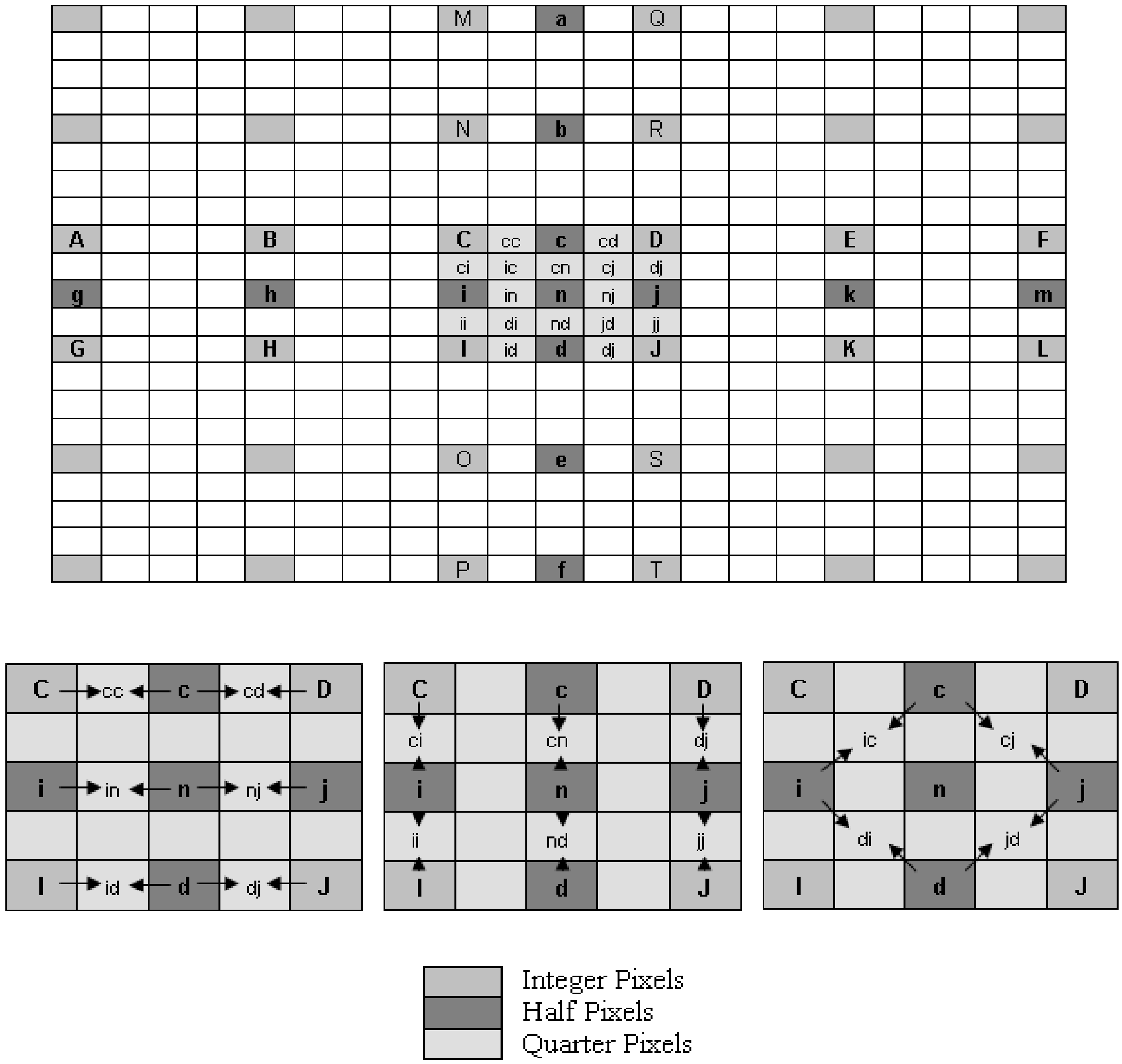


Figure 3.3 Quarter-Pixel Interpolation Example

Before searching for the best quarter-pixel accurate MV, quarter pixels in the quarter-pixel search window are interpolated from neighboring pixels using a bilinear filter. A quarter-pixel interpolation example is shown in Figure 3.3. For example, quarter pixel cc is interpolated from the integer pixel C and half pixel c (cc = (C+c+1)>>1), quarter pixel cn is interpolated from the half pixels c and n (cn = (c+n+1)>>1), and quarter pixel cj is interpolated from the half pixels c and j (cj = (c+j +1)>>1).

3.2.1 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 4x4 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 4x4 block size is shown in Figure 3.4 [15]. The quarter-pixel accurate motion estimation hardwares for other block sizes are similar to this hardware.

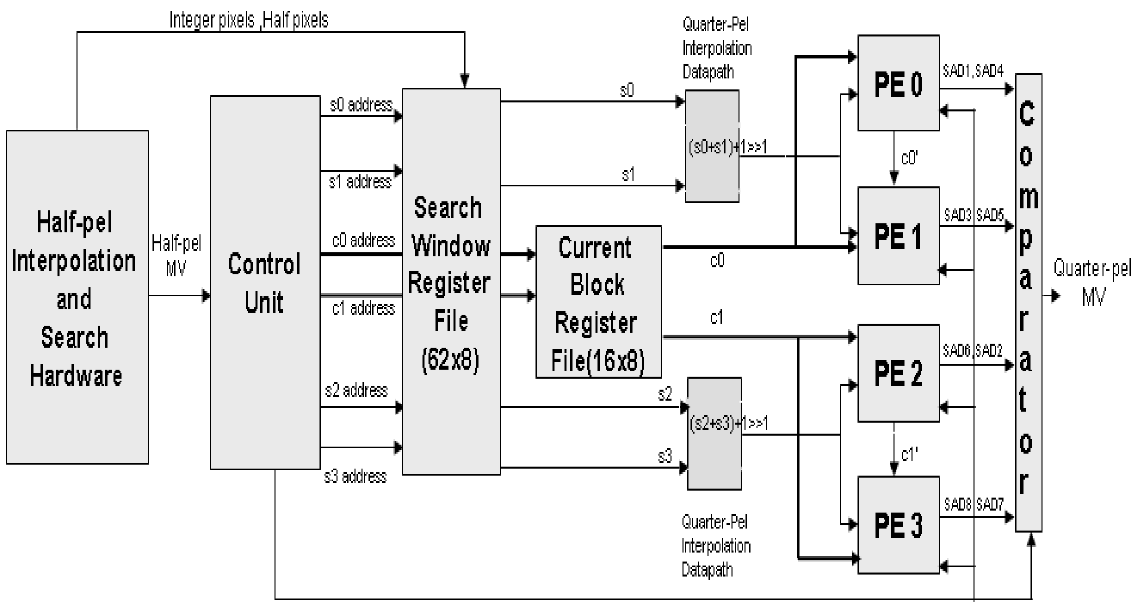


Figure 3.4 Proposed Quarter-Pixel Accurate Motion Estimation Hardware

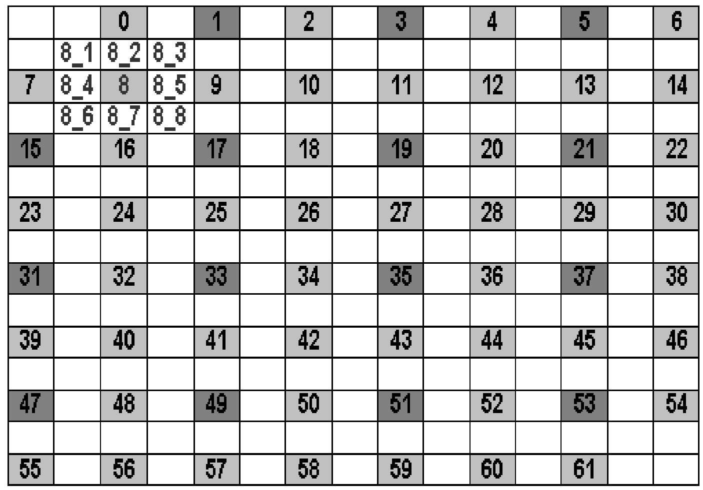
For each 4x4 block in a MB, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 4x4 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed hardware performs quarter-pixel interpolation dynamically, i.e. only the quarter pixels necessary for performing quarter-pixel accurate search at the location pointed by the best half-pixel motion vector are calculated. This reduces the amount of computation performed for quarter-pixel interpolation, and therefore reduces the power consumption of the quarter-pixel accurate motion estimation hardware.

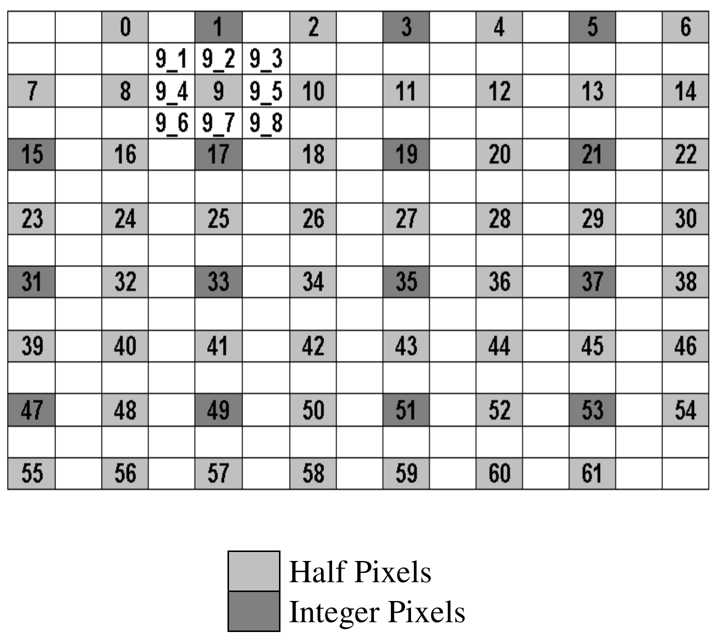
As the half-pixel motion estimation hardware is performing half-pixel interpolation and search, the integer and half pixels necessary for quarter-pixel accurate motion estimation are send to the search window register file by the half-pixel motion estimation hardware. The proposed layout of the integer and half pixels in the 4x4 search window register file, when the location pointed by the best integer-pixel MV is location 17, is shown in Figure 3.5.

Since the half-pixel motion estimation will be performed at the half-pixel search locations 8, 9, 10, 16, 18, 24, 25 and 26, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 8, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 8_1, 8_2, 8_3, 8_4, 8_5, 8_6, 8_7 and 8_8. If the best half-pixel MV points to location 9, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 9_1, 9_2, 9_3, 9_4, 9_5, 9_6, 9_7 and 9_8.

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (8_1, 8_2, 8_3, … , 26_6, 26_7, 26_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.



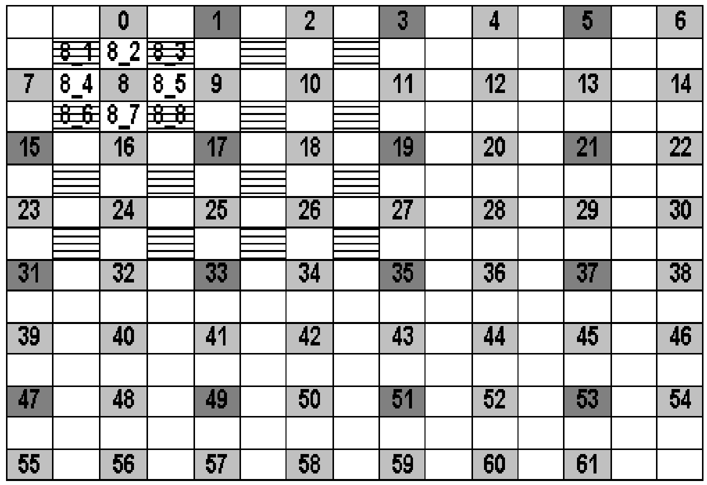
(a) Quarter-Pixel Search Locations around Search Location 8



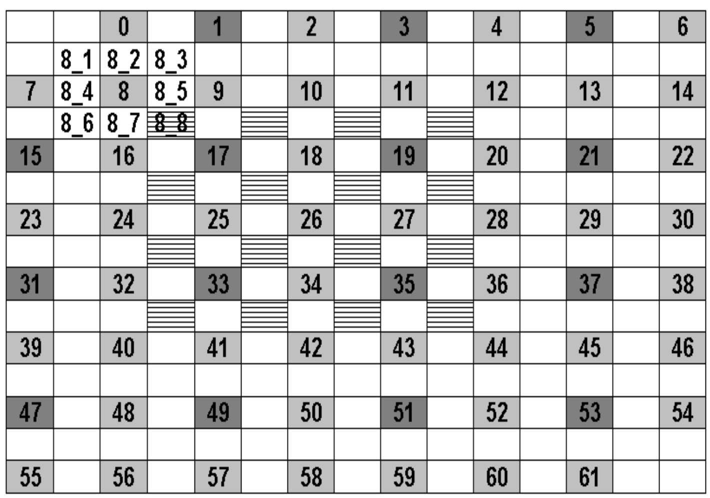
b) Quarter-Pixel Search Locations around Search Location 9

Figure 3.5 Search Window Register File for 4x4 Block Size

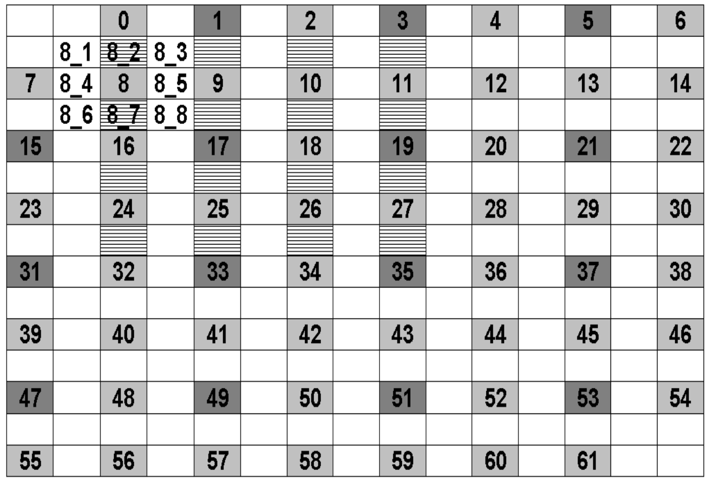
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 8_1, 8_2, 8_7 and 8_8 are shown in Figure 3.6. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 8_1, 8_2, 8_7 and 8_8 are shown in Figure 3.7. For example, the quarter pixels shown in Figure 3.6 (b) are necessary for calculating the SAD value for the quarter-pixel search location 8_8, and the integer and half pixels shown in Figure 3.7 (b) (9, 11, 16, 18, 20, 25, 27, 32, 34, 36, 41, 43) are used for generating these quarter pixels.



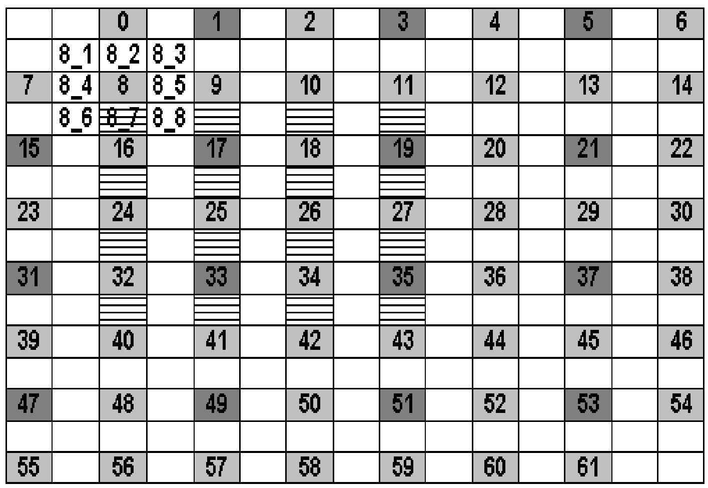
(a) Search location 8_1



(b) Search location 8_8



(c) Search location 8_2



(d) Search location 8_7

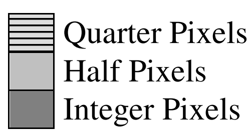
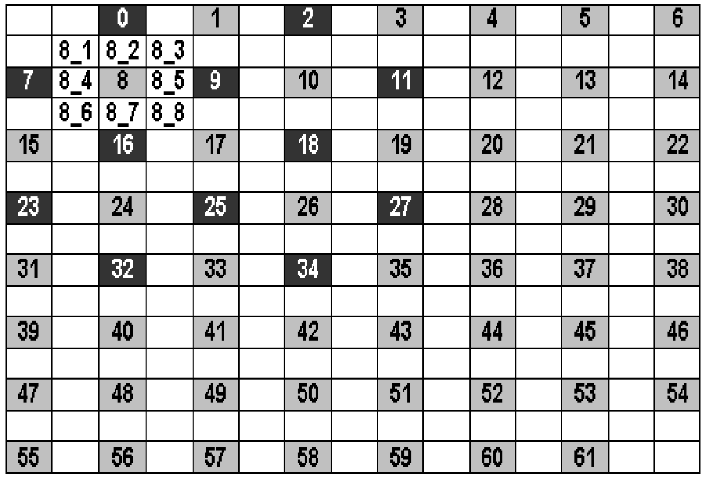
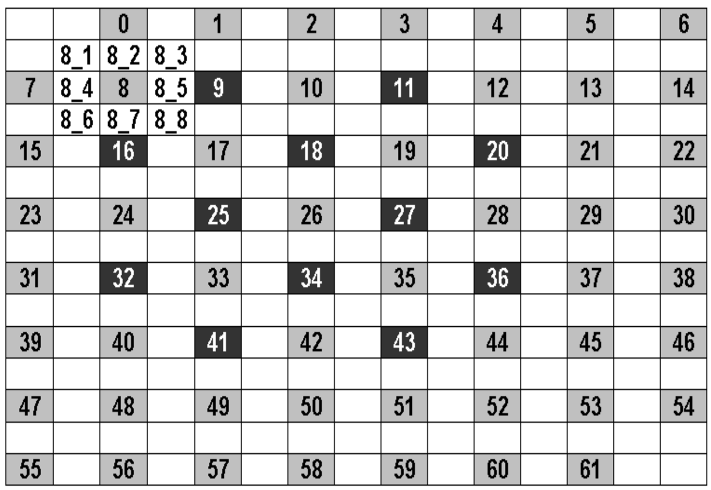


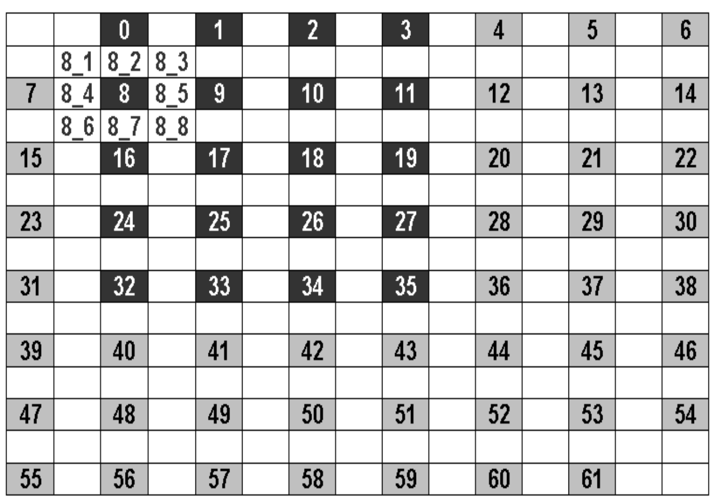
Figure 3.6 Quarter Pixels Necessary for 4x4 Block Size Quarter-Pixel Search Locations 8_1, 8_2, 8_7, 8_8



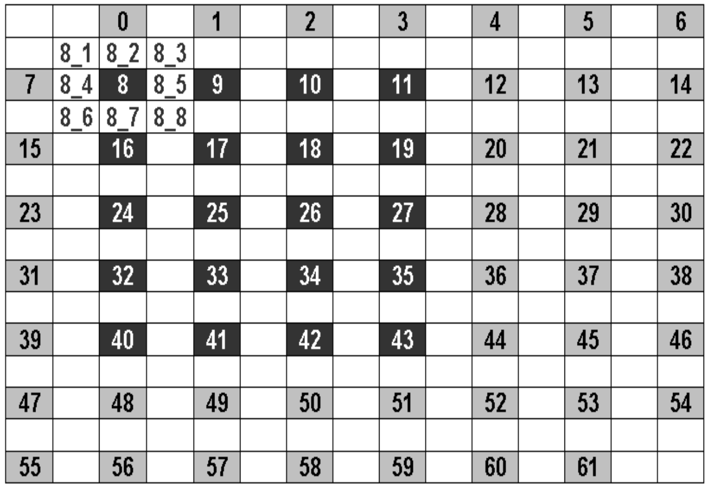
(a) Search location 8_1



(b) Search location 8_8



(c) Search location 8_2



(d) Search location 8_7

Figure 3.7 Integer and Half Pixels Used for 4x4 Block Size Quarter-Pixel Search Locations 8_1, 8_2 8_7, 8_8

The proposed layout of the integer and half pixels in the 4x4 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.8. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_8 are 9 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_1. In Figure 3.8, this read address correlation between quarter-pixel search locations 8_1 and 8_8 is shown by writing the read address for location 8_8 as 8_1 + 9.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
8	8_1	8_2	8_3	8_4	8_4 + 1	8_3 + 7	8_2 + 8	8_1 + 9	row1
9	8_3	8_2 + 1	8_1 + 2	8_4 + 1	8_4 + 2	8_1 + 9	8_2 + 9	8_3 + 9	row2
10	8_1 + 2	8_2 + 2	8_3 + 2	8_4 + 2	8_4 + 3	8_3 + 9	8_2 + 10	8_1 + 11	row1 + 2
17	8_3 + 7	8_2 + 8	8_1 + 9	8_4 + 8	8_4 + 9	8_1 + 16	8_2 + 16	8_3 + 16	row2 + 7
18	8_3 + 9	8_2 + 10	8_1 + 11	8_4 + 10	8_4 + 11	8_1 + 18	8_2 + 18	8_3 + 18	row2 + 9
24	8_1 + 16	8_2 + 16	8_3 + 16	8_4 + 16	8_4 + 17	8_3 + 23	8_2 + 25	8_1 + 26	row1 + 16
25	8_3 + 16	8_2 + 17	8_1 + 18	8_4 + 17	8_4 + 18	8_1 + 25	8_2 + 25	8_3 + 25	row2 + 16
26	8_1 + 18	8_2 + 18	8_3 + 18	8_4 + 18	8_4 + 19	8_3 + 26	8_2 + 26	8_1 + 27	row1 + 18

Figure 3.8 Address Correlation of 4x4 Block Size Quarter-Pixel Search Locations

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 8_1, 8_2, 8_3, 8_4 and the read address correlations of 64 quarter-pixel search locations.

The SAD value for a quarter-pixel search location is calculated by a processing element in 16 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*16=128 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

The proposed quarter-pixel interpolation and search flow for a 4x4 block is shown in Figure A.1. The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0’). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1’). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

3.2.2 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 4x8 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 4x8 block size is similar to 4x4 block size hardware shown in Figure 3.4. For each 4x8 block in a MB, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 4x8 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed layout of the integer and half pixels in the 4x8 search window register file, when the location pointed by the best integer-pixel MV is location 25, is shown in Figure 3.9. Since the half-pixel motion estimation will be performed at the half-pixel search locations 12, 13, 14, 24, 26, 36, 37 and 38, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 12, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 12_1, 12_2, 12_3, 12_4, 12_5, 12_6, 12_7 and 12_8.

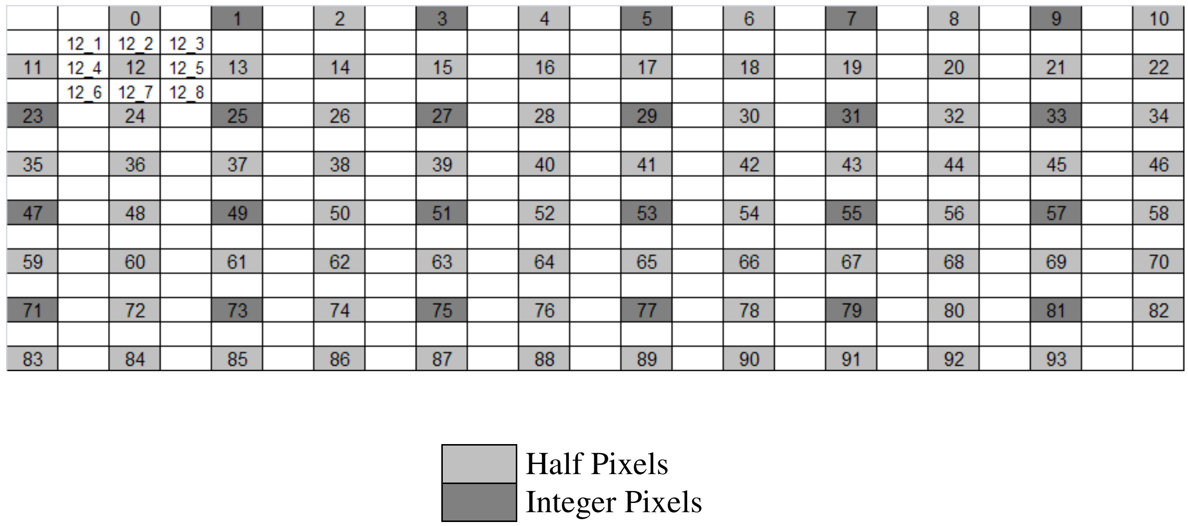
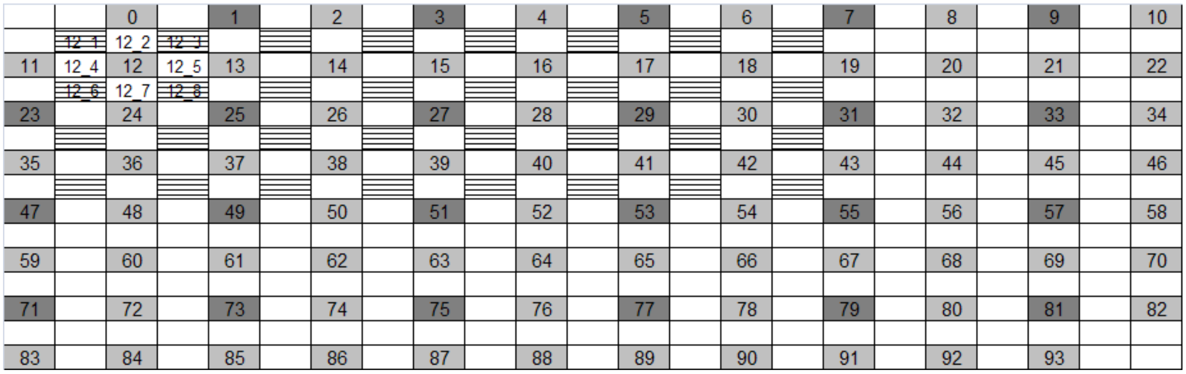


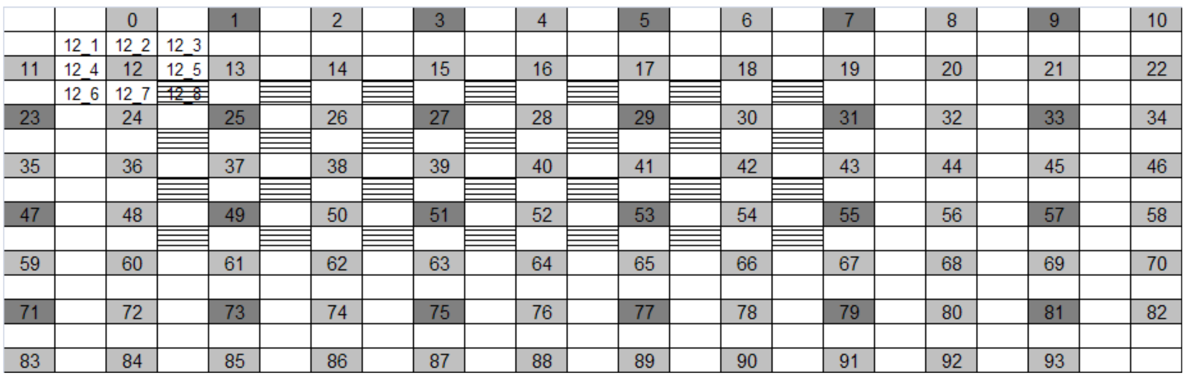
Figure 3.9 Search Window Register File for 4x8 Block Size

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (12_1, 12_2, 12_3, … , 38_6, 38_7, 38_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.

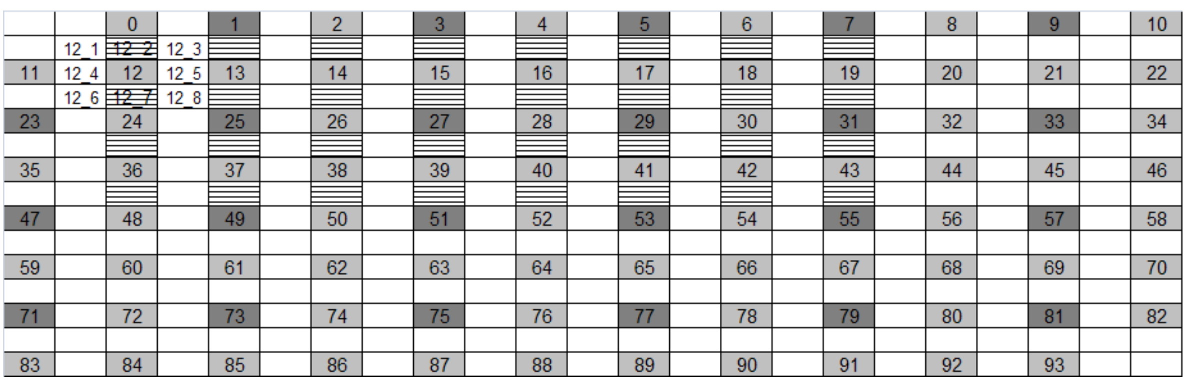
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 12_1, 12_2, 12_7 and 12_8 are shown in Figure 3.10. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 12_1, 12_2, 12_7 and 12_8 are shown in Figure 3.11. For example, the quarter pixels shown in Figure 3.10 (b) are necessary for calculating the SAD value for the quarter-pixel search location 12_8, and the integer and half pixels shown in Figure 3.11 (b) (13, 15, 17, 19, 24, 26, 28, 30, 32, 37, 39, 41, 43, 48, 50, 52, 54, 56, 61, 63, 65, 67) are used for generating these quarter pixels.



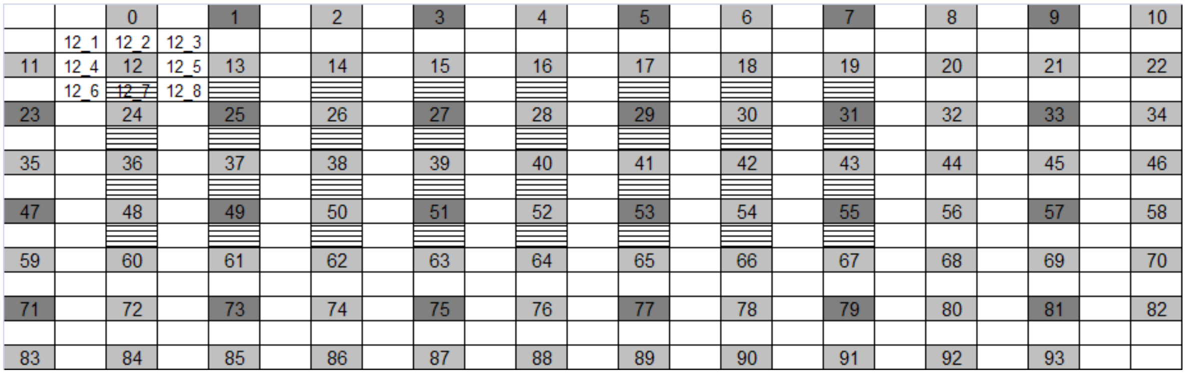
(a) Search location 12_1



(b) Search location 12_8



(c) Search location 12_2



(d) Search location 12_7

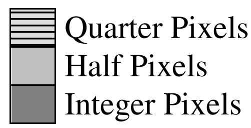
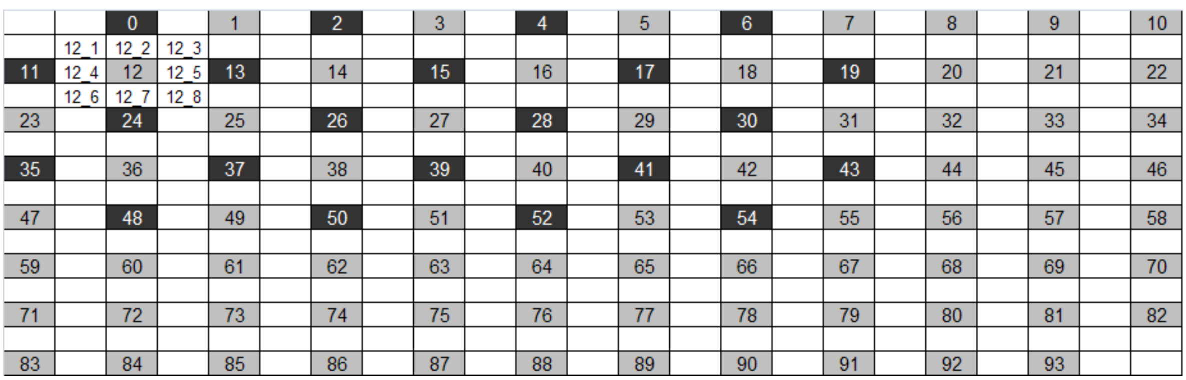
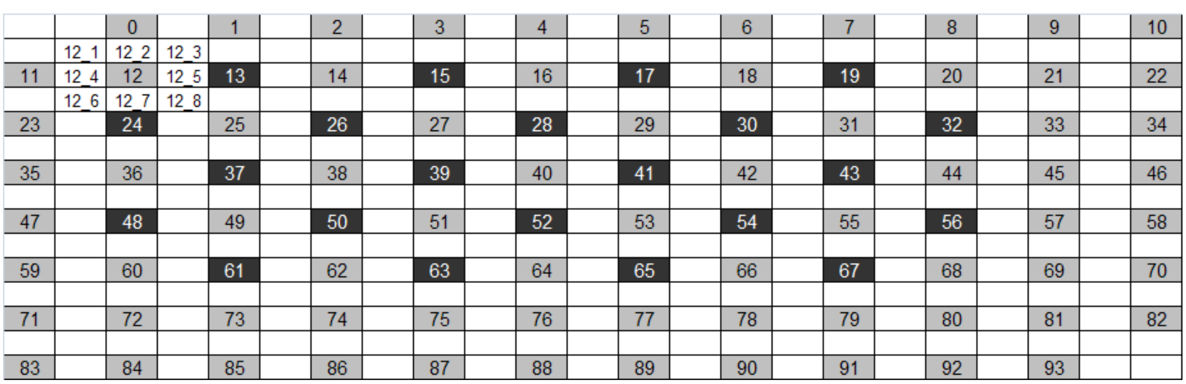


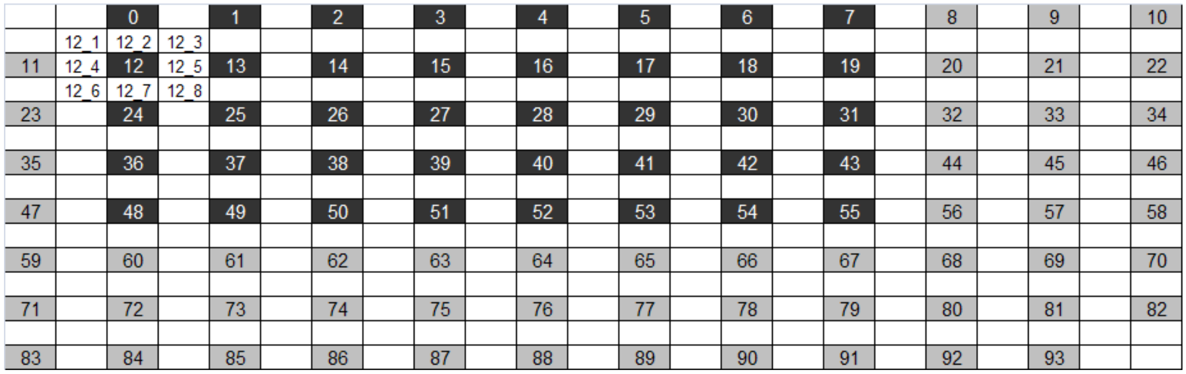
Figure 3.10 Quarter Pixels Necessary for 4x8 Block Size Quarter-Pixel Search Locations 12_1, 12_2, 12_7, 12_8



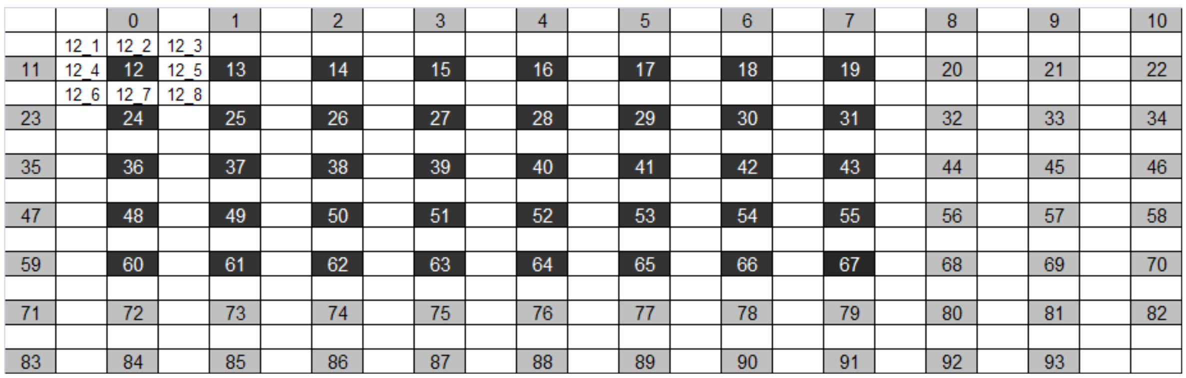
(a) Search location 12_1



(b) Search location 12_8



(c) Search location 12_2



(d) Search location 12_7

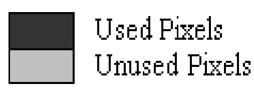


Figure 3.11 Integer and Half Pixels Used for 4x8 Block Size Quarter-Pixel Search Locations 12_1, 12_2, 12_7, 12_8

The proposed layout of the integer and half pixels in the 4x8 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.12. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 12_8 are 13 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 12_1. In Figure 3.12, this read address correlation between quarter-pixel search locations 12_1 and 12_8 is shown by writing the read address for location 12_8 as 12_1 + 13.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
12	12_1	12_2	12_3	12_4	12_4 + 1	12_3 + 11	12_2 + 12	12_1 + 13	row1
13	12_3	12_2 + 1	12_1 + 2	12_4 + 1	12_4 + 2	12_1 + 13	12_2 + 13	12_3 + 13	row2
14	12_1 + 2	12_2 + 2	12_3 + 2	12_4 + 2	12_4 + 3	12_3 + 13	12_2 + 14	12_1 + 15	row1 +2
24	12_3 + 11	12_2 + 12	12_1 + 13	12_4 + 12	12_4 + 13	12_1 + 24	12_2 + 24	12_3 + 24	row2 +11
26	12_3 + 13	12_2 + 14	12_1 + 15	12_4 + 14	12_4 + 15	12_1 + 26	12_2 + 26	12_3 + 26	row2 +13
36	12_1 + 24	12_2 + 24	12_3 + 24	12_4 + 24	12_4 + 25	12_3 + 35	12_2 + 36	12_1 + 37	row1 +24
37	12_3 + 24	12_2 + 25	12_1 + 26	12_4 + 25	12_4 + 26	12_1 + 37	12_2 + 37	12_3 + 37	row2 +24
38	12_1 + 26	12_2 + 26	12_3 + 26	12_4 + 26	12_4 + 27	12_3 + 37	12_2 + 38	12_1 + 39	row1 +26

Figure 3.12 Address Correlation of 4x8 Block Size Quarter-Pixel Search Locations

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 12_1, 12_2, 12_3, 12_4 and the read address correlations of 64 quarter-pixel search locations.

The SAD value for a quarter-pixel search location is calculated by a processing element (PE) in 32 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*32=256 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

The proposed quarter-pixel interpolation and search flow for a 4x8 block is shown in Figure A.2 The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0'). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1'). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

3.2.3 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 8x4 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 8x4 block size is similar to 4x4 block size hardware shown in Figure 3.4. For each 8x4 block in a MB, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 8x4 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed layout of the integer and half pixels in the 8x4 search window register file, when the location pointed by the best integer-pixel MV is location 17, is shown in Figure 3.13. Since the half-pixel motion estimation will be performed at the half-pixel search locations 8, 9, 10, 16, 18, 24, 25 and 26, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 8, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 8_1, 8_2, 8_3, 8_4, 8_5, 8_6, 8_7 and 8_8.

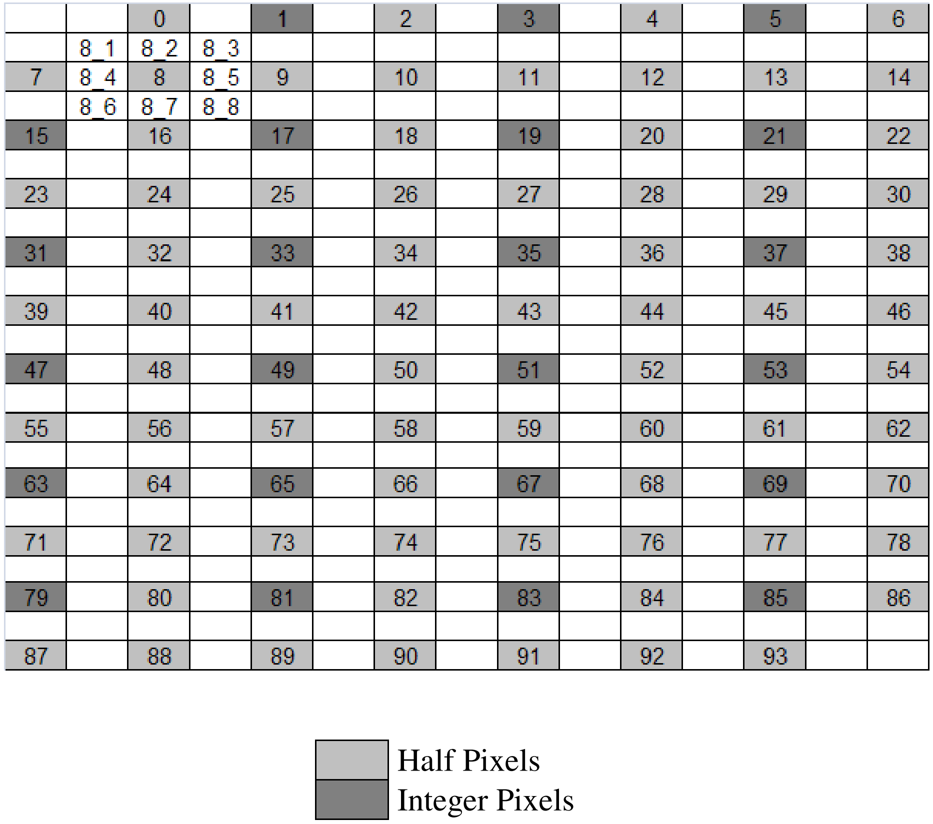
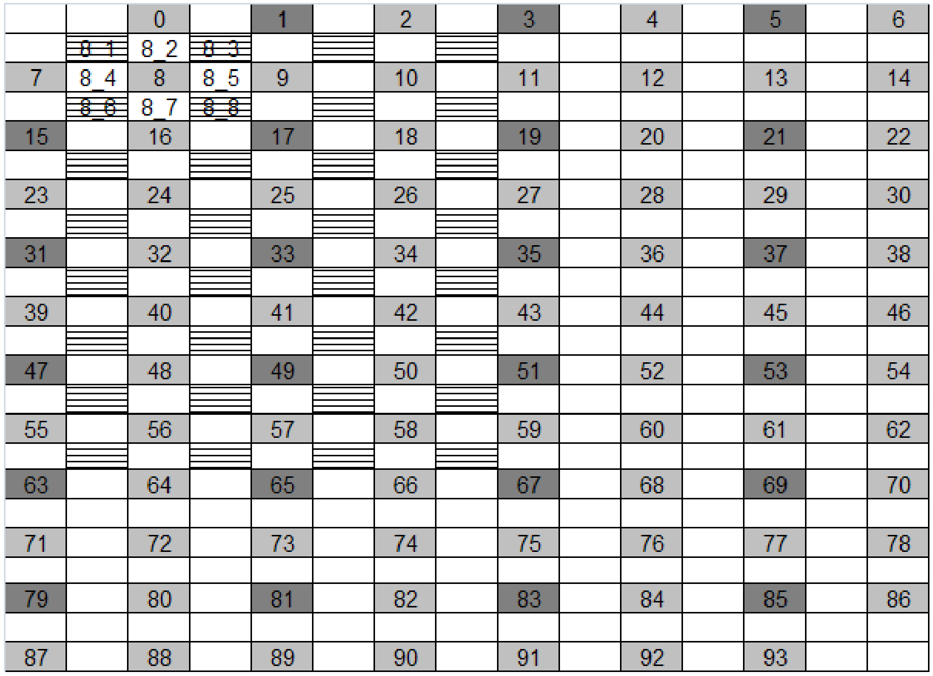


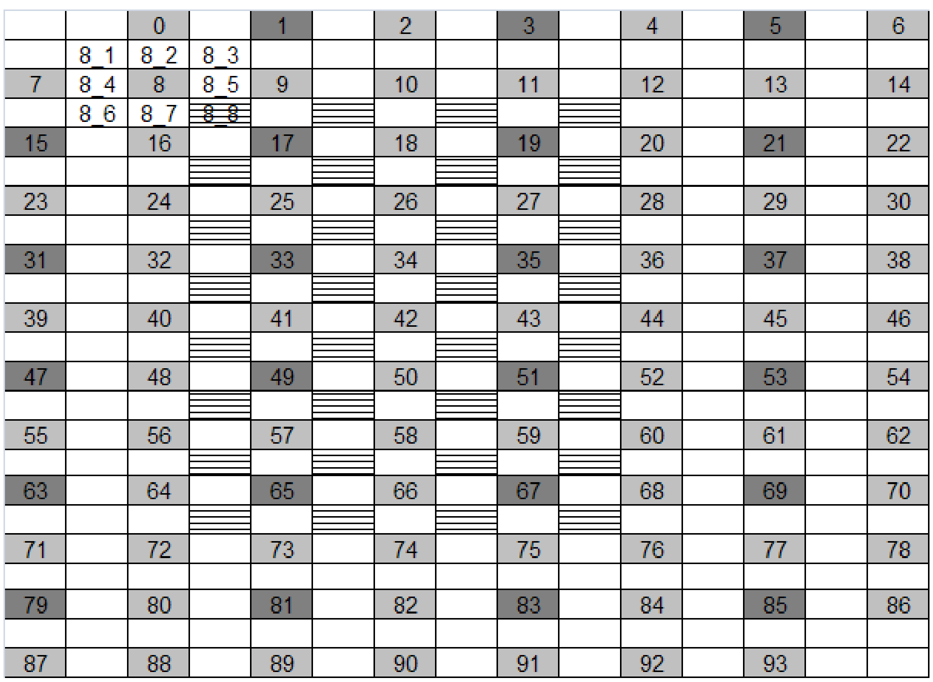
Figure 3.13 Search Window Register File for 8x4 Block Size

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (8_1, 8_2, 8_3, … , 26_6, 26_7, 26_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.

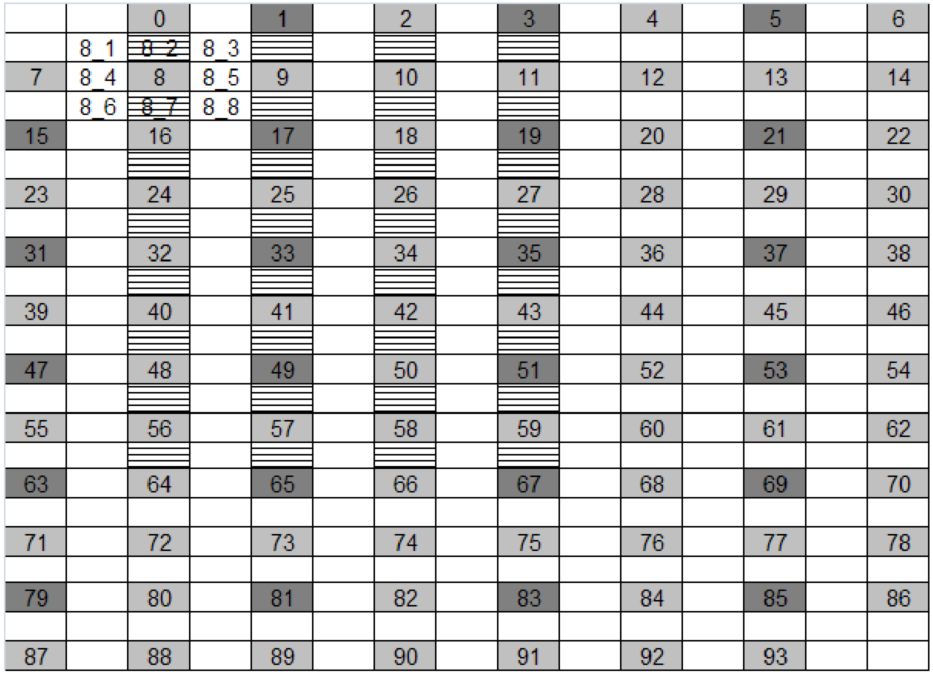
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 8_1, 8_2, 8_7 and 8_8 are shown in Figure 3.14. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 8_1, 8_2, 8_7 and 8_8 are shown in Figure 3.15. For example, the quarter pixels shown in Figure 3.14 (b) are necessary for calculating the SAD value for the quarter-pixel search location 8_8, and the integer and half pixels shown in Figure 3.15 (b) (9, 11, 16, 18, 20, 25, 27, 32, 34, 36, 41, 43) are used for generating these quarter pixels.



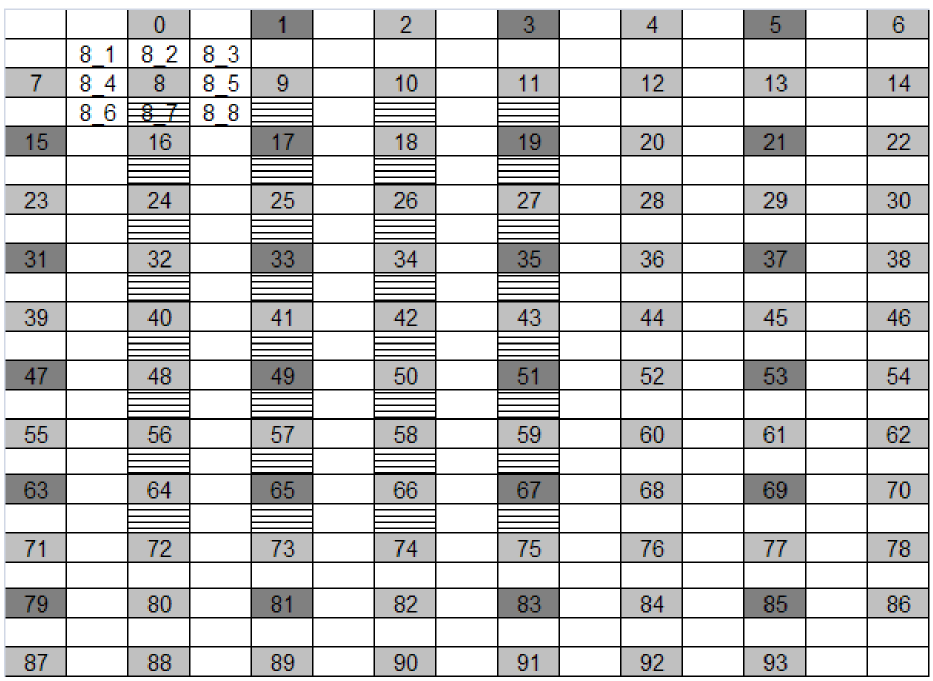
(a) Search location 8_1



(b) Search location 8_8



(c) Search location 8_2



(d) Search location 8_7

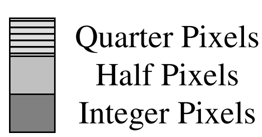
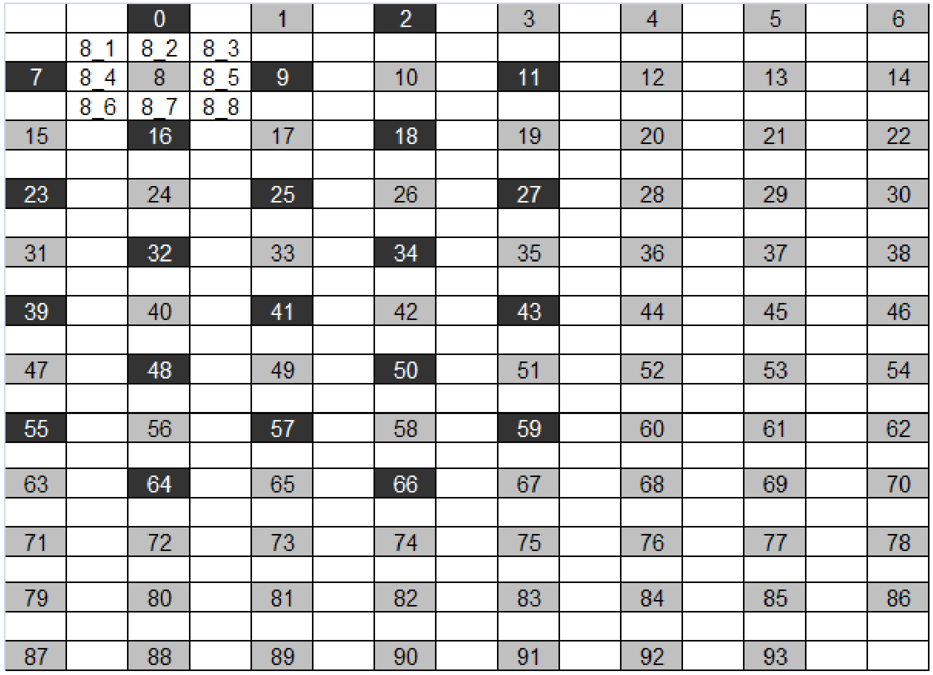
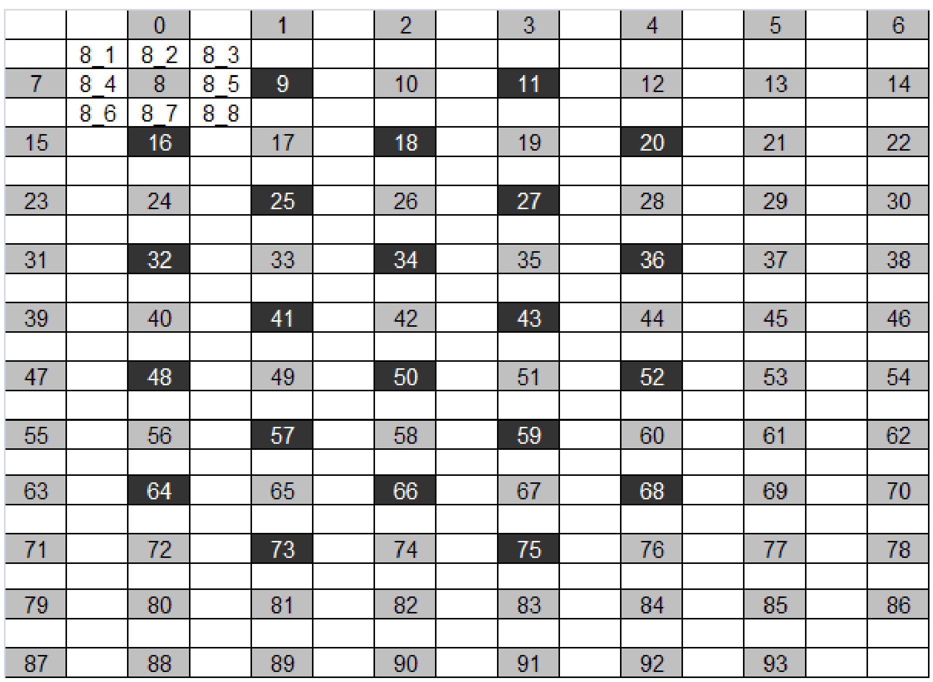


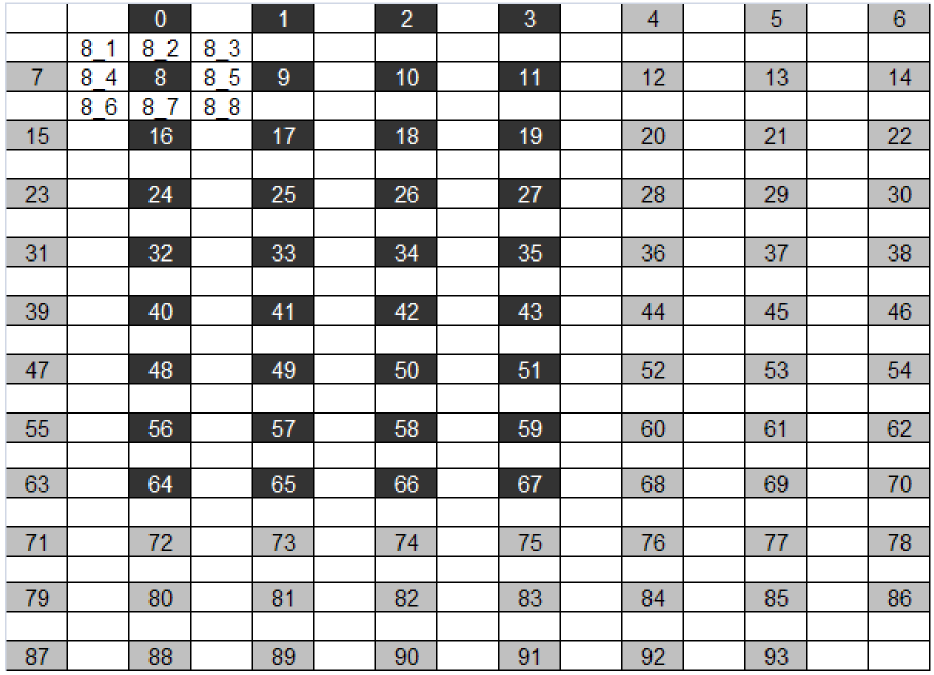
Figure 3.14 Quarter Pixels Necessary for 8x4 Block Size Quarter-Pixel Search Locations 8_1 , 8_2, 8_7, 8_8



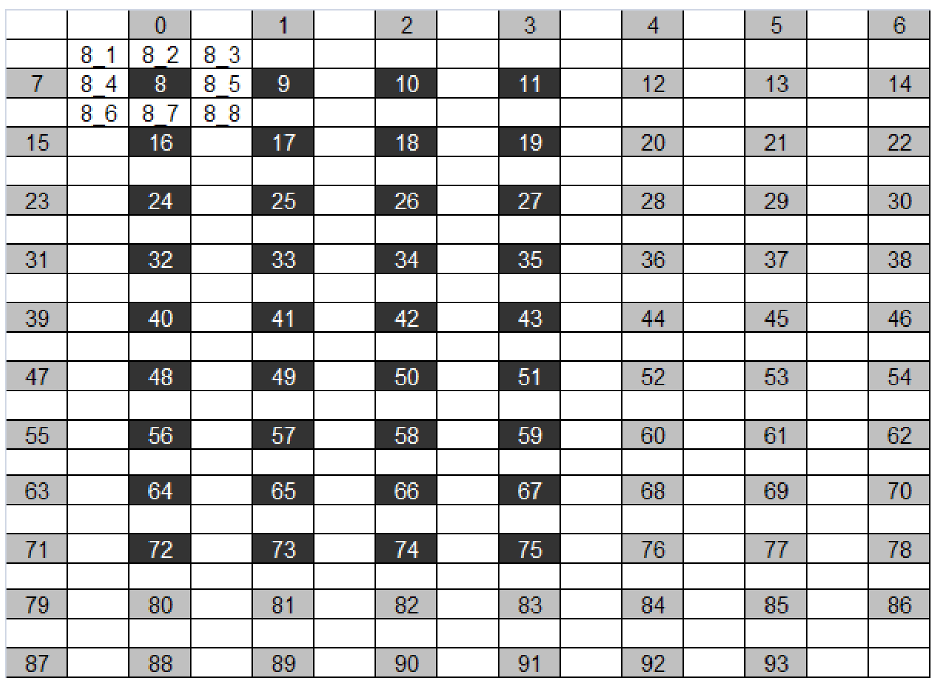
(a) Search location 8_1



(b) Search location 8_8



(c) Search location 8_2



(d) Search location 8_7

Figure 3.15 Integer and Half Pixels Used for 8x4 Block Size Quarter-Pixel Search Locations 8_1, 8_2, 8_7, 8_8

The proposed layout of the integer and half pixels in the 8x4 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.16. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_8 are 9 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_1. In Figure 3.16, this read address correlation between quarter-pixel search locations 8_1 and 8_8 is shown by writing the read address for location 8_8 as 8_1 + 9.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
8	8_1	8_2	8_3	8_4	8_4 + 1	8_3 + 7	8_2 + 8	8_1 + 9	row1
9	8_3	8_2 + 1	8_1 + 2	8_4 + 1	8_4 + 2	8_1 + 9	8_2 + 9	8_3 + 9	row2
10	8_1 + 2	8_2 + 2	8_3 + 2	8_4 + 2	8_4 + 3	8_3 + 9	8_2 + 10	8_1 + 11	row1 + 2
17	8_3 + 7	8_2 + 8	8_1 + 9	8_4 + 8	8_4 + 9	8_1 + 16	8_2 + 16	8_3 + 16	row2 + 7
18	8_3 + 9	8_2 + 10	8_1 + 11	8_4 + 10	8_4 + 11	8_1 + 18	8_2 + 18	8_3 + 18	row2 + 9
24	8_1 + 16	8_2 + 16	8_3 + 16	8_4 + 16	8_4 + 17	8_3 + 23	8_2 + 25	8_1 + 26	row1 + 16
25	8_3 + 16	8_2 + 17	8_1 + 18	8_4 + 17	8_4 + 18	8_1 + 25	8_2 + 25	8_3 + 25	row2 + 16
26	8_1 + 18	8_2 + 18	8_3 + 18	8_4 + 18	8_4 + 19	8_3 + 26	8_2 + 26	8_1 + 27	row1 + 18

Figure 3.16 Address Correlation of 8x4 Block Size Quarter-Pixel Search Locations

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 8_1, 8_2, 8_3, 8_4 and the read address correlations of 64 quarter-pixel search locations.

The SAD value for a quarter-pixel search location is calculated by a processing element in 32 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*32=256 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

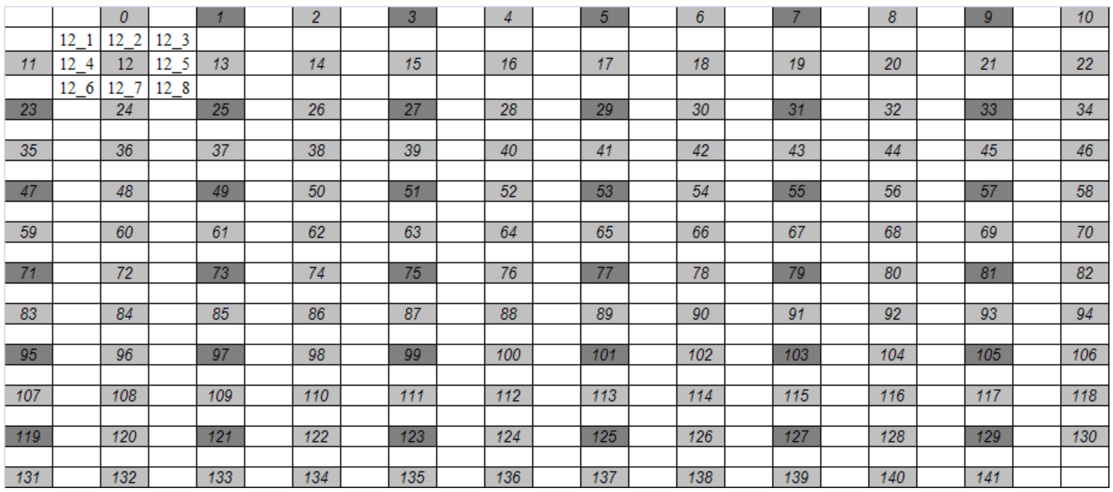
The proposed quarter-pixel interpolation and search flow for a 8x4 block is shown in Figure A.3. The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0'). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1'). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

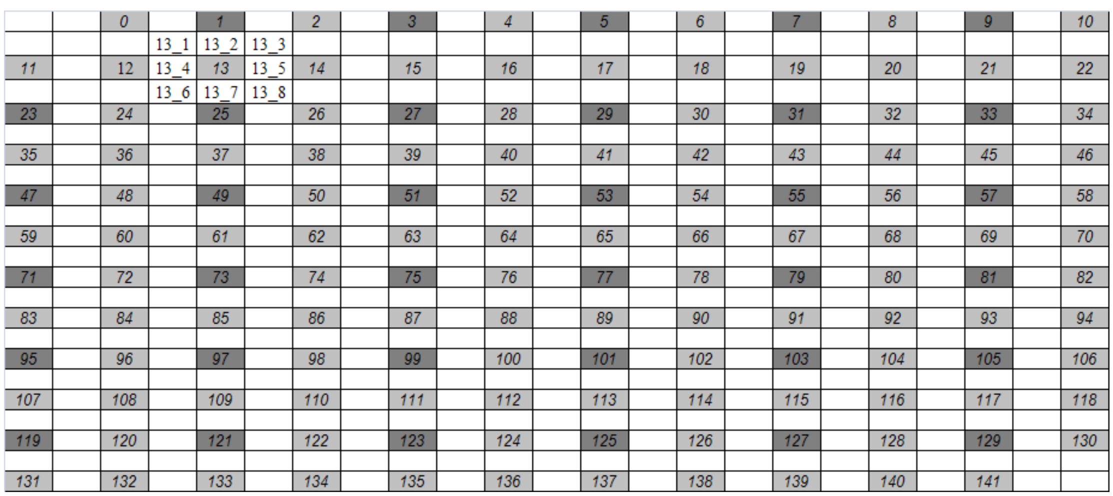
3.2.4 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 8x8 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 8x8 block size is similar to 4x4 block size hardware shown in Figure 3.4. For each 8x8 block in a MB, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 8x8 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed layout of the integer and half pixels in the 8x8 search window register file, when the location pointed by the best integer-pixel MV is location 25, is shown in Figure 3.17. Since the half-pixel motion estimation will be performed at the half-pixel search locations 12, 13, 14, 24, 26, 36, 37 and 38, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 12, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 12_1, 12_2, 12_3, 12_4, 12_5, 12_6, 12_7 and 12_8.



(a) Quarter-Pixel Search Locations around Search Location 12

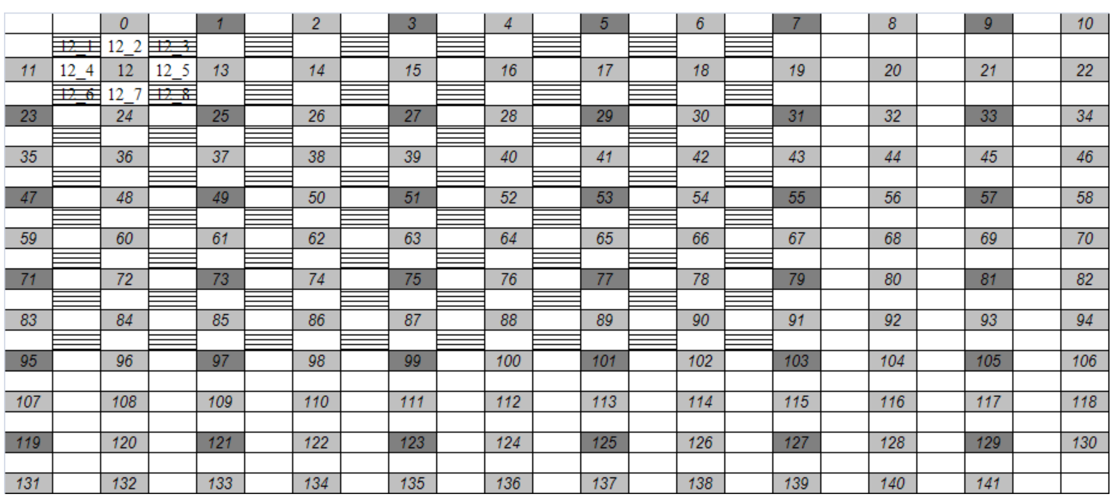


(b) Quarter-Pixel Search Locations around Search Location 13

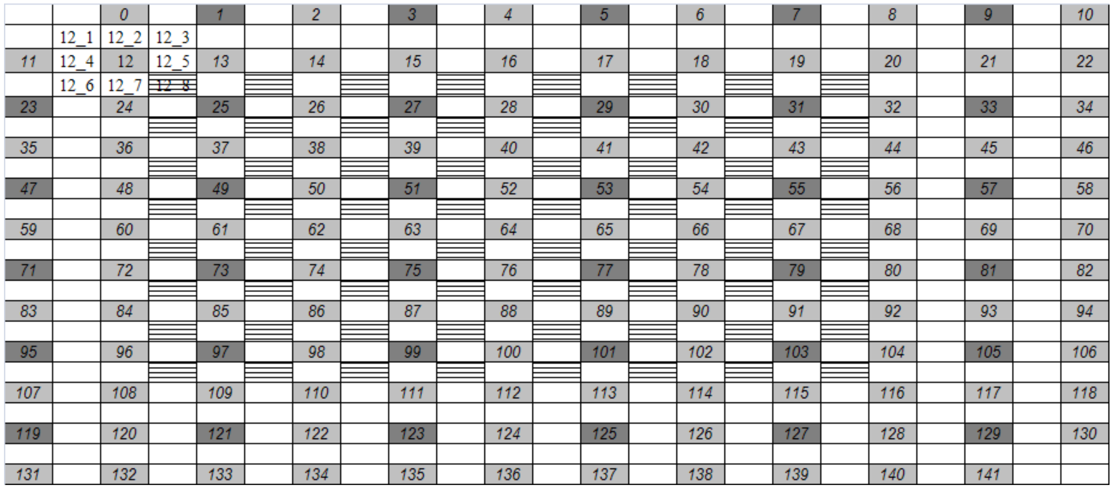
Figure 3.17 Search Window Register File for 8x8 Block Size

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (12_1, 12_2, 12_3, … , 38_6, 38_7, 38_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.

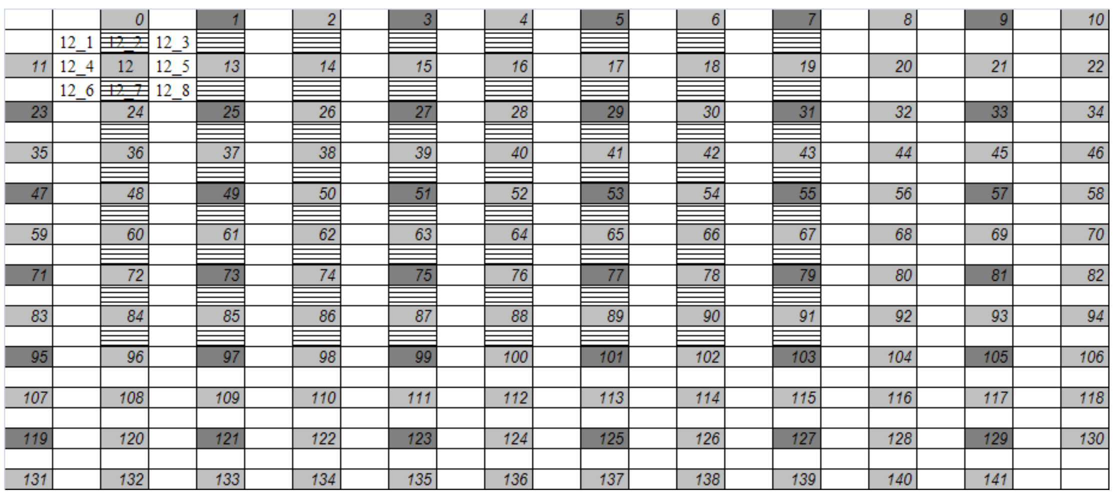
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 12_1, 12_2, 12_7 and 12_8 are shown in Figure 3.18. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 12_1, 12_2, 12_7 and 12_8 are shown in Figure 3.19. For example, the quarter pixels shown in Figure 3.18 (b) are necessary for calculating the SAD value for the quarter-pixel search location 12_8, and the integer and half pixels shown in Figure 3.19 (b) (13, 15, 17, 19, 24, 26, 28, 30, 37, 39, 41, 43, 48, 50, 52, 54, 56, 61, 63, 65, 67, 72, 74, 76, 78, 80, 85, 87, 89, 91, 96, 98, 100, 102, 104, 109, 111, 113, 115) are used for generating these quarter pixels.



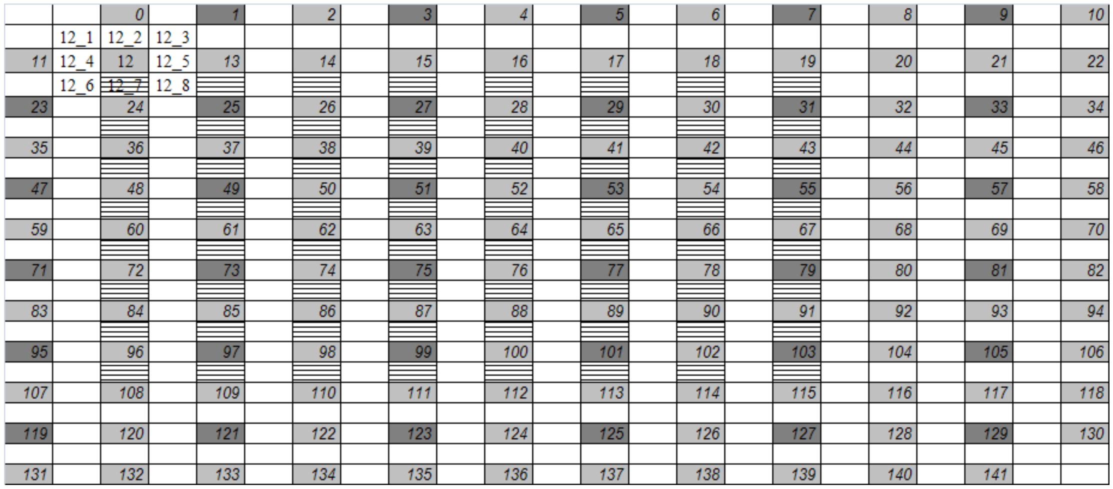
(a) Search location 12_1



(b) Search location 12_8



(c) Search location 12_2



(d) Search location 12_7

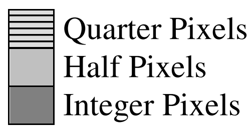
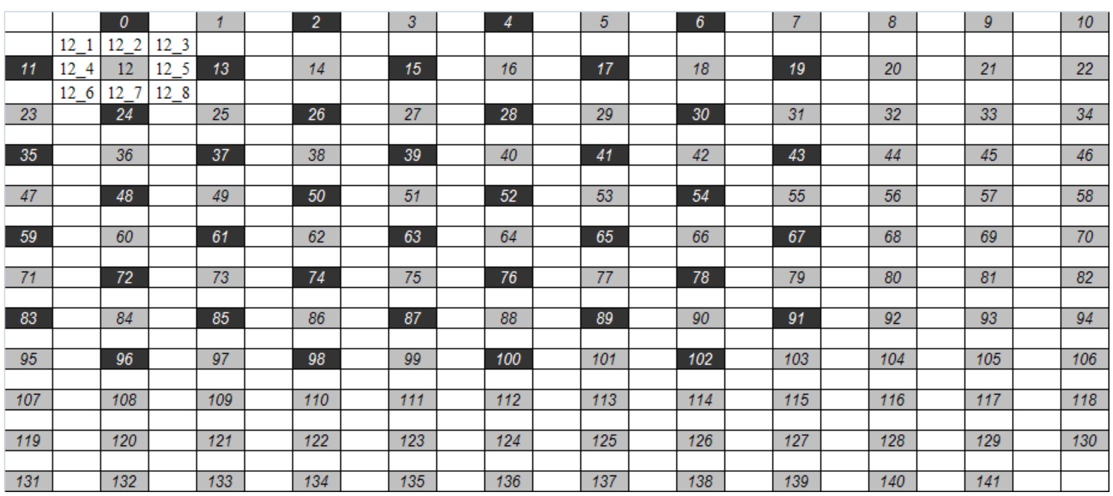
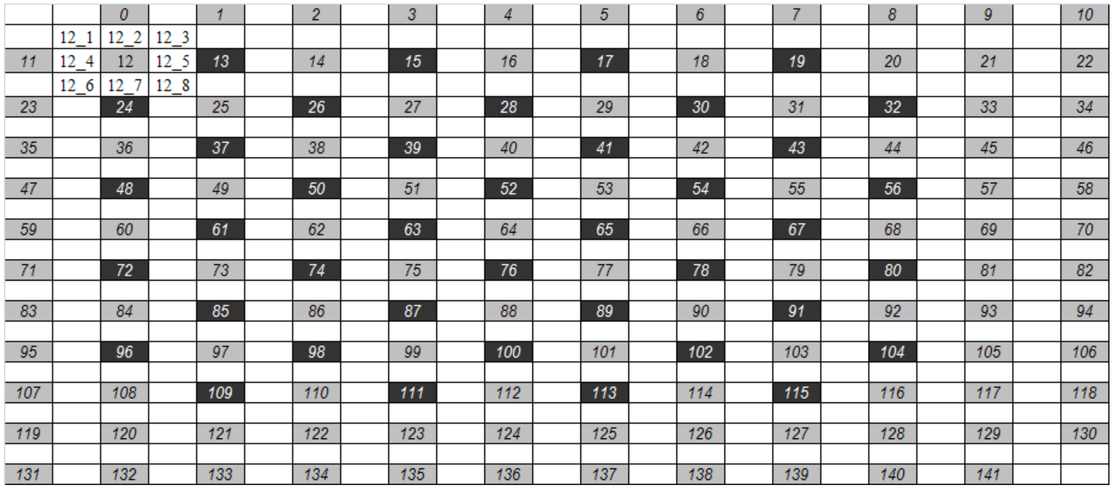


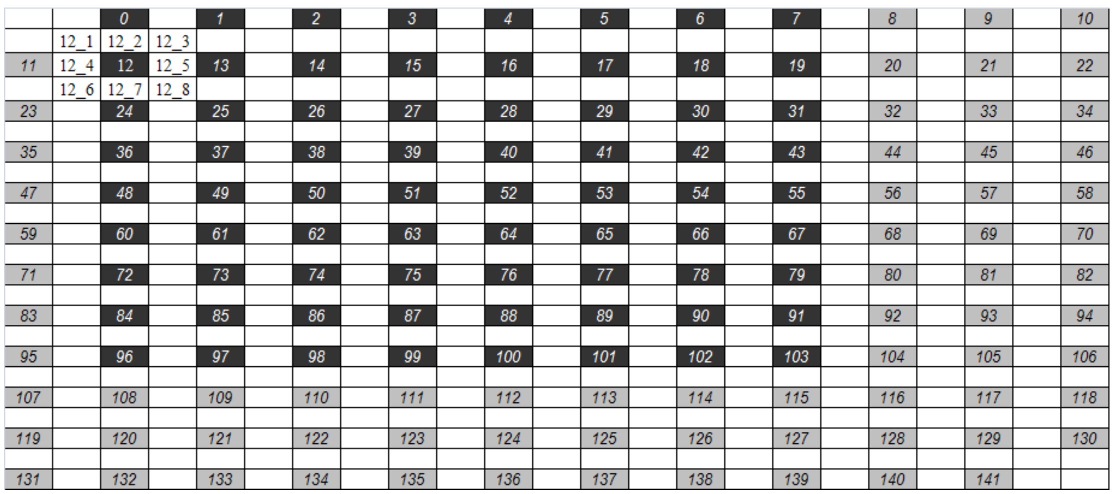
Figure 3.18 Quarter Pixels Necessary for 8x8 Block Size Quarter-pixel Search Locations 12_1, 12_2, 12_7, 12_8



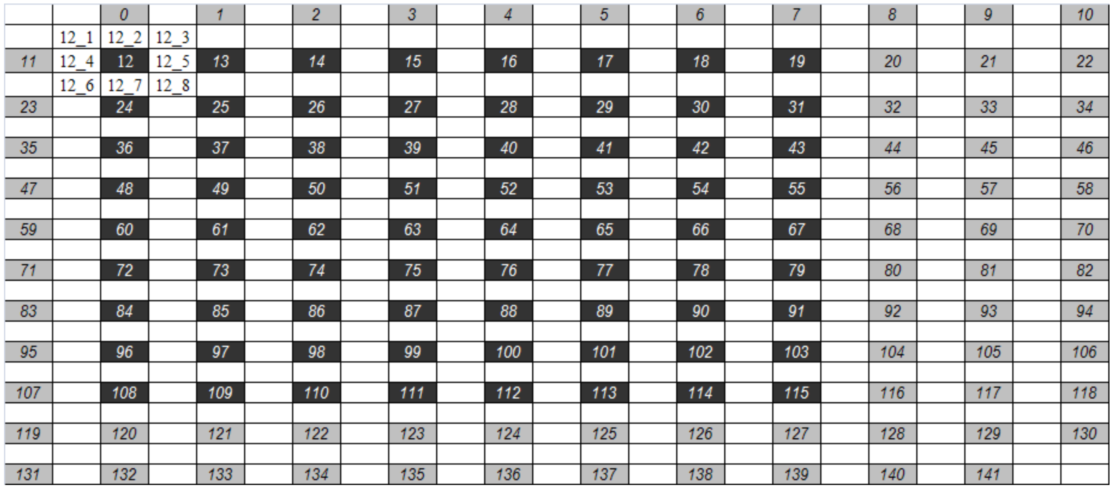
(a) Search location 12_1



(b) Search location 12_8



(c) Search location 12_2



(d) Search location 12_7

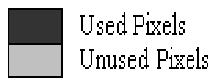


Figure 3.19 Integer and Half Pixels Used for 8x8 Block Size Quarter-pixel Search Locations 12_1, 12_2, 12_7, 12_8

The proposed layout of the integer and half pixels in the 8x8 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.20. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 12_8 are 9 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_1. In Figure 3.20, this read address correlation between quarter-pixel search locations 12_1 and 12_8 is shown by writing the read address for location 12_8 as 12_1 + 9.

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 12_1, 12_2, 12_3, 12_4 and the read address correlations of 64 quarter-pixel search locations.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
12	12_1	12_2	12_3	12_4	12_4 + 1	12_3 + 11	12_2 + 12	12_1 + 13	row1
13	12_3	12_2 + 1	12_1 + 2	12_4 + 1	12_4 + 2	12_1 + 13	12_2 + 13	12_3 + 13	row2
14	12_1 + 2	12_2 + 2	12_3 + 2	12_4 + 2	12_4 + 3	12_3 + 13	12_2 + 14	12_1 + 15	row1 +2
24	12_3 + 11	12_2 + 12	12_1 + 13	12_4 + 12	12_4 + 13	12_1 + 24	12_2 + 24	12_3 + 24	row2 +11
26	12_3 + 13	12_2 + 14	12_1 + 15	12_4 + 14	12_4 + 15	12_1 + 26	12_2 + 26	12_3 + 26	row2 +13
36	12_1 + 24	12_2 + 24	12_3 + 24	12_4 + 24	12_4 + 25	12_3 + 35	12_2 + 36	12_1 + 37	row1 +24
37	12_3 + 24	12_2 + 25	12_1 + 26	12_4 + 25	12_4 + 26	12_1 + 37	12_2 + 37	12_3 + 37	row2 +24
38	12_1 + 26	12_2 + 26	12_3 + 26	12_4 + 26	12_4 + 27	12_3 + 37	12_2 + 38	12_1 + 39	row1 +26

Figure 3.20 Address Correlation of 8x8 Block Size Quarter-pixel Search Locations

The SAD value for a quarter-pixel search location is calculated by a processing element in 64 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*64=512 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

The proposed quarter-pixel interpolation and search flow for a 8x8 block is shown in Figure A.4. The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0'). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1'). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

3.2.5 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 8x16 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 8x16 block size is similar to 4x4 block size hardware shown in Figure 3.4. For each 8x16 block in a MB, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 8x16 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed layout of the integer and half pixels in the 8x16 search window register file, when the location pointed by the best integer-pixel MV is location 41, is shown in Figure 3.21. Since the half-pixel motion estimation will be performed at the half-pixel search locations 20, 21, 22, 40, 42, 60, 61 and 62, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 20, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 20_1, 20_2, 20_3, 20_4, 20_5, 20_6, 20_7 and 20_8.

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (20_1, 20_2, 20_3, … , 62_6, 62_7, 62_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.

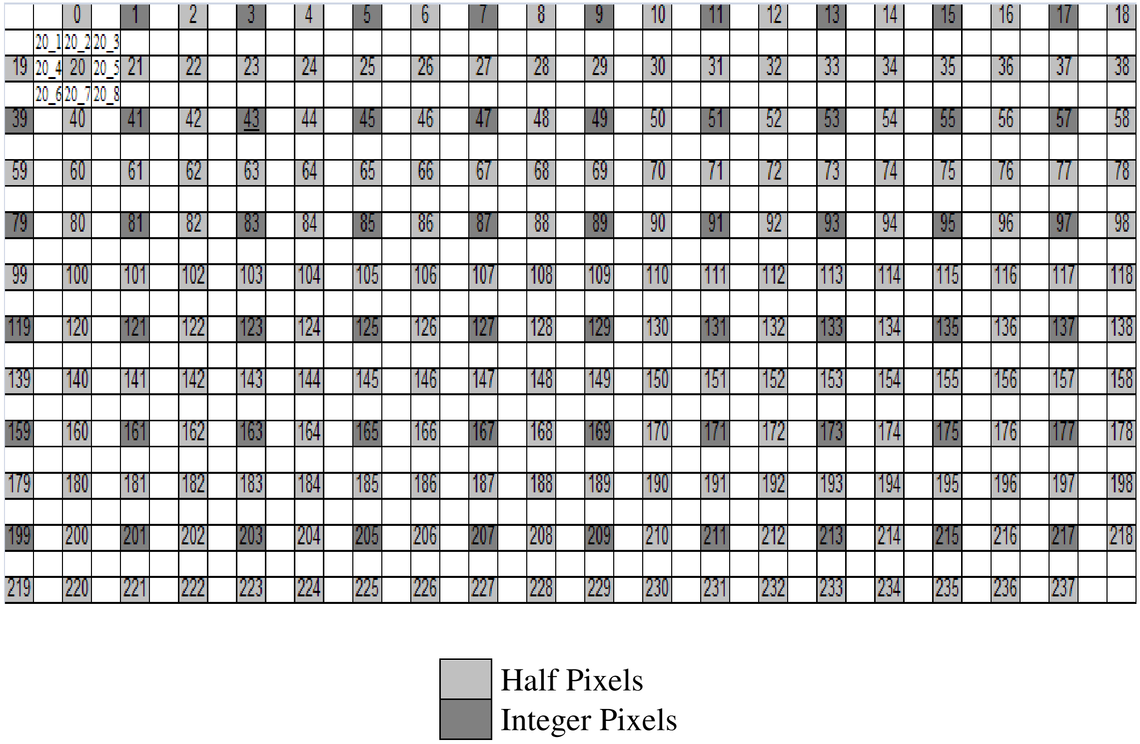
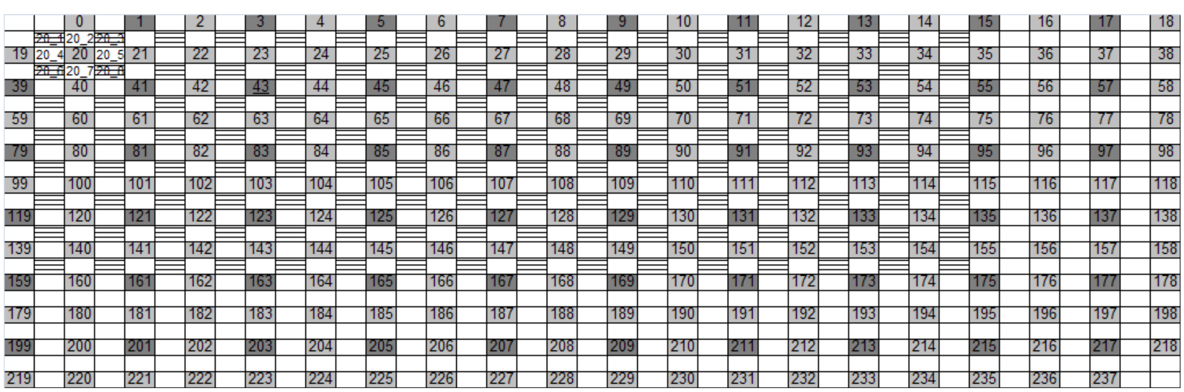
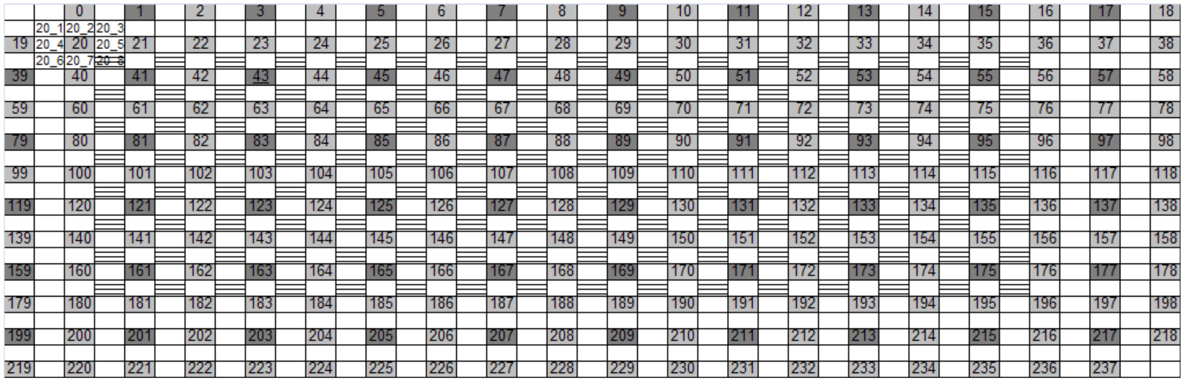


Figure 3.21 Search Window Register File for for 8x16 Block Size

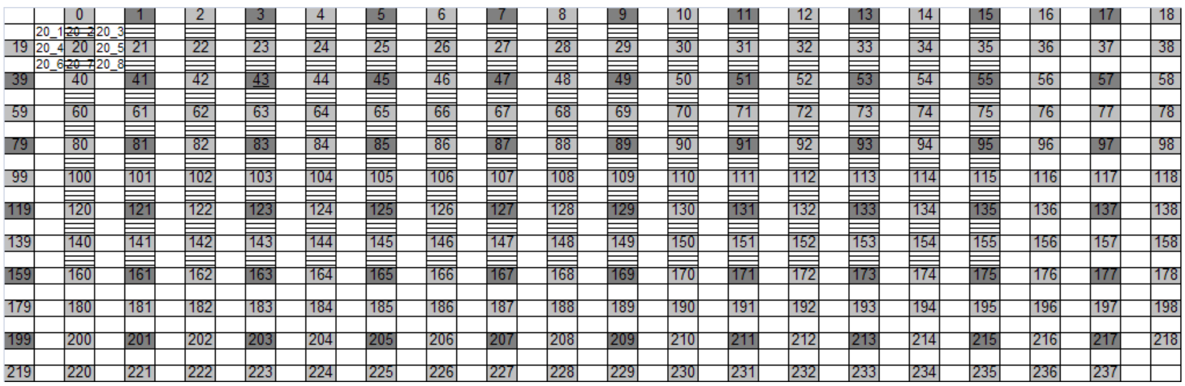
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 20_1, 20_2, 20_7 and 20_8 are shown in Figure 3.22. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 20_1, 20_2, 20_7 and 20_8 are shown in Figure 3.23. For example, the quarter pixels shown in Figure 3.22 (b) are necessary for calculating the SAD value for the quarter-pixel search location 20_8, and the integer and half pixels shown in Figure 3.23 (b) (21, 23, 25, 27, 29, …189, 191, 193,195) are used for generating these quarter pixels.



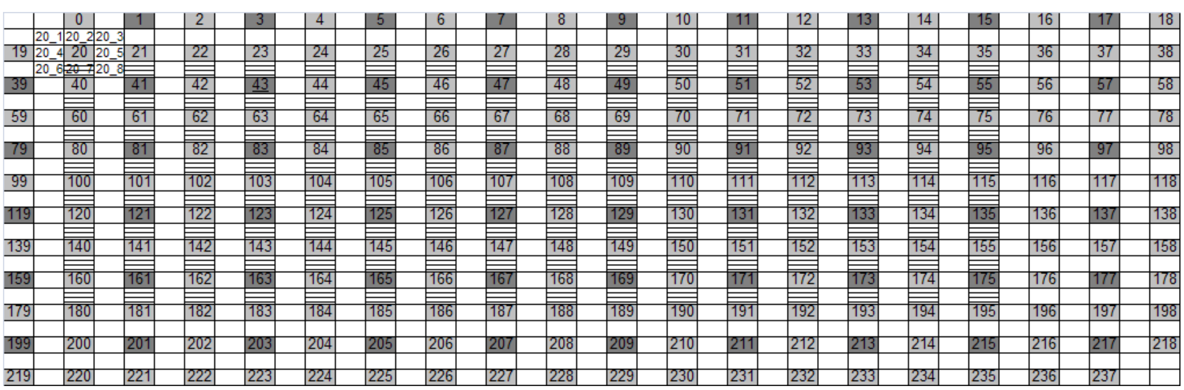
(a) Search location 20_1



(b) Search location 20_8



(c) Search location 20_2



(d) Search location 20_7

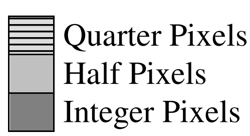
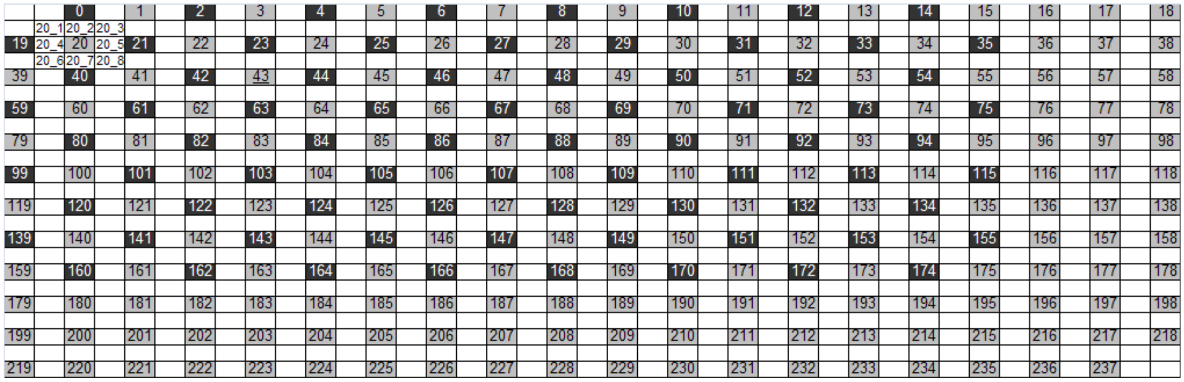
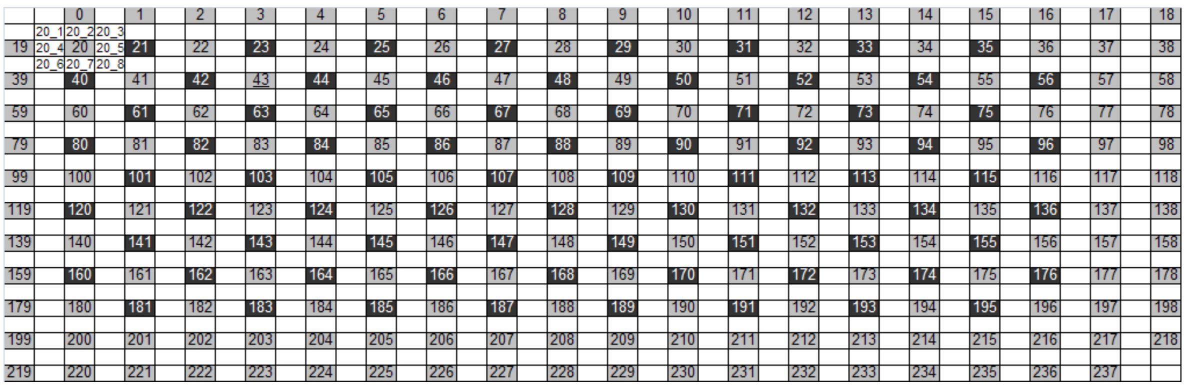


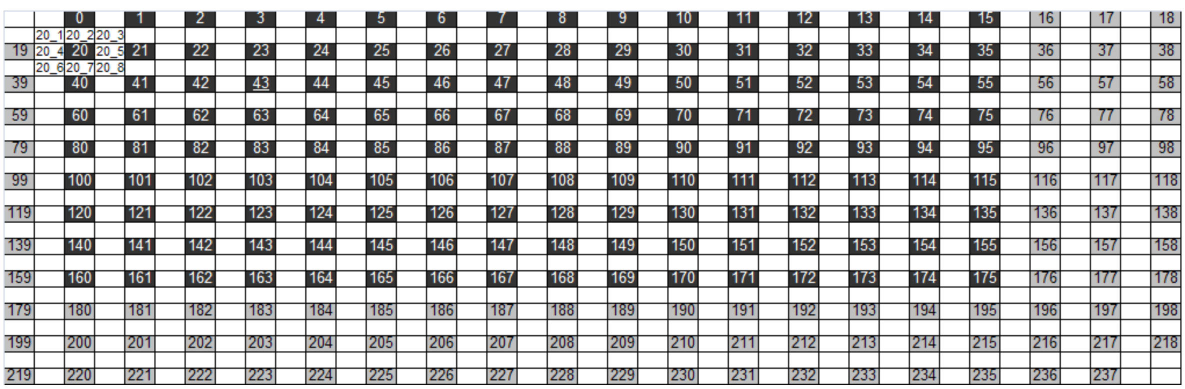
Figure 3.22 Quarter Pixels Necessary for 8x16 Block Size Quarter-Pixel Search Locations 20_1, 20_2, 20_7, 20_8



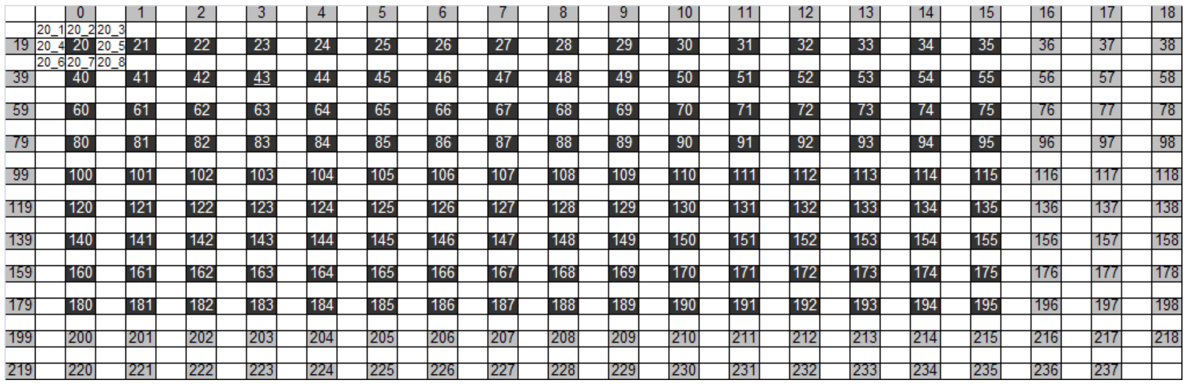
(a) Search location 20_1



(b) Search location 20_8



(c) Search location 20_2



(d) Search location 20_7

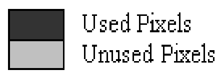


Figure 3.23 Integer and Half Pixels Used for 8x16 Block Size Quarter-Pixel Search Locations 20_1, 20_2, 20_7, 20_8

The proposed layout of the integer and half pixels in the 8x16 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.24. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 20_8 are 9 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_1. In Figure 3.24, this read address correlation between quarter-pixel search locations 20_1 and 20_8 is shown by writing the read address for location 20_8 as 20_1 + 21.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
20	20_1	20_2	20_3	20_4	20_4 + 1	20_3 + 19	20_2 + 20	20_1 + 21	row1
21	20_3	20_2 + 1	20_1 + 2	20_4 + 1	20_4 + 2	20_1 + 21	20_2 + 21	20_3 + 21	row2
22	20_1 + 2	20_2 + 2	20_3 + 2	20_4 + 2	20_4 + 3	20_3 + 21	20_2 + 22	20_1 + 23	row1 +2
40	20_3 + 19	20_2 + 20	20_1 + 21	20_4 + 19	20_4 + 20	20_1 + 38	20_2 + 39	20_3 + 40	row2 +19
42	20_3 + 21	20_2 + 22	20_1 + 23	20_4 + 21	20_4 + 22	20_1 + 40	20_2 + 41	20_3 + 42	row2 +21
60	20_1 + 40	20_2 + 40	20_3 + 40	20_4 + 40	20_4 + 41	20_3 + 59	20_2 + 60	20_1 + 61	row1 +40
61	20_3 + 40	20_2 + 41	20_1 + 42	20_4 + 41	20_4 + 42	20_1 + 61	20_2 + 61	20_3 + 61	row2 +40
62	20_1 + 42	20_2 + 42	20_3 + 42	20_4 + 42	20_4 + 43	20_3 + 61	20_2 + 62	20_1 + 63	row1 +42

Figure 3.24 Address Correlation of 8x16 Block Size Quarter-Pixel Search Locations

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 20_1, 20_2, 20_3, 20_4 and the read address correlations of 64 quarter-pixel search locations.

The SAD value for a quarter-pixel search location is calculated by a processing element in 128 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*128=1024 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

The proposed quarter-pixel interpolation and search flow for a 8x16 block is shown in Figure A.5. The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0’). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1’). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

3.2.6 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 16x8 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 16x8 block size is similar to 4x4 block size hardware shown in Figure 3.4. For each 16x8 block in a MB, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 16x8 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed layout of the integer and half pixels in the 16x8 search window register file, when the location pointed by the best integer-pixel MV is location 25, is shown in Figure 3.25 Since the half-pixel motion estimation will be performed at the half-pixel search locations 12, 13, 14, 24, 26, 36, 37 and 38, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 12, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 12_1, 12_2, 12_3, 12_4, 12_5, 12_6, 12_7 and 12_8.

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (12_1, 12_2, 12_3, … , 38_6, 38_7, 38_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.

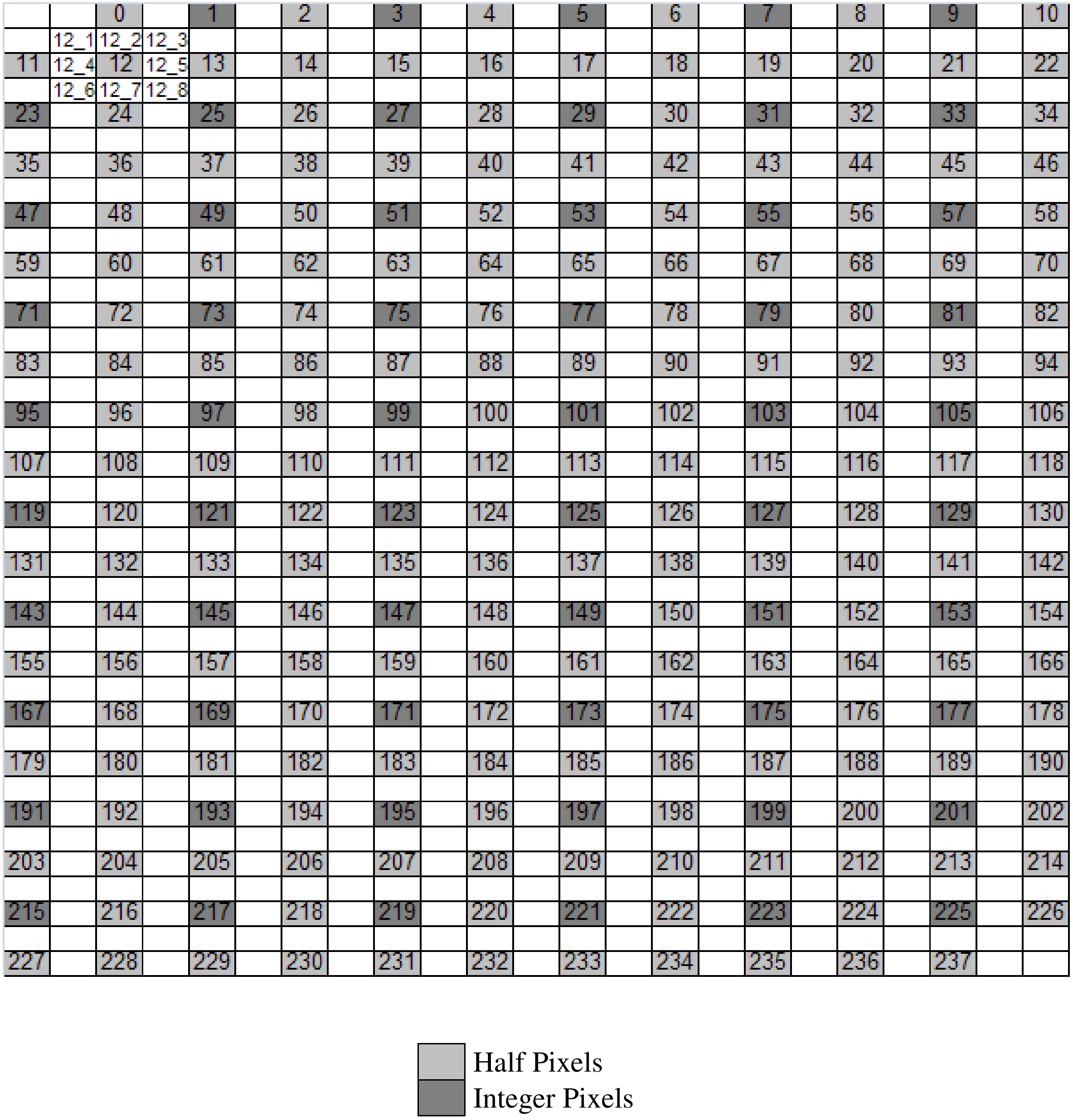
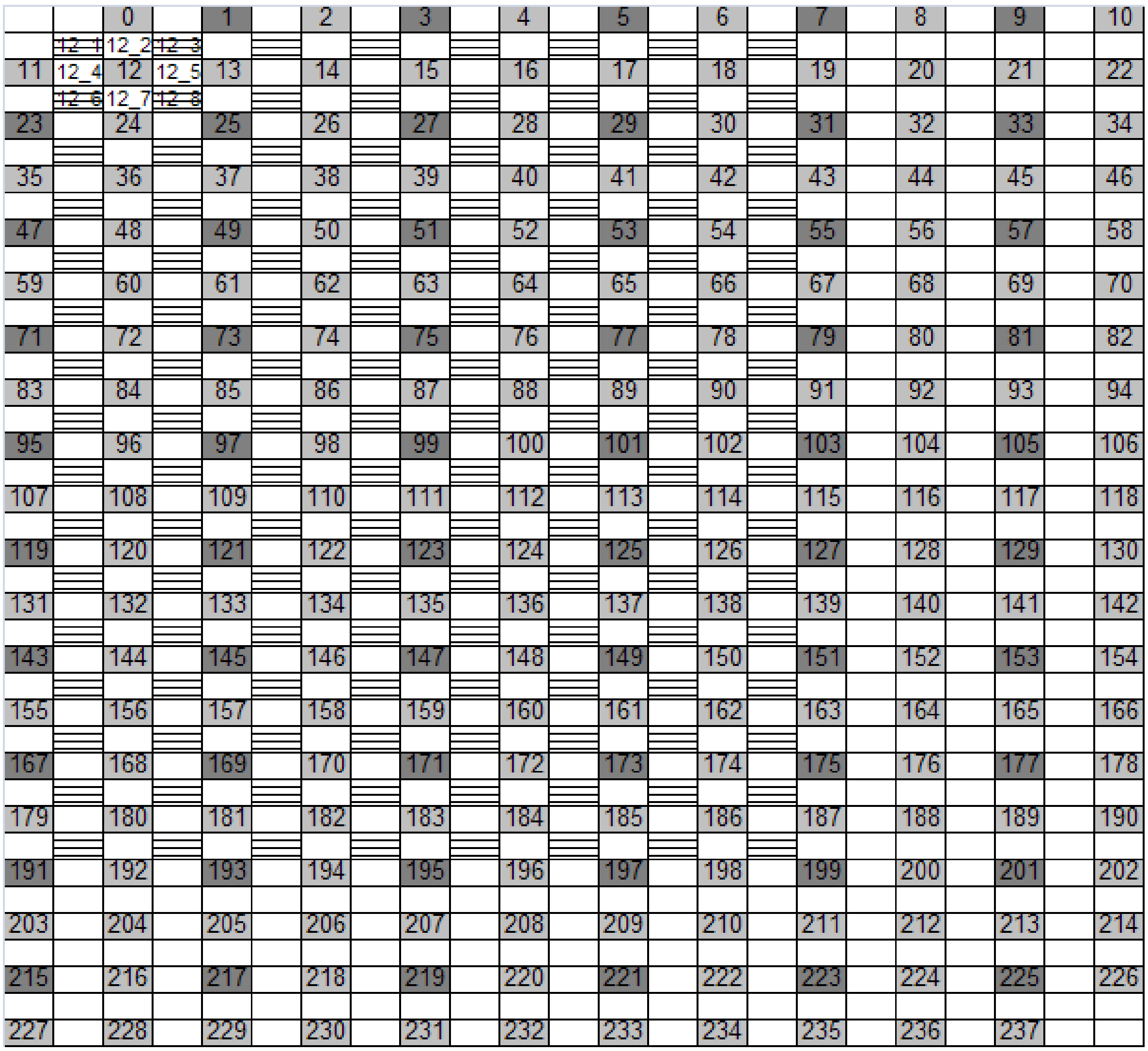
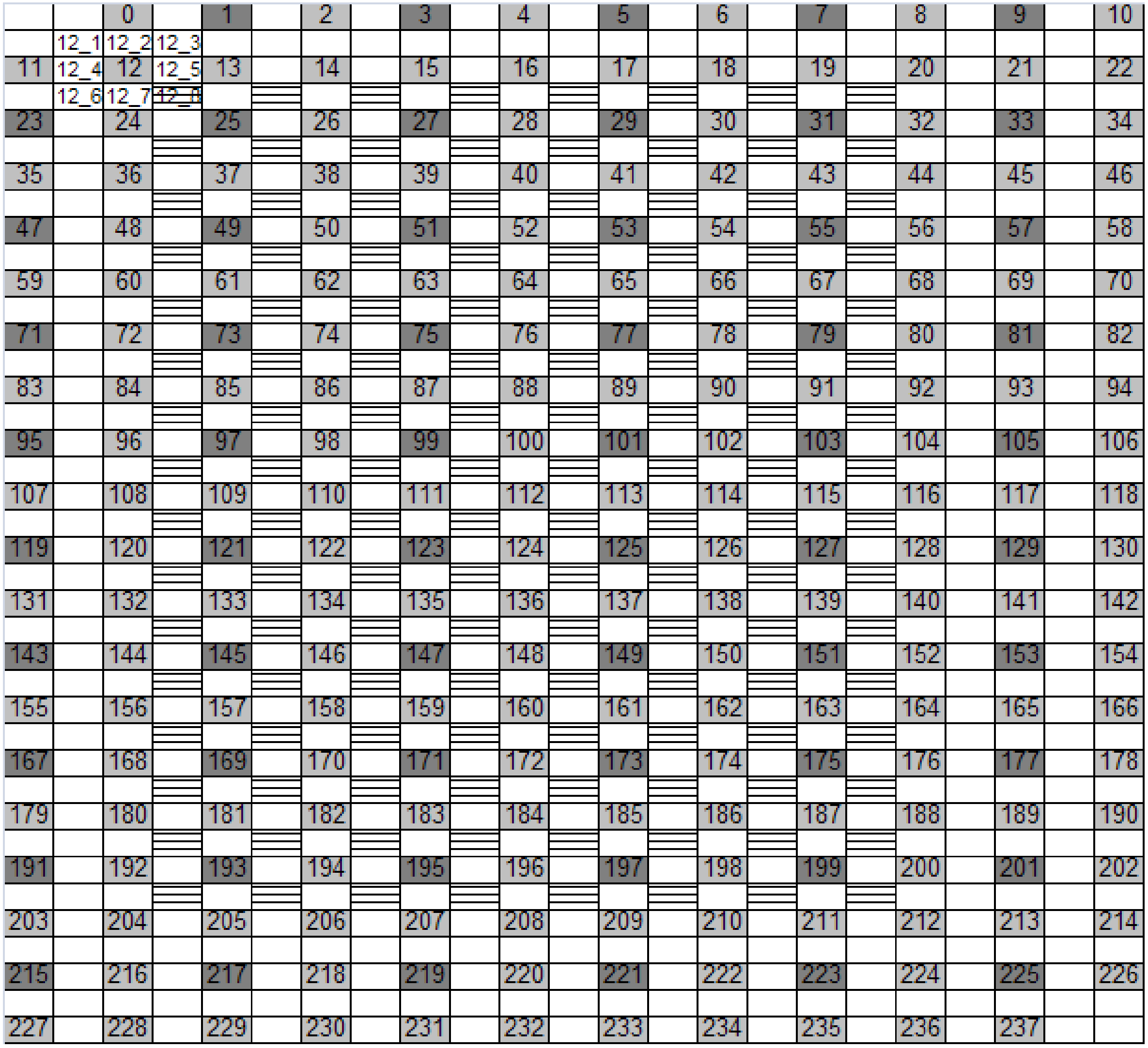


Figure 3.25 Search Window Register File for 16x8 Block Size

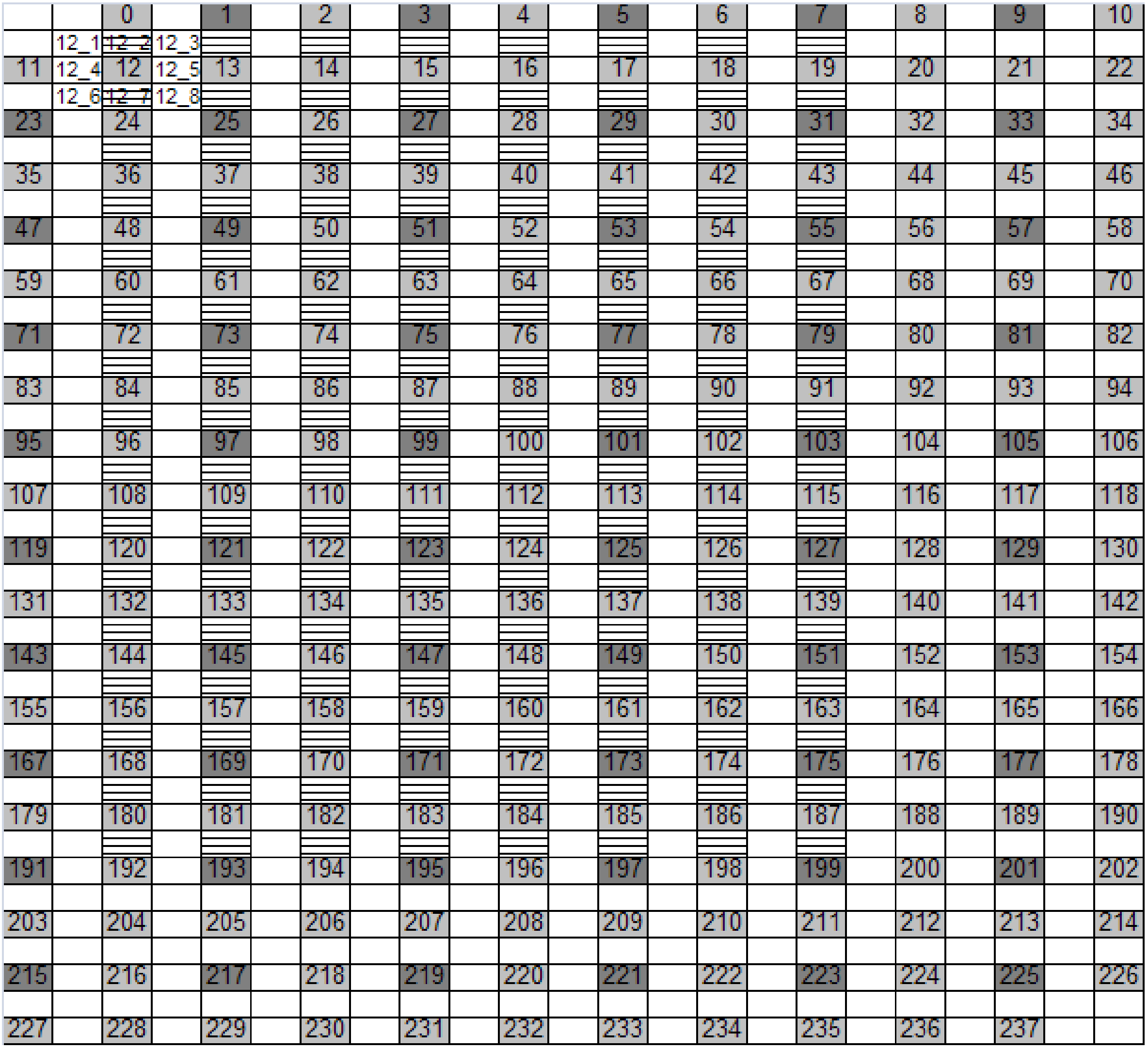
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 12_1, 12_2, 12_7 and 12_8 are shown in Figure 3.26. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 12_1, 12_2, 12_7 and 12_8 are shown in Figure 3.27. For example, the quarter pixels shown in Figure 3.26 (b) are necessary for calculating the SAD value for the quarter-pixel search location 12_8, and the integer and half pixels shown in Figure 3.27 (b) (13, 15, 17, 19, 24, 26, 28, 30, 32, 37, 39, 41, 43, 48, 50, 52, 54, 56, 61, 63, 65, 67, 72, 74, 76, 78, 80, 85, 87, 89, 91, 96, 98, 100, 102, 104, 109, 111, 113, 115, 120, 122, 124, 126, 128, 133, 135, 137, 139, 144, 146, 148, 150, 152, 157, 159, 161, 163, 168, 170, 172, 174, 176, 181, 183, 185, 187, 192, 194, 196, 198, 200, 207, 209, 211) are used for generating these quarter pixels.



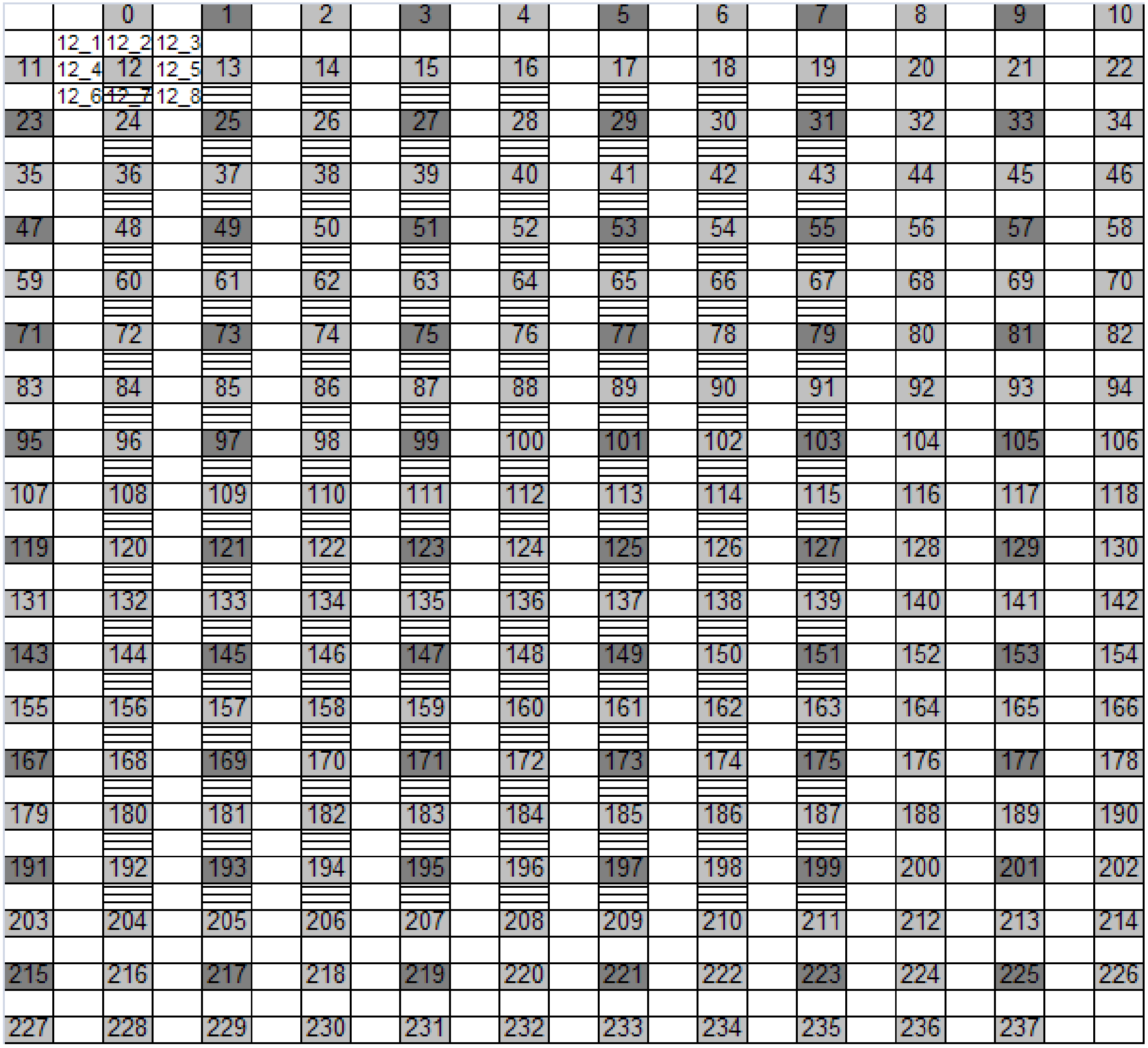
(a) Search location 12_1



(b) Search location 12_8



(c) Search location 12_2



(d) Search location 12_7

-

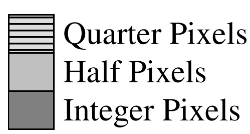
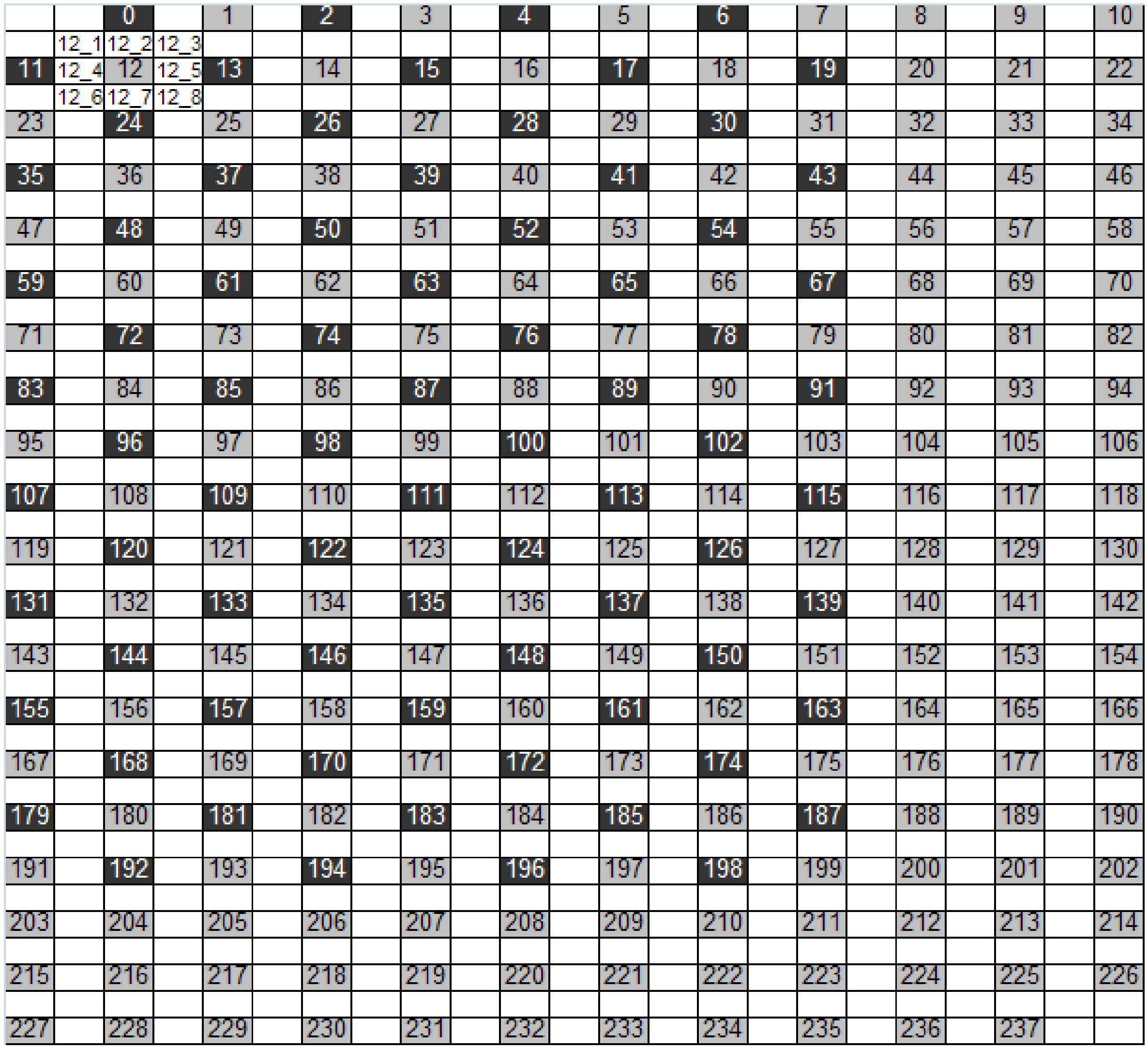


Figure 3.26 Quarter Pixels Necessary for 16x8 Block Size Quarter-Pixel Search Locations 12_1, 12_2, 12_7, 12_8



(a) Search location 12_1

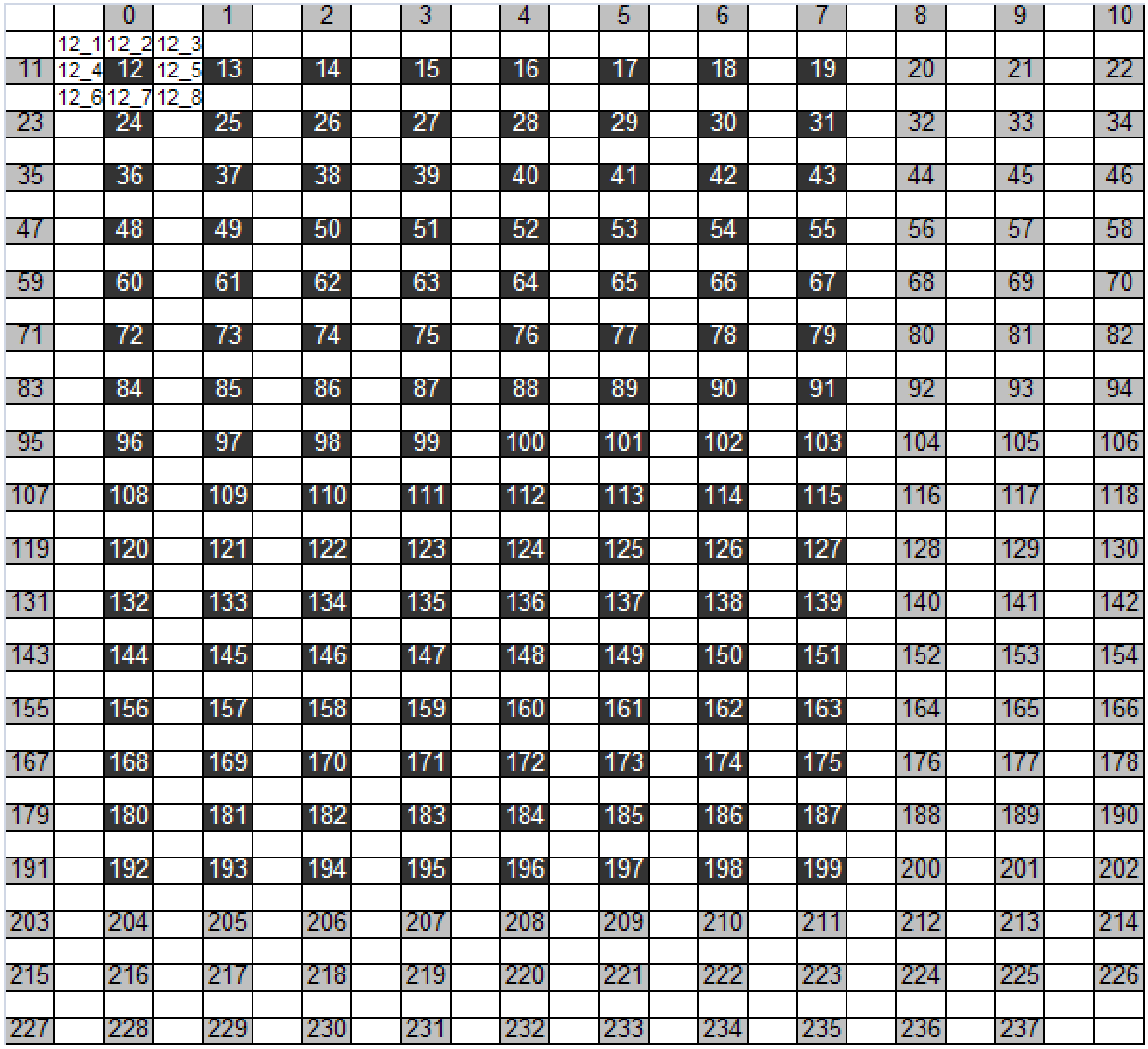
		0		1		2		3		4		5		6		7		8		9		10
	12_1	12_2	12_3																			
11	12_4	12	12_5	13		14		15		16		17		18		19		20		21		22
	12_6	12_7	12_8																			
23		24		25		26		27		28		29		30		31		32		33		34
35		36		37		38		39		40		41		42		43		44		45		46
47		48		49		50		51		52		53		54		55		56		57		58
59		60		61		62		63		64		65		66		67		68		69		70
71		72		73		74		75		76		77		78		79		80		81		82
83		84		85		86		87		88		89		90		91		92		93		94
95		96		97		98		99		100		101		102		103		104		105		106
107		108		109		110		111		112		113		114		115		116		117		118
119		120		121		122		123		124		125		126		127		128		129		130
131		132		133		134		135		136		137		138		139		140		141		142
143		144		145		146		147		148		149		150		151		152		153		154
155		156		157		158		159		160		161		162		163		164		165		166
167		168		169		170		171		172		173		174		175		176		177		178
179		180		181		182		183		184		185		186		187		188		189		190
191		192		193		194		195		196		197		198		199		200		201		202
203		204		205		206		207		208		209		210		211		212		213		214
215		216		217		218		219		220		221		222		223		224		225		226
227		228		229		230		231		232		233		234		235		236		237		

(b) Search location 12_8

		0		1	2	3	4	5	6	7	8	9	10
	12_1	12_2	12_3										
11	12_4	12	12_5	13	14	15	16	17	18	19	20	21	22
	12_6	12_7	12_8										
23		24		25	26	27	28	29	30	31	32	33	34
35		36		37	38	39	40	41	42	43	44	45	46
47		48		49	50	51	52	53	54	55	56	57	58
59		60		61	62	63	64	65	66	67	68	69	70
71		72		73	74	75	76	77	78	79	80	81	82
83		84		85	86	87	88	89	90	91	92	93	94
95		96		97	98	99	100	101	102	103	104	105	106
107		108		109	110	111	112	113	114	115	116	117	118
119		120		121	122	123	124	125	126	127	128	129	130
131		132		133	134	135	136	137	138	139	140	141	142
143		144		145	146	147	148	149	150	151	152	153	154
155		156		157	158	159	160	161	162	163	164	165	166
167		168		169	170	171	172	173	174	175	176	177	178
179		180		181	182	183	184	185	186	187	188	189	190
191		192		193	194	195	196	197	198	199	200	201	202
203		204		205	206	207	208	209	210	211	212	213	214
215		216		217	218	219	220	221	222	223	224	225	226
227		228		229	230	231	232	233	234	235	236	237	

(c) Search location 12_2

-



(d) Search location 12_7

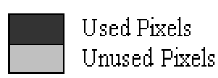


Figure 3.27 Integer and Half Pixels Used for 16x8 Block Size Quarter-Pixel Search Locations 12_1, 12_2, 12_7, 12_8

The proposed layout of the integer and half pixels in the 16x8 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.28. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 12_8 are 13 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 12_1. In Figure 3.28, this read address correlation between quarter-pixel search locations 12_1 and 12_8 is shown by writing the read address for location 12_8 as 12_1 + 13.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
12	12_1	12_2	12_3	12_4	12_4 + 1	12_3 + 11	12_2 + 12	12_1 + 13	row1
13	12_3	12_2 + 1	12_1 + 2	12_4 + 1	12_4 + 2	12_1 + 13	12_2 + 13	12_3 + 13	row2
14	12_1 + 2	12_2 + 2	12_3 + 2	12_4 + 2	12_4 + 3	12_3 + 13	12_2 + 14	12_1 + 15	row1 +2
24	12_3 + 11	12_2 + 12	12_1 + 13	12_4 + 12	12_4 + 13	12_1 + 24	12_2 + 24	12_3 + 24	row2 +11
26	12_3 + 13	12_2 + 14	12_1 + 15	12_4 + 14	12_4 + 15	12_1 + 26	12_2 + 26	12_3 + 26	row2 +13
36	12_1 + 24	12_2 + 24	12_3 + 24	12_4 + 24	12_4 + 25	12_3 + 35	12_2 + 36	12_1 + 37	row1 +24
37	12_3 + 24	12_2 + 25	12_1 + 26	12_4 + 25	12_4 + 26	12_1 + 37	12_2 + 37	12_3 + 37	row2 +24
38	12_1 + 26	12_2 + 26	12_3 + 26	12_4 + 26	12_4 + 27	12_3 + 37	12_2 + 38	12_1 + 39	row1 +26

Figure 3.28 Address Correlation of 16x8 Block Size Quarter-Pixel Search Locations

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 12_1, 12_2, 12_3, 12_4 and the read address correlations of 64 quarter-pixel search locations.

The SAD value for a quarter-pixel search location is calculated by a processing element in 128 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*128=1024 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

The proposed quarter-pixel interpolation and search flow for a 16x8 block is shown in Figure A.6. The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0'). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1'). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

3.2.7 Proposed Quarter-Pixel Accurate Motion Estimation Hardware Architecture for 16x16 Block Size

The proposed quarter-pixel accurate motion estimation hardware for 16x16 block size is similar to 4x4 block size hardware shown in Figure 3.4. For a 16x16 block, first, half-pixel motion estimation hardware finds the best half-pixel MV by performing half-pixel interpolation and half-pixel search and sends this half-pixel MV to quarter-pixel motion estimation hardware. Then, quarter-pixel motion estimation hardware finds the best quarter-pixel MV for that 16x16 block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed layout of the integer and half pixels in the 16x16 search window register file, when the location pointed by the best integer-pixel MV is location 17, is shown in Figure 3.29. Since the half-pixel motion estimation will be performed at the half-pixel search locations 20, 21, 22, 40, 42, 60, 61 and 62, the best half-pixel MV will point to one of these locations and the quarter-pixel motion estimation will be performed at the eight quarter-pixel search locations around that location. For example, if the best half-pixel MV points to location 20, quarter-pixel motion estimation will be performed at the quarter-pixel search locations 20_1, 20_2, 20_3, 20_4, 20_5, 20_6, 20_7 and 20_8.

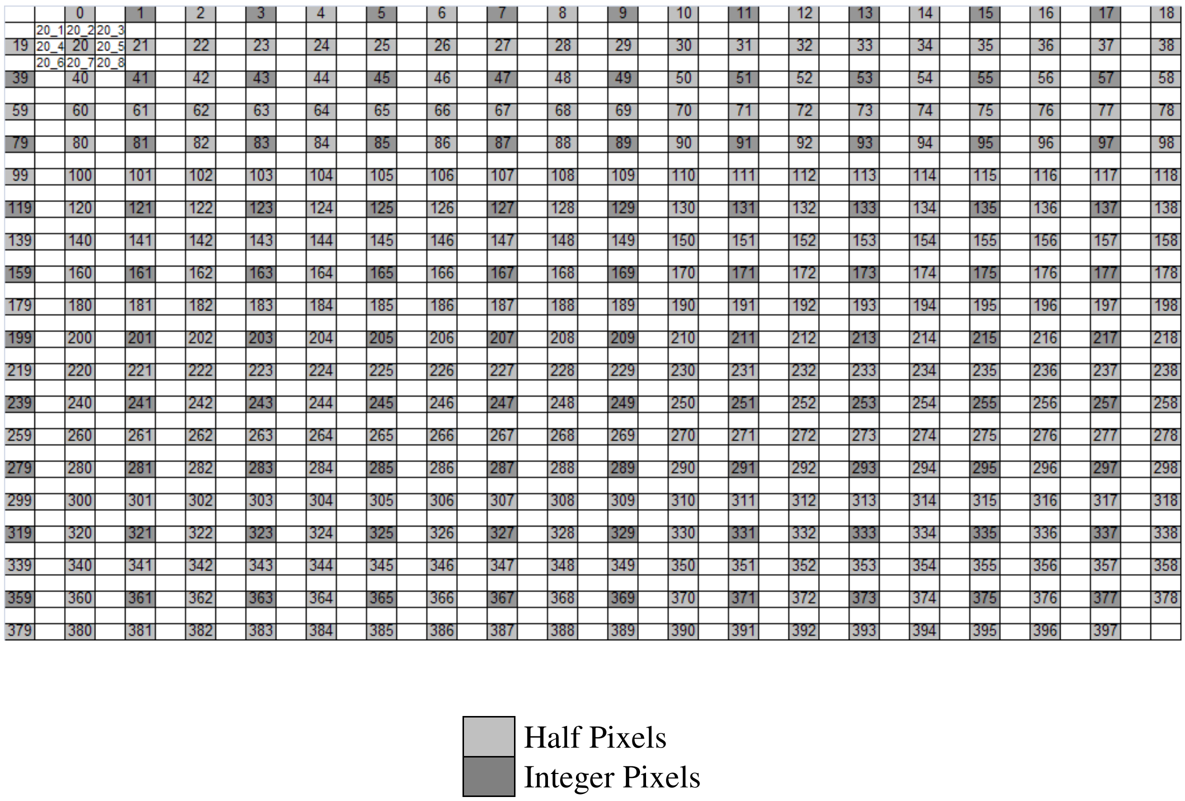
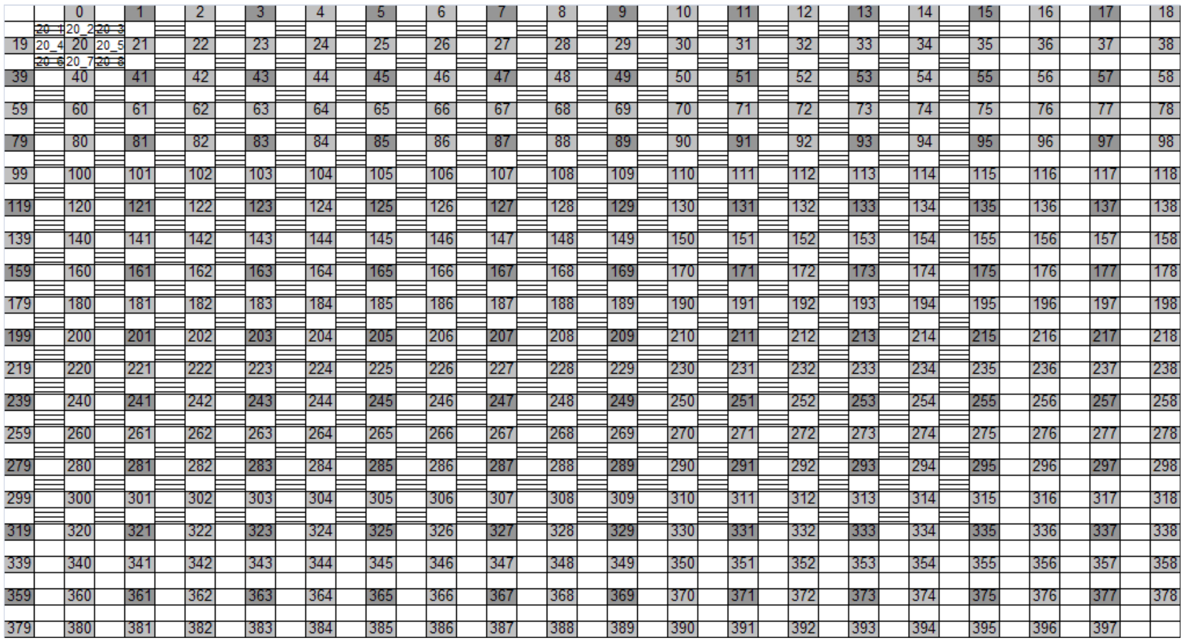


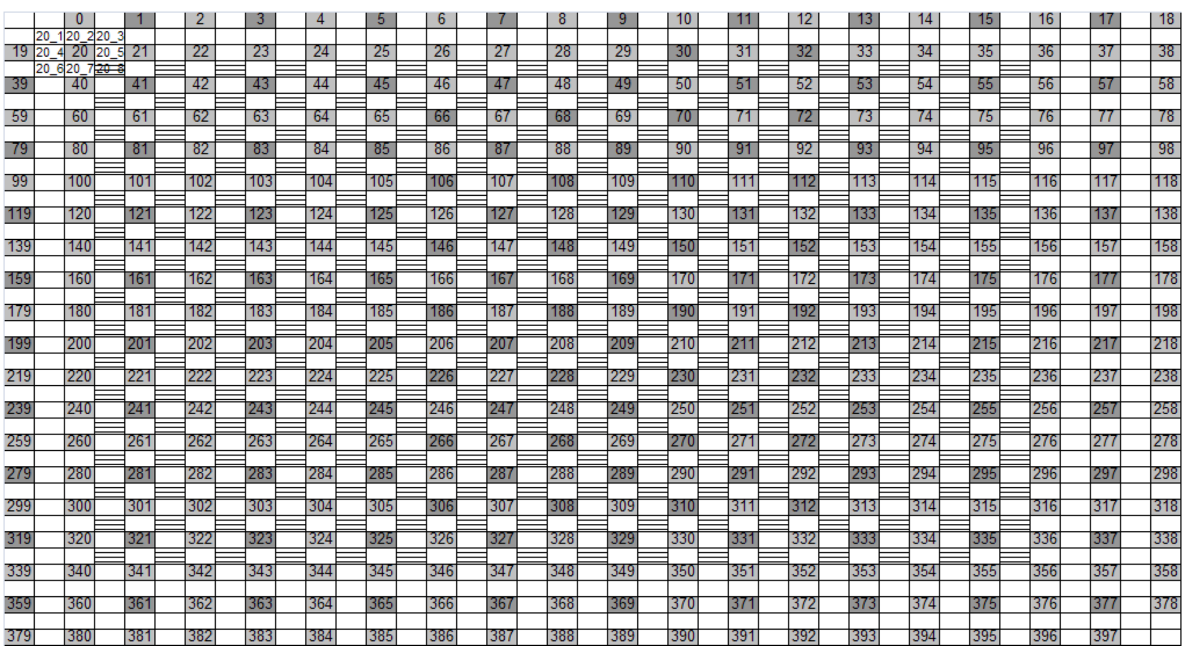
Figure 3.29 Search Window Register File for 16x16 Block Size

The control unit sends the read addresses to search window register file based on the best half-pixel MV for accessing the necessary integer and half pixels. Since there are eight half-pixel search locations and there are eight quarter-pixel search locations for each half-pixel search location, the control unit must be able to generate read addresses for 64 quarter-pixel search locations (20_1, 20_2, 20_3, … , 62_6, 62_7, 62_8). The quarter-pixel interpolation datapaths generate the quarter pixels and send them to processing elements. The SAD values for quarter-pixel search locations are calculated by the processing elements PE0, PE1, PE2 and PE3.

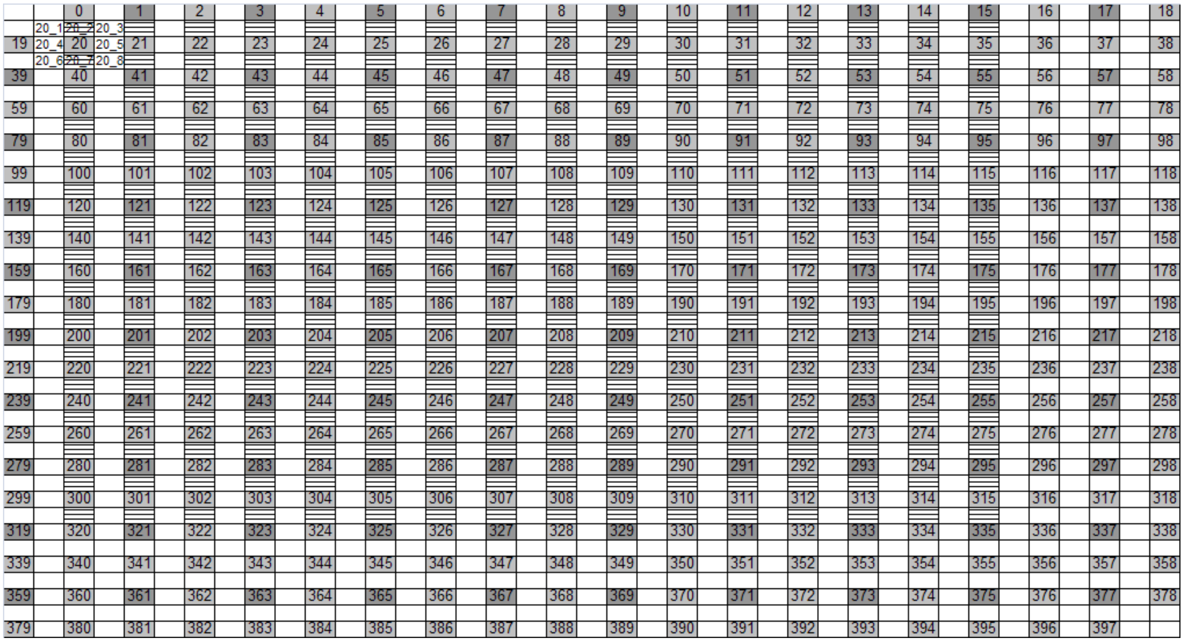
Quarter pixels necessary for calculating the SAD values for the quarter-pixel search locations 20_1, 20_2, 20_7 and 20_8 are shown in Figure 3.30. The integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search locations 20_1, 20_2, 20_7 and 20_8 are shown in Figure 3.31. For example, the quarter pixels shown in Figure 3.30 (b) are necessary for calculating the SAD value for the quarter-pixel search location 20_8, and the integer and half pixels shown in Figure 3.31 (b) (21, 23, 25, 27, 29, …349, 351, 353, 355) are used for generating these quarter pixels.



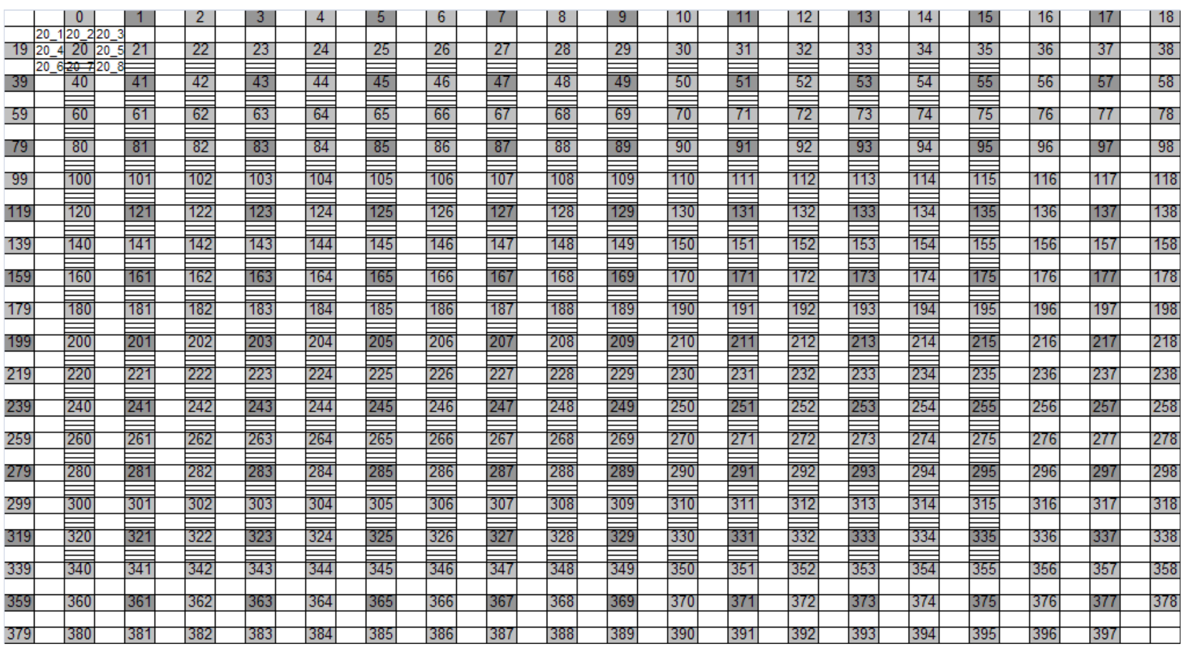
(a) Search location 20_1



(b) Search location 20_8



(c) Search location 20_2



(d) Search location 20_7

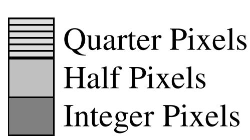
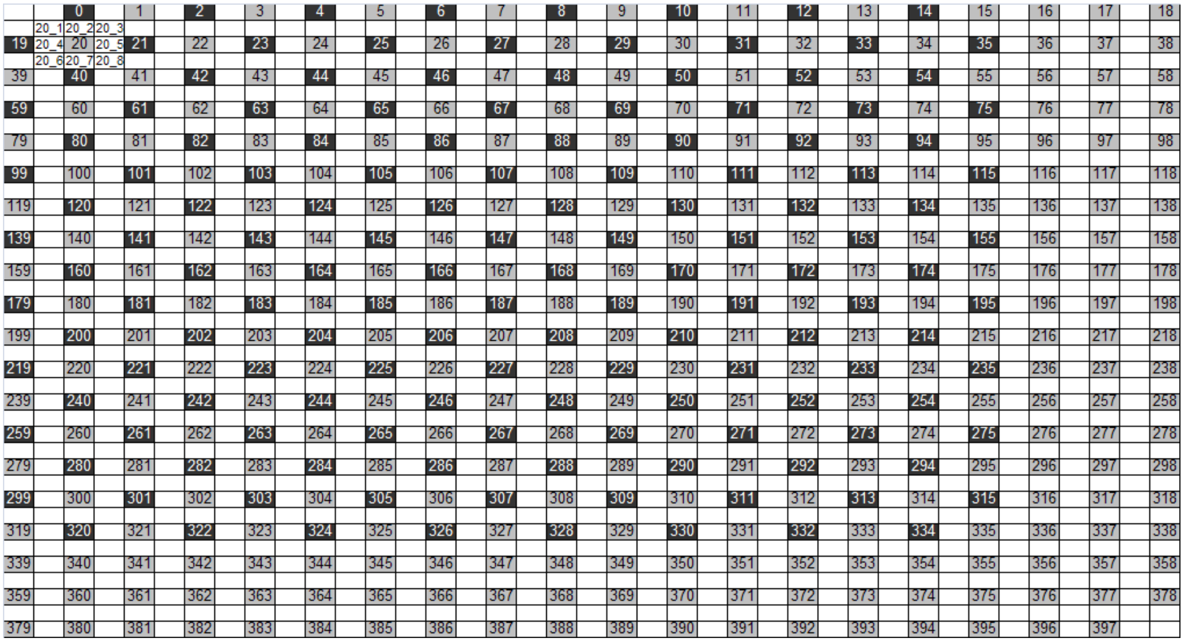
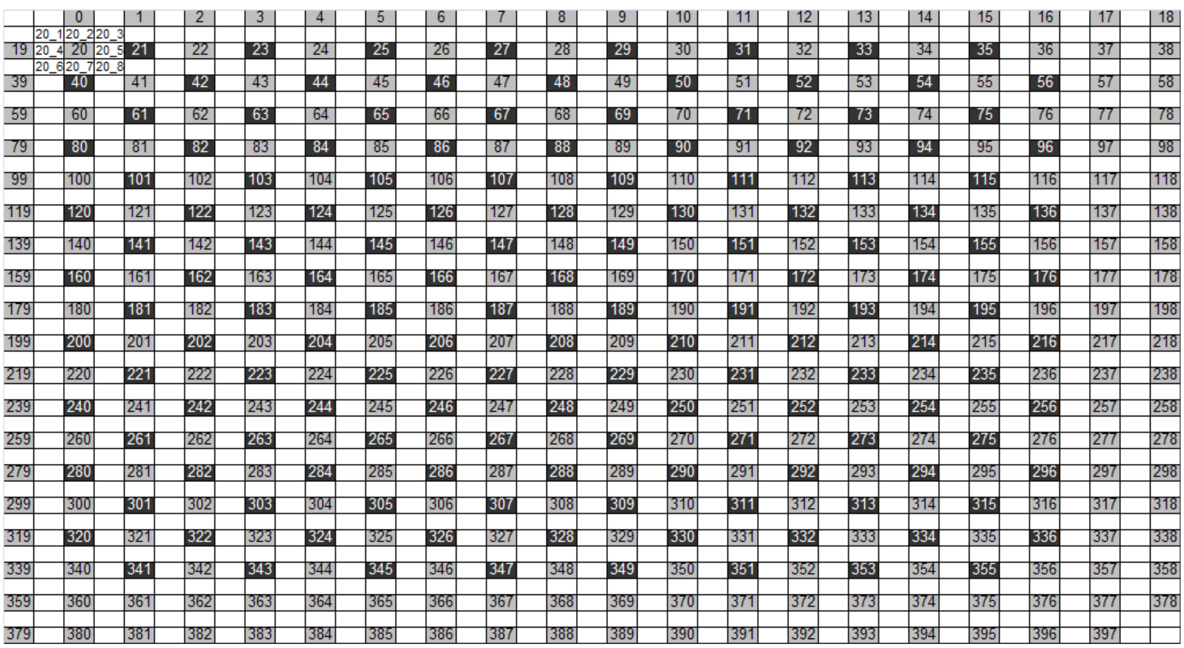


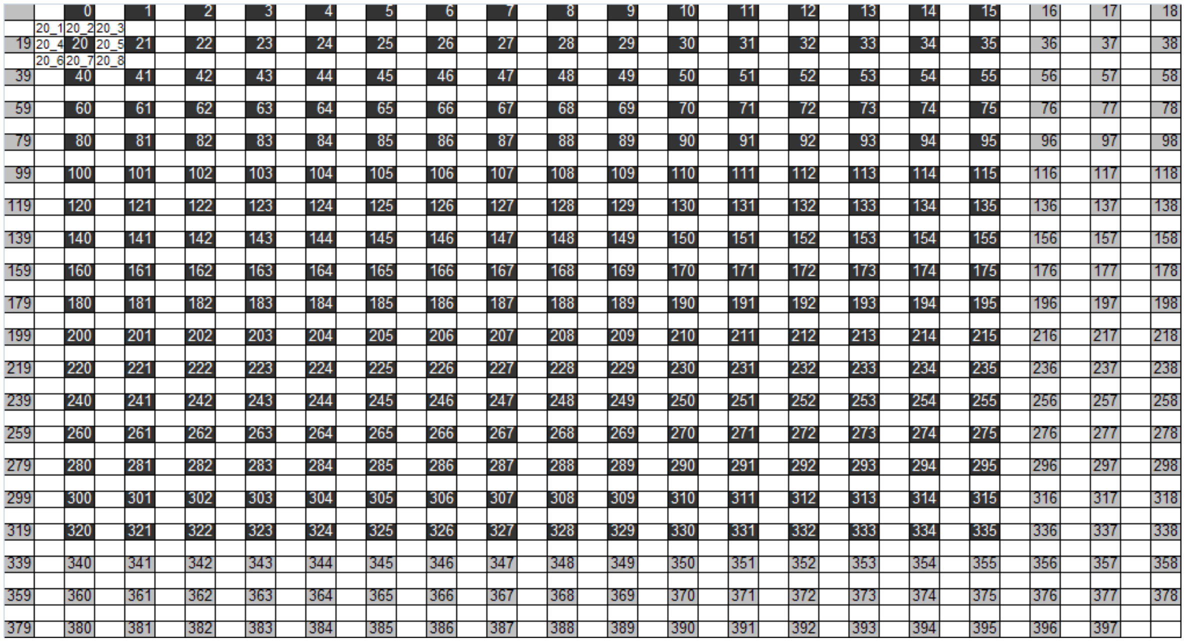
Figure 3.30 Quarter Pixels Necessary for 16x16 Block Size Quarter-Pixel Search Locations 20_1 , 20_2, 20_7, 20_8



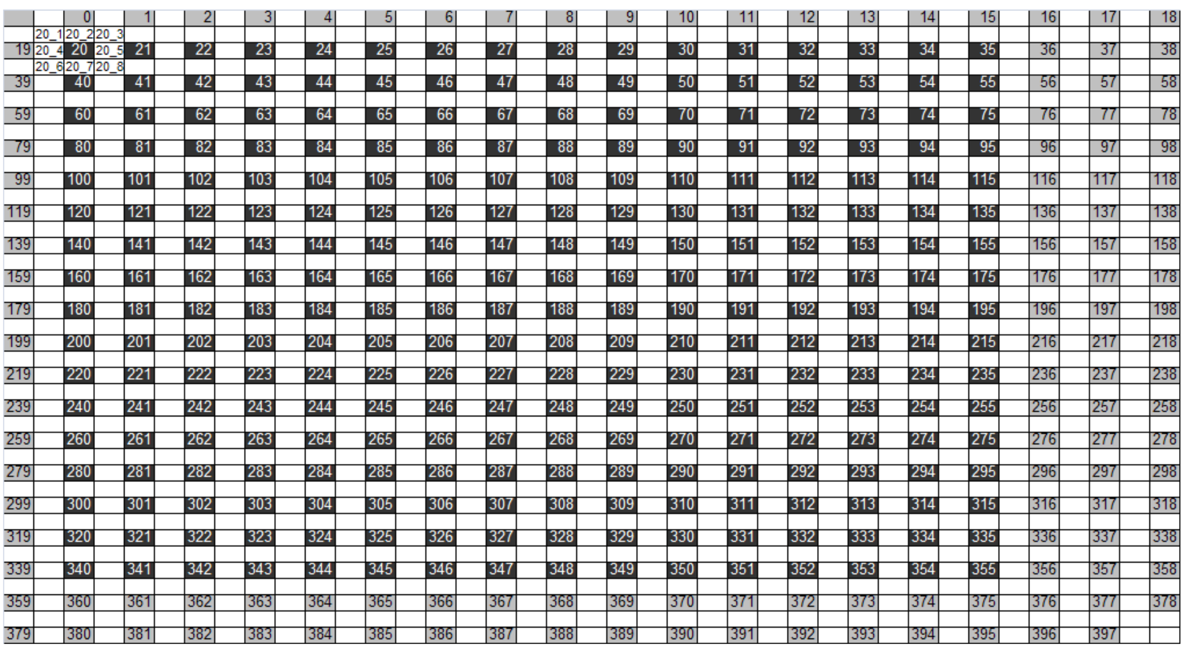
(a) Search location 20_1



(b) Search location 20_8



(c) Search location 20_2



(d) Search location 20_7

Figure 3.31 Integer and Half Pixels Used for 16x16 Block Size Quarter-Pixel Search Locations 20_1, 20_2, 20_7, 20_8

The proposed layout of the integer and half pixels in the 16x16 search window register file provide a good correlation between the read addresses of 64 quarter-pixel search locations. The read address correlations of 64 quarter-pixel search locations are shown in Figure 3.32. For example, the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 20_8 are 9 more than the read addresses of the integer and half pixels used for generating the quarter pixels necessary for quarter-pixel search location 8_1. In Figure 3.32, this read address correlation between quarter-pixel search locations 20_1 and 20_8 is shown by writing the read address for location 20_8 as 20_1 + 21.

Half-Pel Search Locations	Quarter-Pel Search Locations								Address Correlation
	1	2	3	4	5	6	7	8	
20	20_1	20_2	20_3	20_4	20_4 + 1	20_3 + 19	20_2 + 20	20_1 + 21	row1
21	20_3	20_2 + 1	20_1 + 2	20_4 + 1	20_4 + 2	20_1 + 21	20_2 + 21	20_3 + 21	row2
22	20_1 + 2	20_2 + 2	20_3 + 2	20_4 + 2	20_4 + 3	20_3 + 21	20_2 + 22	20_1 + 23	row1 +2
40	20_3 + 19	20_2 + 20	20_1 + 21	20_4 + 19	20_4 + 20	20_1 + 38	20_2 + 39	20_3 + 40	row2 +19
42	20_3 + 21	20_2 + 22	20_1 + 23	20_4 + 21	20_4 + 22	20_1 + 40	20_2 + 41	20_3 + 42	row2 +21
60	20_1 + 40	20_2 + 40	20_3 + 40	20_4 + 40	20_4 + 41	20_3 + 59	20_2 + 60	20_1 + 61	row1 +40
61	20_3 + 40	20_2 + 41	20_1 + 42	20_4 + 41	20_4 + 42	20_1 + 61	20_2 + 61	20_3 + 61	row2 +40
62	20_1 + 42	20_2 + 42	20_3 + 42	20_4 + 42	20_4 + 43	20_3 + 61	20_2 + 62	20_1 + 63	row1 +42

Figure 3.32 Address Correlation of 16x16 Block Size Quarter-Pixel Search Locations

Therefore, the control unit generates the read addresses of 64 quarter-pixel search locations by using the read addresses of the quarter-pixel search locations 20_1, 20_2, 20_3, 20_4 and the read address correlations of 64 quarter-pixel search locations.

The SAD value for a quarter-pixel search location is calculated by a processing element in 256 clock cycles. Since there are 8 quarter-pixel search locations, quarter-pixel search would take 8*256=2048 clock cycles using one PE. We used 4 PEs in order to perform the quarter-pixel search operation faster. Each PE calculates the SAD for two quarter-pixel search locations. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV.

The proposed quarter-pixel interpolation and search flow for a 16x16 block is shown in Figure A.7. The calculations done by each PE in this flow is organized to reduce the number of read ports of the search window and current block register files and to reduce the number of read accesses to these register files.

Because of the proposed allocation of quarter-pixel search locations to PEs and the proposed flow, the search window register file has four 8-bit read ports (s0, s1, s2 and s3), and the current block register file has two 8-bit read ports (c0 and c1). PE0 and PE1 use s0, s1 and c0 ports, PE2 and PE3 use s2, s3 and c1 ports. PE1 can reuse the current block pixel accessed by PE0 in a previous clock cycle (c0'). Similarly, PE3 can reuse the current block pixel accessed by PE2 in a previous clock cycle (c1'). In addition, PE0 and PE1 can use the same search window pixels in the same clock cycle. Similarly, PE2 and PE3 can use the same search window pixels in the same clock cycle. In order to achieve these, PEs do not perform any calculation in some clock cycles.

3.2.8 Implementation Results

The proposed quarter-pixel motion estimation hardware is implemented in Verilog HDL. Quarter-pixel interpolation and search take 44 clock cycles for a 4x4 block. Since there are 16 4x4 blocks in a MB, quarter-pixel motion estimation for a MB for 4x4 block size takes 16*44 = 704 clock cycles. Quarter-pixel interpolation and search for an 8x4 block size take 80 clock cycles. Since there are 8 8x4 blocks in a MB, quarter-pixel motion estimation for a MB for 8x4 block size takes 8*80 = 640 clock cycles. Similarly, quarter-pixel motion estimation for a MB for 4x8, 8x8, 16x8, 8x16 and 16x16 block sizes take 608, 576, 576, 558 and 544 clock cycles respectively. Therefore, 4x4 block size is the bottleneck.

The proposed quarter-pixel motion estimation hardware is designed to be used as part of a complete H.264 video coding system for portable applications together with the half-pixel accurate motion estimation hardware presented in [11]. The half-pixel interpolation and search take 48 clock cycles for a 4x4 block and 4x4 block size is the bottleneck for half-pixel motion estimation hardware as well. Therefore, sub-pixel motion estimation for a 4x4 block takes 48+44 = 92 clock cycles and sub-pixel motion estimation for a MB takes 16*92=1472 clock cycles.

The Verilog HDL implementation of the quarter-pixel motion estimation hardware is verified with RTL simulations using Mentor Graphics ModelSim. The Verilog RTL is then synthesized to a 2V8000ff1152 Xilinx Virtex II FPGA with speed grade 6 using Mentor Graphics Leonardo Spectrum. The resulting netlist is placed and routed to the same FPGA using Xilinx ISE Series 7.1.

The FPGA implementation is verified to work at 60 MHz under worst-case PVT conditions with post place and route simulations. The FPGA implementation can process an VGA frame in 29.32 msec (1200 MB * 1472 cycles per MB * 16.6 ns clock cycle = 29.32 msec). Therefore, it can process 1000/29.32 = 34 VGA frames (640x480) per second.

The FPGA implementation uses the following FPGA resources; 18566 CLB Slices, 37131 Function Generators, and 21339 DFFs, i.e. % 39 of CLB Slices, %39 of Function Generators, and %22 of DFFs.

CHAPTER 4

SUB-PIXEL ACCURATE H.264 MOTION ESTIMATION HARDWARE DESIGN

We presented a half-pixel accurate ME hardware for 4x4 block size in Chapter 2. Sinan Yalcin has implemented a half-pixel accurate variable block size ME hardware by scaling this half-pixel accurate ME hardware for all block sizes [11]. We integrated this half-pixel accurate variable block size ME hardware with the quarter pixel accurate variable block size ME hardware presented in Chapter 3 to implement a sub-pixel accurate variable block size ME hardware.

The sub-pixel accurate variable block size ME hardware performs half-pixel interpolation, half-pixel search, quarter-pixel interpolation, and quarter-pixel search for each block size. For each block in a MB, first, half-pixel interpolation hardware calculates the half pixels in the half-pixel search window of that block. Then, half-pixel search hardware searches the half-pixel search locations and determines the best half-pixel MV for that block and sends this half-pixel MV to quarter-pixel ME hardware. Then, quarter-pixel ME hardware finds the best quarter-pixel MV for that block by performing quarter-pixel search around the location pointed by this half-pixel MV with a search range of [-1, 1].

The proposed hardware includes novel half-pixel and quarter-pixel interpolation and search hardwares designed for each block size. In the proposed hardware, half-pixel interpolation hardwares are shared by half-pixel search hardwares for reducing area. The proposed hardware performs quarter-pixel interpolation dynamically, i.e. only the quarter pixels necessary for performing quarter-pixel accurate search at the location pointed by the best half-pixel motion vector are calculated, for reducing the amount of computation performed for quarter-pixel interpolation and therefore reducing the power consumption.

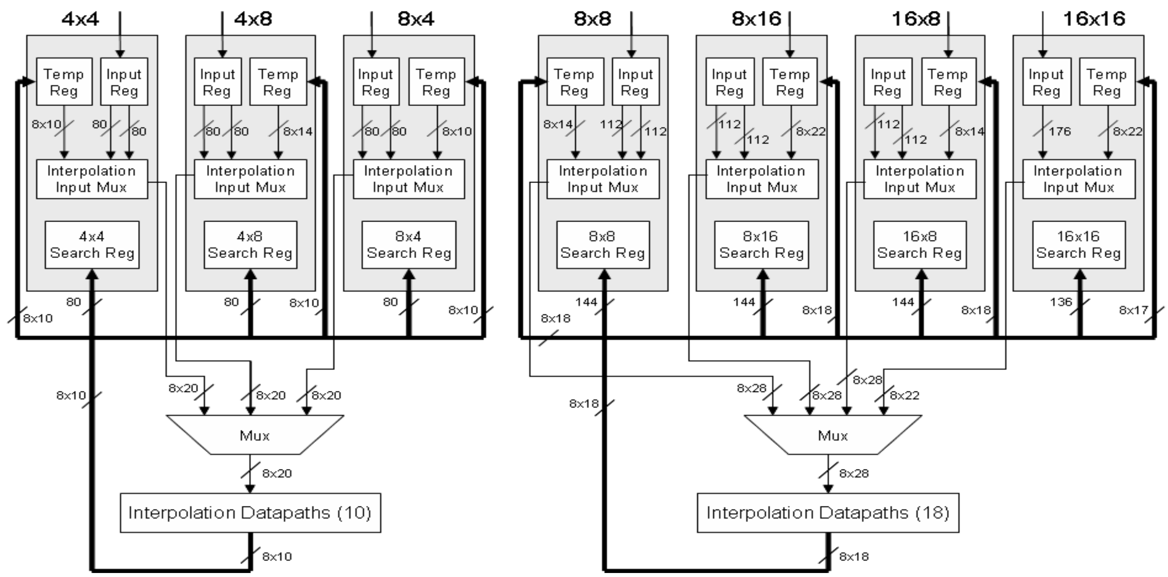


Figure 4.1 Half-Pixel Interpolation Hardware

4.1 Half-Pixel Accurate Motion Estimation Hardware

The half-pixel interpolation hardware for all block sizes is shown in Figure 4.1. If dedicated half-pixel interpolation datapaths are used for each block size, the half-pixel interpolation datapaths would be idle during half-pixel search. Therefore, in the hardware, half-pixel interpolation datapaths are shared by different block sizes. For example, ten half-pixel interpolation datapaths are shared by 4x4, 4x8 and 8x4 block sizes. During the half-pixel search of 4x4 blocks, these half-pixel interpolation datapaths are used for half-pixel interpolation of 4x8 and 8x4 blocks. This reduces the area of the half-pixel interpolation hardware significantly without increasing the cycle count.

0-0		0-1		0-2		0-3		0-4		0-5		0-6		0-7		0-8		0-9		0-10		0-11		0-12		0-13
1--0		1-1		1-2		1-3		1-4		1-5		1-6		1-7		1-8		1-9		1-10		1-11		1-12		1-13
2--0		2-1		2-2		2-3		2-4		2-5		2-6		2-7		2-8		2-9		2-10		2-11		2-12		2-13
A000		*A001*		*A002*	C00	*A003*	C01	*A004*	C02	*A005*	C03	*A006*	C04	*A007*	C05	*A008*	C06	*A009*	C07	*A010*	C08	*A011*		*A012*		*A013*
3-0		3-1		3-2	B00	3-3	B01	3-4	B02	3-5	B03	3-6	B04	3-7	B05	3-8	B06	3-9	B07	3-10	B08	3-11		3-12		3-13
A100		*A101*		*A102*	C10	*A103*	C11	*A104*	C12	*A105*	C13	*A106*	C14	*A107*	C15	*A108*	C16	*A109*	C17	*A110*	C18	*A111*		*A112*		*A113*
4-0		4-1		4-2	B10	4-3	B11	4-4	B12	4-5	B13	4-6	B14	4-7	B15	4-8	B16	4-9	B17	4-10	B18	4-11		4-12		4-13
A200		*A201*		*A202*	C20	*A203*	C21	*A204*	C22	*A205*	C23	*A206*	C24	*A207*	C25	*A208*	C26	*A209*	C27	*A210*	C28	*A211*		*A212*		*A213*
5-0		5-1		5-2	B20	5-3	B21	5-4	B22	5-5	B23	5-6	B24	5-7	B25	5-8	B26	5-9	B27	5-10	B28	5-11		5-12		5-13
A300		*A301*		*A302*	C30	*A303*	C31	*A304*	C32	*A305*	C33	*A306*	C34	*A307*	C35	*A308*	C36	*A309*	C37	*A310*	C38	*A311*		*A312*		*A313*
6--0		6-1		6-2	B30	6-3	B31	6-4	B32	6-5	B33	6-6	B34	6-7	B35	6-8	B36	6-9	B37	6-10	B38	6-11		6-12		6-13
A400		*A401*		*A402*	C40	*A403*	C41	*A404*	C42	*A405*	C43	*A406*	C44	*A407*	C45	*A408*	C46	*A409*	C47	*A410*	C48	*A411*		*A412*		*A413*
7-0		7-1		7-2	B40	7-3	B41	7-4	B42	7-5	B43	7-6	B44	7-7	B45	7-8	B46	7-9	B47	7-10	B48	7-11		7-12		7-13
A500		*A501*		*A502*	C50	*A503*	C51	*A504*	C52	*A505*	C53	*A506*	C54	*A507*	C55	*A508*	C56	*A509*	C57	*A510*	C58	*A511*		*A512*		*A513*
8-0		8-1		8-2	B50	8-3	B51	8-4	B52	8-5	B53	8-6	B54	8-7	B55	8-8	B56	8-9	B57	8-10	B58	8-11		8-12		8-13
A600		*A601*		*A602*	C60	*A603*	C61	*A604*	C62	*A605*	C63	*A606*	C64	*A607*	C65	*A608*	C66	*A609*	C67	*A610*	C68	*A611*		*A612*		*A613*
9-0		9-1		9-2	B60	9-3	B61	9-4	B62	9-5	B63	9-6	B64	9-7	B65	9-8	B66	9-9	B67	9-10	B68	9-11		9-12		9-13
A700		*A701*		*A702*	C70	*A703*	C71	*A704*	C72	*A705*	C73	*A706*	C74	*A707*	C75	*A708*	C76	*A709*	C77	*A710*	C78	*A711*		*A712*		*A713*
10--0		10-1		10-2	B70	10-3	B71	10-4	B72	10-5	B73	10-6	B74	10-7	B75	10-8	B76	10-9	B77	10-10	B78	10-11		10-12		10-13
A800		*A801*		*A802*	C80	*A803*	C81	*A804*	C82	*A805*	C83	*A806*	C84	*A807*	C85	*A808*	C86	*A809*	C87	*A810*	C88	*A811*		*A812*		*A813*
11-0		11-1		11-2		11-3		11-4		11-5		11-6		11-7		11-8		11-9		11-10		11-11		11-12		11-13
12-0		12-1		12-2		12-3		12-4		12-5		12-6		12-7		12-8		12-9		12-10		12-11		12-12		12-13
13-0		13-1		13-2		13-3		13-4		13-5		13-6		13-7		13-8		13-9		13-10		13-11		13-12		13-13

Figure 4.2 8x8 Half-Pixel Interpolation Flow

The half-pixel interpolation flow for a 8x8 block is shown in Figure 4.2. The half-pixel interpolation flows for the other block sizes are similar to this flow. The light gray rectangles denote integer pixels (e.g. 0-0) and dark gray rectangles denote half pixels (e.g. A000, B00, C00). The integer-pixel MV for this 8x8 block points to the integer pixel 3-3. The half-pixel search locations around this integer pixel (C00, A003, C01, B01, C11, A103, C10, B00) have to be searched to determine the best half-pixel accurate MV. Therefore, the top-left corner of the half-pixel search window is C00 and the bottom-right corner is C88. There are 17x17 (integer and half pixels) – 8x8 (integer pixels) = 225 half pixels in the half-pixel search window and 196 integer pixels (0-0 to 13-13) are required to calculate these half pixels. The integer pixels are stored in the input register file and the half pixels in the half-pixel search window are stored in the search register file.

The half pixels are grouped according to their calculation order; first set A half pixels, then set B half pixels and finally set C half pixels are calculated. Set A half pixels are interpolated from 6 corresponding vertical integer pixels; e.g. A00 is interpolated from integer pixels 0-0, 1-0, 2-0, 3-0, 4-0, 5-0. Set B half pixels are interpolated from 6 corresponding horizontal integer pixels; e.g. B00 is interpolated from integer pixels 3-0, 3-1, 3-2, 3-3, 3-4, 3-5. Finally, set C half pixels are interpolated from 6 corresponding horizontal set A half pixels; e.g. C00 is interpolated from set A half pixels A00, A01, A02, A03, A04, A05. Therefore, in addition to the set A half pixels that are in the search window, the set A half pixels that are required to calculate set C half pixels are also calculated and stored in temporary register file.

The half-pixel interpolation datapath for implementing the 6-tap FIR filter round ((A-5B+20C+20D-5E+F) / 32) is presented in [11]. It takes 6 input pixels and calculates the corresponding half pixel.

Since one set A half pixel is interpolated from 6 integer pixels, if we use 1 half-pixel interpolation datapath, 9 set A half pixels will be interpolated in 9 clock cycles by accessing 54 integer pixels. However, since 1 column of set A half pixels (9 pixels) can be calculated using 1 column of integer pixels (14 pixels), if we use 9 half-pixel interpolation datapaths, 9 set A half pixels can be interpolated in 1 clock cycle by accessing 14 integer pixels. This reduces the number of input register file accesses by 4 and the number of clock cycles by 9. We used 18 half-pixel interpolation datapaths to further reduce the clock cycle count. Therefore, 2 columns of set A half pixels (18 pixels) are calculated in 1 clock cycle by accessing 28 integer pixels. Similarly, 2 rows of set B half pixels (18 pixels) are calculated in 1 clock cycle by accessing 28 integer pixels.

However, since set C half pixels are interpolated from set A half pixels and accessing 2 rows of set A half pixels in 1 clock cycle increases the complexity of the register files, 1 row of set C half pixels (9 pixels) are calculated in 1 clock cycle by accessing 1 row of set A half pixels (14 pixels). Half-pixel interpolation hardware, therefore, calculates set A half pixels in 9 clock cycles, set B half pixels in 4 clock cycles and set C half pixels in 7 clock cycles. Half-pixel interpolation for a 8x8 block, therefore, takes 20 clock cycles.

The half-pixel accurate variable block size ME hardware has dedicated half-pixel search hardware for each block size in order to perform the half-pixel search faster. Each half-pixel search hardware has 4 PEs. Since there are 7 block sizes, 28 PEs are used in the half-pixel ME hardware. The SAD value for a search location is calculated by a processing element in 64 clock cycles. Since there are 8 half-pixel search locations, half-pixel search would take 8*64=512 clock cycles using one PE. We used 4 PEs in order to perform the half-pixel search operation faster. Each PE calculates the SAD for two half-pixel search locations in 2*64=128 clock cycles. The SADs calculated by PEs are sent to a comparator, and the comparator determines the minimum SAD and the corresponding best half-pixel accurate MV.

The half-pixel search locations are allocated to PEs as follows. First, 4 search locations that use set C half pixels are searched. PE0 calculates the SAD for the 8x8 block starting with C00 and ending with C77, PE1 calculates the SAD for the 8x8 block starting with C01 and ending with C78, PE2 calculates the SAD for the 8x8 block starting with C10 and ending with C87 and PE3 calculates the SAD for the 8x8 block starting with C11 and ending with C88. Then, 2 search locations that use set A half pixels and 2 search locations that use set B half pixels are searched. PE0 and PE1 searches the two set B search locations and PE2 and PE3 searches the two set A search locations. PE0 calculates the SAD for the 8x8 block starting with B00 and ending with B77, PE1 calculates the SAD for the 8x8 block starting with B01 and ending with B78, PE2 calculates the SAD for the 8x8 block starting with A003 and ending with A710, PE3 calculates the SAD for the 8x8 block starting with A113 and ending with A810.

4.2 Proposed Sub-Pixel Accurate Motion Estimation Hardware

The proposed sub-pixel accurate motion estimation hardware for 4x4 block size is shown in Figure 4.3. The proposed sub-pixel accurate motion estimation hardwares for other block sizes are similar to this hardware. For each block size, 4 PEs are used in the half-pixel search and quarter-pixel search hardware. Since there are 7 block sizes, 28 PEs are used in the sub-pixel accurate variable block size ME hardware. These 28 PEs are shared by quarter-pixel and half-pixel search hardwares for reducing area and power consumption. The inputs to PEs are selected by a multiplexer.

As the half-pixel ME hardware is performing half-pixel interpolation for a block, the integer and half pixels necessary for half-pixel accurate ME and quarter-pixel accurate ME are stored in half-pel search window register file and quarter-pel search window register file respectively.

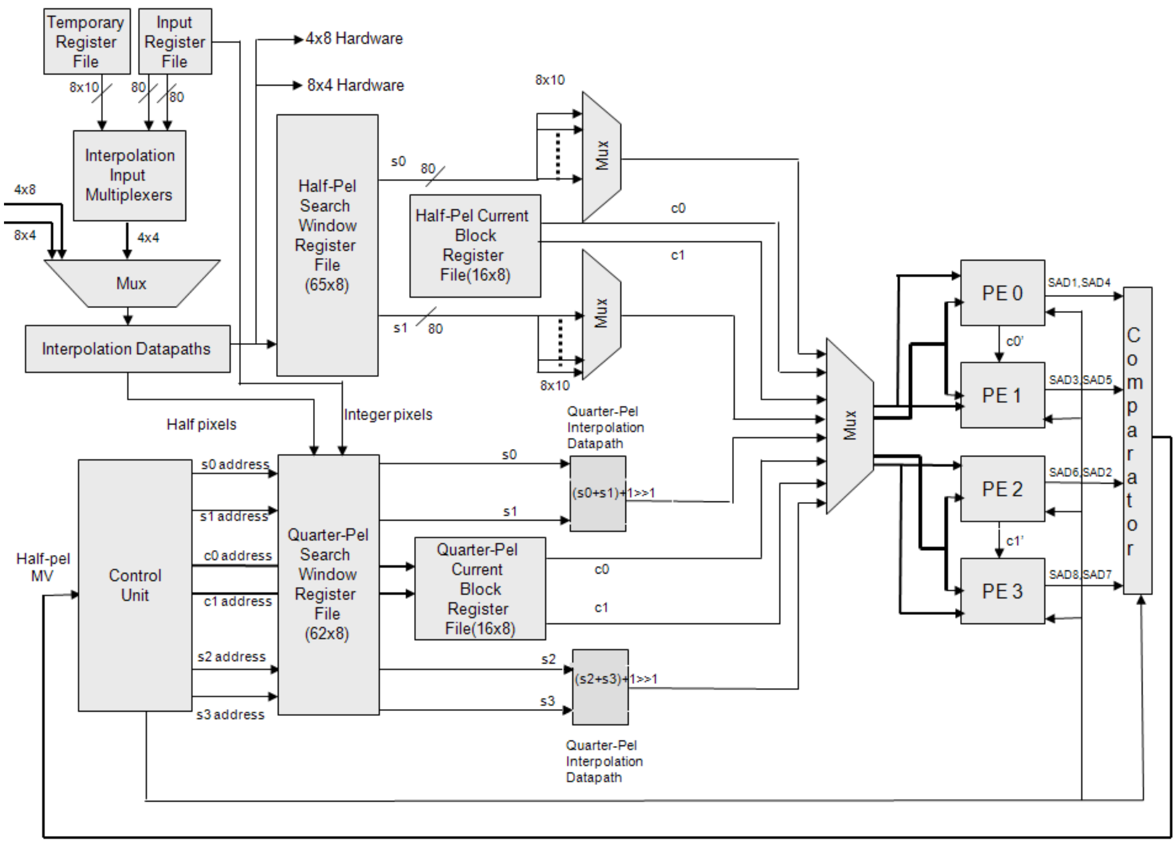


Figure 4.3 Proposed Sub-Pixel Accurate Motion Estimation Hardware for 4x4 Block Size

The SADs calculated by PEs are sent to the comparator. The comparator determines the minimum SAD and the corresponding best half-pixel accurate MV for that block and sends this MV to the control unit. The control unit sends read addresses to the quarter-pel search window register file based on this MV. The control unit generates read addresses dynamically, i.e. only the quarter pixels necessary for performing quarter-pixel search at the location pointed by the best half-pixel MV are calculated. This reduces the amount of computation performed for quarter-pixel interpolation, and therefore reduces the power consumption of the sub-pixel accurate ME hardware.

The quarter-pixel interpolation datapaths generates the quarter pixels and send them to shared PEs. The SAD values for quarter-pixel search locations are calculated by the shared PEs and send to the comparator. The comparator determines the minimum SAD and the corresponding best quarter-pixel accurate MV for that block. This process is performed for all blocks.

4.3 Implementation Results

The number of clock cycles required for sub-pixel (half-pixel and quarter-pixel) ME for each block size are given in Figure 4.4. For example, for a 4x4 block, half-pixel ME takes 48 clock cycles (half-pixel interpolation takes 14 clock cycles and half-pixel search takes 34 clock cycles), quarter-pixel ME takes 44 clock cycles, and therefore, sub-pixel ME takes 48+44 = 92 clock cycles. Since there are 16 4x4 blocks in a MB, sub-pixel ME for a MB for 4x4 block size takes 16*92=1472 clock cycles. As it can be seen from Figure 4.4, 4x4 block size is the bottleneck for sub-pixel ME hardware, and therefore, sub-pixel ME for a MB for all block sizes take 1472 clock cycles.

The proposed sub-pixel (half-pixel and quarter-pixel) ME hardware is implemented in Verilog HDL. The Verilog HDL implementation of the sub-pixel ME hardware is verified with RTL simulations using Mentor Graphics ModelSim. A software model for sub-pixel accurate H.264 variable block size ME algorithm is implemented in C. The software model is used for verifying the RTL design by comparing their outputs for a randomly generated current frame and a reference frame. Both the software model and the RTL design are simulated for all block sizes. The outputs of the software and hardware simulations exactly matched, verifying the RTL design.

The Verilog RTL is then synthesized to a 2V8000ff1152 Xilinx Virtex II FPGA with speed grade 6 using Mentor Graphics Leonardo Spectrum. The resulting netlist is placed and routed to the same FPGA using Xilinx ISE Series 7.1. The FPGA implementation is verified to work at 60 MHz under worst-case PVT conditions with post place and route simulations. The FPGA implementation can process a VGA frame in 29.32 msec (1200 MB * 1472 cycles per MB * 16.6 ns clock cycle = 29.32 msec). Therefore, it can process 1000/29.32 = 34 VGA frames (640x480) per second.

The FPGA resources used by the sub-pixel (half-pixel and quarter-pixel) ME implementation are shown in Figure 4.5.

Block Size	Number of Clock Cycles		
	1 Block		1 MB
	Half-pel ME	Quarter-pel ME	Sub-pel ME
4x4	14+34=48	44	16*(48+44)=1472
4x8	16+66=82	76	8*(82+76)=1264
8x4	23+66=89	80	8*(89+80)=1352
8x8	20+130=150	144	4*(150+144)=1176
8x16	28+258=286	280	2*(286+280)=1132
16x8	39+258=297	288	2*(297+288)=1170
16x16	55+514=569	544	1*(569+544)=1113

Figure 4.4 Performance of Sub-pixel ME Hardware

Area	Half-pel ME	Quarter-pel ME	Sub-pel ME
CLB Slices	11020(%24)	18566(%39)	29586(%63,5)
Function Generators	22040(%24)	37131(%39)	59171(%63,5)
DFFs	14243(%15,5)	21339(%22)	35582(%37,51)

Figure 4.5 Area of Sub-pixel ME Hardware

Several hardware architectures for real-time implementation of sub-pixel accurate variable block size ME for H.264 video coding are presented in the literature [13, 14]. The hardware architecture presented in [13] uses less hardware than our hardware design and has lower performance than our hardware design. The hardware architecture presented in [14] achieves higher performance than our hardware design at the expense of a much higher hardware cost. It uses much more FIR filters (64 vs. 28) and processing elements (144 vs. 56) than our hardware design in order to process 30 HDTV frames (1280x720) per second. Our hardware design is a more cost-effective solution for portable applications.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

In this thesis, we presented a half-pixel accurate ME hardware for 4x4 block size in Chapter 2 and a quarter-pixel accurate variable block size ME hardware in Chapter 3. We integrated this quarter pixel accurate ME hardware with the half pixel accurate variable block size ME hardware presented in [11] to implement a sub-pixel accurate variable block size ME hardware for H.264 video coding. This hardware is designed to be used as part of a complete H.264 video coding system for portable applications. The proposed hardware architecture is implemented in Verilog HDL. The Verilog RTL code is verified to work at 60 MHz in a Xilinx Virtex II FPGA. The FPGA implementation can process 34 VGA frames (640x480) per second.

As future work, the hardware design can be implemented as an ASIC in order to increase the operating frequency and therefore increase the number of frames processed per second. Improved sub-pixel motion estimation algorithms, e.g. algorithms with different motion vector selection criterion or with different search ranges, can also be implemented by using the proposed datapath with a new control unit. The power consumption of the hardware can be analyzed. Based on this analysis, low-power techniques such as clock gating and glitch reduction can be used to reduce its power consumption. A variable block size H.264 motion estimation hardware can be implemented by integrating this sub-pixel accurate H.264 variable block size motion estimation hardware with an integer-pixel H.264 variable block size motion estimation hardware [12].

REFERENCES

[1] Joint Video Team (JVT) of ITU-T VCEG and ISO/IEC MPEG, Draft ITU-T. Recommendation and Final Draft International Standard of Joint Video Specification, ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, May 2003

[2] R. Schäfer, T. Wiegand and H. Schwarz, "The Emerging H.264/AVC Standard", *EBU Technical Review*, January 2003

[3] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra "Overview of the H.264/AVC Video Coding Standard", *IEEE Trans. on Circuits and Systems for Video Technology vol.* 13, no. 7, pp. 560–576, July 2003

[4] I. Richardson, H.264 and MPEG-4 Video Compression, Wiley, 2003

[5] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards: Algorithms and Architectures*, Kluwer Academic Publishers, 2nd Edition, 1997

[6] S. Yalcin, "H.264 Motion Estimator Design", MS Thesis, Sabanci University, August 2005

[7] H. Mahdavi-Nasab, S. Kasaei, "Half-Pixel Accuracy Block Matching Motion Estimation Algorithms for Low Bitrate Video Communications", *Proc. IEEE Int Conf. on Internet*, 2005

[8] M. Rehan, P. Agathoklis, "Half-pixel Accurate Motion Estimation Using a Flexible Triangle Search", *Proc. IEEE Conf. on Comm., Comp. and Signal Proc.*, pp. 257–260, 2005

[9] T. Dias, N. Roma, L. Sousa, "Efficient motion vector refinement architecture for sub-pixel motion estimation systems", *Proc. IEEE Workshop on Signal Proc. System Design and Implementation*, pp. 313-318, November 2005

[10] K. Minoo, T. Q. Nquyen, “Reverse, Sub-Pixel Block Matching: Applications within H.264 and Analysis of Limitations”, *IEEE Int. Conf. on Image Processing,* pp. 3161-3164, 2006

[11] S. Yalcin, I. Hamzaoglu, “A High Performance Hardware Architecture for Half-Pixel Accurate H.264 Motion Estimation”, *14th Int. Conf. on VLSI-SoC,* October 2006

[12] H. S. Yalcin, H. Ates, I. Hamzaoglu, “A High Performance Hardware Architecture for an SAD Reuse based Hierarchical Motion Estimation Algorithm for H.264 Video Coding”, *Int. Conf. on Field Programmable Logic and Applications*, pp. 509–514, August 2005

[13] AT. C.Chen, Y.W.Huang, L.G.Chen, “Fully untilized and reusable architecture for fractional motion estimation of H.264/AVC", *IEEE ICASSP*, 2004

[14] C. Yang, S. Goto, T. Ikenaga, “High performace architecture for fractional motion estimation in H.264 for HDTV", *IEEE ISCAS*, 2006

[15] S. Oktem, I. Hamzaoglu, “An Efficient Hardware Architecture for Quarter-Pixel Accurate H.264 Motion Estimation”, *10th Euromicro Conference on Digital System Design*, August 2007

APPENDIX A

QUARTER-PIXEL INTERPOLATION AND SEARCH FLOWS

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw0--sw9)		cb0-(sw16--sw9)	
2	cb1-(sw9--sw2)	cb0-(sw9--sw2)	cb1-(sw9--sw18)	cb0-(sw9--sw18)
3	cb2-(sw2--sw11)	cb1-(sw2--sw11)	cb2-(sw18--sw11)	cb1-(sw18--sw11)
4	cb3-(sw11--sw4)	cb2-(sw11--sw4)	cb3-(sw11--sw20)	cb2-(sw11--sw20)
5	cb4-(sw16--sw9)		cb4-(sw16--sw25)	
6	cb5-(sw9--sw18)	cb4-(sw9--sw18)	cb5-(sw25--sw18)	cb4-(sw25--sw18)
7	cb6-(sw18--sw11)	cb5-(sw18--sw11)	cb6-(sw18--sw27)	cb5--(sw18--sw27)
8	cb7-(sw11--sw20)	cb6-(sw11--sw20)	cb7-(sw27--sw20)	cb6-(sw27--sw20)
9	cb8-(sw16--sw25)		cb8-(sw32--sw25)	
10	cb9-(sw25--sw18)	cb8-(sw25--sw18)	cb9-(sw25--sw34)	cb8--(sw25--sw34)
11	cb10-(sw18--sw27)	cb9-(sw18--sw27)	cb10-(sw34--sw27)	cb9-(sw34--sw27)
12	cb11-(sw27--sw20)	cb10-(sw27--sw20	cb11-(sw27--sw36)	cb10-(sw27--sw36)
13	cb12-(sw32--sw25)		cb12-(sw32--sw41)	
14	cb13-(sw25--sw34)	cb12-(sw25--sw34)	cb13-(sw41--sw34)	cb12-(sw41--sw34)
15	cb14-(sw34--sw27)	cb13-(sw34--sw27)	cb14-(sw34--sw43)	cb13-(sw34--sw43
16	cb15-(sw27--sw36)	cb14-(sw27--sw36)	cb15-(sw43--sw36)	cb14-(sw43--sw36)
17		cb3-(sw4--sw13)		cb15-(sw36--sw45)
18		cb7-(sw13--sw20)		cb3-(sw13--sw20)
19		cb11-(sw20--sw29)		cb7-(sw20--sw29)
20		cb15-(sw29--sw36)		cb11-(sw29--sw36)
21	cb0-(sw8--sw9)		cb0-(sw1--sw9)	
22	cb1-(sw9--sw10)	cb0-(sw9--sw10)	cb4-(sw9--sw17)	cb0-(sw9--sw17)
23	cb2-(sw10--sw11)	cb1-(sw10--sw11)	cb8-(sw17--sw25)	cb4-(sw17--sw25)
24	cb3-(sw11--sw12)	cb2-(sw11--sw12)	cb12-(sw25--sw33)	cb8-(sw25--sw33)
25	cb4-(sw16--sw17)		cb1-(sw2--sw10)	
26	cb5-(sw17--sw18)	cb4-(sw17--sw18)	cb5-(sw10--sw18)	cb1-(sw10--sw18)
27	cb6-(sw18--sw19)	cb5-(sw18--sw19)	cb9-(sw18--sw26)	cb5-(sw18--sw26)
28	cb7-(sw19--sw20)	cb6-(sw19--sw20)	cb13-(sw26--sw34)	cb9-(sw26--sw34)
29	cb8-(sw24--sw25)		cb2-(sw3--sw11)	
30	cb9-(sw25--sw26)	cb8-(sw25--sw26)	cb6-(sw11--sw19)	cb2-(sw11--sw19)
31	cb10-(sw26--sw27)	cb9-(sw26--sw27)	cb10-(sw19--sw27)	cb6-(sw19--sw27)
32	cb11-(sw27--sw28)	cb10-(sw27--sw28)	cb14-(sw27--sw35)	cb10-(sw27--sw35)
33	cb12-(sw32--sw33)		cb3-(sw4--sw12)	
34	cb13-(sw33--sw34)	cb12-(sw33--sw34)	cb7-(sw12--sw20)	cb3-(sw12--sw20)
35	cb14-(sw34--sw35)	cb13-(sw34--sw35)	cb11-(sw20--sw28)	cb7-(sw20--sw28)
36	cb15-(sw35--sw36)	cb14-(sw35--sw36)	cb15-(sw28--sw36)	cb11-(sw28--sw36)
37		cb3-(sw12--sw13		cb12-(sw33--sw41)
38		cb7-(sw20--sw21)		cb13-(sw34--sw42)
39		cb1-(sw28--sw29)		cb14-(sw35--sw43)
40		cb15-(sw36--sw37		cb15-(sw36--sw44)

Figure A.1 Quarter-Pixel Interpolation and Search Flow for 4x4 Block Size

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw11--sw0)		cb0-(sw11--sw24)	
2	cb1-(sw0--sw13)	cb0-(sw0--sw13)	cb1-(sw24--sw13)	cb0-(sw24--sw13)
3	cb2-(sw13--sw2)	cb1-(sw13--sw2)	cb2-(sw13--sw26)	cb1-(sw13--sw26)
4	cb3-(sw2--sw15)	cb2-(sw2--sw15)	cb3-(sw26--sw15)	cb2-(sw26--sw15)
5	cb4-(sw15--sw4)	cb3-(sw15--sw4)	cb4-(sw15--sw28)	cb3-(sw15--sw28)
6	cb5-(sw4--sw17)	cb4-(sw4--sw17)	cb5-(sw28--sw17)	cb4-(sw28--sw17)
7	cb6-(sw17--sw6)	cb5-(sw17--sw6)	cb6-(sw17--sw30)	cb5-(sw17--sw30)
8	cb7-(sw6--sw19)	cb6-(sw6--sw19)	cb7-(sw30--sw19)	cb6-(sw30--sw19)
9	cb8-(sw11--sw24)		cb8-(sw35--sw24)	
10	cb9-(sw24--sw13)	cb8-(sw24--sw13)	cb9-(sw24--sw37)	cb8-(sw24--sw37)
11	cb10-(sw13--sw26)	cb9-(sw13--sw26)	cb10-(sw37--sw26)	cb9-(sw37--sw26)
12	cb11-(sw26--sw15)	cb10-(sw26--sw15)	cb11-(sw26--sw39)	cb10-(sw26--sw39)
13	cb12-(sw15--sw28)	cb11-(sw15--sw28)	cb12-(sw39--sw28)	cb11-(sw39--sw28)
14	cb13-(sw28--sw17)	cb12-(sw28--sw17)	cb13-(sw28--sw41)	cb12-(sw28--sw41)
15	cb14-(sw17--sw30)	cb13-(sw17--sw30)	cb14-(sw41--sw30)	cb13-(sw41--sw30)
16	cb15-(sw30--sw19)	cb14-(sw30--sw19)	cb15-(sw30--sw43)	cb14-(sw30--sw43)
17	cb16-(sw35--sw24)		cb16-(sw35--sw48)	
18	cb17-(sw24--sw37)	cb16-(sw24--sw37)	cb17-(sw48--sw37)	cb16-(sw48--sw37)
19	cb18-(sw37--sw26)	cb17-(sw37--sw26)	cb18-(sw37--sw50)	cb17-(sw37--sw50)
20	cb19-(sw26--sw39)	cb18-(sw26--sw39)	cb19-(sw50--sw39)	cb18-(sw50--sw39)
21	cb20-(sw39--sw28)	cb19-(sw39--sw28)	cb20-(sw39--sw52)	cb19-(sw39--sw52)
22	cb21-(sw28--sw41)	cb20-(sw28--sw41)	cb21-(sw52--sw41)	cb20-(sw52--sw41)
23	cb22-(sw41--sw30)	cb21-(sw41--sw30)	cb22-(sw41--sw54)	cb21-(sw41--sw54)
24	cb23-(sw30--sw43)	cb22-(sw30--sw43)	cb23-(sw54--sw43)	cb22-(sw54--sw43)
25	cb24-(sw35--sw48)		cb24-(sw59--sw48)	
26	cb25-(sw48--sw37)	cb24-(sw48--sw37)	cb25-(sw48--sw61)	cb24-(sw48--sw61)
27	cb26-(sw37--sw50)	cb25-(sw37--sw50)	cb26-(sw61--sw50)	cb25-(sw61--sw50)
28	cb27-(sw50--sw39)	cb26-(sw50--sw39)	cb27-(sw50--sw63)	cb26-(sw50--sw63)
29	cb28-(sw39--sw52)	cb27-(sw39--sw52)	cb28-(sw63--sw52)	cb27-(sw63--sw52)
30	cb29-(sw52--sw41)	cb28-(sw52--sw41)	cb29-(sw52--sw65)	cb28-(sw52--sw65)
31	cb30-(sw41--sw54)	cb29-(sw41--sw54)	cb30-(sw65--sw54)	cb29-(sw65--sw54)
32	cb31-(sw54--sw43)	cb30-(sw54--sw43)	cb31-(sw54--sw67)	cb30-(sw54--sw67)
33		cb31-(sw43--sw56)	-(sw--sw)	cb31-(sw67--sw56)
34		cb7-(sw19--sw8)	-(sw--sw)	cb7-(sw19--sw32)
35		cb15-(sw19--sw32)	-(sw--sw)	cb15-(sw43--sw32)
36		cb23-(sw43--sw32)	-(sw--sw)	cb23-(sw43--sw56)
37	cb0-(sw11--sw12)		cb0-(sw0--sw12)	
38	cb1-(sw12--sw13)	cb0-(sw12--sw13)	cb1-(sw1--sw13)	
39	cb2-(sw13--sw14)	cb1-(sw13--sw14)	cb2-(sw2--sw14)	
40	cb3-(sw14--sw15)	cb2-(sw14--sw15)	cb3-(sw3--sw15)	
41	cb4-(sw15--sw16)	cb3-(sw15--sw16)	cb4-(sw4--sw16)	
42	cb5-(sw16--sw17)	cb4-(sw16--sw17)	cb5-(sw5--sw17)	
43	cb6-(sw17--sw18)	cb5-(sw17--sw18)	cb6-(sw6--sw18)	
44	cb7-(sw18--sw19)	cb6-(sw18--sw19)	cb7-(sw7--sw19)	
45	cb8-(sw23--sw24)		cb8-(sw12--sw24)	cb0-(sw12--sw24)
46	cb9-(sw24--sw25)	cb8-(sw24--sw25)	cb9-(sw13--sw25)	cb1-(sw13--sw25)
47	cb10-(sw25--sw26)	cb9-(sw25--sw26)	cb10-(sw14--sw26)	cb2-(sw14--sw26)

48	cb11-(sw26--sw27)	cb10-(sw26--sw27)	cb11-(sw15--sw27)	cb3-(sw15--sw27)
49	cb12-(sw27--sw28)	cb11-(sw27--sw28)	cb12-(sw16--sw28)	cb4-(sw16--sw28)
50	cb13-(sw28--sw29)	cb12-(sw28--sw29)	cb13-(sw17--sw29)	cb5-(sw17--sw29)
51	cb14-(sw29--sw30)	cb13-(sw29--sw30)	cb14-(sw18--sw30)	cb6-(sw18--sw30)
52	cb15-(sw30--sw31)	cb14-(sw30--sw31)	cb15-(sw19--sw31)	cb7-(sw19--sw31)
53	cb16-(sw35--sw36)		cb16-(sw24--sw36)	cb8-(sw24--sw36)
54	cb17-(sw36--sw37)	cb16-(sw36--sw37)	cb17-(sw25--sw37)	cb9-(sw25--sw37)
55	cb18-(sw37--sw38)	cb17-(sw37--sw38)	cb18-(sw26--sw38)	cb10-(sw26--sw38)
56	cb19-(sw38--sw39)	cb18-(sw38--sw39)	cb19-(sw27--sw39)	cb11-(sw27--sw39)
57	cb20-(sw39--sw40)	cb19-(sw39--sw40)	cb20-(sw28--sw40)	cb12-(sw28--sw40)
58	cb21-(sw40--sw41)	cb20-(sw40--sw41)	cb21-(sw29--sw41)	cb13-(sw29--sw41)
59	cb22-(sw41--sw42)	cb21-(sw41--sw42)	cb22-(sw30--sw42)	cb14-(sw30--sw42)
60	cb23-(sw42--sw43)	cb22-(sw42--sw43)	cb23-(sw31--sw43)	cb15-(sw31--sw43)
61	cb24-(sw47--sw48)		cb24-(sw36--sw48)	cb16-(sw36--sw48)
62	cb25-(sw48--sw49)	cb24-(sw48--sw49)	cb25-(sw37--sw49)	cb17-(sw37--sw49)
63	cb26-(sw49--sw50)	cb25-(sw49--sw50)	cb26-(sw38--sw50)	cb18-(sw38--sw50)
64	cb27-(sw50--sw51)	cb26-(sw50--sw51)	cb27-(sw39--sw51)	cb19-(sw39--sw51)
65	cb28-(sw51--sw52)	cb27-(sw51--sw52)	cb28-(sw40--sw52)	cb20-(sw40--sw52)
66	cb29-(sw52--sw53)	cb28-(sw52--sw53)	cb29-(sw41--sw53)	cb21-(sw41--sw53)
67	cb30-(sw53--sw54)	cb29-(sw53--sw54)	cb30-(sw42--sw54)	cb22-(sw42--sw54)
68	cb31-(sw54--sw55)	cb30-(sw54--sw55)	cb31-(sw43--sw55)	cb23-(sw43--sw55)
69		cb31-(sw55--sw56)		cb24-(sw48--sw60)
70		cb7-(sw19--sw20)		cb25-(sw49--sw61)
71		cb15-(sw31--sw32)		cb26-(sw50--sw62)
72		cb23-(sw43--sw44)		cb27-(sw51--sw63)
73				cb28-(sw52--sw64)
74				cb29-(sw53--sw65)
75				cb30-(sw54--sw66)
76				cb31-(sw55--sw67)

Figure A.2 Quarter-Pixel Interpolation and Search Flow for 4x8 Block Size

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw7--sw0)		cb0-(sw7--sw16)	
2	cb1-(sw0--sw9)	cb0-(sw0--sw9)	cb1-(sw16--sw9)	cb0-(sw16--sw9)
3	cb2-(sw9--sw2)	cb1-(sw9--sw2)	cb2-(sw9--sw18)	cb1-(sw9--sw18)
4	cb3-(sw2--sw11)	cb2-(sw2--sw11)	cb3-(sw18--sw11)	cb2-(sw18--sw11)
5	cb4-(sw7--sw16)		cb4-(sw23--sw16)	
6	cb5-(sw16--sw9)	cb4-(sw16--sw9)	cb5-(sw16--sw25)	cb4-(sw16--sw25)
7	cb6-(sw9--sw18)	cb5-(sw9--sw18)	cb6-(sw25--sw18)	cb5-(sw25--sw18)
8	cb7-(sw18--sw11)	cb6-(sw18--sw11)	cb7-(sw18--sw27)	cb6-(sw18--sw27)
9	cb8-(sw23--sw16)		cb8-(sw23--sw32)	
10	cb9-(sw16--sw25)	cb8-(sw16--sw25)	cb9-(sw32--sw25)	cb8-(sw32--sw25)
11	cb10-(sw25--sw18)	cb9-(sw25--sw18)	cb10-(sw25--sw34)	cb9-(sw25--sw34)
12	cb11-(sw18--sw27)	cb10-(sw18--sw27)	cb11-(sw34--sw27)	cb10-(sw34--sw27)
13	cb12-(sw23--sw32)		cb12-(sw39--sw32)	
14	cb13-(sw32--sw25)	cb12-(sw32--sw25)	cb13-(sw32--sw41)	cb12-(sw32--sw41)
15	cb14-(sw25--sw34)	cb13-(sw25--sw34)	cb14-(sw41--sw34)	cb13-(sw41--sw34)
16	cb15-(sw34--sw27)	cb14-(sw34--sw27)	cb15-(sw34--sw43)	cb14-(sw34--sw43)
17	cb16-(sw39--sw32)		cb16-(sw39--sw48)	
18	cb17-(sw32--sw41)	cb16-(sw32--sw41)	cb17-(sw48--sw41)	cb16-(sw48--sw41)
19	cb18-(sw41--sw34)	cb17-(sw41--sw34)	cb18-(sw41--sw50)	cb17-(sw41--sw50)
20	cb19-(sw34--sw43)	cb18-(sw34--sw43)	cb19-(sw50--sw43)	cb18-(sw50--sw43)
21	cb20-(sw39--sw48)		cb20-(sw55--sw48)	
22	cb21-(sw48--sw41)	cb20-(sw48--sw41)	cb21-(sw48--sw57)	cb20-(sw48--sw57)
23	cb22-(sw41--sw50)	cb21-(sw41--sw50)	cb22-(sw57--sw50)	cb21-(sw57--sw50)
24	cb23-(sw50--sw43)	cb22-(sw50--sw43)	cb23-(sw50--sw59)	cb22-(sw50--sw59)
25	cb24-(sw55--sw48)		cb24-(sw55--sw64)	
26	cb25-(sw48--sw57)	cb24-(sw48--sw57)	cb25-(sw64--sw57)	cb24-(sw64--sw57)
27	cb26-(sw57--sw50)	cb25-(sw57--sw50)	cb26-(sw57--sw66)	cb25-(sw57--sw66)
28	cb27-(sw50--sw59)	cb26-(sw50--sw59)	cb27-(sw66--sw59)	cb26-(sw66--sw59)
29	cb28-(sw55--sw64)		cb28-(sw71--sw64)	
30	cb29-(sw64--sw57)	cb28-(sw64--sw57)	cb29-(sw64--sw73)	cb28-(sw64--sw73)
31	cb30-(sw57--sw66)	cb29-(sw57--sw66)	cb30-(sw73--sw66)	cb29-(sw73--sw66)
32	cb31-(sw66--sw59)	cb30-(sw66--sw59)	cb31-(sw66--sw75)	cb30-(sw66--sw75)
33		cb31-(sw59--sw68)		cb31-(sw75--sw68)
34		cb3-(sw11--sw4)		cb3-(sw11--sw20)
35		cb7-(sw11--sw20)		cb7-(sw27--sw20)
36		cb11-(sw27--sw20)		cb11-(sw27--sw36)
37		cb15-(sw27--sw36)		cb15-(sw43--sw36)
38		cb19-(sw43--sw36)		cb19-(sw43--sw52)
39		cb23-(sw43--sw52)		cb23-(sw59--sw52)
40		cb27-(sw59--sw52)		cb27-(sw59--sw68)
41	cb0-(sw7--sw8)		cb0-(sw0--sw8)	
42	cb1-(sw8--sw9)	cb0-(sw8--sw9)	cb1-(sw1--sw9)	
43	cb2-(sw9--sw10)	cb1-(sw9--sw10)	cb2-(sw2--sw10)	
44	cb3-(sw10--sw11)	cb2-(sw10--sw11)	cb3-(sw3--sw11)	
45	cb4-(sw15--sw16)		cb4-(sw8--sw16)	cb0-(sw8--sw16)
46	cb5-(sw16--sw17)	cb4-(sw16--sw17)	cb5-(sw9--sw17)	cb1-(sw9--sw17)
47	cb6-(sw17--sw18)	cb5-(sw17--sw18)	cb6-(sw10--sw18)	cb2-(sw10--sw18)

48	cb7-(sw18--sw19)	cb6-(sw18--sw19)	cb7-(sw11--sw19)	cb3-(sw11--sw19)
49	cb8-(sw23--sw24)		cb8-(sw16--sw24)	cb4-(sw16--sw24)
50	cb9-(sw24--sw25)	cb8-(sw24--sw25)	cb9-(sw17--sw25)	cb5-(sw17--sw25)
51	cb10-(sw25--sw26)	cb9-(sw25--sw26)	cb10-(sw18--sw26)	cb6-(sw18--sw26)
52	cb11-(sw26--sw27)	cb10-(sw26--sw27)	cb11-(sw19--sw27)	cb7-(sw19--sw27)
53	cb12-(sw31--sw32)		cb12-(sw24--sw32)	cb8-(sw24--sw32)
54	cb13-(sw32--sw33)	cb12-(sw32--sw33)	cb13-(sw25--sw33)	cb9-(sw25--sw33)
55	cb14-(sw33--sw34)	cb13-(sw33--sw34)	cb14-(sw26--sw34)	cb10-(sw26--sw34)
56	cb15-(sw34--sw35)	cb14-(sw34--sw35)	cb15-(sw27--sw35)	cb11-(sw27--sw35)
57	cb16-(sw39--sw40)		cb16-(sw32--sw40)	cb12-(sw32--sw40)
58	cb17-(sw40--sw41)	cb16-(sw40--sw41)	cb17-(sw33--sw41)	cb13-(sw33--sw41)
59	cb18-(sw41--sw42)	cb17-(sw41--sw42)	cb18-(sw34--sw42)	cb14-(sw34--sw42)
60	cb19-(sw42--sw43)	cb18-(sw42--sw43)	cb19-(sw35--sw43)	cb15-(sw35--sw43)
61	cb20-(sw47--sw48)		cb20-(sw40--sw48)	cb16-(sw40--sw48)
62	cb21-(sw48--sw49)	cb20-(sw48--sw49)	cb21-(sw41--sw49)	cb17-(sw41--sw49)
63	cb22-(sw49--sw50)	cb21-(sw49--sw50)	cb22-(sw42--sw50)	cb18-(sw42--sw50)
64	cb23-(sw50--sw51)	cb22-(sw50--sw51)	cb23-(sw43--sw51)	cb19-(sw43--sw51)
65	cb24-(sw55--sw56)		cb24-(sw48--sw56)	cb20-(sw48--sw56)
66	cb25-(sw56--sw57)	cb24-(sw56--sw57)	cb25-(sw49--sw57)	cb21-(sw49--sw57)
67	cb26-(sw57--sw58)	cb25-(sw57--sw58)	cb26-(sw50--sw58)	cb22-(sw50--sw58)
68	cb27-(sw58--sw59)	cb26-(sw58--sw59)	cb27-(sw51--sw59)	cb23-(sw51--sw59)
69	cb28-(sw63--sw64)		cb28-(sw56--sw64)	cb24-(sw56--sw64)
70	cb29-(sw64--sw65)	cb28-(sw64--sw65)	cb29-(sw57--sw65)	cb25-(sw57--sw65)
71	cb30-(sw65--sw66)	cb29-(sw65--sw66)	cb30-(sw58--sw66)	cb26-(sw58--sw66)
72	cb31-(sw66--sw67)	cb30-(sw66--sw67)	cb31-(sw59--sw67)	cb27-(sw59--sw67)
73		cb31-(sw67--sw68)		cb28-(sw64--sw72)
74		cb3-(sw11--sw12)		cb29-(sw65--sw73)
75		cb7-(sw19--sw20)		cb30-(sw66--sw74)
76		cb11-(sw27--sw28)		cb31-(sw67--sw75)
77		cb15-(sw35--sw36)		
78		cb19-(sw43--sw44)		
79		cb23-(sw51--sw52)		
80		cb27-(sw59--sw60)		

Figure A.3 Quarter-Pixel Interpolation and Search Flow for 8x4 Block Size

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw11--sw0)		cb0-(sw11--sw24)	
2	cb1-(sw0--sw13)	cb0-(sw0--sw13)	cb1-(sw24--sw13)	cb0-(sw24--sw13)
3	cb2-(sw13--sw2)	cb1-(sw13--sw2)	cb2-(sw13--sw26)	cb1-(sw13--sw26)
4	cb3-(sw2--sw15)	cb2-(sw2--sw15)	cb3-(sw26--sw15)	cb2-(sw26--sw15)
5	cb4-(sw15--sw4)	cb3-(sw15--sw4)	cb4-(sw15--sw28)	cb3-(sw15--sw28)
6	cb5-(sw4--sw17)	cb4-(sw4--sw17)	cb5-(sw28--sw17)	cb4-(sw28--sw17)
7	cb6-(sw17--sw6)	cb5-(sw17--sw6)	cb6-(sw17--sw30)	cb5-(sw17--sw30)
8	cb7-(sw6--sw19)	cb6-(sw6--sw19)	cb7-(sw30--sw19)	cb6-(sw30--sw19)
9	cb8-(sw11--sw24)		cb8-(sw35--sw24)	
10	cb9-(sw24--sw13)	cb8-(sw24--sw13)	cb9-(sw24--sw37)	cb8-(sw24--sw37)
11	cb10-(sw13--sw26)	cb9-(sw13--sw26)	cb10-(sw37--sw26)	cb9-(sw37--sw26)
12	cb11-(sw26--sw15)	cb10-(sw26--sw15)	cb11-(sw26--sw39)	cb10-(sw26--sw39)
13	cb12-(sw15--sw28)	cb11-(sw15--sw28)	cb12-(sw39--sw28)	cb11-(sw39--sw28)
14	cb13-(sw28--sw17)	cb12-(sw28--sw17)	cb13-(sw28--sw41)	cb12-(sw28--sw41)
15	cb14-(sw17--sw30)	cb13-(sw17--sw30)	cb14-(sw41--sw30)	cb13-(sw41--sw30)
16	cb15-(sw30--sw19)	cb14-(sw30--sw19)	cb15-(sw30--sw43)	cb14-(sw30--sw43)
17	cb16-(sw35--sw24)		cb16-(sw35--sw48)	
18	cb17-(sw24--sw37)	cb16-(sw24--sw37)	cb17-(sw48--sw37)	cb16-(sw48--sw37)
19	cb18-(sw37--sw26)	cb17-(sw37--sw26)	cb18-(sw37--sw50)	cb17-(sw37--sw50)
20	cb19-(sw26--sw39)	cb18-(sw26--sw39)	cb19-(sw50--sw39)	cb18-(sw50--sw39)
21	cb20-(sw39--sw28)	cb19-(sw39--sw28)	cb20-(sw39--sw52)	cb19-(sw39--sw52)
22	cb21-(sw28--sw41)	cb20-(sw28--sw41)	cb21-(sw52--sw41)	cb20-(sw52--sw41)
23	cb22-(sw41--sw30)	cb21-(sw41--sw30)	cb22-(sw41--sw54)	cb21-(sw41--sw54)
24	cb23-(sw30--sw43)	cb22-(sw30--sw43)	cb23-(sw54--sw43)	cb22-(sw54--sw43)
25	cb24-(sw35--sw48)		cb24-(sw59--sw48)	
26	cb25-(sw48--sw37)	cb24-(sw48--sw37)	cb25-(sw48--sw61)	cb24-(sw48--sw61)
27	cb26-(sw37--sw50)	cb25-(sw37--sw50)	cb26-(sw61--sw50)	cb25-(sw61--sw50)
28	cb27-(sw50--sw39)	cb26-(sw50--sw39)	cb27-(sw50--sw63)	cb26-(sw50--sw63)
29	cb28-(sw39--sw52)	cb27-(sw39--sw52)	cb28-(sw63--sw52)	cb27-(sw63--sw52)
30	cb29-(sw52--sw41)	cb28-(sw52--sw41)	cb29-(sw52--sw65)	cb28-(sw52--sw65)
31	cb30-(sw41--sw54)	cb29-(sw41--sw54)	cb30-(sw65--sw54)	cb29-(sw65--sw54)
32	cb31-(sw54--sw43)	cb30-(sw54--sw43)	cb31-(sw54--sw67)	cb30-(sw54--sw67)
33	cb32-(sw59--sw48)		cb32-(sw59--sw72)	
34	cb33-(sw48--sw61)	cb32-(sw48--sw61)	cb33-(sw72--sw61)	cb32-(sw72--sw61)
35	cb34-(sw61--sw50)	cb33-(sw61--sw50)	cb34-(sw61--sw74)	cb33-(sw61--sw74)
36	cb35-(sw50--sw63)	cb34-(sw50--sw63)	cb35-(sw74--sw63)	cb34-(sw74--sw63)
37	cb36-(sw63--sw52)	cb35-(sw63--sw52)	cb36-(sw63--sw76)	cb35-(sw63--sw76)
38	cb37-(sw52--sw65)	cb36-(sw52--sw65)	cb37-(sw76--sw65)	cb36-(sw76--sw65)
39	cb38-(sw65--sw54)	cb37-(sw65--sw54)	cb38-(sw65--sw78)	cb37-(sw65--sw78)
40	cb39-(sw54--sw67)	cb38-(sw54--sw67)	cb39-(sw78--sw67)	cb38-(sw78--sw67)
41	cb40-(sw59--sw72)		cb40-(sw83--sw72)	
42	cb41-(sw72--sw61)	cb40-(sw72--sw61)	cb41-(sw72--sw85)	cb40-(sw72--sw85)
43	cb42-(sw61--sw74)	cb41-(sw61--sw74)	cb42-(sw85--sw74)	cb41-(sw85--sw74)
44	cb43-(sw74--sw63)	cb42-(sw74--sw63)	cb43-(sw74--sw87)	cb42-(sw74--sw87)
45	cb44-(sw63--sw76)	cb43-(sw63--sw76)	cb44-(sw87--sw76)	cb43-(sw87--sw76)
46	cb45-(sw76--sw65)	cb44-(sw76--sw65)	cb45-(sw76--sw89)	cb44-(sw76--sw89)
47	cb46-(sw65--sw78)	cb45-(sw65--sw78)	cb46-(sw89--sw78)	cb45-(sw89--sw78)
48	cb47-(sw78--sw67)	cb46-(sw78--sw67)	cb47-(sw78--sw91)	cb46-(sw78--sw91)

49	cb48-(sw83--sw72)		cb48-(sw83--sw96)	
50	cb49-(sw72--sw85)	cb48-(sw72--sw85)	cb49-(sw96--sw85)	cb48-(sw96--sw85)
51	cb50-(sw85--sw74)	cb49-(sw85--sw74)	cb50-(sw85--sw98)	cb49-(sw85--sw98)
52	cb51-(sw74--sw87)	cb50-(sw74--sw87)	cb51-(sw98--sw87)	cb50-(sw98--sw87)
53	cb52-(sw87--sw76)	cb51-(sw87--sw76)	cb52-(sw87--sw100)	cb51-(sw87--sw100)
54	cb53-(sw76--sw89)	cb52-(sw76--sw89)	cb53-(sw100--sw89)	cb52-(sw100--sw89)
55	cb54-(sw89--sw78)	cb53-(sw89--sw78)	cb54-(sw89--sw102)	cb53-(sw89--sw102)
56	cb55-(sw78--sw91)	cb54-(sw78--sw91)	cb55-(sw102--sw91)	cb54-(sw102--sw91)
57	cb56-(sw83--sw96)		cb56-(sw107--sw96)	
58	cb57-(sw96--sw85)	cb56-(sw96--sw85)	cb57-(sw96--sw109)	cb56-(sw96--sw109)
59	cb58-(sw85--sw98)	cb57-(sw85--sw98)	cb58-(sw109--sw98)	cb57-(sw109--sw98)
60	cb59-(sw98--sw87)	cb58-(sw98--sw87)	cb59-(sw98--sw111)	cb58-(sw98--sw111)
61	cb60-(sw87--sw100)	cb59-(sw87--sw100)	cb60-(sw111--sw100)	cb59-(sw111--sw100)
62	cb61-(sw100--sw89)	cb60-(sw100--sw89)	cb61-(sw100--sw113)	cb60-(sw100--sw113)
63	cb62-(sw89--sw102)	cb61-(sw89--sw102)	cb62-(sw113--sw102)	cb61-(sw113--sw102)
64	cb63-(sw102--sw91)	cb62-(sw102--sw91)	cb63-(sw102--sw115)	cb62-(sw102--sw115)
65		cb63-(sw91--sw104)		cb63-(sw115--sw104)
66		cb7-(sw19--sw8)		cb7-(sw19--sw32)
67		cb15-(sw19--sw32)		cb15-(sw43--sw32)
68		cb23-(sw43--sw32)		cb23-(sw43--sw56)
69		cb31-(sw43--sw56)		cb31-(sw67--sw56)
70		cb39-(sw67--sw56)		cb39-(sw67--sw80)
71		cb47-(sw67--sw80)		cb47-(sw91--sw80)
72		cb55-(sw91--sw80)		cb55-(sw91--sw104)
73	cb0-(sw11--sw12)		cb0-(sw0--sw12)	
74	cb1-(sw12--sw13)	cb0-(sw12--sw13)	cb1-(sw1--sw13)	
75	cb2-(sw13--sw14)	cb1-(sw13--sw14)	cb2-(sw2--sw14)	
76	cb3-(sw14--sw15)	cb2-(sw14--sw15)	cb3-(sw3--sw15)	
77	cb4-(sw15--sw16)	cb3-(sw15--sw16)	cb4-(sw4--sw16)	
78	cb5-(sw16--sw17)	cb4-(sw16--sw17)	cb5-(sw5--sw17)	
79	cb6-(sw17--sw18)	cb5-(sw17--sw18)	cb6-(sw6--sw18)	
80	cb7-(sw18--sw19)	cb6-(sw18--sw19)	cb7-(sw7--sw19)	
81	cb8-(sw23--sw24)		cb8-(sw12--sw24)	cb0-(sw12--sw24)
82	cb9-(sw24--sw25)	cb8-(sw24--sw25)	cb9-(sw13--sw25)	cb1-(sw13--sw25)
83	cb10-(sw25--sw26)	cb9-(sw25--sw26)	cb10-(sw14--sw26)	cb2-(sw14--sw26)
84	cb11-(sw26--sw27)	cb10-(sw26--sw27)	cb11-(sw15--sw27)	cb3-(sw15--sw27)
85	cb12-(sw27--sw28)	cb11-(sw27--sw28)	cb12-(sw16--sw28)	cb4-(sw16--sw28)
86	cb13-(sw28--sw29)	cb12-(sw28--sw29)	cb13-(sw17--sw29)	cb5-(sw17--sw29)
87	cb14-(sw29--sw30)	cb13-(sw29--sw30)	cb14-(sw18--sw30)	cb6-(sw18--sw30)
88	cb15-(sw30--sw31)	cb14-(sw30--sw31)	cb15-(sw19--sw31)	cb7-(sw19--sw31)
89	cb16-(sw35--sw36)		cb16-(sw24--sw36)	cb8-(sw24--sw36)
90	cb17-(sw36--sw37)	cb16-(sw36--sw37)	cb17-(sw25--sw37)	cb9-(sw25--sw37)
91	cb18-(sw37--sw38)	cb17-(sw37--sw38)	cb18-(sw26--sw38)	cb10-(sw26--sw38)
92	cb19-(sw38--sw39)	cb18-(sw38--sw39)	cb19-(sw27--sw39)	cb11-(sw27--sw39)
93	cb20-(sw39--sw40)	cb19-(sw39--sw40)	cb20-(sw28--sw40)	cb12-(sw28--sw40)
94	cb21-(sw40--sw41)	cb20-(sw40--sw41)	cb21-(sw29--sw41)	cb13-(sw29--sw41)
95	cb22-(sw41--sw42)	cb21-(sw41--sw42)	cb22-(sw30--sw42)	cb14-(sw30--sw42)
96	cb23-(sw42--sw43)	cb22-(sw42--sw43)	cb23-(sw31--sw43)	cb15-(sw31--sw43)
97	cb24-(sw47--sw48)		cb24-(sw36--sw48)	cb16-(sw36--sw48)

98	cb25-(sw48--sw49)	cb24-(sw48--sw49)	cb25-(sw37--sw49)	cb17-(sw37--sw49)
99	cb26-(sw49--sw50)	cb25-(sw49--sw50)	cb26-(sw38--sw50)	cb18-(sw38--sw50)
100	cb27-(sw50--sw51)	cb26-(sw50--sw51)	cb27-(sw39--sw51)	cb19-(sw39--sw51)
101	cb28-(sw51--sw52)	cb27-(sw51--sw52)	cb28-(sw40--sw52)	cb20-(sw40--sw52)
102	cb29-(sw52--sw53)	cb28-(sw52--sw53)	cb29-(sw41--sw53)	cb21-(sw41--sw53)
103	cb30-(sw53--sw54)	cb29-(sw53--sw54)	cb30-(sw42--sw54)	cb22-(sw42--sw54)
104	cb31-(sw54--sw55)	cb30-(sw54--sw55)	cb31-(sw43--sw55)	cb23-(sw43--sw55)
105	cb32-(sw59--sw60)		cb32-(sw48--sw60)	cb24-(sw48--sw60)
106	cb33-(sw60--sw61)	cb32-(sw60--sw61)	cb33-(sw49--sw61)	cb25-(sw49--sw61)
107	cb34-(sw61--sw62)	cb33-(sw61--sw62)	cb34-(sw50--sw62)	cb26-(sw50--sw62)
108	cb35-(sw62--sw63)	cb34-(sw62--sw63)	cb35-(sw51--sw63)	cb27-(sw51--sw63)
109	cb36-(sw63--sw64)	cb35-(sw63--sw64)	cb36-(sw52--sw64)	cb28-(sw52--sw64)
110	cb37-(sw64--sw65)	cb36-(sw64--sw65)	cb37-(sw53--sw65)	cb29-(sw53--sw65)
111	cb38-(sw65--sw66)	cb37-(sw65--sw66)	cb38-(sw54--sw66)	cb30-(sw54--sw66)
112	cb39-(sw66--sw67)	cb38-(sw66--sw67)	cb39-(sw55--sw67)	cb31-(sw55--sw67)
113	cb40-(sw71--sw72)		cb40-(sw60--sw72)	cb32-(sw60--sw72)
114	cb41-(sw72--sw73)	cb40-(sw72--sw73)	cb41-(sw61--sw73)	cb33-(sw61--sw73)
115	cb42-(sw73--sw74)	cb41-(sw73--sw74)	cb42-(sw62--sw74)	cb34-(sw62--sw74)
116	cb43-(sw74--sw75)	cb42-(sw74--sw75)	cb43-(sw63--sw75)	cb35-(sw63--sw75)
117	cb44-(sw75--sw76)	cb43-(sw75--sw76)	cb44-(sw64--sw76)	cb36-(sw64--sw76)
118	cb45-(sw76--sw77)	cb44-(sw76--sw77)	cb45-(sw65--sw77)	cb37-(sw65--sw77)
119	cb46-(sw77--sw78)	cb45-(sw77--sw78)	cb46-(sw66--sw78)	cb38-(sw66--sw78)
120	cb47-(sw78--sw79)	cb46-(sw78--sw79)	cb47-(sw67--sw79)	cb39-(sw67--sw79)
121	cb48-(sw83--sw84)		cb48-(sw72--sw84)	cb40-(sw72--sw84)
122	cb49-(sw84--sw85)	cb48-(sw84--sw85)	cb49-(sw73--sw85)	cb41-(sw73--sw85)
123	cb50-(sw85--sw86)	cb49-(sw85--sw86)	cb50-(sw74--sw86)	cb42-(sw74--sw86)
124	cb51-(sw86--sw87)	cb50-(sw86--sw87)	cb51-(sw75--sw87)	cb43-(sw75--sw87)
125	cb52-(sw87--sw88)	cb51-(sw87--sw88)	cb52-(sw76--sw88)	cb44-(sw76--sw88)
126	cb53-(sw88--sw89)	cb52-(sw88--sw89)	cb53-(sw77--sw89)	cb45-(sw77--sw89)
127	cb54-(sw89--sw90)	cb53-(sw89--sw90)	cb54-(sw78--sw90)	cb46-(sw78--sw90)
128	cb55-(sw90--sw91)	cb54-(sw90--sw91)	cb55-(sw79--sw91)	cb47-(sw79--sw91)
129	cb56-(sw95--sw96)		cb56-(sw84--sw96)	cb48-(sw84--sw96)
130	cb57-(sw96--sw97)	cb56-(sw96--sw97)	cb57-(sw85--sw97)	cb49-(sw85--sw97)
131	cb58-(sw97--sw98)	cb57-(sw97--sw98)	cb58-(sw86--sw98)	cb50-(sw86--sw98)
132	cb59-(sw98--sw99)	cb58-(sw98--sw99)	cb59-(sw87--sw99)	cb51-(sw87--sw99)
133	cb60-(sw99--sw100)	cb59-(sw99--sw100)	cb60-(sw88--sw100)	cb52-(sw88--sw100)
134	cb61-(sw100--sw101)	cb60-(sw100--sw101)	cb61-(sw89--sw101)	cb53-(sw89--sw101)
135	cb62-(sw101--sw102)	cb61-(sw101--sw102)	cb62-(sw90--sw102)	cb54-(sw90--sw102)
136	cb63-(sw102--sw103)	cb62-(sw102--sw103)	cb63-(sw91--sw103)	cb55-(sw91--sw103)
137		cb63-(sw103--sw104)		cb56-(sw96--sw108)
138		cb7-(sw19--sw20)		cb57-(sw97--sw109)
139		cb15-(sw31--sw32)		cb58-(sw98--sw110)
140		cb23-(sw43--sw44)		cb59-(sw99--sw111)
141		cb31-(sw55--sw56)		cb60-(sw100--sw112)
142		cb39-(sw67--sw68)		cb61-(sw101--sw113)
143		cb47-(sw79--sw80)		cb62-(sw102--sw114)
144		cb55-(sw91--sw92)		cb63-(sw103--sw115)

Figure A.4 Quarter-Pixel Interpolation and Search Flow for 8x8 block size

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw19--sw0)		cb0-(sw19--sw40)	
2	cb1-(sw0--sw21)	cb0-(sw0--sw21)	cb1-(sw40--sw21)	cb0-(sw40--sw21)
3	cb2-(sw21--sw2)	cb1-(sw21--sw2)	cb2-(sw21--sw42)	cb1-(sw21--sw42)
4	cb3-(sw2--sw23)	cb2-(sw2--sw23)	cb3-(sw42--sw23)	cb2-(sw42--sw23)
5	cb4-(sw23--sw4)	cb3-(sw23--sw4)	cb4-(sw23--sw44)	cb3-(sw23--sw44)
6	cb5-(sw4--sw25)	cb4-(sw4--sw25)	cb5-(sw44--sw25)	cb4-(sw44--sw25)
7	cb6-(sw25--sw6)	cb5-(sw25--sw6)	cb6-(sw25--sw46)	cb5-(sw25--sw46)
8	cb7-(sw6--sw27)	cb6-(sw6--sw27)	cb7-(sw46--sw27)	cb6-(sw46--sw27)
9	cb8-(sw27--sw8)	cb7-(sw27--sw8)	cb8-(sw27--sw48)	cb7-(sw27--sw48)
10	cb9-(sw8--sw29)	cb8-(sw8--sw29)	cb9-(sw48--sw29)	cb8-(sw48--sw29)
11	cb10-(sw29--sw10)	cb9-(sw29--sw10)	cb10-(sw29--sw50)	cb9-(sw29--sw50)
12	cb11-(sw10--sw31)	cb10-(sw10--sw31)	cb11-(sw50--sw31)	cb10-(sw50--sw31)
13	cb12-(sw31--sw12)	cb11-(sw31--sw12)	cb12-(sw31--sw52)	cb11-(sw31--sw52)
14	cb13-(sw12--sw33)	cb12-(sw12--sw33)	cb13-(sw52--sw33)	cb12-(sw52--sw33)
15	cb14-(sw33--sw14)	cb13-(sw33--sw14)	cb14-(sw33--sw54)	cb13-(sw33--sw54)
16	cb15-(sw14--sw35)	cb14-(sw14--sw35)	cb15-(sw54--sw35)	cb14-(sw54--sw35)
17	cb16-(sw19--sw40)		cb16-(sw59--sw40)	
18	cb17-(sw40--sw21)	cb16-(sw40--sw21)	cb17-(sw40--sw61)	cb16-(sw40--sw61)
19	cb18-(sw21--sw42)	cb17-(sw21--sw42)	cb18-(sw61--sw42)	cb17-(sw61--sw42)
20	cb19-(sw42--sw23)	cb18-(sw42--sw23)	cb19-(sw42--sw63)	cb18-(sw42--sw63)
21	cb20-(sw23--sw44)	cb19-(sw23--sw44)	cb20-(sw63--sw44)	cb19-(sw63--sw44)
22	cb21-(sw44--sw25)	cb20-(sw44--sw25)	cb21-(sw44--sw65)	cb20-(sw44--sw65)
23	cb22-(sw25--sw46)	cb21-(sw25--sw46)	cb22-(sw65--sw46)	cb21-(sw65--sw46)
24	cb23-(sw46--sw27)	cb22-(sw46--sw27)	cb23-(sw46--sw67)	cb22-(sw46--sw67)
25	cb24-(sw27--sw48)	cb23-(sw27--sw48)	cb24-(sw67--sw48)	cb23-(sw67--sw48)
26	cb25-(sw48--sw29)	cb24-(sw48--sw29)	cb25-(sw48--sw69)	cb24-(sw48--sw69)
27	cb26-(sw29--sw50)	cb25-(sw29--sw50)	cb26-(sw69--sw50)	cb25-(sw69--sw50)
28	cb27-(sw50--sw31)	cb26-(sw50--sw31)	cb27-(sw50--sw71)	cb26-(sw50--sw71)
29	cb28-(sw31--sw52)	cb27-(sw31--sw52)	cb28-(sw71--sw52)	cb27-(sw71--sw52)
30	cb29-(sw52--sw33)	cb28-(sw52--sw33)	cb29-(sw52--sw73)	cb28-(sw52--sw73)
31	cb30-(sw33--sw54)	cb29-(sw33--sw54)	cb30-(sw73--sw54)	cb29-(sw73--sw54)
32	cb31-(sw54--sw35)	cb30-(sw54--sw35)	cb31-(sw54--sw75)	cb30-(sw54--sw75)
33	cb32-(sw59--sw40)		cb32-(sw59--sw80)	
34	cb33-(sw40--sw61)	cb32-(sw40--sw61)	cb33-(sw80--sw61)	cb32-(sw80--sw61)
35	cb34-(sw61--sw42)	cb33-(sw61--sw42)	cb34-(sw61--sw82)	cb33-(sw61--sw82)
36	cb35-(sw42--sw63)	cb34-(sw42--sw63)	cb35-(sw82--sw63)	cb34-(sw82--sw63)
37	cb36-(sw63--sw44)	cb35-(sw63--sw44)	cb36-(sw63--sw84)	cb35-(sw63--sw84)
38	cb37-(sw44--sw65)	cb36-(sw44--sw65)	cb37-(sw84--sw65)	cb36-(sw84--sw65)
39	cb38-(sw65--sw46)	cb37-(sw65--sw46)	cb38-(sw65--sw86)	cb37-(sw65--sw86)
40	cb39-(sw46--sw67)	cb38-(sw46--sw67)	cb39-(sw86--sw67)	cb38-(sw86--sw67)
41	cb40-(sw67--sw48)	cb39-(sw67--sw48)	cb40-(sw67--sw88)	cb39-(sw67--sw88)
42	cb41-(sw48--sw69)	cb40-(sw48--sw69)	cb41-(sw88--sw69)	cb40-(sw88--sw69)

43	cb42-(sw69--sw50)	cb41-(sw69--sw50)	cb42-(sw69--sw90)	cb41-(sw69--sw90)
44	cb43-(sw50--sw71)	cb42-(sw50--sw71)	cb43-(sw90--sw71)	cb42-(sw90--sw71)
45	cb44-(sw71--sw52)	cb43-(sw71--sw52)	cb44-(sw71--sw92)	cb43-(sw71--sw92)
46	cb45-(sw52--sw73)	cb44-(sw52--sw73)	cb45-(sw92--sw73)	cb44-(sw92--sw73)
47	cb46-(sw73--sw54)	cb45-(sw73--sw54)	cb46-(sw73--sw94)	cb45-(sw73--sw94)
48	cb47-(sw54--sw75)	cb46-(sw54--sw75)	cb47-(sw94--sw75)	cb46-(sw94--sw75)
49	cb48-(sw59--sw80)		cb48-(sw99--sw80)	
50	cb49-(sw80--sw61)	cb48-(sw80--sw61)	cb49-(sw80--sw101)	cb48-(sw80--sw101)
51	cb50-(sw61--sw82)	cb49-(sw61--sw82)	cb50-(sw101--sw82)	cb49-(sw101--sw82)
52	cb51-(sw82--sw63)	cb50-(sw82--sw63)	cb51-(sw82--sw103)	cb50-(sw82--sw103)
53	cb52-(sw63--sw84)	cb51-(sw63--sw84)	cb52-(sw103--sw84)	cb51-(sw103--sw84)
54	cb53-(sw84--sw65)	cb52-(sw84--sw65)	cb53-(sw84--sw105)	cb52-(sw84--sw105)
55	cb54-(sw65--sw86)	cb53-(sw65--sw86)	cb54-(sw105--sw86)	cb53-(sw105--sw86)
56	cb55-(sw86--sw67)	cb54-(sw86--sw67)	cb55-(sw86--sw107)	cb54-(sw86--sw107)
57	cb56-(sw67--sw88)	cb55-(sw67--sw88)	cb56-(sw107--sw88)	cb55-(sw107--sw88)
58	cb57-(sw88--sw69)	cb56-(sw88--sw69)	cb57-(sw88--sw109)	cb56-(sw88--sw109)
59	cb58-(sw69--sw90)	cb57-(sw69--sw90)	cb58-(sw109--sw90)	cb57-(sw109--sw90)
60	cb59-(sw90--sw71)	cb58-(sw90--sw71)	cb59-(sw90--sw111)	cb58-(sw90--sw111)
61	cb60-(sw71--sw92)	cb59-(sw71--sw92)	cb60-(sw111--sw92)	cb59-(sw111--sw92)
62	cb61-(sw92--sw73)	cb60-(sw92--sw73)	cb61-(sw92--sw113)	cb60-(sw92--sw113)
63	cb62-(sw73--sw94)	cb61-(sw73--sw94)	cb62-(sw113--sw94)	cb61-(sw113--sw94)
64	cb63-(sw94--sw75)	cb62-(sw94--sw75)	cb63-(sw94--sw115)	cb62-(sw94--sw115)
65	cb64-(sw99--sw80)		cb64-(sw99--sw120)	
66	cb65-(sw80--sw101)	cb64-(sw80--sw101)	cb65-(sw120--sw101)	cb64-(sw120--sw101)
67	cb66-(sw101--sw82)	cb65-(sw101--sw82)	cb66-(sw101--sw122)	cb65-(sw101--sw122)
68	cb67-(sw82--sw103)	cb66-(sw82--sw103)	cb67-(sw122--sw103)	cb66-(sw122--sw103)
69	cb68-(sw103--sw84)	cb67-(sw103--sw84)	cb68-(sw103--sw124)	cb67-(sw103--sw124)
70	cb69-(sw84--sw105)	cb68-(sw84--sw105)	cb69-(sw124--sw105)	cb68-(sw124--sw105)
71	cb70-(sw105--sw86)	cb69-(sw105--sw86)	cb70-(sw105--sw126)	cb69-(sw105--sw126)
72	cb71-(sw86--sw107)	cb70-(sw86--sw107)	cb71-(sw126--sw107)	cb70-(sw126--sw107)
73	cb72-(sw107--sw88)	cb71-(sw107--sw88)	cb72-(sw107--sw128)	cb71-(sw107--sw128)
74	cb73-(sw88--sw109)	cb72-(sw88--sw109)	cb73-(sw128--sw109)	cb72-(sw128--sw109)
75	cb74-(sw109--sw90)	cb73-(sw109--sw90)	cb74-(sw109--sw130)	cb73-(sw109--sw130)
76	cb75-(sw90--sw111)	cb74-(sw90--sw111)	cb75-(sw130--sw111)	cb74-(sw130--sw111)
77	cb76-(sw111--sw92)	cb75-(sw111--sw92)	cb76-(sw111--sw132)	cb75-(sw111--sw132)
78	cb77-(sw92--sw113)	cb76-(sw92--sw113)	cb77-(sw132--sw113)	cb76-(sw132--sw113)
79	cb78-(sw113--sw94)	cb77-(sw113--sw94)	cb78-(sw113--sw134)	cb77-(sw113--sw134)
80	cb79-(sw94--sw115)	cb78-(sw94--sw115)	cb79-(sw134--sw115)	cb78-(sw134--sw115)
81	cb80-(sw99--sw120)		cb80-(sw139--sw120)	
82	cb81-(sw120--sw101)	cb80-(sw120--sw101)	cb81-(sw120--sw141)	cb80-(sw120--sw141)
83	cb82-(sw101--sw122)	cb81-(sw101--sw122)	cb82-(sw141--sw122)	cb81-(sw141--sw122)
84	cb83-(sw122--sw103)	cb82-(sw122--sw103)	cb83-(sw122--sw143)	cb82-(sw122--sw143)
85	cb84-(sw103--sw124)	cb83-(sw103--sw124)	cb84-(sw143--sw124)	cb83-(sw143--sw124)
86	cb85-(sw124--sw105)	cb84-(sw124--sw105)	cb85-(sw124--sw145)	cb84-(sw124--sw145)
87	cb86-(sw105--sw126)	cb85-(sw105--sw126)	cb86-(sw145--sw126)	cb85-(sw145--sw126)

88	cb87-(sw126--sw107)	cb86-(sw126--sw107)	cb87-(sw126--sw147)	cb86-(sw126--sw147)
89	cb88-(sw107--sw128)	cb87-(sw107--sw128)	cb88-(sw147--sw128)	cb87-(sw147--sw128)
90	cb89-(sw128--sw109)	cb88-(sw128--sw109)	cb89-(sw128--sw149)	cb88-(sw128--sw149)
91	cb90-(sw109--sw130)	cb89-(sw109--sw130)	cb90-(sw149--sw130)	cb89-(sw149--sw130)
92	cb91-(sw130--sw111)	cb90-(sw130--sw111)	cb91-(sw130--sw151)	cb90-(sw130--sw151)
93	cb92-(sw111--sw132)	cb91-(sw111--sw132)	cb92-(sw151--sw132)	cb91-(sw151--sw132)
94	cb93-(sw132--sw113)	cb92-(sw132--sw113)	cb93-(sw132--sw153)	cb92-(sw132--sw153)
95	cb94-(sw113--sw134)	cb93-(sw113--sw134)	cb94-(sw153--sw134)	cb93-(sw153--sw134)
96	cb95-(sw134--sw115)	cb94-(sw134--sw115)	cb95-(sw134--sw155)	cb94-(sw134--sw155)
97	cb96-(sw139--sw120)		cb96-(sw139--sw160)	
98	cb97-(sw120--sw141)	cb96-(sw120--sw141)	cb97-(sw160--sw141)	cb96-(sw160--sw141)
99	cb98-(sw141--sw122)	cb97-(sw141--sw122)	cb98-(sw141--sw162)	cb97-(sw141--sw162)
100	cb99-(sw122--sw143)	cb98-(sw122--sw143)	cb99-(sw162--sw143)	cb98-(sw162--sw143)
101	cb100-(sw143--sw124)	cb99-(sw143--sw124)	cb100-(sw143--sw164)	cb99-(sw143--sw164)
102	cb101-(sw124--sw145)	cb100-(sw124--sw145)	cb101-(sw164--sw145)	cb100-(sw164--sw145)
103	cb102-(sw145--sw126)	cb101-(sw145--sw126)	cb102-(sw145--sw166)	cb101-(sw145--sw166)
104	cb103-(sw126--sw147)	cb102-(sw126--sw147)	cb103-(sw166--sw147)	cb102-(sw166--sw147)
105	cb104-(sw147--sw128)	cb103-(sw147--sw128)	cb104-(sw147--sw168)	cb103-(sw147--sw168)
106	cb105-(sw128--sw149)	cb104-(sw128--sw149)	cb105-(sw168--sw149)	cb104-(sw168--sw149)
107	cb106-(sw149--sw130)	cb105-(sw149--sw130)	cb106-(sw149--sw170)	cb105-(sw149--sw170)
108	cb107-(sw130--sw151)	cb106-(sw130--sw151)	cb107-(sw170--sw151)	cb106-(sw170--sw151)
109	cb108-(sw151--sw132)	cb107-(sw151--sw132)	cb108-(sw151--sw172)	cb107-(sw151--sw172)
110	cb109-(sw132--sw153)	cb108-(sw132--sw153)	cb109-(sw172--sw153)	cb108-(sw172--sw153)
111	cb110-(sw153--sw134)	cb109-(sw153--sw134)	cb110-(sw153--sw174)	cb109-(sw153--sw174)
112	cb111-(sw134--sw155)	cb110-(sw134--sw155)	cb111-(sw174--sw155)	cb110-(sw174--sw155)
113	cb112-(sw139--sw160)		cb112-(sw179--sw160)	
114	cb113-(sw160--sw141)	cb112-(sw160--sw141)	cb113-(sw160--sw181)	cb112-(sw160--sw181)
115	cb114-(sw141--sw162)	cb113-(sw141--sw162)	cb114-(sw181--sw162)	cb113-(sw181--sw162)
116	cb115-(sw162--sw143)	cb114-(sw162--sw143)	cb115-(sw162--sw183)	cb114-(sw162--sw183)
117	cb116-(sw143--sw164)	cb115-(sw143--sw164)	cb116-(sw183--sw164)	cb115-(sw183--sw164)
118	cb117-(sw164--sw145)	cb116-(sw164--sw145)	cb117-(sw164--sw185)	cb116-(sw164--sw185)
119	cb118-(sw145--sw166)	cb117-(sw145--sw166)	cb118-(sw185--sw166)	cb117-(sw185--sw166)
120	cb119-(sw166--sw147)	cb118-(sw166--sw147)	cb119-(sw166--sw187)	cb118-(sw166--sw187)
121	cb120-(sw147--sw168)	cb119-(sw147--sw168)	cb120-(sw187--sw168)	cb119-(sw187--sw168)
122	cb121-(sw168--sw149)	cb120-(sw168--sw149)	cb121-(sw168--sw189)	cb120-(sw168--sw189)
123	cb122-(sw149--sw170)	cb121-(sw149--sw170)	cb122-(sw189--sw170)	cb121-(sw189--sw170)
124	cb123-(sw170--sw151)	cb122-(sw170--sw151)	cb123-(sw170--sw191)	cb122-(sw170--sw191)
125	cb124-(sw151--sw172)	cb123-(sw151--sw172)	cb124-(sw191--sw172)	cb123-(sw191--sw172)
126	cb125-(sw172--sw153)	cb124-(sw172--sw153)	cb125-(sw172--sw193)	cb124-(sw172--sw193)
127	cb126-(sw153--sw174)	cb125-(sw153--sw174)	cb126-(sw193--sw174)	cb125-(sw193--sw174)
128	cb127-(sw174--sw155)	cb126-(sw174--sw155)	cb127-(sw174--sw195)	cb126-(sw174--sw195)
129		cb127-(sw155--sw176)		cb127-(sw195--sw176)
130		cb15-(sw35--sw16)		cb15-(sw35--sw56)
131		cb31-(sw35--sw56)		cb31-(sw75--sw56)
132		cb47-(sw75--sw56)		cb47-(sw75--sw96)

133		cb63-(sw75--sw96)		cb63-(sw115--sw96)
134		cb79-(sw115--sw96)		cb79-(sw115--sw136)
135		cb95-(sw115--sw136)		cb95-(sw155--sw136)
136		cb111-(sw155--sw136)		cb111-(sw155--sw176)
137	cb0-(sw19--sw20)		cb0-(sw0--sw20)	
138	cb1-(sw20--sw21)	cb0-(sw0--sw21)	cb1-(sw1--sw21)	
139	cb2-(sw21--sw22)	cb1-(sw21--sw2)	cb2-(sw2--sw22)	
140	cb3-(sw22--sw23)	cb2-(sw2--sw23)	cb3-(sw3--sw23)	
141	cb4-(sw23--sw24)	cb3-(sw23--sw4)	cb4-(sw4--sw24)	
142	cb5-(sw24--sw25)	cb4-(sw4--sw25)	cb5-(sw5--sw25)	
143	cb6-(sw25--sw26)	cb5-(sw25--sw6)	cb6-(sw6--sw26)	
144	cb7-(sw26--sw27)	cb6-(sw6--sw27)	cb7-(sw7--sw27)	
145	cb8-(sw27--sw28)	cb7-(sw27--sw8)	cb8-(sw8--sw28)	
146	cb9-(sw28--sw29)	cb8-(sw8--sw29)	cb9-(sw9--sw29)	
147	cb10-(sw29--sw30)	cb9-(sw29--sw10)	cb10-(sw10--sw30)	
148	cb11-(sw30--sw31)	cb10-(sw10--sw31)	cb11-(sw11--sw31)	
149	cb12-(sw31--sw32)	cb11-(sw31--sw12)	cb12-(sw12--sw32)	
150	cb13-(sw32--sw33)	cb12-(sw12--sw33)	cb13-(sw13--sw33)	
151	cb14-(sw33--sw34)	cb13-(sw33--sw14)	cb14-(sw14--sw34)	
152	cb15-(sw34--sw35)	cb14-(sw14--sw35)	cb15-(sw15--sw35)	
153	cb16-(sw39--sw40)		cb16-(sw20--sw40)	cb0-(sw20--sw40)
154	cb17-(sw40--sw41)	cb16-(sw40--sw21)	cb17-(sw21--sw41)	cb1-(sw21--sw41)
155	cb18-(sw41--sw42)	cb17-(sw21--sw42)	cb18-(sw22--sw42)	cb2-(sw22--sw42)
156	cb19-(sw42--sw43)	cb18-(sw42--sw23)	cb19-(sw23--sw43)	cb3-(sw23--sw43)
157	cb20-(sw43--sw44)	cb19-(sw23--sw44)	cb20-(sw24--sw44)	cb4-(sw24--sw44)
158	cb21-(sw44--sw45)	cb20-(sw44--sw25)	cb21-(sw25--sw45)	cb5-(sw25--sw45)
159	cb22-(sw45--sw46)	cb21-(sw25--sw46)	cb22-(sw26--sw46)	cb6-(sw26--sw46)
160	cb23-(sw46--sw47)	cb22-(sw46--sw27)	cb23-(sw27--sw47)	cb7-(sw27--sw47)
161	cb24-(sw47--sw48)	cb23-(sw27--sw48)	cb24-(sw28--sw48)	cb8-(sw28--sw48)
162	cb25-(sw48--sw49)	cb24-(sw48--sw29)	cb25-(sw29--sw49)	cb9-(sw29--sw49)
163	cb26-(sw49--sw50)	cb25-(sw29--sw50)	cb26-(sw30--sw50)	cb10-(sw30--sw50)
164	cb27-(sw50--sw51)	cb26-(sw50--sw31)	cb27-(sw31--sw51)	cb11-(sw31--sw51)
165	cb28-(sw51--sw52)	cb27-(sw31--sw52)	cb28-(sw32--sw52)	cb12-(sw32--sw52)
166	cb29-(sw52--sw53)	cb28-(sw52--sw33)	cb29-(sw33--sw53)	cb13-(sw33--sw53)
167	cb30-(sw53--sw54)	cb29-(sw33--sw54)	cb30-(sw34--sw54)	cb14-(sw34--sw54)
168	cb31-(sw54--sw55)	cb30-(sw54--sw35)	cb31-(sw35--sw55)	cb15-(sw35--sw55)
169	cb32-(sw59--sw60)		cb32-(sw40--sw60)	cb16-(sw40--sw60)
170	cb33-(sw60--sw61)	cb32-(sw40--sw61)	cb33-(sw41--sw61)	cb17-(sw41--sw61)
171	cb34-(sw61--sw62)	cb33-(sw61--sw42)	cb34-(sw42--sw62)	cb18-(sw42--sw62)
172	cb35-(sw62--sw63)	cb34-(sw42--sw63)	cb35-(sw43--sw63)	cb19-(sw43--sw63)
173	cb36-(sw63--sw64)	cb35-(sw63--sw44)	cb36-(sw44--sw64)	cb20-(sw44--sw64)
174	cb37-(sw64--sw65)	cb36-(sw44--sw65)	cb37-(sw45--sw65)	cb21-(sw45--sw65)
175	cb38-(sw65--sw66)	cb37-(sw65--sw46)	cb38-(sw46--sw66)	cb22-(sw46--sw66)
176	cb39-(sw66--sw67)	cb38-(sw46--sw67)	cb39-(sw47--sw67)	cb23-(sw47--sw67)
177	cb40-(sw67--sw68)	cb39-(sw67--sw48)	cb40-(sw48--sw68)	cb24-(sw48--sw68)

178	cb41-(sw68--sw69)	cb40-(sw48--sw69)	cb41-(sw49--sw69)	cb25-(sw49--sw69)
179	cb42-(sw69--sw70)	cb41-(sw69--sw50)	cb42-(sw50--sw70)	cb26-(sw50--sw70)
180	cb43-(sw70--sw71)	cb42-(sw50--sw71)	cb43-(sw51--sw71)	cb27-(sw51--sw71)
181	cb44-(sw71--sw72)	cb43-(sw71--sw52)	cb44-(sw52--sw72)	cb28-(sw52--sw72)
182	cb45-(sw72--sw73)	cb44-(sw52--sw73)	cb45-(sw53--sw73)	cb29-(sw53--sw73)
183	cb46-(sw73--sw74)	cb45-(sw73--sw54)	cb46-(sw54--sw74)	cb30-(sw54--sw74)
184	cb47-(sw74--sw75)	cb46-(sw54--sw75)	cb47-(sw55--sw75)	cb31-(sw55--sw75)
185	cb48-(sw79--sw80)		cb48-(sw60--sw80)	cb32-(sw60--sw80)
186	cb49-(sw80--sw81)	cb48-(sw80--sw61)	cb49-(sw61--sw81)	cb33-(sw61--sw81)
187	cb50-(sw81--sw82)	cb49-(sw61--sw82)	cb50-(sw62--sw82)	cb34-(sw62--sw82)
188	cb51-(sw82--sw83)	cb50-(sw82--sw63)	cb51-(sw63--sw83)	cb35-(sw63--sw83)
189	cb52-(sw83--sw84)	cb51-(sw63--sw84)	cb52-(sw64--sw84)	cb36-(sw64--sw84)
190	cb53-(sw84--sw85)	cb52-(sw84--sw65)	cb53-(sw65--sw85)	cb37-(sw65--sw85)
191	cb54-(sw85--sw86)	cb53-(sw65--sw86)	cb54-(sw66--sw86)	cb38-(sw66--sw86)
192	cb55-(sw86--sw87)	cb54-(sw86--sw67)	cb55-(sw67--sw87)	cb39-(sw67--sw87)
193	cb56-(sw87--sw88)	cb55-(sw67--sw88)	cb56-(sw68--sw88)	cb40-(sw68--sw88)
194	cb57-(sw88--sw89)	cb56-(sw88--sw69)	cb57-(sw69--sw89)	cb41-(sw69--sw89)
195	cb58-(sw89--sw90)	cb57-(sw69--sw90)	cb58-(sw70--sw90)	cb42-(sw70--sw90)
196	cb59-(sw90--sw91)	cb58-(sw90--sw71)	cb59-(sw71--sw91)	cb43-(sw71--sw91)
197	cb60-(sw91--sw92)	cb59-(sw71--sw92)	cb60-(sw72--sw92)	cb44-(sw72--sw92)
198	cb61-(sw92--sw93)	cb60-(sw92--sw73)	cb61-(sw73--sw93)	cb45-(sw73--sw93)
199	cb62-(sw93--sw94)	cb61-(sw73--sw94)	cb62-(sw74--sw94)	cb46-(sw74--sw94)
200	cb63-(sw94--sw95)	cb62-(sw94--sw75)	cb63-(sw75--sw95)	cb47-(sw75--sw95)
201	cb64-(sw99--sw100)		cb64-(sw80--sw100)	cb48-(sw80--sw100)
202	cb65-(sw100--sw101)	cb64-(sw80--sw101)	cb65-(sw81--sw101)	cb49-(sw81--sw101)
203	cb66-(sw101--sw102)	cb65-(sw101--sw82)	cb66-(sw82--sw102)	cb50-(sw82--sw102)
204	cb67-(sw102--sw103)	cb66-(sw82--sw103)	cb67-(sw83--sw103)	cb51-(sw83--sw103)
205	cb68-(sw103--sw104)	cb67-(sw103--sw84)	cb68-(sw84--sw104)	cb52-(sw84--sw104)
206	cb69-(sw104--sw105)	cb68-(sw84--sw105)	cb69-(sw85--sw105)	cb53-(sw85--sw105)
207	cb70-(sw105--sw106)	cb69-(sw105--sw86)	cb70-(sw86--sw106)	cb54-(sw86--sw106)
208	cb71-(sw106--sw107)	cb70-(sw86--sw107)	cb71-(sw87--sw107)	cb55-(sw87--sw107)
209	cb72-(sw107--sw108)	cb71-(sw107--sw88)	cb72-(sw88--sw108)	cb56-(sw88--sw108)
210	cb73-(sw108--sw109)	cb72-(sw88--sw109)	cb73-(sw89--sw109)	cb57-(sw89--sw109)
211	cb74-(sw109--sw110)	cb73-(sw109--sw90)	cb74-(sw90--sw110)	cb58-(sw90--sw110)
212	cb75-(sw110--sw111)	cb74-(sw90--sw111)	cb75-(sw91--sw111)	cb59-(sw91--sw111)
213	cb76-(sw111--sw112)	cb75-(sw111--sw92)	cb76-(sw92--sw112)	cb60-(sw92--sw112)
214	cb77-(sw112--sw113)	cb76-(sw92--sw113)	cb77-(sw93--sw113)	cb61-(sw93--sw113)
215	cb78-(sw113--sw114)	cb77-(sw113--sw94)	cb78-(sw94--sw114)	cb62-(sw94--sw114)
216	cb79-(sw114--sw115)	cb78-(sw94--sw115)	cb79-(sw95--sw115)	cb63-(sw95--sw115)
217	cb80-(sw119--sw120)		cb80-(sw100--sw120)	cb64-(sw100--sw120)
218	cb81-(sw120--sw121)	cb80-(sw120--sw101)	cb81-(sw101--sw121)	cb65-(sw101--sw121)
219	cb82-(sw121--sw122)	cb81-(sw101--sw122)	cb82-(sw102--sw122)	cb66-(sw102--sw122)
220	cb83-(sw122--sw123)	cb82-(sw122--sw103)	cb83-(sw103--sw123)	cb67-(sw103--sw123)
221	cb84-(sw123--sw124)	cb83-(sw103--sw124)	cb84-(sw104--sw124)	cb68-(sw104--sw124)
222	cb85-(sw124--sw125)	cb84-(sw124--sw105)	cb85-(sw105--sw125)	cb69-(sw105--sw125)

223	cb86-(sw125--sw126)	cb85-(sw105--sw126)	cb86-(sw106--sw126)	cb70-(sw106--sw126)
224	cb87-(sw126--sw127)	cb86-(sw126--sw107)	cb87-(sw107--sw127)	cb71-(sw107--sw127)
225	cb88-(sw127--sw128)	cb87-(sw107--sw128)	cb88-(sw108--sw128)	cb72-(sw108--sw128)
226	cb89-(sw128--sw129)	cb88-(sw128--sw109)	cb89-(sw109--sw129)	cb73-(sw109--sw129)
227	cb90-(sw129--sw130)	cb89-(sw109--sw130)	cb90-(sw110--sw130)	cb74-(sw110--sw130)
228	cb91-(sw130--sw131)	cb90-(sw130--sw111)	cb91-(sw111--sw131)	cb75-(sw111--sw131)
229	cb92-(sw131--sw132)	cb91-(sw111--sw132)	cb92-(sw112--sw132)	cb76-(sw112--sw132)
230	cb93-(sw132--sw133)	cb92-(sw132--sw113)	cb93-(sw113--sw133)	cb77-(sw113--sw133)
231	cb94-(sw133--sw134)	cb93-(sw113--sw134)	cb94-(sw114--sw134)	cb78-(sw114--sw134)
232	cb95-(sw134--sw135)	cb94-(sw134--sw115)	cb95-(sw115--sw135)	cb79-(sw115--sw135)
233	cb96-(sw139--sw140)		cb96-(sw120--sw140)	cb80-(sw120--sw140)
234	cb97-(sw140--sw141)	cb96-(sw120--sw141)	cb97-(sw121--sw141)	cb81-(sw121--sw141)
235	cb98-(sw141--sw142)	cb97-(sw141--sw122)	cb98-(sw122--sw142)	cb82-(sw122--sw142)
236	cb99-(sw142--sw143)	cb98-(sw122--sw143)	cb99-(sw123--sw143)	cb83-(sw123--sw143)
237	cb100-(sw143--sw144)	cb99-(sw143--sw124)	cb100-(sw124--sw144)	cb84-(sw124--sw144)
238	cb101-(sw144--sw145)	cb100-(sw124--sw145)	cb101-(sw125--sw145)	cb85-(sw125--sw145)
239	cb102-(sw145--sw146)	cb101-(sw145--sw126)	cb102-(sw126--sw146)	cb86-(sw126--sw146)
240	cb103-(sw146--sw147)	cb102-(sw126--sw147)	cb103-(sw127--sw147)	cb87-(sw127--sw147)
241	cb104-(sw147--sw148)	cb103-(sw147--sw128)	cb104-(sw128--sw148)	cb88-(sw128--sw148)
242	cb105-(sw148--sw149)	cb104-(sw128--sw149)	cb105-(sw129--sw149)	cb89-(sw129--sw149)
243	cb106-(sw149--sw150)	cb105-(sw149--sw130)	cb106-(sw130--sw150)	cb90-(sw130--sw150)
244	cb107-(sw150--sw151)	cb106-(sw130--sw151)	cb107-(sw131--sw151)	cb91-(sw131--sw151)
245	cb108-(sw151--sw152)	cb107-(sw151--sw132)	cb108-(sw132--sw152)	cb92-(sw132--sw152)
246	cb109-(sw152--sw153)	cb108-(sw132--sw153)	cb109-(sw133--sw153)	cb93-(sw133--sw153)
247	cb110-(sw153--sw154)	cb109-(sw153--sw134)	cb110-(sw134--sw154)	cb94-(sw134--sw154)
248	cb111-(sw154--sw155)	cb110-(sw134--sw155)	cb111-(sw135--sw155)	cb95-(sw135--sw155)
249	cb112-(sw159--sw160)		cb112-(sw140--sw160)	cb96-(sw140--sw160)
250	cb113-(sw160--sw161)	cb112-(sw160--sw141)	cb113-(sw141--sw161)	cb97-(sw141--sw161)
251	cb114-(sw161--sw162)	cb113-(sw141--sw162)	cb114-(sw142--sw162)	cb98-(sw142--sw162)
252	cb115-(sw162--sw163)	cb114-(sw162--sw143)	cb115-(sw143--sw163)	cb99-(sw143--sw163)
253	cb116-(sw163--sw164)	cb115-(sw143--sw164)	cb116-(sw144--sw164)	cb100-(sw144--sw164)
254	cb117-(sw164--sw165)	cb116-(sw164--sw145)	cb117-(sw145--sw165)	cb101-(sw145--sw165)
255	cb118-(sw165--sw166)	cb117-(sw145--sw166)	cb118-(sw146--sw166)	cb102-(sw146--sw166)
256	cb119-(sw166--sw167)	cb118-(sw166--sw147)	cb119-(sw147--sw167)	cb103-(sw147--sw167)
257	cb120-(sw167--sw168)	cb119-(sw147--sw168)	cb120-(sw148--sw168)	cb104-(sw148--sw168)
258	cb121-(sw168--sw169)	cb120-(sw168--sw149)	cb121-(sw149--sw169)	cb105-(sw149--sw169)
259	cb122-(sw169--sw170)	cb121-(sw149--sw170)	cb122-(sw150--sw170)	cb106-(sw150--sw170)
260	cb123-(sw170--sw171)	cb122-(sw170--sw151)	cb123-(sw151--sw171)	cb107-(sw151--sw171)
261	cb124-(sw171--sw172)	cb123-(sw151--sw172)	cb124-(sw152--sw172)	cb108-(sw152--sw172)
262	cb125-(sw172--sw173)	cb124-(sw172--sw153)	cb125-(sw153--sw173)	cb109-(sw153--sw173)
263	cb126-(sw173--sw174)	cb125-(sw153--sw174)	cb126-(sw154--sw174)	cb110-(sw154--sw174)
264	cb127-(sw174--sw175)	cb126-(sw174--sw155)	cb127-(sw155--sw175)	cb111-(sw155--sw175)
265		cb127-(sw155--sw176)		cb112-(sw160--sw180)
266		cb15-(sw35--sw16)		cb113-(sw161--sw181)
267		cb31-(sw35--sw56)		cb114-(sw162--sw182)

268		cb47-(sw75--sw56)		cb115-(sw163--sw183)
269		cb63-(sw75--sw96)		cb116-(sw164--sw184)
270		cb79-(sw115--sw96)		cb117-(sw165--sw185)
271		cb95-(sw115--sw136)		cb118-(sw166--sw186)
272		cb111-(sw155--sw136)		cb119-(sw167--sw187)
273				cb120-(sw168--sw188)
274				cb121-(sw169--sw189)
275				cb122-(sw170--sw190)
276				cb123-(sw171--sw191)
277				cb124-(sw172--sw192)
278				cb125-(sw173--sw193)
279				cb126-(sw174--sw194)
280				cb127-(sw175--sw195)

Figure A.5 Quarter-Pixel Interpolation and Search Flow for 8x16 Block Size

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw11--sw0)		cb0-(sw11--sw24)	
2	cb1-(sw0--sw13)	cb0-(sw0--sw13)	cb1-(sw24--sw13)	cb0-(sw24--sw13)
3	cb2-(sw13--sw2)	cb1-(sw13--sw2)	cb2-(sw13--sw26)	cb1-(sw13--sw26)
4	cb3-(sw2--sw15)	cb2-(sw2--sw15)	cb3-(sw26--sw15)	cb2-(sw26--sw15)
5	cb4-(sw15--sw4)	cb3-(sw15--sw4)	cb4-(sw15--sw28)	cb3-(sw15--sw28)
6	cb5-(sw4--sw17)	cb4-(sw4--sw17)	cb5-(sw28--sw17)	cb4-(sw28--sw17)
7	cb6-(sw17--sw6)	cb5-(sw17--sw6)	cb6-(sw17--sw30)	cb5-(sw17--sw30)
8	cb7-(sw6--sw19)	cb6-(sw6--sw19)	cb7-(sw30--sw19)	cb6-(sw30--sw19)
9	cb8-(sw11--sw24)		cb8-(sw35--sw24)	
10	cb9-(sw24--sw13)	cb8-(sw24--sw13)	cb9-(sw24--sw37)	cb8-(sw24--sw37)
11	cb10-(sw13--sw26)	cb9-(sw13--sw26)	cb10-(sw37--sw26)	cb9-(sw37--sw26)
12	cb11-(sw26--sw15)	cb10-(sw26--sw15)	cb11-(sw26--sw39)	cb10-(sw26--sw39)
13	cb12-(sw15--sw28)	cb11-(sw15--sw28)	cb12-(sw39--sw28)	cb11-(sw39--sw28)
14	cb13-(sw28--sw17)	cb12-(sw28--sw17)	cb13-(sw28--sw41)	cb12-(sw28--sw41)
15	cb14-(sw17--sw30)	cb13-(sw17--sw30)	cb14-(sw41--sw30)	cb13-(sw41--sw30)
16	cb15-(sw30--sw19)	cb14-(sw30--sw19)	cb15-(sw30--sw43)	cb14-(sw30--sw43)
17	cb16-(sw35--sw24)		cb16-(sw35--sw48)	
18	cb17-(sw24--sw37)	cb16-(sw24--sw37)	cb17-(sw48--sw37)	cb16-(sw48--sw37)
19	cb18-(sw37--sw26)	cb17-(sw37--sw26)	cb18-(sw37--sw50)	cb17-(sw37--sw50)
20	cb19-(sw26--sw39)	cb18-(sw26--sw39)	cb19-(sw50--sw39)	cb18-(sw50--sw39)
21	cb20-(sw39--sw28)	cb19-(sw39--sw28)	cb20-(sw39--sw52)	cb19-(sw39--sw52)
22	cb21-(sw28--sw41)	cb20-(sw28--sw41)	cb21-(sw52--sw41)	cb20-(sw52--sw41)
23	cb22-(sw41--sw30)	cb21-(sw41--sw30)	cb22-(sw41--sw54)	cb21-(sw41--sw54)
24	cb23-(sw30--sw43)	cb22-(sw30--sw43)	cb23-(sw54--sw43)	cb22-(sw54--sw43)
25	cb24-(sw35--sw48)		cb24-(sw59--sw48)	
26	cb25-(sw48--sw37)	cb24-(sw48--sw37)	cb25-(sw48--sw61)	cb24-(sw48--sw61)
27	cb26-(sw37--sw50)	cb25-(sw37--sw50)	cb26-(sw61--sw50)	cb25-(sw61--sw50)
28	cb27-(sw50--sw39)	cb26-(sw50--sw39)	cb27-(sw50--sw63)	cb26-(sw50--sw63)
29	cb28-(sw39--sw52)	cb27-(sw39--sw52)	cb28-(sw63--sw52)	cb27-(sw63--sw52)
30	cb29-(sw52--sw41)	cb28-(sw52--sw41)	cb29-(sw52--sw65)	cb28-(sw52--sw65)
31	cb30-(sw41--sw54)	cb29-(sw41--sw54)	cb30-(sw65--sw54)	cb29-(sw65--sw54)
32	cb31-(sw54--sw43)	cb30-(sw54--sw43)	cb31-(sw54--sw67)	cb30-(sw54--sw67)
33	cb32-(sw59--sw48)		cb32-(sw59--sw72)	
34	cb33-(sw48--sw61)	cb32-(sw48--sw61)	cb33-(sw72--sw61)	cb32-(sw72--sw61)
35	cb34-(sw61--sw50)	cb33-(sw61--sw50)	cb34-(sw61--sw74)	cb33-(sw61--sw74)
36	cb35-(sw50--sw63)	cb34-(sw50--sw63)	cb35-(sw74--sw63)	cb34-(sw74--sw63)
37	cb36-(sw63--sw52)	cb35-(sw63--sw52)	cb36-(sw63--sw76)	cb35-(sw63--sw76)
38	cb37-(sw52--sw65)	cb36-(sw52--sw65)	cb37-(sw76--sw65)	cb36-(sw76--sw65)
39	cb38-(sw65--sw54)	cb37-(sw65--sw54)	cb38-(sw65--sw78)	cb37-(sw65--sw78)
40	cb39-(sw54--sw67)	cb38-(sw54--sw67)	cb39-(sw78--sw67)	cb38-(sw78--sw67)
41	cb40-(sw59--sw72)		cb40-(sw83--sw72)	
42	cb41-(sw72--sw61)	cb40-(sw72--sw61)	cb41-(sw72--sw85)	cb40-(sw72--sw85)
43	cb42-(sw61--sw74)	cb41-(sw61--sw74)	cb42-(sw85--sw74)	cb41-(sw85--sw74)
44	cb43-(sw74--sw63)	cb42-(sw74--sw63)	cb43-(sw74--sw87)	cb42-(sw74--sw87)
45	cb44-(sw63--sw76)	cb43-(sw63--sw76)	cb44-(sw87--sw76)	cb43-(sw87--sw76)
46	cb45-(sw76--sw65)	cb44-(sw76--sw65)	cb45-(sw76--sw89)	cb44-(sw76--sw89)
47	cb46-(sw65--sw78)	cb45-(sw65--sw78)	cb46-(sw89--sw78)	cb45-(sw89--sw78)

48	cb47-(sw78--sw67)	cb46-(sw78--sw67)	cb47-(sw78--sw91)	cb46-(sw78--sw91)
49	cb48-(sw83--sw72)		cb48-(sw83--sw96)	
50	cb49-(sw72--sw85)	cb48-(sw72--sw85)	cb49-(sw96--sw85)	cb48-(sw96--sw85)
51	cb50-(sw85--sw74)	cb49-(sw85--sw74)	cb50-(sw85--sw98)	cb49-(sw85--sw98)
52	cb51-(sw74--sw87)	cb50-(sw74--sw87)	cb51-(sw98--sw87)	cb50-(sw98--sw87)
53	cb52-(sw87--sw76)	cb51-(sw87--sw76)	cb52-(sw87--sw100)	cb51-(sw87--sw100)
54	cb53-(sw76--sw89)	cb52-(sw76--sw89)	cb53-(sw100--sw89)	cb52-(sw100--sw89)
55	cb54-(sw89--sw78)	cb53-(sw89--sw78)	cb54-(sw89--sw102)	cb53-(sw89--sw102)
56	cb55-(sw78--sw91)	cb54-(sw78--sw91)	cb55-(sw102--sw91)	cb54-(sw102--sw91)
57	cb56-(sw83--sw96)		cb56-(sw107--sw96)	
58	cb57-(sw96--sw85)	cb56-(sw96--sw85)	cb57-(sw96--sw109)	cb56-(sw96--sw109)
59	cb58-(sw85--sw98)	cb57-(sw85--sw98)	cb58-(sw109--sw98)	cb57-(sw109--sw98)
60	cb59-(sw98--sw87)	cb58-(sw98--sw87)	cb59-(sw98--sw111)	cb58-(sw98--sw111)
61	cb60-(sw87--sw100)	cb59-(sw87--sw100)	cb60-(sw111--sw100)	cb59-(sw111--sw100)
62	cb61-(sw100--sw89)	cb60-(sw100--sw89)	cb61-(sw100--sw113)	cb60-(sw100--sw113)
63	cb62-(sw89--sw102)	cb61-(sw89--sw102)	cb62-(sw113--sw102)	cb61-(sw113--sw102)
64	cb63-(sw102--sw91)	cb62-(sw102--sw91)	cb63-(sw102--sw115)	cb62-(sw102--sw115)
65	cb64-(sw107--sw96)		cb64-(sw107--sw120)	
66	cb65-(sw96--sw109)	cb64-(sw96--sw109)	cb65-(sw120--sw109)	cb64-(sw120--sw109)
67	cb66-(sw109--sw98)	cb65-(sw109--sw98)	cb66-(sw109--sw122)	cb65-(sw109--sw122)
68	cb67-(sw98--sw111)	cb66-(sw98--sw111)	cb67-(sw122--sw111)	cb66-(sw122--sw111)
69	cb68-(sw111--sw100)	cb67-(sw111--sw100)	cb68-(sw111--sw124)	cb67-(sw111--sw124)
70	cb69-(sw100--sw113)	cb68-(sw100--sw113)	cb69-(sw124--sw113)	cb68-(sw124--sw113)
71	cb70-(sw113--sw102)	cb69-(sw113--sw102)	cb70-(sw113--sw126)	cb69-(sw113--sw126)
72	cb71-(sw102--sw115)	cb70-(sw102--sw115)	cb71-(sw126--sw115)	cb70-(sw126--sw115)
73	cb72-(sw107--sw120)		cb72-(sw131--sw120)	
74	cb73-(sw120--sw109)	cb72-(sw120--sw109)	cb73-(sw120--sw133)	cb72-(sw120--sw133)
75	cb74-(sw109--sw122)	cb73-(sw109--sw122)	cb74-(sw133--sw122)	cb73-(sw133--sw122)
76	cb75-(sw122--sw111)	cb74-(sw122--sw111)	cb75-(sw122--sw135)	cb74-(sw122--sw135)
77	cb76-(sw111--sw124)	cb75-(sw111--sw124)	cb76-(sw135--sw124)	cb75-(sw135--sw124)
78	cb77-(sw124--sw113)	cb76-(sw124--sw113)	cb77-(sw124--sw137)	cb76-(sw124--sw137)
79	cb78-(sw113--sw126)	cb77-(sw113--sw126)	cb78-(sw137--sw126)	cb77-(sw137--sw126)
80	cb79-(sw126--sw115)	cb78-(sw126--sw115)	cb79-(sw126--sw139)	cb78-(sw126--sw139)
81	cb80-(sw131--sw120)		cb80-(sw131--sw144)	
82	cb81-(sw120--sw133)	cb80-(sw120--sw133)	cb81-(sw144--sw133)	cb80-(sw144--sw133)
83	cb82-(sw133--sw122)	cb81-(sw133--sw122)	cb82-(sw133--sw146)	cb81-(sw133--sw146)
84	cb83-(sw122--sw135)	cb82-(sw122--sw135)	cb83-(sw146--sw135)	cb82-(sw146--sw135)
85	cb84-(sw135--sw124)	cb83-(sw135--sw124)	cb84-(sw135--sw148)	cb83-(sw135--sw148)
86	cb85-(sw124--sw137)	cb84-(sw124--sw137)	cb85-(sw148--sw137)	cb84-(sw148--sw137)
87	cb86-(sw137--sw126)	cb85-(sw137--sw126)	cb86-(sw137--sw150)	cb85-(sw137--sw150)
88	cb87-(sw126--sw139)	cb86-(sw126--sw139)	cb87-(sw150--sw139)	cb86-(sw150--sw139)
89	cb88-(sw131--sw144)		cb88-(sw155--sw144)	
90	cb89-(sw144--sw133)	cb88-(sw144--sw133)	cb89-(sw144--sw157)	cb88-(sw144--sw157)
91	cb90-(sw133--sw146)	cb89-(sw133--sw146)	cb90-(sw157--sw146)	cb89-(sw157--sw146)
92	cb91-(sw146--sw135)	cb90-(sw146--sw135)	cb91-(sw146--sw159)	cb90-(sw146--sw159)
93	cb92-(sw135--sw148)	cb91-(sw135--sw148)	cb92-(sw159--sw148)	cb91-(sw159--sw148)
94	cb93-(sw148--sw137)	cb92-(sw148--sw137)	cb93-(sw148--sw161)	cb92-(sw148--sw161)
95	cb94-(sw137--sw150)	cb93-(sw137--sw150)	cb94-(sw161--sw150)	cb93-(sw161--sw150)
96	cb95-(sw150--sw139)	cb94-(sw150--sw139)	cb95-(sw150--sw163)	cb94-(sw150--sw163)

97	cb96-(sw155--sw144)		cb96-(sw155--sw168)	
98	cb97-(sw144--sw157)	cb96-(sw144--sw157)	cb97-(sw168--sw157)	cb96-(sw168--sw157)
99	cb98-(sw157--sw146)	cb97-(sw157--sw146)	cb98-(sw157--sw170)	cb97-(sw157--sw170)
100	cb99-(sw146--sw159)	cb98-(sw146--sw159)	cb99-(sw170--sw159)	cb98-(sw170--sw159)
101	cb100-(sw159--sw148)	cb99-(sw159--sw148)	cb100-(sw159--sw172)	cb99-(sw159--sw172)
102	cb101-(sw148--sw161)	cb100-(sw148--sw161)	cb101-(sw172--sw161)	cb100-(sw172--sw161)
103	cb102-(sw161--sw150)	cb101-(sw161--sw150)	cb102-(sw161--sw174)	cb101-(sw161--sw174)
104	cb103-(sw150--sw163)	cb102-(sw150--sw163)	cb103-(sw174--sw163)	cb102-(sw174--sw163)
105	cb104-(sw155--sw168)		cb104-(sw179--sw168)	
106	cb105-(sw168--sw157)	cb104-(sw168--sw157)	cb105-(sw168--sw181)	cb104-(sw168--sw181)
107	cb106-(sw157--sw170)	cb105-(sw157--sw170)	cb106-(sw181--sw170)	cb105-(sw181--sw170)
108	cb107-(sw170--sw159)	cb106-(sw170--sw159)	cb107-(sw170--sw183)	cb106-(sw170--sw183)
109	cb108-(sw159--sw172)	cb107-(sw159--sw172)	cb108-(sw183--sw172)	cb107-(sw183--sw172)
110	cb109-(sw172--sw161)	cb108-(sw172--sw161)	cb109-(sw172--sw185)	cb108-(sw172--sw185)
111	cb110-(sw161--sw174)	cb109-(sw161--sw174)	cb110-(sw185--sw174)	cb109-(sw185--sw174)
112	cb111-(sw174--sw163)	cb110-(sw174--sw163)	cb111-(sw174--sw187)	cb110-(sw174--sw187)
113	cb112-(sw179--sw168)		cb112-(sw179--sw192)	
114	cb113-(sw168--sw181)	cb112-(sw168--sw181)	cb113-(sw192--sw181)	cb112-(sw192--sw181)
115	cb114-(sw181--sw170)	cb113-(sw181--sw170)	cb114-(sw181--sw194)	cb113-(sw181--sw194)
116	cb115-(sw170--sw183)	cb114-(sw170--sw183)	cb115-(sw194--sw183)	cb114-(sw194--sw183)
117	cb116-(sw183--sw172)	cb115-(sw183--sw172)	cb116-(sw183--sw196)	cb115-(sw183--sw196)
118	cb117-(sw172--sw185)	cb116-(sw172--sw185)	cb117-(sw196--sw185)	cb116-(sw196--sw185)
119	cb118-(sw185--sw174)	cb117-(sw185--sw174)	cb118-(sw185--sw198)	cb117-(sw185--sw198)
120	cb119-(sw174--sw187)	cb118-(sw174--sw187)	cb119-(sw198--sw187)	cb118-(sw198--sw187)
121	cb120-(sw179--sw192)		cb120-(sw203--sw192)	
122	cb121-(sw192--sw181)	cb120-(sw192--sw181)	cb121-(sw192--sw205)	cb120-(sw192--sw205)
123	cb122-(sw181--sw194)	cb121-(sw181--sw194)	cb122-(sw205--sw194)	cb121-(sw205--sw194)
124	cb123-(sw194--sw183)	cb122-(sw194--sw183)	cb123-(sw194--sw207)	cb122-(sw194--sw207)
125	cb124-(sw183--sw196)	cb123-(sw183--sw196)	cb124-(sw207--sw196)	cb123-(sw207--sw196)
126	cb125-(sw196--sw185)	cb124-(sw196--sw185)	cb125-(sw196--sw209)	cb124-(sw196--sw209)
127	cb126-(sw185--sw198)	cb125-(sw185--sw198)	cb126-(sw209--sw198)	cb125-(sw209--sw198)
128	cb127-(sw198--sw187)	cb126-(sw198--sw187)	cb127-(sw198--sw211)	cb126-(sw198--sw211)
129		cb127-(sw187--sw200)		cb127-(sw211--sw200)
130		cb7-(sw19--sw8)		cb7-(sw19--sw32)
131		cb15-(sw19--sw32)		cb15-(sw43--sw32)
132		cb23-(sw43--sw32)		cb23-(sw43--sw56)
133		cb31-(sw43--sw56)		cb31-(sw67--sw56)
134		cb39-(sw67--sw56)		cb39-(sw67--sw80)
135		cb47-(sw67--sw80)		cb47-(sw91--sw80)
136		cb55-(sw91--sw80)		cb55-(sw91--sw104)
137		cb63-(sw91--sw104)		cb63-(sw115--sw104)
138		cb71-(sw115--sw104)		cb71-(sw115--sw128)
139		cb79-(sw115--sw128)		cb79-(sw139--sw128)
140		cb87-(sw139--sw128)		cb87-(sw139--sw152)
141		cb95-(sw139--sw152)		cb95-(sw163--sw152)
142		cb103-(sw163--sw152)		cb103-(sw163--sw176)
143		cb111-(sw163--sw176)		cb111-(sw187--sw176)
144		cb119-(sw187--sw176)		cb119-(sw187--sw200)
145	cb0-(sw11--sw12)		cb0-(sw0--sw12)	

146	cb1-(sw12--sw13)	cb0-(sw12--sw13)	cb1-(sw1--sw13)	
147	cb2-(sw13--sw14)	cb1-(sw13--sw14)	cb2-(sw2--sw14)	
148	cb3-(sw14--sw14)	cb2-(sw14--sw15)	cb3-(sw3--sw15)	
149	cb4-(sw15--sw16)	cb3-(sw14--sw15)	cb4-(sw4--sw16)	
150	cb5-(sw16--sw16)	cb4-(sw16--sw17)	cb5-(sw5--sw17)	
151	cb6-(sw17--sw18)	cb5-(sw16--sw17)	cb6-(sw6--sw18)	
152	cb7-(sw18--sw19)	cb6-(sw18--sw19)	cb7-(sw7--sw19)	
153	cb8-(sw23--sw24)		cb8-(sw12--sw24)	cb0-(sw12--sw24)
154	cb9-(sw24--sw25)	cb8-(sw24--sw25)	cb9-(sw13--sw25)	cb1-(sw13--sw25)
155	cb10-(sw25--sw26)	cb9-(sw25--sw26)	cb10-(sw14--sw26)	cb2-(sw14--sw26)
156	cb11-(sw26--sw26)	cb10-(sw26--sw27)	cb11-(sw15--sw27)	cb3-(sw15--sw27)
157	cb12-(sw27--sw28)	cb11-(sw26--sw27)	cb12-(sw16--sw28)	cb4-(sw16--sw28)
158	cb13-(sw28--sw28)	cb12-(sw28--sw29)	cb13-(sw17--sw29)	cb5-(sw17--sw29)
159	cb14-(sw29--sw30)	cb13-(sw28--sw29)	cb14-(sw18--sw30)	cb6-(sw18--sw30)
160	cb15-(sw30--sw31)	cb14-(sw30--sw31)	cb15-(sw19--sw31)	cb7-(sw19--sw31)
161	cb16-(sw35--sw36)		cb16-(sw24--sw36)	cb8-(sw24--sw36)
162	cb17-(sw36--sw37)	cb16-(sw36--sw37)	cb17-(sw25--sw37)	cb9-(sw25--sw37)
163	cb18-(sw37--sw38)	cb17-(sw37--sw38)	cb18-(sw26--sw38)	cb10-(sw26--sw38)
164	cb19-(sw38--sw39)	cb18-(sw38--sw39)	cb19-(sw27--sw39)	cb11-(sw27--sw39)
165	cb20-(sw39--sw40)	cb19-(sw39--sw40)	cb20-(sw28--sw40)	cb12-(sw28--sw40)
166	cb21-(sw40--sw41)	cb20-(sw40--sw41)	cb21-(sw29--sw41)	cb13-(sw29--sw41)
167	cb22-(sw41--sw42)	cb21-(sw41--sw42)	cb22-(sw30--sw42)	cb14-(sw30--sw42)
168	cb23-(sw42--sw43)	cb22-(sw42--sw43)	cb23-(sw31--sw43)	cb15-(sw31--sw43)
169	cb24-(sw47--sw48)		cb24-(sw36--sw48)	cb16-(sw36--sw48)
170	cb25-(sw48--sw49)	cb24-(sw48--sw49)	cb25-(sw37--sw49)	cb17-(sw37--sw49)
171	cb26-(sw49--sw50)	cb25-(sw49--sw50)	cb26-(sw38--sw50)	cb18-(sw38--sw50)
172	cb27-(sw50--sw50)	cb26-(sw50--sw51)	cb27-(sw39--sw51)	cb19-(sw39--sw51)
173	cb28-(sw51--sw52)	cb27-(sw50--sw51)	cb28-(sw40--sw52)	cb20-(sw40--sw52)
174	cb29-(sw52--sw52)	cb28-(sw52--sw53)	cb29-(sw41--sw53)	cb21-(sw41--sw53)
175	cb30-(sw53--sw54)	cb29-(sw52--sw53)	cb30-(sw42--sw54)	cb22-(sw42--sw54)
176	cb31-(sw54--sw55)	cb30-(sw54--sw55)	cb31-(sw43--sw55)	cb23-(sw43--sw55)
177	cb32-(sw59--sw60)		cb32-(sw48--sw60)	cb24-(sw48--sw60)
178	cb33-(sw60--sw61)	cb32-(sw60--sw61)	cb33-(sw49--sw61)	cb25-(sw49--sw61)
179	cb34-(sw61--sw62)	cb33-(sw61--sw62)	cb34-(sw50--sw62)	cb26-(sw50--sw62)
180	cb35-(sw62--sw62)	cb34-(sw62--sw63)	cb35-(sw51--sw63)	cb27-(sw51--sw63)
181	cb36-(sw63--sw64)	cb35-(sw63--sw64)	cb36-(sw52--sw64)	cb28-(sw52--sw64)
182	cb37-(sw64--sw64)	cb36-(sw64--sw65)	cb37-(sw53--sw65)	cb29-(sw53--sw65)
183	cb38-(sw65--sw66)	cb37-(sw65--sw66)	cb38-(sw54--sw66)	cb30-(sw54--sw66)
184	cb39-(sw66--sw67)	cb38-(sw66--sw67)	cb39-(sw55--sw67)	cb31-(sw55--sw67)
185	cb40-(sw71--sw72)		cb40-(sw60--sw72)	cb32-(sw60--sw72)
186	cb41-(sw72--sw73)	cb40-(sw72--sw73)	cb41-(sw61--sw73)	cb33-(sw61--sw73)
187	cb42-(sw73--sw74)	cb41-(sw73--sw74)	cb42-(sw62--sw74)	cb34-(sw62--sw74)
188	cb43-(sw74--sw74)	cb42-(sw74--sw75)	cb43-(sw63--sw75)	cb35-(sw63--sw75)
189	cb44-(sw75--sw76)	cb43-(sw74--sw75)	cb44-(sw64--sw76)	cb36-(sw64--sw76)
190	cb45-(sw76--sw76)	cb44-(sw76--sw77)	cb45-(sw65--sw77)	cb37-(sw65--sw77)
191	cb46-(sw77--sw78)	cb45-(sw76--sw77)	cb46-(sw66--sw78)	cb38-(sw66--sw78)
192	cb47-(sw78--sw79)	cb46-(sw78--sw79)	cb47-(sw67--sw79)	cb39-(sw67--sw79)
193	cb48-(sw83--sw84)		cb48-(sw72--sw84)	cb40-(sw72--sw84)
194	cb49-(sw84--sw85)	cb48-(sw84--sw85)	cb49-(sw73--sw85)	cb41-(sw73--sw85)

195	cb50-(sw85--sw86)	cb49-(sw85--sw86)	cb50-(sw74--sw86)	cb42-(sw74--sw86)
196	cb51-(sw86--sw86)	cb50-(sw86--sw87)	cb51-(sw75--sw87)	cb43-(sw75--sw87)
197	cb52-(sw87--sw88)	cb51-(sw86--sw87)	cb52-(sw76--sw88)	cb44-(sw76--sw88)
198	cb53-(sw88--sw88)	cb52-(sw88--sw89)	cb53-(sw77--sw89)	cb45-(sw77--sw89)
199	cb54-(sw89--sw90)	cb53-(sw88--sw89)	cb54-(sw78--sw90)	cb46-(sw78--sw90)
200	cb55-(sw90--sw91)	cb54-(sw90--sw91)	cb55-(sw79--sw91)	cb47-(sw79--sw91)
201	cb56-(sw95--sw96)		cb56-(sw84--sw96)	cb48-(sw84--sw96)
202	cb57-(sw96--sw97)	cb56-(sw96--sw97)	cb57-(sw85--sw97)	cb49-(sw85--sw97)
203	cb58-(sw97--sw98)	cb57-(sw97--sw98)	cb58-(sw86--sw98)	cb50-(sw86--sw98)
204	cb59-(sw98--sw98)	cb58-(sw98--sw99)	cb59-(sw87--sw99)	cb51-(sw87--sw99)
205	cb60-(sw99--sw100)	cb59-(sw99--sw100)	cb60-(sw88--sw100)	cb52-(sw88--sw100)
206	cb61-(sw100--sw101)	cb60-(sw100--sw101)	cb61-(sw89--sw101)	cb53-(sw89--sw101)
207	cb62-(sw101--sw102)	cb61-(sw101--sw102)	cb62-(sw90--sw102)	cb54-(sw90--sw102)
208	cb63-(sw102--sw103)	cb62-(sw102--sw103)	cb63-(sw91--sw103)	cb55-(sw91--sw103)
209	cb64-(sw107--sw108)		cb64-(sw96--sw108)	cb56-(sw96--sw108)
210	cb65-(sw108--sw109)	cb64-(sw108--sw109)	cb65-(sw97--sw109)	cb57-(sw97--sw109)
211	cb66-(sw109--sw110)	cb65-(sw109--sw110)	cb66-(sw98--sw110)	cb58-(sw98--sw110)
212	cb67-(sw110--sw110)	cb66-(sw110--sw111)	cb67-(sw99--sw111)	cb59-(sw99--sw111)
213	cb68-(sw111--sw112)	cb67-(sw111--sw112)	cb68-(sw100--sw112)	cb60-(sw100--sw112)
214	cb69-(sw112--sw112)	cb68-(sw112--sw113)	cb69-(sw101--sw113)	cb61-(sw101--sw113)
215	cb70-(sw113--sw114)	cb69-(sw113--sw114)	cb70-(sw102--sw114)	cb62-(sw102--sw114)
216	cb71-(sw114--sw115)	cb70-(sw114--sw115)	cb71-(sw103--sw115)	cb63-(sw103--sw115)
217	cb72-(sw119--sw120)		cb72-(sw108--sw120)	cb64-(sw108--sw120)
218	cb73-(sw120--sw121)	cb72-(sw120--sw121)	cb73-(sw109--sw121)	cb65-(sw109--sw121)
219	cb74-(sw121--sw122)	cb73-(sw121--sw122)	cb74-(sw110--sw122)	cb66-(sw110--sw122)
220	cb75-(sw122--sw123)	cb74-(sw122--sw123)	cb75-(sw111--sw123)	cb67-(sw111--sw123)
221	cb76-(sw123--sw124)	cb75-(sw123--sw124)	cb76-(sw112--sw124)	cb68-(sw112--sw124)
222	cb77-(sw124--sw125)	cb76-(sw124--sw125)	cb77-(sw113--sw125)	cb69-(sw113--sw125)
223	cb78-(sw125--sw126)	cb77-(sw125--sw126)	cb78-(sw114--sw126)	cb70-(sw114--sw126)
224	cb79-(sw126--sw127)	cb78-(sw126--sw127)	cb79-(sw115--sw127)	cb71-(sw115--sw127)
225	cb80-(sw131--sw132)		cb80-(sw120--sw132)	cb72-(sw120--sw132)
226	cb81-(sw132--sw133)	cb80-(sw132--sw133)	cb81-(sw121--sw133)	cb73-(sw121--sw133)
227	cb82-(sw133--sw134)	cb81-(sw133--sw134)	cb82-(sw122--sw134)	cb74-(sw122--sw134)
228	cb83-(sw134--sw135)	cb82-(sw134--sw135)	cb83-(sw123--sw135)	cb75-(sw123--sw135)
229	cb84-(sw135--sw136)	cb83-(sw135--sw136)	cb84-(sw124--sw136)	cb76-(sw124--sw136)
230	cb85-(sw136--sw137)	cb84-(sw136--sw137)	cb85-(sw125--sw137)	cb77-(sw125--sw137)
231	cb86-(sw137--sw138)	cb85-(sw137--sw138)	cb86-(sw126--sw138)	cb78-(sw126--sw138)
232	cb87-(sw138--sw139)	cb86-(sw138--sw139)	cb87-(sw127--sw139)	cb79-(sw127--sw139)
233	cb88-(sw143--sw144)		cb88-(sw132--sw144)	cb80-(sw132--sw144)
234	cb89-(sw144--sw145)	cb88-(sw144--sw145)	cb89-(sw133--sw145)	cb81-(sw133--sw145)
235	cb90-(sw145--sw146)	cb89-(sw145--sw146)	cb90-(sw134--sw146)	cb82-(sw134--sw146)
236	cb91-(sw146--sw147)	cb90-(sw146--sw147)	cb91-(sw135--sw147)	cb83-(sw135--sw147)
237	cb92-(sw147--sw148)	cb91-(sw147--sw148)	cb92-(sw136--sw148)	cb84-(sw136--sw148)
238	cb93-(sw148--sw149)	cb92-(sw148--sw149)	cb93-(sw137--sw149)	cb85-(sw137--sw149)
239	cb94-(sw149--sw150)	cb93-(sw149--sw150)	cb94-(sw138--sw150)	cb86-(sw138--sw150)
240	cb95-(sw150--sw151)	cb94-(sw150--sw151)	cb95-(sw139--sw151)	cb87-(sw139--sw151)
241	cb96-(sw155--sw156)		cb96-(sw144--sw156)	cb88-(sw144--sw156)
242	cb97-(sw156--sw157)	cb96-(sw156--sw157)	cb97-(sw145--sw157)	cb89-(sw145--sw157)
243	cb98-(sw157--sw158)	cb97-(sw157--sw158)	cb98-(sw146--sw158)	cb90-(sw146--sw158)

244	cb99-(sw158--sw159)	cb98-(sw158--sw159)	cb99-(sw147--sw159)	cb91-(sw147--sw159)
245	cb100-(sw159--sw160)	cb99-(sw159--sw160)	cb100-(sw148--sw160)	cb92-(sw148--sw160)
246	cb101-(sw160--sw161)	cb100-(sw160--sw161)	cb101-(sw149--sw161)	cb93-(sw149--sw161)
247	cb102-(sw161--sw162)	cb101-(sw161--sw162)	cb102-(sw150--sw162)	cb94-(sw150--sw162)
248	cb103-(sw162--sw163)	cb102-(sw162--sw163)	cb103-(sw151--sw163)	cb95-(sw151--sw163)
249	cb104-(sw167--sw168)		cb104-(sw156--sw168)	cb96-(sw156--sw168)
250	cb105-(sw168--sw169)	cb104-(sw168--sw169)	cb105-(sw157--sw169)	cb97-(sw157--sw169)
251	cb106-(sw169--sw170)	cb105-(sw169--sw170)	cb106-(sw158--sw170)	cb98-(sw158--sw170)
252	cb107-(sw170--sw171)	cb106-(sw170--sw171)	cb107-(sw159--sw171)	cb99-(sw159--sw171)
253	cb108-(sw171--sw172)	cb107-(sw171--sw172)	cb108-(sw160--sw172)	cb100-(sw160--sw172)
254	cb109-(sw172--sw173)	cb108-(sw172--sw173)	cb109-(sw161--sw173)	cb101-(sw161--sw173)
255	cb110-(sw173--sw174)	cb109-(sw173--sw174)	cb110-(sw162--sw174)	cb102-(sw162--sw174)
256	cb111-(sw174--sw175)	cb110-(sw174--sw175)	cb111-(sw163--sw175)	cb103-(sw163--sw175)
257	cb112-(sw179--sw180)		cb112-(sw168--sw180)	cb104-(sw168--sw180)
258	cb113-(sw180--sw181)	cb112-(sw180--sw181)	cb113-(sw169--sw181)	cb105-(sw169--sw181)
259	cb114-(sw181--sw182)	cb113-(sw181--sw182)	cb114-(sw170--sw182)	cb106-(sw170--sw182)
260	cb115-(sw182--sw182)	cb114-(sw182--sw183)	cb115-(sw171--sw183)	cb107-(sw171--sw183)
261	cb116-(sw183--sw184)	cb115-(sw183--sw184)	cb116-(sw172--sw184)	cb108-(sw172--sw184)
262	cb117-(sw184--sw185)	cb116-(sw184--sw185)	cb117-(sw173--sw185)	cb109-(sw173--sw185)
263	cb118-(sw185--sw186)	cb117-(sw185--sw186)	cb118-(sw174--sw186)	cb110-(sw174--sw186)
264	cb119-(sw186--sw187)	cb118-(sw186--sw187)	cb119-(sw175--sw187)	cb111-(sw175--sw187)
265	cb120-(sw191--sw192)		cb120-(sw180--sw192)	cb112-(sw180--sw192)
266	cb121-(sw192--sw193)	cb120-(sw192--sw193)	cb121-(sw181--sw193)	cb113-(sw181--sw193)
267	cb122-(sw193--sw194)	cb121-(sw193--sw194)	cb122-(sw182--sw194)	cb114-(sw182--sw194)
268	cb123-(sw194--sw195)	cb122-(sw194--sw195)	cb123-(sw183--sw195)	cb115-(sw183--sw195)
269	cb124-(sw195--sw196)	cb123-(sw195--sw196)	cb124-(sw184--sw196)	cb116-(sw184--sw196)
270	cb125-(sw196--sw197)	cb124-(sw196--sw197)	cb125-(sw185--sw197)	cb117-(sw185--sw197)
271	cb126-(sw197--sw198)	cb125-(sw197--sw198)	cb126-(sw186--sw198)	cb118-(sw186--sw198)
272	cb127-(sw198--sw199)	cb126-(sw198--sw199)	cb127-(sw187--sw199)	cb119-(sw187--sw199)
273		cb127-(sw199--sw200)		cb120-(sw192--sw204)
274		cb7-(sw19--sw20)		cb121-(sw193--sw205)
275		cb15-(sw31--sw32)		cb122-(sw194--sw206)
276		cb23-(sw43--sw44)		cb123-(sw195--sw207)
277		cb31-(sw55--sw56)		cb124-(sw196--sw208)
278		cb39-(sw67--sw68)		cb125-(sw197--sw209)
279		cb47-(sw79--sw80)		cb126-(sw198--sw210)
280		cb55-(sw91--sw92)		cb127-(sw199--sw211)
281		cb63-(sw103--sw104)		
282		cb71-(sw115--sw116)		
283		cb79-(sw127--sw128)		
284		cb87-(sw139--sw140)		
285		cb95-(sw151--sw152)		
286		cb103-(sw163--sw164)		
287		cb111-(sw175--sw176)		
288		cb119-(sw187--sw188)		

Figure A.6 Quarter-Pixel Interpolation and Search Flow for 16x8 Block Size

clock cycle	PE0	PE1	PE2	PE3
1	cb0-(sw19--sw0)		cb0-(sw19--sw40)	
2	cb1-(sw0--sw21)	cb0-(sw0--sw21)	cb1-(sw40--sw21)	cb0-(sw40--sw21)
3	cb2-(sw21--sw2)	cb1-(sw21--sw2)	cb2-(sw21--sw42)	cb1-(sw21--sw42)
4	cb3-(sw2--sw23)	cb2-(sw2--sw23)	cb3-(sw42--sw23)	cb2-(sw42--sw23)
5	cb4-(sw23--sw4)	cb3-(sw23--sw4)	cb4-(sw23--sw44)	cb3-(sw23--sw44)
6	cb5-(sw4--sw25)	cb4-(sw4--sw25)	cb5-(sw44--sw25)	cb4-(sw44--sw25)
7	cb6-(sw25--sw6)	cb5-(sw25--sw6)	cb6-(sw25--sw46)	cb5-(sw25--sw46)
8	cb7-(sw6--sw27)	cb6-(sw6--sw27)	cb7-(sw46--sw27)	cb6-(sw46--sw27)
9	cb8-(sw27--sw8)	cb7-(sw27--sw8)	cb8-(sw27--sw48)	cb7-(sw27--sw48)
10	cb9-(sw8--sw29)	cb8-(sw8--sw29)	cb9-(sw48--sw29)	cb8-(sw48--sw29)
11	cb10-(sw29--sw10)	cb9-(sw29--sw10)	cb10-(sw29--sw50)	cb9-(sw29--sw50)
12	cb11-(sw10--sw31)	cb10-(sw10--sw31)	cb11-(sw50--sw31)	cb10-(sw50--sw31)
13	cb12-(sw31--sw12)	cb11-(sw31--sw12)	cb12-(sw31--sw52)	cb11-(sw31--sw52)
14	cb13-(sw12--sw33)	cb12-(sw12--sw33)	cb13-(sw52--sw33)	cb12-(sw52--sw33)
15	cb14-(sw33--sw14)	cb13-(sw33--sw14)	cb14-(sw33--sw54)	cb13-(sw33--sw54)
16	cb15-(sw14--sw35)	cb14-(sw14--sw35)	cb15-(sw54--sw35)	cb14-(sw54--sw35)
17	cb16-(sw19--sw40)		cb16-(sw59--sw40)	
18	cb17-(sw40--sw21)	cb16-(sw40--sw21)	cb17-(sw40--sw61)	cb16-(sw40--sw61)
19	cb18-(sw21--sw42)	cb17-(sw21--sw42)	cb18-(sw61--sw42)	cb17-(sw61--sw42)
20	cb19-(sw42--sw23)	cb18-(sw42--sw23)	cb19-(sw42--sw63)	cb18-(sw42--sw63)
21	cb20-(sw23--sw44)	cb19-(sw23--sw44)	cb20-(sw63--sw44)	cb19-(sw63--sw44)
22	cb21-(sw44--sw25)	cb20-(sw44--sw25)	cb21-(sw44--sw65)	cb20-(sw44--sw65)
23	cb22-(sw25--sw46)	cb21-(sw25--sw46)	cb22-(sw65--sw46)	cb21-(sw65--sw46)
24	cb23-(sw46--sw27)	cb22-(sw46--sw27)	cb23-(sw46--sw67)	cb22-(sw46--sw67)
25	cb24-(sw27--sw48)	cb23-(sw27--sw48)	cb24-(sw67--sw48)	cb23-(sw67--sw48)
26	cb25-(sw48--sw29)	cb24-(sw48--sw29)	cb25-(sw48--sw69)	cb24-(sw48--sw69)
27	cb26-(sw29--sw50)	cb25-(sw29--sw50)	cb26-(sw69--sw50)	cb25-(sw69--sw50)
28	cb27-(sw50--sw31)	cb26-(sw50--sw31)	cb27-(sw50--sw71)	cb26-(sw50--sw71)
29	cb28-(sw31--sw52)	cb27-(sw31--sw52)	cb28-(sw71--sw52)	cb27-(sw71--sw52)
30	cb29-(sw52--sw33)	cb28-(sw52--sw33)	cb29-(sw52--sw73)	cb28-(sw52--sw73)
31	cb30-(sw33--sw54)	cb29-(sw33--sw54)	cb30-(sw73--sw54)	cb29-(sw73--sw54)
32	cb31-(sw54--sw35)	cb30-(sw54--sw35)	cb31-(sw54--sw75)	cb30-(sw54--sw75)
33	cb32-(sw59--sw40)		cb32-(sw59--sw80)	
34	cb33-(sw40--sw61)	cb32-(sw40--sw61)	cb33-(sw80--sw61)	cb32-(sw80--sw61)
35	cb34-(sw61--sw42)	cb33-(sw61--sw42)	cb34-(sw61--sw82)	cb33-(sw61--sw82)
36	cb35-(sw42--sw63)	cb34-(sw42--sw63)	cb35-(sw82--sw63)	cb34-(sw82--sw63)
37	cb36-(sw63--sw44)	cb35-(sw63--sw44)	cb36-(sw63--sw84)	cb35-(sw63--sw84)
38	cb37-(sw44--sw65)	cb36-(sw44--sw65)	cb37-(sw84--sw65)	cb36-(sw84--sw65)
39	cb38-(sw65--sw46)	cb37-(sw65--sw46)	cb38-(sw65--sw86)	cb37-(sw65--sw86)
40	cb39-(sw46--sw67)	cb38-(sw46--sw67)	cb39-(sw86--sw67)	cb38-(sw86--sw67)
41	cb40-(sw67--sw48)	cb39-(sw67--sw48)	cb40-(sw67--sw88)	cb39-(sw67--sw88)
42	cb41-(sw48--sw69)	cb40-(sw48--sw69)	cb41-(sw88--sw69)	cb40-(sw88--sw69)
43	cb42-(sw69--sw50)	cb41-(sw69--sw50)	cb42-(sw69--sw90)	cb41-(sw69--sw90)
44	cb43-(sw50--sw71)	cb42-(sw50--sw71)	cb43-(sw90--sw71)	cb42-(sw90--sw71)
45	cb44-(sw71--sw52)	cb43-(sw71--sw52)	cb44-(sw71--sw92)	cb43-(sw71--sw92)
46	cb45-(sw52--sw73)	cb44-(sw52--sw73)	cb45-(sw92--sw73)	cb44-(sw92--sw73)
47	cb46-(sw73--sw54)	cb45-(sw73--sw54)	cb46-(sw73--sw94)	cb45-(sw73--sw94)

48	cb47-(sw54--sw75)	cb46-(sw54--sw75)	cb47-(sw94--sw75)	cb46-(sw94--sw75)
49	cb48-(sw59--sw80)		cb48-(sw99--sw80)	
50	cb49-(sw80--sw61)	cb48-(sw80--sw61)	cb49-(sw80--sw101)	cb48-(sw80--sw101)
51	cb50-(sw61--sw82)	cb49-(sw61--sw82)	cb50-(sw101--sw82)	cb49-(sw101--sw82)
52	cb51-(sw82--sw63)	cb50-(sw82--sw63)	cb51-(sw82--sw103)	cb50-(sw82--sw103)
53	cb52-(sw63--sw84)	cb51-(sw63--sw84)	cb52-(sw103--sw84)	cb51-(sw103--sw84)
54	cb53-(sw84--sw65)	cb52-(sw84--sw65)	cb53-(sw84--sw105)	cb52-(sw84--sw105)
55	cb54-(sw65--sw86)	cb53-(sw65--sw86)	cb54-(sw105--sw86)	cb53-(sw105--sw86)
56	cb55-(sw86--sw67)	cb54-(sw86--sw67)	cb55-(sw86--sw107)	cb54-(sw86--sw107)
57	cb56-(sw67--sw88)	cb55-(sw67--sw88)	cb56-(sw107--sw88)	cb55-(sw107--sw88)
58	cb57-(sw88--sw69)	cb56-(sw88--sw69)	cb57-(sw88--sw109)	cb56-(sw88--sw109)
59	cb58-(sw69--sw90)	cb57-(sw69--sw90)	cb58-(sw109--sw90)	cb57-(sw109--sw90)
60	cb59-(sw90--sw71)	cb58-(sw90--sw71)	cb59-(sw90--sw111)	cb58-(sw90--sw111)
61	cb60-(sw71--sw92)	cb59-(sw71--sw92)	cb60-(sw111--sw92)	cb59-(sw111--sw92)
62	cb61-(sw92--sw73)	cb60-(sw92--sw73)	cb61-(sw92--sw113)	cb60-(sw92--sw113)
63	cb62-(sw73--sw94)	cb61-(sw73--sw94)	cb62-(sw113--sw94)	cb61-(sw113--sw94)
64	cb63-(sw94--sw75)	cb62-(sw94--sw75)	cb63-(sw94--sw115)	cb62-(sw94--sw115)
65	cb64-(sw99--sw80)		cb64-(sw99--sw120)	
66	cb65-(sw80--sw101)	cb64-(sw80--sw101)	cb65-(sw120--sw101)	cb64-(sw120--sw101)
67	cb66-(sw101--sw82)	cb65-(sw101--sw82)	cb66-(sw101--sw122)	cb65-(sw101--sw122)
68	cb67-(sw82--sw103)	cb66-(sw82--sw103)	cb67-(sw122--sw103)	cb66-(sw122--sw103)
69	cb68-(sw103--sw84)	cb67-(sw103--sw84)	cb68-(sw103--sw124)	cb67-(sw103--sw124)
70	cb69-(sw84--sw105)	cb68-(sw84--sw105)	cb69-(sw124--sw105)	cb68-(sw124--sw105)
71	cb70-(sw105--sw86)	cb69-(sw105--sw86)	cb70-(sw105--sw126)	cb69-(sw105--sw126)
72	cb71-(sw86--sw107)	cb70-(sw86--sw107)	cb71-(sw126--sw107)	cb70-(sw126--sw107)
73	cb72-(sw107--sw88)	cb71-(sw107--sw88)	cb72-(sw107--sw128)	cb71-(sw107--sw128)
74	cb73-(sw88--sw109)	cb72-(sw88--sw109)	cb73-(sw128--sw109)	cb72-(sw128--sw109)
75	cb74-(sw109--sw90)	cb73-(sw109--sw90)	cb74-(sw109--sw130)	cb73-(sw109--sw130)
76	cb75-(sw90--sw111)	cb74-(sw90--sw111)	cb75-(sw130--sw111)	cb74-(sw130--sw111)
77	cb76-(sw111--sw92)	cb75-(sw111--sw92)	cb76-(sw111--sw132)	cb75-(sw111--sw132)
78	cb77-(sw92--sw113)	cb76-(sw92--sw113)	cb77-(sw132--sw113)	cb76-(sw132--sw113)
79	cb78-(sw113--sw94)	cb77-(sw113--sw94)	cb78-(sw113--sw134)	cb77-(sw113--sw134)
80	cb79-(sw94--sw115)	cb78-(sw94--sw115)	cb79-(sw134--sw115)	cb78-(sw134--sw115)
81	cb80-(sw99--sw120)		cb80-(sw139--sw120)	
82	cb81-(sw120--sw101)	cb80-(sw120--sw101)	cb81-(sw120--sw141)	cb80-(sw120--sw141)
83	cb82-(sw101--sw122)	cb81-(sw101--sw122)	cb82-(sw141--sw122)	cb81-(sw141--sw122)
84	cb83-(sw122--sw103)	cb82-(sw122--sw103)	cb83-(sw122--sw143)	cb82-(sw122--sw143)
85	cb84-(sw103--sw124)	cb83-(sw103--sw124)	cb84-(sw143--sw124)	cb83-(sw143--sw124)
86	cb85-(sw124--sw105)	cb84-(sw124--sw105)	cb85-(sw124--sw145)	cb84-(sw124--sw145)
87	cb86-(sw105--sw126)	cb85-(sw105--sw126)	cb86-(sw145--sw126)	cb85-(sw145--sw126)
88	cb87-(sw126--sw107)	cb86-(sw126--sw107)	cb87-(sw126--sw147)	cb86-(sw126--sw147)
89	cb88-(sw107--sw128)	cb87-(sw107--sw128)	cb88-(sw147--sw128)	cb87-(sw147--sw128)
90	cb89-(sw128--sw109)	cb88-(sw128--sw109)	cb89-(sw128--sw149)	cb88-(sw128--sw149)
91	cb90-(sw109--sw130)	cb89-(sw109--sw130)	cb90-(sw149--sw130)	cb89-(sw149--sw130)
92	cb91-(sw130--sw111)	cb90-(sw130--sw111)	cb91-(sw130--sw151)	cb90-(sw130--sw151)
93	cb92-(sw111--sw132)	cb91-(sw111--sw132)	cb92-(sw151--sw132)	cb91-(sw151--sw132)
94	cb93-(sw132--sw113)	cb92-(sw132--sw113)	cb93-(sw132--sw153)	cb92-(sw132--sw153)
95	cb94-(sw113--sw134)	cb93-(sw113--sw134)	cb94-(sw153--sw134)	cb93-(sw153--sw134)
96	cb95-(sw134--sw115)	cb94-(sw134--sw115)	cb95-(sw134--sw155)	cb94-(sw134--sw155)

97	cb96-(sw139--sw120)		cb96-(sw139--sw160)	
98	cb97-(sw120--sw141)	cb96-(sw120--sw141)	cb97-(sw160--sw141)	cb96-(sw160--sw141)
99	cb98-(sw141--sw122)	cb97-(sw141--sw122)	cb98-(sw141--sw162)	cb97-(sw141--sw162)
100	cb99-(sw122--sw143)	cb98-(sw122--sw143)	cb99-(sw162--sw143)	cb98-(sw162--sw143)
101	cb100-(sw143--sw124)	cb99-(sw143--sw124)	cb100-(sw143--sw164)	cb99-(sw143--sw164)
102	cb101-(sw124--sw145)	cb100-(sw124--sw145)	cb101-(sw164--sw145)	cb100-(sw164--sw145)
103	cb102-(sw145--sw126)	cb101-(sw145--sw126)	cb102-(sw145--sw166)	cb101-(sw145--sw166)
104	cb103-(sw126--sw147)	cb102-(sw126--sw147)	cb103-(sw166--sw147)	cb102-(sw166--sw147)
105	cb104-(sw147--sw128)	cb103-(sw147--sw128)	cb104-(sw147--sw168)	cb103-(sw147--sw168)
106	cb105-(sw128--sw149)	cb104-(sw128--sw149)	cb105-(sw168--sw149)	cb104-(sw168--sw149)
107	cb106-(sw149--sw130)	cb105-(sw149--sw130)	cb106-(sw149--sw170)	cb105-(sw149--sw170)
108	cb107-(sw130--sw151)	cb106-(sw130--sw151)	cb107-(sw170--sw151)	cb106-(sw170--sw151)
109	cb108-(sw151--sw132)	cb107-(sw151--sw132)	cb108-(sw151--sw172)	cb107-(sw151--sw172)
110	cb109-(sw132--sw153)	cb108-(sw132--sw153)	cb109-(sw172--sw153)	cb108-(sw172--sw153)
111	cb110-(sw153--sw134)	cb109-(sw153--sw134)	cb110-(sw153--sw174)	cb109-(sw153--sw174)
112	cb111-(sw134--sw155)	cb110-(sw134--sw155)	cb111-(sw174--sw155)	cb110-(sw174--sw155)
113	cb112-(sw139--sw160)		cb112-(sw179--sw160)	
114	cb113-(sw160--sw141)	cb112-(sw160--sw141)	cb113-(sw160--sw181)	cb112-(sw160--sw181)
115	cb114-(sw141--sw162)	cb113-(sw141--sw162)	cb114-(sw181--sw162)	cb113-(sw181--sw162)
116	cb115-(sw162--sw143)	cb114-(sw162--sw143)	cb115-(sw162--sw183)	cb114-(sw162--sw183)
117	cb116-(sw143--sw164)	cb115-(sw143--sw164)	cb116-(sw183--sw164)	cb115-(sw183--sw164)
118	cb117-(sw164--sw145)	cb116-(sw164--sw145)	cb117-(sw164--sw185)	cb116-(sw164--sw185)
119	cb118-(sw145--sw166)	cb117-(sw145--sw166)	cb118-(sw185--sw166)	cb117-(sw185--sw166)
120	cb119-(sw166--sw147)	cb118-(sw166--sw147)	cb119-(sw166--sw187)	cb118-(sw166--sw187)
121	cb120-(sw147--sw168)	cb119-(sw147--sw168)	cb120-(sw187--sw168)	cb119-(sw187--sw168)
122	cb121-(sw168--sw149)	cb120-(sw168--sw149)	cb121-(sw168--sw189)	cb120-(sw168--sw189)
123	cb122-(sw149--sw170)	cb121-(sw149--sw170)	cb122-(sw189--sw170)	cb121-(sw189--sw170)
124	cb123-(sw170--sw151)	cb122-(sw170--sw151)	cb123-(sw170--sw191)	cb122-(sw170--sw191)
125	cb124-(sw151--sw172)	cb123-(sw151--sw172)	cb124-(sw191--sw172)	cb123-(sw191--sw172)
126	cb125-(sw172--sw153)	cb124-(sw172--sw153)	cb125-(sw172--sw193)	cb124-(sw172--sw193)
127	cb126-(sw153--sw174)	cb125-(sw153--sw174)	cb126-(sw193--sw174)	cb125-(sw193--sw174)
128	cb127-(sw174--sw155)	cb126-(sw174--sw155)	cb127-(sw174--sw195)	cb126-(sw174--sw195)
129	cb128-(sw179--sw160)		cb128-(sw179--sw200)	
130	cb129-(sw160--sw181)	cb128-(sw160--sw181)	cb129-(sw200--sw181)	cb128-(sw200--sw181)
131	cb130-(sw181--sw162)	cb129-(sw181--sw162)	cb130-(sw181--sw202)	cb129-(sw181--sw202)
132	cb131-(sw162--sw183)	cb130-(sw162--sw183)	cb131-(sw202--sw183)	cb130-(sw202--sw183)
133	cb132-(sw183--sw164)	cb131-(sw183--sw164)	cb132-(sw183--sw204)	cb131-(sw183--sw204)
134	cb133-(sw164--sw185)	cb132-(sw164--sw185)	cb133-(sw204--sw185)	cb132-(sw204--sw185)
135	cb134-(sw185--sw166)	cb133-(sw185--sw166)	cb134-(sw185--sw206)	cb133-(sw185--sw206)
136	cb135-(sw166--sw187)	cb134-(sw166--sw187)	cb135-(sw206--sw187)	cb134-(sw206--sw187)
137	cb136-(sw187--sw168)	cb135-(sw187--sw168)	cb136-(sw187--sw208)	cb135-(sw187--sw208)
138	cb137-(sw168--sw189)	cb136-(sw168--sw189)	cb137-(sw208--sw189)	cb136-(sw208--sw189)
139	cb138-(sw189--sw170)	cb137-(sw189--sw170)	cb138-(sw189--sw210)	cb137-(sw189--sw210)
140	cb139-(sw170--sw191)	cb138-(sw170--sw191)	cb139-(sw210--sw191)	cb138-(sw210--sw191)
141	cb140-(sw191--sw172)	cb139-(sw191--sw172)	cb140-(sw191--sw212)	cb139-(sw191--sw212)
142	cb141-(sw172--sw193)	cb140-(sw172--sw193)	cb141-(sw212--sw193)	cb140-(sw212--sw193)
143	cb142-(sw193--sw174)	cb141-(sw193--sw174)	cb142-(sw193--sw214)	cb141-(sw193--sw214)
144	cb143-(sw174--sw195)	cb142-(sw174--sw195)	cb143-(sw214--sw195)	cb142-(sw214--sw195)
145	cb144-(sw179--sw200)		cb144-(sw219--sw200)	

146	cb145-(sw200--sw181)	cb144-(sw200--sw181)	cb145-(sw200--sw221)	cb144-(sw200--sw221)
147	cb146-(sw181--sw202)	cb145-(sw181--sw202)	cb146-(sw221--sw202)	cb145-(sw221--sw202)
148	cb147-(sw202--sw183)	cb146-(sw202--sw183)	cb147-(sw202--sw223)	cb146-(sw202--sw223)
149	cb148-(sw183--sw204)	cb147-(sw183--sw204)	cb148-(sw223--sw204)	cb147-(sw223--sw204)
150	cb149-(sw204--sw185)	cb148-(sw204--sw185)	cb149-(sw204--sw225)	cb148-(sw204--sw225)
151	cb150-(sw185--sw206)	cb149-(sw185--sw206)	cb150-(sw225--sw206)	cb149-(sw225--sw206)
152	cb151-(sw206--sw187)	cb150-(sw206--sw187)	cb151-(sw206--sw227)	cb150-(sw206--sw227)
153	cb152-(sw187--sw208)	cb151-(sw187--sw208)	cb152-(sw227--sw208)	cb151-(sw227--sw208)
154	cb153-(sw208--sw189)	cb152-(sw208--sw189)	cb153-(sw208--sw229)	cb152-(sw208--sw229)
155	cb154-(sw189--sw210)	cb153-(sw189--sw210)	cb154-(sw229--sw210)	cb153-(sw229--sw210)
156	cb155-(sw210--sw191)	cb154-(sw210--sw191)	cb155-(sw210--sw231)	cb154-(sw210--sw231)
157	cb156-(sw191--sw212)	cb155-(sw191--sw212)	cb156-(sw231--sw212)	cb155-(sw231--sw212)
158	cb157-(sw212--sw193)	cb156-(sw212--sw193)	cb157-(sw212--sw233)	cb156-(sw212--sw233)
159	cb158-(sw193--sw214)	cb157-(sw193--sw214)	cb158-(sw233--sw214)	cb157-(sw233--sw214)
160	cb159-(sw214--sw195)	cb158-(sw214--sw195)	cb159-(sw214--sw235)	cb158-(sw214--sw235)
161	cb160-(sw219--sw200)		cb160-(sw219--sw240)	
162	cb161-(sw200--sw221)	cb160-(sw200--sw221)	cb161-(sw240--sw221)	cb160-(sw240--sw221)
163	cb162-(sw221--sw202)	cb161-(sw221--sw202)	cb162-(sw221--sw242)	cb161-(sw221--sw242)
164	cb163-(sw202--sw223)	cb162-(sw202--sw223)	cb163-(sw242--sw223)	cb162-(sw242--sw223)
165	cb164-(sw223--sw204)	cb163-(sw223--sw204)	cb164-(sw223--sw244)	cb163-(sw223--sw244)
166	cb165-(sw204--sw225)	cb164-(sw204--sw225)	cb165-(sw244--sw225)	cb164-(sw244--sw225)
167	cb166-(sw225--sw206)	cb165-(sw225--sw206)	cb166-(sw225--sw246)	cb165-(sw225--sw246)
168	cb167-(sw206--sw227)	cb166-(sw206--sw227)	cb167-(sw246--sw227)	cb166-(sw246--sw227)
169	cb168-(sw227--sw208)	cb167-(sw227--sw208)	cb168-(sw227--sw248)	cb167-(sw227--sw248)
170	cb169-(sw208--sw229)	cb168-(sw208--sw229)	cb169-(sw248--sw229)	cb168-(sw248--sw229)
171	cb170-(sw229--sw210)	cb169-(sw229--sw210)	cb170-(sw229--sw250)	cb169-(sw229--sw250)
172	cb171-(sw210--sw231)	cb170-(sw210--sw231)	cb171-(sw250--sw231)	cb170-(sw250--sw231)
173	cb172-(sw231--sw212)	cb171-(sw231--sw212)	cb172-(sw231--sw252)	cb171-(sw231--sw252)
174	cb173-(sw212--sw233)	cb172-(sw212--sw233)	cb173-(sw252--sw233)	cb172-(sw252--sw233)
175	cb174-(sw233--sw214)	cb173-(sw233--sw214)	cb174-(sw233--sw254)	cb173-(sw233--sw254)
176	cb175-(sw214--sw235)	cb174-(sw214--sw235)	cb175-(sw254--sw235)	cb174-(sw254--sw235)
177	cb176-(sw219--sw240)		cb176-(sw259--sw240)	
178	cb177-(sw240--sw221)	cb176-(sw240--sw221)	cb177-(sw240--sw261)	cb176-(sw240--sw261)
179	cb178-(sw221--sw242)	cb177-(sw221--sw242)	cb178-(sw261--sw242)	cb177-(sw261--sw242)
180	cb179-(sw242--sw223)	cb178-(sw242--sw223)	cb179-(sw242--sw263)	cb178-(sw242--sw263)
181	cb180-(sw223--sw244)	cb179-(sw223--sw244)	cb180-(sw263--sw244)	cb179-(sw263--sw244)
182	cb181-(sw244--sw225)	cb180-(sw244--sw225)	cb181-(sw244--sw265)	cb180-(sw244--sw265)
183	cb182-(sw225--sw246)	cb181-(sw225--sw246)	cb182-(sw265--sw246)	cb181-(sw265--sw246)
184	cb183-(sw246--sw227)	cb182-(sw246--sw227)	cb183-(sw246--sw267)	cb182-(sw246--sw267)
185	cb184-(sw227--sw248)	cb183-(sw227--sw248)	cb184-(sw267--sw248)	cb183-(sw267--sw248)
186	cb185-(sw248--sw229)	cb184-(sw248--sw229)	cb185-(sw248--sw269)	cb184-(sw248--sw269)
187	cb186-(sw229--sw250)	cb185-(sw229--sw250)	cb186-(sw269--sw250)	cb185-(sw269--sw250)
188	cb187-(sw250--sw231)	cb186-(sw250--sw231)	cb187-(sw250--sw271)	cb186-(sw250--sw271)
189	cb188-(sw231--sw252)	cb187-(sw231--sw252)	cb188-(sw271--sw252)	cb187-(sw271--sw252)
190	cb189-(sw252--sw233)	cb188-(sw252--sw233)	cb189-(sw252--sw273)	cb188-(sw252--sw273)
191	cb190-(sw233--sw254)	cb189-(sw233--sw254)	cb190-(sw273--sw254)	cb189-(sw273--sw254)
192	cb191-(sw254--sw235)	cb190-(sw254--sw235)	cb191-(sw254--sw275)	cb190-(sw254--sw275)
193	cb192-(sw259--sw240)		cb192-(sw259--sw280)	
194	cb193-(sw240--sw261)	cb192-(sw240--sw261)	cb193-(sw280--sw261)	cb192-(sw280--sw261)

195	cb194-(sw261--sw242)	cb193-(sw261--sw242)	cb194-(sw261--sw282)	cb193-(sw261--sw282)
196	cb195-(sw242--sw263)	cb194-(sw242--sw263)	cb195-(sw282--sw263)	cb194-(sw282--sw263)
197	cb196-(sw263--sw244)	cb195-(sw263--sw244)	cb196-(sw263--sw284)	cb195-(sw263--sw284)
198	cb197-(sw244--sw265)	cb196-(sw244--sw265)	cb197-(sw284--sw265)	cb196-(sw284--sw265)
199	cb198-(sw265--sw246)	cb197-(sw265--sw246)	cb198-(sw265--sw286)	cb197-(sw265--sw286)
200	cb199-(sw246--sw267)	cb198-(sw246--sw267)	cb199-(sw286--sw267)	cb198-(sw286--sw267)
201	cb200-(sw267--sw248)	cb199-(sw267--sw248)	cb200-(sw267--sw288)	cb199-(sw267--sw288)
202	cb201-(sw248--sw269)	cb200-(sw248--sw269)	cb201-(sw288--sw269)	cb200-(sw288--sw269)
203	cb202-(sw269--sw250)	cb201-(sw269--sw250)	cb202-(sw269--sw290)	cb201-(sw269--sw290)
204	cb203-(sw250--sw271)	cb202-(sw250--sw271)	cb203-(sw290--sw271)	cb202-(sw290--sw271)
205	cb204-(sw271--sw252)	cb203-(sw271--sw252)	cb204-(sw271--sw292)	cb203-(sw271--sw292)
206	cb205-(sw252--sw273)	cb204-(sw252--sw273)	cb205-(sw292--sw273)	cb204-(sw292--sw273)
207	cb206-(sw273--sw254)	cb205-(sw273--sw254)	cb206-(sw273--sw294)	cb205-(sw273--sw294)
208	cb207-(sw254--sw275)	cb206-(sw254--sw275)	cb207-(sw294--sw275)	cb206-(sw294--sw275)
209	cb208-(sw259--sw280)		cb208-(sw299--sw280)	
210	cb209-(sw280--sw261)	cb208-(sw280--sw261)	cb209-(sw280--sw301)	cb208-(sw280--sw301)
211	cb210-(sw261--sw282)	cb209-(sw261--sw282)	cb210-(sw301--sw282)	cb209-(sw301--sw282)
212	cb211-(sw282--sw263)	cb210-(sw282--sw263)	cb211-(sw282--sw303)	cb210-(sw282--sw303)
213	cb212-(sw263--sw284)	cb211-(sw263--sw284)	cb212-(sw303--sw284)	cb211-(sw303--sw284)
214	cb213-(sw284--sw265)	cb212-(sw284--sw265)	cb213-(sw284--sw305)	cb212-(sw284--sw305)
215	cb214-(sw265--sw286)	cb213-(sw265--sw286)	cb214-(sw305--sw286)	cb213-(sw305--sw286)
216	cb215-(sw286--sw267)	cb214-(sw286--sw267)	cb215-(sw286--sw307)	cb214-(sw286--sw307)
217	cb216-(sw267--sw288)	cb215-(sw267--sw288)	cb216-(sw307--sw288)	cb215-(sw307--sw288)
218	cb217-(sw288--sw269)	cb216-(sw288--sw269)	cb217-(sw288--sw309)	cb216-(sw288--sw309)
219	cb218-(sw269--sw290)	cb217-(sw269--sw290)	cb218-(sw309--sw290)	cb217-(sw309--sw290)
220	cb219-(sw290--sw271)	cb218-(sw290--sw271)	cb219-(sw290--sw311)	cb218-(sw290--sw311)
221	cb220-(sw271--sw292)	cb219-(sw271--sw292)	cb220-(sw311--sw292)	cb219-(sw311--sw292)
222	cb221-(sw292--sw273)	cb220-(sw292--sw273)	cb221-(sw292--sw313)	cb220-(sw292--sw313)
223	cb222-(sw273--sw294)	cb221-(sw273--sw294)	cb222-(sw313--sw294)	cb221-(sw313--sw294)
224	cb223-(sw294--sw275)	cb222-(sw294--sw275)	cb223-(sw294--sw315)	cb222-(sw294--sw315)
225	cb224-(sw299--sw280)		cb224-(sw299--sw320)	
226	cb225-(sw280--sw301)	cb224-(sw280--sw301)	cb225-(sw320--sw301)	cb224-(sw320--sw301)
227	cb226-(sw301--sw282)	cb225-(sw301--sw282)	cb226-(sw301--sw322)	cb225-(sw301--sw322)
228	cb227-(sw282--sw303)	cb226-(sw282--sw303)	cb227-(sw322--sw303)	cb226-(sw322--sw303)
229	cb228-(sw303--sw284)	cb227-(sw303--sw284)	cb228-(sw303--sw324)	cb227-(sw303--sw324)
230	cb229-(sw284--sw305)	cb228-(sw284--sw305)	cb229-(sw324--sw305)	cb228-(sw324--sw305)
231	cb230-(sw305--sw286)	cb229-(sw305--sw286)	cb230-(sw305--sw326)	cb229-(sw305--sw326)
232	cb231-(sw286--sw307)	cb230-(sw286--sw307)	cb231-(sw326--sw307)	cb230-(sw326--sw307)
233	cb232-(sw307--sw288)	cb231-(sw307--sw288)	cb232-(sw307--sw328)	cb231-(sw307--sw328)
234	cb233-(sw288--sw309)	cb232-(sw288--sw309)	cb233-(sw328--sw309)	cb232-(sw328--sw309)
235	cb234-(sw309--sw290)	cb233-(sw309--sw290)	cb234-(sw311--sw330)	cb233-(sw309--sw330)
236	cb235-(sw290--sw311)	cb234-(sw290--sw311)	cb235-(sw330--sw313)	cb234-(sw330--sw311)
237	cb236-(sw311--sw292)	cb235-(sw311--sw292)	cb236-(sw313--sw332)	cb235-(sw311--sw332)
238	cb237-(sw292--sw313)	cb236-(sw292--sw313)	cb237-(sw332--sw315)	cb236-(sw332--sw313)
239	cb238-(sw313--sw294)	cb237-(sw313--sw294)	cb238-(sw315--sw334)	cb237-(sw313--sw334)
240	cb239-(sw294--sw315)	cb238-(sw294--sw315)	cb239-(sw334--sw315)	cb238-(sw334--sw315)
241	cb240-(sw299--sw320)		cb240-(sw339--sw320)	
242	cb241-(sw320--sw301)	cb240-(sw320--sw301)	cb241-(sw320--sw341)	cb240-(sw320--sw341)
243	cb242-(sw301--sw322)	cb241-(sw301--sw322)	cb242-(sw341--sw322)	cb241-(sw341--sw322)

244	cb243-(sw322--sw303)	cb242-(sw322--sw303)	cb243-(sw322--sw343)	cb242-(sw322--sw343)
245	cb244-(sw303--sw324)	cb243-(sw303--sw324)	cb244-(sw343--sw324)	cb243-(sw343--sw324)
246	cb245-(sw324--sw305)	cb244-(sw324--sw305)	cb245-(sw324--sw345)	cb244-(sw324--sw345)
247	cb246-(sw305--sw326)	cb245-(sw305--sw326)	cb246-(sw345--sw326)	cb245-(sw345--sw326)
248	cb247-(sw326--sw307)	cb246-(sw326--sw307)	cb247-(sw326--sw347)	cb246-(sw326--sw347)
249	cb248-(sw307--sw328)	cb247-(sw307--sw328)	cb248-(sw347--sw328)	cb247-(sw347--sw328)
250	cb249-(sw328--sw309)	cb248-(sw328--sw309)	cb249-(sw328--sw349)	cb248-(sw328--sw349)
251	cb250-(sw309--sw330)	cb249-(sw309--sw330)	cb250-(sw349--sw330)	cb249-(sw349--sw330)
252	cb251-(sw330--sw311)	cb250-(sw330--sw311)	cb251-(sw330--sw351)	cb250-(sw330--sw351)
253	cb252-(sw311--sw332)	cb251-(sw311--sw332)	cb252-(sw353--sw332)	cb251-(sw351--sw332)
254	cb253-(sw332--sw313)	cb252-(sw332--sw313)	cb253-(sw332--sw355)	cb252-(sw332--sw353)
255	cb254-(sw313--sw334)	cb253-(sw313--sw334)	cb254-(sw355--sw334)	cb253-(sw353--sw334)
256	cb255-(sw334--sw315)	cb254-(sw334--sw315)	cb255-(sw334--sw355)	cb254-(sw334--sw355)
257		cb255-(sw315--sw336)		cb255-(sw355--sw336)
258		cb15-(sw35--sw16)		cb15-(sw35--sw56)
259		cb31-(sw35--sw56)		cb31-(sw75--sw56)
260		cb47-(sw75--sw56)		cb47-(sw75--sw96)
261		cb63-(sw75--sw96)		cb63-(sw115--sw96)
262		cb79-(sw115--sw96)		cb79-(sw115--sw136)
263		cb95-(sw115--sw136)		cb95-(sw155--sw136)
264		cb111-(sw155--sw136)		cb111-(sw155--sw176)
265		cb127-(sw155--sw176)		cb127-(sw195--sw176)
266		cb143-(sw195--sw176)		cb143-(sw195--sw216)
267		cb159-(sw195--sw216)		cb159-(sw235--sw216)
268		cb175-(sw235--sw216)		cb175-(sw235--sw256)
269		cb191-(sw235--sw256)		cb191-(sw275--sw256)
270		cb207-(sw275--sw256)		cb207-(sw275--sw296)
271		cb223-(sw275--sw296)		cb223-(sw315--sw296)
272		cb239-(sw315--sw296)		cb239-(sw315--336)
273	cb0-(sw19--sw20)		cb0-(sw0--sw20)	
274	cb1-(sw20--sw21)	cb0-(sw20--sw21)	cb1-(sw1--sw21)	
275	cb2-(sw21--sw22)	cb1-(sw21--sw22)	cb2-(sw2--sw22)	
276	cb3-(sw22--sw23)	cb2-(sw22--sw23)	cb3-(sw3--sw23)	
277	cb4-(sw23--sw24)	cb3-(sw23--sw24)	cb4-(sw4--sw24)	
278	cb5-(sw24--sw25)	cb4-(sw24--sw25)	cb5-(sw5--sw25)	
279	cb6-(sw25--sw26)	cb5-(sw25--sw26)	cb6-(sw6--sw26)	
280	cb7-(sw26--sw27)	cb6-(sw26--sw27)	cb7-(sw7--sw27)	
281	cb8-(sw27--sw28)	cb7-(sw27--sw28)	cb8-(sw8--sw28)	
282	cb9-(sw28--sw29)	cb8-(sw28--sw29)	cb9-(sw9--sw29)	
283	cb10-(sw29--sw30)	cb9-(sw29--sw30)	cb10-(sw10--sw30)	
284	cb11-(sw30--sw31)	cb10-(sw30--sw31)	cb11-(sw11--sw31)	
285	cb12-(sw31--sw32)	cb11-(sw31--sw32)	cb12-(sw12--sw32)	
286	cb13-(sw32--sw33)	cb12-(sw32--sw33)	cb13-(sw13--sw33)	
287	cb14-(sw33--sw34)	cb13-(sw33--sw34)	cb14-(sw14--sw34)	
288	cb15-(sw34--sw35)	cb14-(sw34--sw35)	cb15-(sw15--sw35)	
289	cb16-(sw39--sw40)		cb16-(sw20--sw40)	cb0-(sw20--sw40)
290	cb17-(sw40--sw41)	cb16-(sw40--sw41)	cb17-(sw21--sw41)	cb1-(sw21--sw41)
291	cb18-(sw41--sw42)	cb17-(sw41--sw42)	cb18-(sw22--sw42)	cb2-(sw22--sw42)
292	cb19-(sw42--sw43)	cb18-(sw42--sw43)	cb19-(sw23--sw43)	cb3-(sw23--sw43)

293	cb20-(sw43--sw44)	cb19-(sw43--sw44)	cb20-(sw24--sw44)	cb4-(sw24--sw44)
294	cb21-(sw44--sw45)	cb20-(sw44--sw45)	cb21-(sw25--sw45)	cb5-(sw25--sw45)
295	cb22-(sw45--sw46)	cb21-(sw45--sw46)	cb22-(sw26--sw46)	cb6-(sw26--sw46)
296	cb23-(sw46--sw47)	cb22-(sw46--sw47)	cb23-(sw27--sw47)	cb7-(sw27--sw47)
297	cb24-(sw47--sw48)	cb23-(sw47--sw48)	cb24-(sw28--sw48)	cb8-(sw28--sw48)
298	cb25-(sw48--sw49)	cb24-(sw48--sw49)	cb25-(sw29--sw49)	cb9-(sw29--sw49)
299	cb26-(sw49--sw50)	cb25-(sw49--sw50)	cb26-(sw30--sw50)	cb10-(sw30--sw50)
300	cb27-(sw50--sw51)	cb26-(sw50--sw51)	cb27-(sw31--sw51)	cb11-(sw31--sw51)
301	cb28-(sw51--sw52)	cb27-(sw51--sw52)	cb28-(sw32--sw52)	cb12-(sw32--sw52)
302	cb29-(sw52--sw53)	cb28-(sw52--sw53)	cb29-(sw33--sw53)	cb13-(sw33--sw53)
303	cb30-(sw53--sw54)	cb29-(sw53--sw54)	cb30-(sw34--sw54)	cb14-(sw34--sw54)
304	cb31-(sw54--sw55)	cb30-(sw54--sw55)	cb31-(sw35--sw55)	cb15-(sw35--sw55)
305	cb32-(sw59--sw60)		cb32-(sw40--sw60)	cb16-(sw40--sw60)
306	cb33-(sw60--sw61)	cb32-(sw60--sw61)	cb33-(sw41--sw61)	cb17-(sw41--sw61)
307	cb34-(sw61--sw62)	cb33-(sw61--sw62)	cb34-(sw42--sw62)	cb18-(sw42--sw62)
308	cb35-(sw62--sw63)	cb34-(sw62--sw63)	cb35-(sw43--sw63)	cb19-(sw43--sw63)
309	cb36-(sw63--sw64)	cb35-(sw63--sw64)	cb36-(sw44--sw64)	cb20-(sw44--sw64)
310	cb37-(sw64--sw65)	cb36-(sw64--sw65)	cb37-(sw45--sw65)	cb21-(sw45--sw65)
311	cb38-(sw65--sw66)	cb37-(sw65--sw66)	cb38-(sw46--sw66)	cb22-(sw46--sw66)
312	cb39-(sw66--sw67)	cb38-(sw66--sw67)	cb39-(sw47--sw67)	cb23-(sw47--sw67)
313	cb40-(sw67--sw68)	cb39-(sw67--sw68)	cb40-(sw48--sw68)	cb24-(sw48--sw68)
314	cb41-(sw68--sw69)	cb40-(sw68--sw69)	cb41-(sw49--sw69)	cb25-(sw49--sw69)
315	cb42-(sw69--sw70)	cb41-(sw69--sw70)	cb42-(sw50--sw70)	cb26-(sw50--sw70)
316	cb43-(sw70--sw71)	cb42-(sw70--sw71)	cb43-(sw51--sw71)	cb27-(sw51--sw71)
317	cb44-(sw71--sw72)	cb43-(sw71--sw72)	cb44-(sw52--sw72)	cb28-(sw52--sw72)
318	cb45-(sw72--sw73)	cb44-(sw72--sw73)	cb45-(sw53--sw73)	cb29-(sw53--sw73)
319	cb46-(sw73--sw74)	cb45-(sw73--sw74)	cb46-(sw54--sw74)	cb30-(sw54--sw74)
320	cb47-(sw74--sw75)	cb46-(sw74--sw75)	cb47-(sw55--sw75)	cb31-(sw55--sw75)
321	cb48-(sw79--sw80)		cb48-(sw60--sw80)	cb32-(sw60--sw80)
322	cb49-(sw80--sw81)	cb48-(sw80--sw81)	cb49-(sw61--sw81)	cb33-(sw61--sw81)
323	cb50-(sw81--sw82)	cb49-(sw81--sw82)	cb50-(sw62--sw82)	cb34-(sw62--sw82)
324	cb51-(sw82--sw83)	cb50-(sw82--sw83)	cb51-(sw63--sw83)	cb35-(sw63--sw83)
325	cb52-(sw83--sw84)	cb51-(sw83--sw84)	cb52-(sw64--sw84)	cb36-(sw64--sw84)
326	cb53-(sw84--sw85)	cb52-(sw84--sw85)	cb53-(sw65--sw85)	cb37-(sw65--sw85)
327	cb54-(sw85--sw86)	cb53-(sw85--sw86)	cb54-(sw66--sw86)	cb38-(sw66--sw86)
328	cb55-(sw86--sw87)	cb54-(sw86--sw87)	cb55-(sw67--sw87)	cb39-(sw67--sw87)
329	cb56-(sw87--sw88)	cb55-(sw87--sw88)	cb56-(sw68--sw88)	cb40-(sw68--sw88)
330	cb57-(sw88--sw89)	cb56-(sw88--sw89)	cb57-(sw69--sw89)	cb41-(sw69--sw89)
331	cb58-(sw89--sw90)	cb57-(sw89--sw90)	cb58-(sw70--sw90)	cb42-(sw70--sw90)
332	cb59-(sw90--sw91)	cb58-(sw90--sw91)	cb59-(sw71--sw91)	cb43-(sw71--sw91)
333	cb60-(sw91--sw92)	cb59-(sw91--sw92)	cb60-(sw72--sw92)	cb44-(sw72--sw92)
334	cb61-(sw92--sw93)	cb60-(sw92--sw93)	cb61-(sw73--sw93)	cb45-(sw73--sw93)
335	cb62-(sw93--sw94)	cb61-(sw93--sw94)	cb62-(sw74--sw94)	cb46-(sw74--sw94)
336	cb63-(sw94--sw95)	cb62-(sw94--sw95)	cb63-(sw75--sw95)	cb47-(sw75--sw95)
337	cb64-(sw99--sw100)		cb64-(sw80--sw100)	cb48-(sw80--sw100)
338	cb65-(sw100--sw101)	cb64-(sw100--sw101)	cb65-(sw81--sw101)	cb49-(sw81--sw101)
339	cb66-(sw101--sw102)	cb65-(sw101--sw102)	cb66-(sw82--sw102)	cb50-(sw82--sw102)
340	cb67-(sw102--sw103)	cb66-(sw102--sw103)	cb67-(sw83--sw103)	cb51-(sw83--sw103)
341	cb68-(sw103--sw104)	cb67-(sw103--sw104)	cb68-(sw84--sw104)	cb52-(sw84--sw104)

342	cb69-(sw104--sw105)	cb68-(sw104--sw105)	cb69-(sw85--sw105)	cb53-(sw85--sw105)
343	cb70-(sw105--sw106)	cb69-(sw105--sw106)	cb70-(sw86--sw106)	cb54-(sw86--sw106)
344	cb71-(sw106--sw107)	cb70-(sw106--sw107)	cb71-(sw87--sw107)	cb55-(sw87--sw107)
345	cb72-(sw107--sw108)	cb71-(sw107--sw108)	cb72-(sw88--sw108)	cb56-(sw88--sw108)
346	cb73-(sw108--sw109)	cb72-(sw108--sw109)	cb73-(sw89--sw109)	cb57-(sw89--sw109)
347	cb74-(sw109--sw110)	cb73-(sw109--sw110)	cb74-(sw90--sw110)	cb58-(sw90--sw110)
348	cb75-(sw110--sw111)	cb74-(sw110--sw111)	cb75-(sw91--sw111)	cb59-(sw91--sw111)
349	cb76-(sw111--sw112)	cb75-(sw111--sw112)	cb76-(sw92--sw112)	cb60-(sw92--sw112)
350	cb77-(sw112--sw113)	cb76-(sw112--sw113)	cb77-(sw93--sw113)	cb61-(sw93--sw113)
351	cb78-(sw113--sw114)	cb77-(sw113--sw114)	cb78-(sw94--sw114)	cb62-(sw94--sw114)
352	cb79-(sw114--sw115)	cb78-(sw114--sw115)	cb79-(sw95--sw115)	cb63-(sw95--sw115)
353	cb80-(sw119--sw120)		cb80-(sw100--sw120)	cb64-(sw100--sw120)
354	cb81-(sw120--sw121)	cb80-(sw120--sw121)	cb81-(sw101--sw121)	cb65-(sw101--sw121)
355	cb82-(sw121--sw122)	cb81-(sw121--sw122)	cb82-(sw102--sw122)	cb66-(sw102--sw122)
356	cb83-(sw122--sw123)	cb82-(sw122--sw123)	cb83-(sw103--sw123)	cb67-(sw103--sw123)
357	cb84-(sw123--sw124)	cb83-(sw123--sw124)	cb84-(sw104--sw124)	cb68-(sw104--sw124)
358	cb85-(sw124--sw125)	cb84-(sw124--sw125)	cb85-(sw105--sw125)	cb69-(sw105--sw125)
359	cb86-(sw125--sw126)	cb85-(sw125--sw126)	cb86-(sw106--sw126)	cb70-(sw106--sw126)
360	cb87-(sw126--sw127)	cb86-(sw126--sw127)	cb87-(sw107--sw127)	cb71-(sw107--sw127)
361	cb88-(sw127--sw128)	cb87-(sw127--sw128)	cb88-(sw108--sw128)	cb72-(sw108--sw128)
362	cb89-(sw128--sw129)	cb88-(sw128--sw129)	cb89-(sw109--sw129)	cb73-(sw109--sw129)
363	cb90-(sw129--sw130)	cb89-(sw129--sw130)	cb90-(sw110--sw130)	cb74-(sw110--sw130)
364	cb91-(sw130--sw131)	cb90-(sw130--sw131)	cb91-(sw111--sw131)	cb75-(sw111--sw131)
365	cb92-(sw131--sw132)	cb91-(sw131--sw132)	cb92-(sw112--sw132)	cb76-(sw112--sw132)
366	cb93-(sw132--sw133)	cb92-(sw132--sw133)	cb93-(sw113--sw133)	cb77-(sw113--sw133)
367	cb94-(sw133--sw134)	cb93-(sw133--sw134)	cb94-(sw114--sw134)	cb78-(sw114--sw134)
368	cb95-(sw134--sw135)	cb94-(sw134--sw135)	cb95-(sw115--sw135)	cb79-(sw115--sw135)
369	cb96-(sw139--sw140)		cb96-(sw120--sw140)	cb80-(sw120--sw140)
370	cb97-(sw140--sw141)	cb96-(sw140--sw141)	cb97-(sw121--sw141)	cb81-(sw121--sw141)
371	cb98-(sw141--sw142)	cb97-(sw141--sw142)	cb98-(sw122--sw142)	cb82-(sw122--sw142)
372	cb99-(sw142--sw143)	cb98-(sw142--sw143)	cb99-(sw123--sw143)	cb83-(sw123--sw143)
373	cb100-(sw143--sw144)	cb99-(sw143--sw144)	cb100-(sw124--sw144)	cb84-(sw124--sw144)
374	cb101-(sw144--sw145)	cb100-(sw144--sw145)	cb101-(sw125--sw145)	cb85-(sw125--sw145)
375	cb102-(sw145--sw146)	cb101-(sw145--sw146)	cb102-(sw126--sw146)	cb86-(sw126--sw146)
376	cb103-(sw146--sw147)	cb102-(sw146--sw147)	cb103-(sw127--sw147)	cb87-(sw127--sw147)
377	cb104-(sw147--sw148)	cb103-(sw147--sw148)	cb104-(sw128--sw148)	cb88-(sw128--sw148)
378	cb105-(sw148--sw149)	cb104-(sw148--sw149)	cb105-(sw129--sw149)	cb89-(sw129--sw149)
379	cb106-(sw149--sw150)	cb105-(sw149--sw150)	cb106-(sw130--sw150)	cb90-(sw130--sw150)
380	cb107-(sw150--sw151)	cb106-(sw150--sw151)	cb107-(sw131--sw151)	cb91-(sw131--sw151)
381	cb108-(sw151--sw152)	cb107-(sw151--sw152)	cb108-(sw132--sw152)	cb92-(sw132--sw152)
382	cb109-(sw152--sw153)	cb108-(sw152--sw153)	cb109-(sw133--sw153)	cb93-(sw133--sw153)
383	cb110-(sw153--sw154)	cb109-(sw153--sw154)	cb110-(sw134--sw154)	cb94-(sw134--sw154)
384	cb111-(sw154--sw155)	cb110-(sw154--sw155)	cb111-(sw135--sw155)	cb95-(sw135--sw155)
385	cb112-(sw159--sw160)		cb112-(sw140--sw160)	cb96-(sw140--sw160)
386	cb113-(sw160--sw161)	cb112-(sw160--sw161)	cb113-(sw141--sw161)	cb97-(sw141--sw161)
387	cb114-(sw161--sw162)	cb113-(sw161--sw162)	cb114-(sw142--sw162)	cb98-(sw142--sw162)
388	cb115-(sw162--sw163)	cb114-(sw162--sw163)	cb115-(sw143--sw163)	cb99-(sw143--sw163)
389	cb116-(sw163--sw164)	cb115-(sw163--sw164)	cb116-(sw144--sw164)	cb100-(sw144--sw164)
390	cb117-(sw164--sw165)	cb116-(sw164--sw165)	cb117-(sw145--sw165)	cb101-(sw145--sw165)

391	cb118-(sw165--sw166)	cb117-(sw165--sw166)	cb118-(sw146--sw166)	cb102-(sw146--sw166)
392	cb119-(sw166--sw167)	cb118-(sw166--sw167)	cb119-(sw147--sw167)	cb103-(sw147--sw167)
393	cb120-(sw167--sw168)	cb119-(sw167--sw168)	cb120-(sw148--sw168)	cb104-(sw148--sw168)
394	cb121-(sw168--sw169)	cb120-(sw168--sw169)	cb121-(sw149--sw169)	cb105-(sw149--sw169)
395	cb122-(sw169--sw170)	cb121-(sw169--sw170)	cb122-(sw150--sw170)	cb106-(sw150--sw170)
396	cb123-(sw170--sw171)	cb122-(sw170--sw171)	cb123-(sw151--sw171)	cb107-(sw151--sw171)
397	cb124-(sw171--sw172)	cb123-(sw171--sw172)	cb124-(sw152--sw172)	cb108-(sw152--sw172)
398	cb125-(sw172--sw173)	cb124-(sw172--sw173)	cb125-(sw153--sw173)	cb109-(sw153--sw173)
399	cb126-(sw173--sw174)	cb125-(sw173--sw174)	cb126-(sw154--sw174)	cb110-(sw154--sw174)
400	cb127-(sw174--sw175)	cb126-(sw174--sw175)	cb127-(sw155--sw175)	cb111-(sw155--sw175)
401	cb128-(sw179--sw180)		cb128-(sw160--sw180)	cb112-(sw160--sw180)
402	cb129-(sw180--sw181)	cb128-(sw180--sw181)	cb129-(sw161--sw181)	cb113-(sw161--sw181)
403	cb130-(sw181--sw182)	cb129-(sw181--sw182)	cb130-(sw162--sw182)	cb114-(sw162--sw182)
404	cb131-(sw182--sw183)	cb130-(sw182--sw183)	cb131-(sw163--sw183)	cb115-(sw163--sw183)
405	cb132-(sw183--sw184)	cb131-(sw183--sw184)	cb132-(sw164--sw184)	cb116-(sw164--sw184)
406	cb133-(sw184--sw185)	cb132-(sw184--sw185)	cb133-(sw165--sw185)	cb117-(sw165--sw185)
407	cb134-(sw185--sw186)	cb133-(sw185--sw186)	cb134-(sw166--sw186)	cb118-(sw166--sw186)
408	cb135-(sw186--sw187)	cb134-(sw186--sw187)	cb135-(sw167--sw187)	cb119-(sw167--sw187)
409	cb136-(sw187--sw188)	cb135-(sw187--sw188)	cb136-(sw168--sw188)	cb120-(sw168--sw188)
410	cb137-(sw188--sw189)	cb136-(sw188--sw189)	cb137-(sw169--sw189)	cb121-(sw169--sw189)
411	cb138-(sw189--sw190)	cb137-(sw189--sw190)	cb138-(sw170--sw190)	cb122-(sw170--sw190)
412	cb139-(sw190--sw191)	cb138-(sw190--sw191)	cb139-(sw171--sw191)	cb123-(sw171--sw191)
413	cb140-(sw191--sw192)	cb139-(sw191--sw192)	cb140-(sw172--sw192)	cb124-(sw172--sw192)
414	cb141-(sw192--sw193)	cb140-(sw192--sw193)	cb141-(sw173--sw193)	cb125-(sw173--sw193)
415	cb142-(sw193--sw194)	cb141-(sw193--sw194)	cb142-(sw174--sw194)	cb126-(sw174--sw194)
416	cb143-(sw194--sw195)	cb142-(sw194--sw195)	cb143-(sw175--sw195)	cb127-(sw175--sw195)
417	cb144-(sw199--sw200)		cb144-(sw180--sw200)	cb128-(sw180--sw200)
418	cb145-(sw200--sw201)	cb144-(sw200--sw201)	cb145-(sw181--sw201)	cb129-(sw181--sw201)
419	cb146-(sw201--sw202)	cb145-(sw201--sw202)	cb146-(sw182--sw202)	cb130-(sw182--sw202)
420	cb147-(sw202--sw203)	cb146-(sw202--sw203)	cb147-(sw183--sw203)	cb131-(sw183--sw203)
421	cb148-(sw203--sw204)	cb147-(sw203--sw204)	cb148-(sw184--sw204)	cb132-(sw184--sw204)
422	cb149-(sw204--sw205)	cb148-(sw204--sw205)	cb149-(sw185--sw205)	cb133-(sw185--sw205)
423	cb150-(sw205--sw206)	cb149-(sw205--sw206)	cb150-(sw186--sw206)	cb134-(sw186--sw206)
424	cb151-(sw206--sw207)	cb150-(sw206--sw207)	cb151-(sw187--sw207)	cb135-(sw187--sw207)
425	cb152-(sw207--sw208)	cb151-(sw207--sw208)	cb152-(sw188--sw208)	cb136-(sw188--sw208)
426	cb153-(sw208--sw209)	cb152-(sw208--sw209)	cb153-(sw189--sw209)	cb137-(sw189--sw209)
427	cb154-(sw209--sw210)	cb153-(sw209--sw210)	cb154-(sw190--sw210)	cb138-(sw190--sw210)
428	cb155-(sw210--sw211)	cb154-(sw210--sw211)	cb155-(sw191--sw211)	cb139-(sw191--sw211)
429	cb156-(sw211--sw212)	cb155-(sw211--sw212)	cb156-(sw192--sw212)	cb140-(sw192--sw212)
430	cb157-(sw212--sw213)	cb156-(sw212--sw213)	cb157-(sw193--sw213)	cb141-(sw193--sw213)
431	cb158-(sw213--sw214)	cb157-(sw213--sw214)	cb158-(sw194--sw214)	cb142-(sw194--sw214)
432	cb159-(sw214--sw215)	cb158-(sw214--sw215)	cb159-(sw195--sw215)	cb143-(sw195--sw215)
433	cb160-(sw219--sw220)		cb160-(sw200--sw220)	cb144-(sw200--sw220)
434	cb161-(sw220--sw221)	cb160-(sw220--sw221)	cb161-(sw201--sw221)	cb145-(sw201--sw221)
435	cb162-(sw221--sw222)	cb161-(sw221--sw222)	cb162-(sw202--sw222)	cb146-(sw202--sw222)
436	cb163-(sw222--sw223)	cb162-(sw222--sw223)	cb163-(sw203--sw223)	cb147-(sw203--sw223)
437	cb164-(sw223--sw224)	cb163-(sw223--sw224)	cb164-(sw204--sw224)	cb148-(sw204--sw224)
438	cb165-(sw224--sw225)	cb164-(sw224--sw225)	cb165-(sw205--sw225)	cb149-(sw205--sw225)
439	cb166-(sw225--sw226)	cb165-(sw225--sw226)	cb166-(sw206--sw226)	cb150-(sw206--sw226)

440	cb167-(sw226--sw227)	cb166-(sw226--sw227)	cb167-(sw207--sw227)	cb151-(sw207--sw227)
441	cb168-(sw227--sw228)	cb167-(sw227--sw228)	cb168-(sw208--sw228)	cb152-(sw208--sw228)
442	cb169-(sw228--sw229)	cb168-(sw228--sw229)	cb169-(sw209--sw229)	cb153-(sw209--sw229)
443	cb170-(sw229--sw230)	cb169-(sw229--sw230)	cb170-(sw210--sw230)	cb154-(sw210--sw230)
444	cb171-(sw230--sw231)	cb170-(sw230--sw231)	cb171-(sw211--sw231)	cb155-(sw211--sw231)
445	cb172-(sw231--sw232)	cb171-(sw231--sw232)	cb172-(sw212--sw232)	cb156-(sw212--sw232)
446	cb173-(sw232--sw233)	cb172-(sw232--sw233)	cb173-(sw213--sw233)	cb157-(sw213--sw233)
447	cb174-(sw233--sw234)	cb173-(sw233--sw234)	cb174-(sw214--sw234)	cb158-(sw214--sw234)
448	cb175-(sw234--sw235)	cb174-(sw234--sw235)	cb175-(sw215--sw235)	cb159-(sw215--sw235)
449	cb176-(sw239--sw240)		cb176-(sw220--sw240)	cb160-(sw220--sw240)
450	cb177-(sw240--sw241)	cb176-(sw240--sw241)	cb177-(sw221--sw241)	cb161-(sw221--sw241)
451	cb178-(sw241--sw242)	cb177-(sw241--sw242)	cb178-(sw222--sw242)	cb162-(sw222--sw242)
452	cb179-(sw242--sw243)	cb178-(sw242--sw243)	cb179-(sw223--sw243)	cb163-(sw223--sw243)
453	cb180-(sw243--sw244)	cb179-(sw243--sw244)	cb180-(sw224--sw244)	cb164-(sw224--sw244)
454	cb181-(sw244--sw245)	cb180-(sw244--sw245)	cb181-(sw225--sw245)	cb165-(sw225--sw245)
455	cb182-(sw245--sw246)	cb181-(sw245--sw246)	cb182-(sw226--sw246)	cb166-(sw226--sw246)
456	cb183-(sw246--sw247)	cb182-(sw246--sw247)	cb183-(sw227--sw247)	cb167-(sw227--sw247)
457	cb184-(sw247--sw248)	cb183-(sw247--sw248)	cb184-(sw228--sw248)	cb168-(sw228--sw248)
458	cb185-(sw248--sw249)	cb184-(sw248--sw249)	cb185-(sw229--sw249)	cb169-(sw229--sw249)
459	cb186-(sw249--sw250)	cb185-(sw249--sw250)	cb186-(sw230--sw250)	cb170-(sw230--sw250)
460	cb187-(sw250--sw251)	cb186-(sw250--sw251)	cb187-(sw231--sw251)	cb171-(sw231--sw251)
461	cb188-(sw251--sw252)	cb187-(sw251--sw252)	cb188-(sw232--sw252)	cb172-(sw232--sw252)
462	cb189-(sw252--sw253)	cb188-(sw252--sw253)	cb189-(sw233--sw253)	cb173-(sw233--sw253)
463	cb190-(sw253--sw254)	cb189-(sw253--sw254)	cb190-(sw234--sw254)	cb174-(sw234--sw254)
464	cb191-(sw254--sw255)	cb190-(sw254--sw255)	cb191-(sw235--sw255)	cb175-(sw235--sw255)
465	cb192-(sw259--sw260)		cb192-(sw240--sw260)	cb176-(sw240--sw260)
466	cb193-(sw260--sw261)	cb192-(sw260--sw261)	cb193-(sw241--sw261)	cb177-(sw241--sw261)
467	cb194-(sw261--sw262)	cb193-(sw261--sw262)	cb194-(sw242--sw262)	cb178-(sw242--sw262)
468	cb195-(sw262--sw263)	cb194-(sw262--sw263)	cb195-(sw243--sw263)	cb179-(sw243--sw263)
469	cb196-(sw263--sw264)	cb195-(sw263--sw264)	cb196-(sw244--sw264)	cb180-(sw244--sw264)
470	cb197-(sw264--sw265)	cb196-(sw264--sw265)	cb197-(sw245--sw265)	cb181-(sw245--sw265)
471	cb198-(sw265--sw266)	cb197-(sw265--sw266)	cb198-(sw246--sw266)	cb182-(sw246--sw266)
472	cb199-(sw266--sw267)	cb198-(sw266--sw267)	cb199-(sw247--sw267)	cb183-(sw247--sw267)
473	cb200-(sw267--sw268)	cb199-(sw267--sw268)	cb200-(sw248--sw268)	cb184-(sw248--sw268)
474	cb201-(sw268--sw269)	cb200-(sw268--sw269)	cb201-(sw249--sw269)	cb185-(sw249--sw269)
475	cb202-(sw269--sw270)	cb201-(sw269--sw270)	cb202-(sw250--sw270)	cb186-(sw250--sw270)
476	cb203-(sw270--sw271)	cb202-(sw270--sw271)	cb203-(sw251--sw271)	cb187-(sw251--sw271)
477	cb204-(sw271--sw272)	cb203-(sw271--sw272)	cb204-(sw252--sw272)	cb188-(sw252--sw272)
478	cb205-(sw272--sw273)	cb204-(sw272--sw273)	cb205-(sw253--sw273)	cb189-(sw253--sw273)
479	cb206-(sw273--sw274)	cb205-(sw273--sw274)	cb206-(sw254--sw274)	cb190-(sw254--sw274)
480	cb207-(sw274--sw275)	cb206-(sw274--sw275)	cb207-(sw255--sw275)	cb191-(sw255--sw275)
481	cb208-(sw279--sw280)		cb208-(sw260--sw280)	cb192-(sw260--sw280)
482	cb209-(sw280--sw281)	cb208-(sw280--sw281)	cb209-(sw261--sw281)	cb193-(sw261--sw281)
483	cb210-(sw281--sw282)	cb209-(sw281--sw282)	cb210-(sw262--sw282)	cb194-(sw262--sw282)
484	cb211-(sw282--sw283)	cb210-(sw282--sw283)	cb211-(sw263--sw283)	cb195-(sw263--sw283)
485	cb212-(sw283--sw284)	cb211-(sw283--sw284)	cb212-(sw264--sw284)	cb196-(sw264--sw284)
486	cb213-(sw284--sw285)	cb212-(sw284--sw285)	cb213-(sw265--sw285)	cb197-(sw265--sw285)
487	cb214-(sw285--sw286)	cb213-(sw285--sw286)	cb214-(sw266--sw286)	cb198-(sw266--sw286)
488	cb215-(sw286--sw287)	cb214-(sw286--sw287)	cb215-(sw267--sw287)	cb199-(sw267--sw287)

489	cb216-(sw287--sw288)	cb215-(sw287--sw288)	cb216-(sw268--sw288)	cb200-(sw268--sw288)
490	cb217-(sw288--sw289)	cb216-(sw288--sw289)	cb217-(sw269--sw289)	cb201-(sw269--sw289)
491	cb218-(sw289--sw290)	cb217-(sw289--sw290)	cb218-(sw270--sw290)	cb202-(sw270--sw290)
492	cb219-(sw290--sw291)	cb218-(sw290--sw291)	cb219-(sw271--sw291)	cb203-(sw271--sw291)
493	cb220-(sw291--sw292)	cb219-(sw291--sw292)	cb220-(sw272--sw292)	cb204-(sw272--sw292)
494	cb221-(sw292--sw293)	cb220-(sw292--sw293)	cb221-(sw273--sw293)	cb205-(sw273--sw293)
495	cb222-(sw293--sw294)	cb221-(sw293--sw294)	cb222-(sw274--sw294)	cb206-(sw274--sw294)
496	cb223-(sw294--sw295)	cb222-(sw294--sw295)	cb223-(sw275--sw295)	cb207-(sw275--sw295)
497	cb224-(sw299--sw300)		cb224-(sw280--sw300)	cb208-(sw280--sw300)
498	cb225-(sw300--sw301)	cb224-(sw300--sw301)	cb225-(sw281--sw301)	cb209-(sw281--sw301)
499	cb226-(sw301--sw302)	cb225-(sw301--sw302)	cb226-(sw282--sw302)	cb210-(sw282--sw302)
500	cb227-(sw302--sw303)	cb226-(sw302--sw303)	cb227-(sw283--sw303)	cb211-(sw283--sw303)
501	cb228-(sw303--sw304)	cb227-(sw303--sw304)	cb228-(sw284--sw304)	cb212-(sw284--sw304)
502	cb229-(sw304--sw305)	cb228-(sw304--sw305)	cb229-(sw285--sw305)	cb213-(sw285--sw305)
503	cb230-(sw305--sw306)	cb229-(sw305--sw306)	cb230-(sw286--sw306)	cb214-(sw286--sw306)
504	cb231-(sw306--sw307)	cb230-(sw306--sw307)	cb231-(sw287--sw307)	cb215-(sw287--sw307)
505	cb232-(sw307--sw308)	cb231-(sw307--sw308)	cb232-(sw288--sw308)	cb216-(sw288--sw308)
506	cb233-(sw308--sw309)	cb232-(sw308--sw309)	cb233-(sw289--sw309)	cb217-(sw289--sw309)
507	cb234-(sw309--sw310)	cb233-(sw309--sw310)	cb234-(sw290--sw310)	cb218-(sw290--sw310)
508	cb235-(sw310--sw311)	cb234-(sw310--sw311)	cb235-(sw291--sw311)	cb219-(sw291--sw311)
509	cb236-(sw311--sw312)	cb235-(sw311--sw312)	cb236-(sw292--sw312)	cb220-(sw292--sw312)
510	cb237-(sw312--sw313)	cb236-(sw312--sw313)	cb237-(sw293--sw313)	cb221-(sw293--sw313)
511	cb238-(sw313--sw314)	cb237-(sw313--sw314)	cb238-(sw294--sw314)	cb222-(sw294--sw314)
512	cb239-(sw314--sw315)	cb238-(sw314--sw315)	cb239-(sw295--sw315)	cb223-(sw295--sw315)
513	cb240-(sw319--sw320)		cb240-(sw300--sw320)	cb224-(sw300--sw320)
514	cb241-(sw320--sw321)	cb240-(sw320--sw321)	cb241-(sw301--sw321)	cb225-(sw301--sw321)
515	cb242-(sw321--sw322)	cb241-(sw321--sw322)	cb242-(sw302--sw322)	cb226-(sw302--sw322)
516	cb243-(sw322--sw323)	cb242-(sw322--sw323)	cb243-(sw303--sw323)	cb227-(sw303--sw323)
517	cb244-(sw323--sw324)	cb243-(sw323--sw324)	cb244-(sw304--sw324)	cb228-(sw304--sw324)
518	cb245-(sw324--sw325)	cb244-(sw324--sw325)	cb245-(sw305--sw325)	cb229-(sw305--sw325)
519	cb246-(sw325--sw326)	cb245-(sw325--sw326)	cb246-(sw306--sw326)	cb230-(sw306--sw326)
520	cb247-(sw326--sw327)	cb246-(sw326--sw327)	cb247-(sw307--sw327)	cb231-(sw307--sw327)
521	cb248-(sw327--sw328)	cb247-(sw327--sw328)	cb248-(sw308--sw328)	cb232-(sw308--sw328)
522	cb249-(sw328--sw329)	cb248-(sw328--sw329)	cb249-(sw309--sw329)	cb233-(sw309--sw329)
523	cb250-(sw329--sw330)	cb249-(sw329--sw330)	cb250-(sw310--sw330)	cb234-(sw310--sw330)
524	cb251-(sw330--sw331)	cb250-(sw330--sw331)	cb251-(sw311--sw331)	cb235-(sw311--sw331)
525	cb252-(sw331--sw332)	cb251-(sw331--sw332)	cb252-(sw312--sw332)	cb236-(sw312--sw332)
526	cb253-(sw332--sw333)	cb252-(sw332--sw333)	cb253-(sw313--sw333)	cb237-(sw313--sw333)
527	cb254-(sw333--sw334)	cb253-(sw333--sw334)	cb254-(sw314--sw334)	cb238-(sw314--sw334)
528	cb255-(sw334--sw335)	cb254-(sw334--sw335)	cb255-(sw315--sw335)	cb239-(sw315--sw335)
529		cb255-(sw335--sw336)		cb240-(sw320--sw340)
530		cb15-(sw35--sw36)		cb241-(sw321--sw341)
531		cb31-(sw55--sw56)		cb242-(sw322--sw342)
532		cb47-(sw75--sw76)		cb243-(sw323--sw343)
533		cb63-(sw95--sw96)		cb244-(sw324--sw344)
534		cb79-(sw115--sw116)		cb245-(sw325--sw345)
535		cb95-(sw135--sw136)		cb246-(sw326--sw346)
536		cb111-(sw155--sw156)		cb247-(sw327--sw347)
537		cb127-(sw175--sw176)		cb248-(sw328--sw348)

538		cb143-(sw195--sw196)		cb249-(sw329--sw349)
539		cb159-(sw215--sw216)		cb250-(sw330--sw350)
540		cb175-(sw235--sw236)		cb251-(sw331--sw351)
541		cb191-(sw255--sw256)		cb252-(sw332--sw352)
542		cb207-(sw275--sw276)		cb253-(sw333--sw353)
543		cb223-(sw295--sw296)		cb254-(sw334--sw354)
544		cb239-(sw315--sw316)		cb255-(sw335--sw355)

Figure A.7 Quarter-Pixel Interpolation and Search Flow for 16x16 Block Size