논문 2008-45SP-5-10

H.264/AVC의 인트라 예측 병렬 파이프라인 실행 알고리즘

(A Parallel Pipeline Execution Algorithm for H.264/AVC Intra Prediction)

허 가 열*, 조 효 문**, 조 상 복***

(Jia-Yue Xu, Hyo-Moon Cho, and Sang-Bock Cho)

요 약

H.264/AVC는 ITU-T와 ISO/IEC 표준화 단체에서 개발한 차세대 국제 영상압축 표준규격으로 이는 H.261, H.263, MPEG-4 등에 비해 더 좋은 압축 효율을 제공한다. 그러나 전체 인트라 모드에 대해 검색이 수행되므로 연산복잡성이 더욱 증가하는 문제와 하드웨어 자원의 낭비가 발생한다. 따라서 본 논문은 두 개의 프로세서 유닛 기반의 병렬 파이프라인 구조로 표준 모델에 비해 연산 복잡 도를 67% 감소시켰고, 부호화 순서를 병렬 파이프라인 구조에 적합하도록 변화시켜 기존 병렬구조에 비해 하드웨어 자원 낭비를 3% 감소시켰다.

Abstract

H.264/AVC is the newest international video coding standard developed by the joint ITU-T and ISO/IEC standards organizations. This newest video coding standard offers much higher coding efficiency than the H.261, H.263 and MPEG-4. But it has high computing complexity and high H/W resources wasting problem. This paper described the two unit parallel pipeline structure. This new structure comparing with standard model decreased the computing complexity of 67% and the H/W resources waste of 3%.

Keywords : H.264/AVC, intra mode, parallel, pipeline structure

I. Introduction

H.264/AVC the newest international video coding standard is developed by the joint ITU-T and ISO/IEC. H.264/AVC used many new technologies such as the intra spatial prediction, the variable block motion compensation, the multiple reference frames and so on. By this reason, the H.264/AVC offers much higher coding efficiency than the H.261 and H.263[1]. Comparing with other codec, H264/AVC saves the average bitrate up to 50% and has significantly better performance in terms of PSNR.

In H.264/AVC, the MD(mode decision) operation is performed to increase compression rate by finding minimum RDcost(Rate Distortion cost) of all possible modes.

Nowadays many algorithms have been proposed to reduce the complexity such as Pan et al[3]. Pan's algorithms reduced the computing complexity of intra MD by using the edge detection histogram and the local edge detection. However, it needed the additional coding time to calculate the edge direction information.

* 학생회원, ** 정회원, 울산대학교 전기전자정보시스템 공학부
(Department of electrical engineering, University of ulsan)
*** 평생회원, 울산대학교 전기전자정보시스템 공학부
(Department of electrical engineering, University of ulsan)

※ 이 논문은 2006년 울산대학교 교내연구비의 지원을 받은 연구이며, 칩제작은 IDEC의 지원을 받았음.
접수일자: 2008년5월6일, 수정완료일: 2008년8월5일

The hardware waste problems are occurred when the intra predictions and reconstructions are performed for 4x4 blocks serialized[7]. To solve this problem, many algorithms adopted the changing of encoding order. However, there are two prediction modes not used in some blocks.

In this paper, we proposed the parallel pipelined execution algorithm for H.264/AVC intra prediction. The proposed algorithm combined above two methods and dramatically reduced a processing time of 67% compared with standard model.

The rest of this paper is organized as follows: Section II described the intra mode decision of H.264/AVC. In section III, we proposed an intra prediction algorithm in detail. In section IV, the analysis results are described. Finally, conclusion is described in section V.

II. Intra 4x4 prediction of H.264/AVC

1. RDO(Rate-distortion optimization) technique for an intra prediction

In H.264/AVC, the RDO technique used to select the best mode with the smallest RDcost for all macroblocks. But this technique costs high computational complexity. The Lagrangian function of RDcost for the best intra mode is defined as follows:

$$J(s,c,MODE\mid QP,\lambda_{MODE}) = SSD(s,c,MODE\mid QP) + \lambda_{MODE}\cdot R(s,c,MODE\mid QP) \qquad (1)$$

Where QP is the macroblock quantization parameter, $\lambda_{MODE} = 0.85\times 2^{(QP-12)/3}$ the Lagrange multiplier for mode decision, SSD the sum of squared difference between the original signal s and its reconstruction c, and $R(s,c,MODE|QP)$ the number of bits associated with the chosen mode[2]. And the SSD is defined as follow:

$$SSD(s,c,MODE/QP) = \sum_{x=1,y=1}^{4,4}[s(x.y)-c(x,y,MODE/QP)]^2 \qquad (2)$$

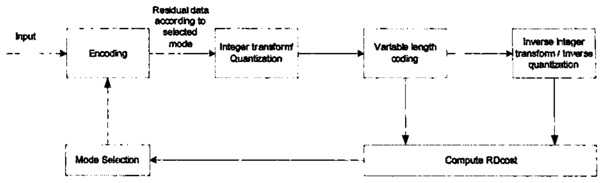

Input
Encoding
Residual data according to selected mode
Integer transform/ Quantization
Variable length coding
Inverse Integer transform / inverse quantization
Mode Selection
Compute RDcost

그림 1. RDcost의 계산
Fig. 1. Computation of RDcost.

Figure 1 shows the RDO process, the RDO technique requires a lot of computations because it calculates all possible coding modes of intra to select the minimum cost.

Therefore, the number of mode prediction for luminance and chrominance of an MB was defined as M8x(M4x16+M16) where the M8, M4 and M16 is the number of prediction modes. For an MB, the RDO calculation number is 4x(9x16+4)=592 before decide a best mode. It needs a high computation[3].

2. 4x4 intra prediction mode

There are nine prediction modes in the each 4x4 block show in figure 2. The values of each block of luminance samples are predicted from the neighboring pixels above or left of a 4x4 block. The pixels from samples a to p are predicted from samples A-L and M. The eight directions mode and a DC mode are defined as below:

Mode 0: Vertical Prediction
Mode 1: Horizontal Prediction
Mode 2: DC Prediction
Mode 3: Diagonal down left Prediction
Mode 4: Diagonal down right Prediction
Mode 5: Vertical right Prediction
Mode 6: Horizontal down Prediction
Mode 7: Vertical left Prediction
Mode 8: Horizontal up Prediction

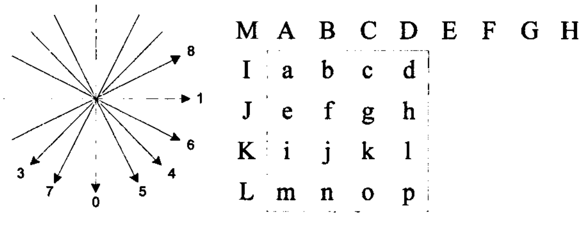


그림 2. 4x4 블록의 인트라 예측
Fig. 2. Intra prediction of 4x4 block.

3. Previous intra prediction methods

Currently, in order to reduce the computational complexity of RDcost for intra mode, many methods have been proposed during these years. There are two approaches in generally. One approach is to reduce the candidate modes, and another approach is simplification of cost function.

For the first approach, the representative approach is a fast mode decision algorithm. It uses the current MB and the feature pixel of neighbor to exclude some modes which have small possibility for prediction modes. As we known, Pan et al proposed a fast intra mode decision algorithms based on edge detection histogram and local edge detection. Although it can save much calculate time and with a negligible loss of PSNR. However, Pan's algorithms still need a coding time to detect the edge direction and to classify it into a limited direction.

The second approach is efficient hardware architecture[8~10]. To reduce the calculated time, some methods were proposed to improve the hardware architecture efficiently. Such as [4], there is an interleaved intra prediction that divided the 16x16 intra prediction into sixteen parts when the corresponding 4x4 image is reconstructed. But it need to be stored in the register and the cycles are also not accurate as the intra 4x4 predication reconstruction cycles, so the hardware computational complexity would be increased. And some methods change the encoding order but there were not compatible with the standard. In Wonjae Lee et al[5], a novel pipelined intra prediction is proposed for intra 4x4 prediction and it is compatible with the standard. However, it need to exhaustively searching all possible modes, this made the computational complexity increased.

Ⅲ. Proposed intra prediction algorithm

In this paper, we proposed an efficient parallel execution algorithm for H.264/AVC intra prediction. Through this method, the processing time can be reduced by changing the encoding order. We do not exclude some modes for pipelined processing and this changed encoding order is also compatible with the standard.

1. Decision of the intra mode block size

In this paper, the pipelined method is based on the intra sixteen 4x4 blocks. Because of the encoding order is for sixteen 4x4 blocks in a macroblock show in figure 3. At first, we decide the intra prediction block which is 4x4 block.

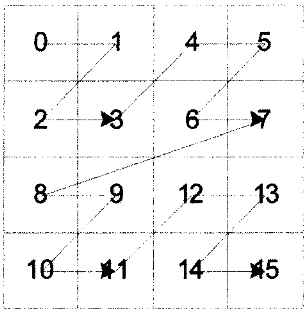


그림 3. MB 내 16개의 4x4 블록의 부호화 순서
Fig. 3. Encoding order of 16 4x4 blocks in one MB.

In order to decide the block size we must calculate all modes in H.264/AVC, it takes high computational complexity. The sixteen 4x4 blocks are encoded for more details with 4x4 intra prediction; and the 16x16 intra prediction for the 16x16 block is applied for region with less spatial detail. If the region is not smoothness, we will use the 4x4 intra mode. In this paper, we determine a discrepancy of a macroblock to scale the block whether 4x4 or not. The expressions as below:

$$\mathit{Discrepanc}\ \ y = \sqrt{\sum_{x=0}^{15}\sum_{y=0}^{15}\left[L_{xy} - \sum_{x=0}^{15}\sum_{y=0}^{15}(L_{xy})/256\right]^2} \qquad (3)$$

In expression (3), L_{xy} denotes a pixel of luma in a macroblock. But this expression has squared root that will makes the calculation become complexity. So we use the expression (4) instead of (3) to avoid the square.

$$D = \sum_{x=0}^{15}\sum_{y=0}^{15}\left[L_{xy} - \sum_{x=0}^{15}\sum_{y=0}^{15}(L_{xy})/256\right]^2 \qquad (4)$$

This discrepancy of a macroblock can describe the difference of block. In order to select the block size is 4x4 or 16x16, we adopt this discrepancy to decide the threshold value[6].

2. Fast mode decision for 4x4 intra prediction

In order to reduce the processing time, we select the mode before encoding process, so that the reconstruction process of the previous 4x4 block need not to calculate all modes. Because it has to perform 4x(9x16+4)=592 different RDO calculations before selected a best RDO mode.

In this paper, we improve Pan's algorithm. Since Pan's algorithm is a fast intra mode decision based on edge detection histogram and local edge detection. Therefore, it can decrease the number of candidate modes and the RDO calculations form 592 to 132 or 198. But it also take high complexity, so in this paper, our improved algorithm only need 100 for RDO calculation. For 4x4 intra mode, it just needs 3 modes instead of 9. It is shown in table 1.

In Pan's algorithm, they use the Sobel edge operators to obtain the edge information of each pixel. Analyzing the local edge information then constitute an edge directional histogram which was distributed of each pixel in a block. The edge direction histogram of a 4x4 luma block is decided by (5)

$$a_0 = (-103.3°,-76.7°];\ \ a_1 = (-13.3°,13.3°]$$
$$a_3 = (35.8°,54.2°];\ \ a_4 = (-54.2°,-35.8°]$$
$$a_5 = (-76.6°,-54.2°];a_6 = (-35.8°,-13.3°]$$
$$a_7 = (54.2°,-76.7°];\ \ \ a_8 = (13.3°,35.8°] \qquad (5)$$

It can be made of edge directional histogram about 8 directional prediction modes. For 4x4 MB, there are 4 modes that one is the best candidate of edge directional histogram and its two neighbors plus DC mode. Because of DC mode would be chosen as one of the candidate all the time. Using this method, we can perform 4 modes RDO calculation instead of 9 for 4x4 MB. In this paper, only performs 3 modes instead of 9 modes for 4x4 MB.

표 1. 후보 모드의 수
Table 1. Number of candidate modes.

Block size	Total No. of modes	Pan's algorithm	Proposed algorithm
4x4	9	4	3
16x16	4	2	2
8x8	4	3 or 2	1 or 2

표 2. 4x4 휘도 블록모드에 대한 각도결정
Table 2. The angles for modes of 4x4 luma block.

Mode 1 & 6	$-26.6° < \theta \le 0$
Mode 1 & 8	$0 < \theta \le 26.6°$
Mode 4 & 6	$-45° < \theta \le -26.6°$
Mode 4 & 5	$-63.4° < \theta \le -45°$
Mode 0 & 5	$-90° < \theta \le -63.4°$
Mode 0 & 7	$63.4° < \theta < 90°$
Mode 3 & 7	$45° < \theta \le 63.4°$
Mode 3 & 8	$26.6° < \theta \le 35.8°$

The fast intra mode decision algorithm for 4x4 MB the table was shown as follows:

Form the above table, if the amplitude is in one of these angles we just chose two modes instead of 3 modes. For example, if the angles belong to $-45° < \theta \le -26.6°$, we will decide mode 4 and mode 6 for this pixel and plus DC mode.

3. Parallel pipelined execution algorithm

Recently, the efficient hardware architectures were proposed for the fast intra prediction of H.246/AVC. In this paper, we propose pipelined intra prediction method that can reduce much more processing time compare with the other pipelined intra prediction method. The encoding order has been changed, but it compatibles with the standard. Because of the changed order is based on the decoding process of standard.

Figure 5 shows the new encoding order which has been changed and the steps will be shown in the figure 4.

In order to increase the hardware utilization we proposed a parallel method. As shown in figure 3, the encoding steps have two pipelined processing in parallel execution of intra prediction. One is 0, 1, 2, 3, 8, 9, 10, 11 and the other is 0, 1, 4, 5, 6, 7, 12, 13, 14, 15. From the flow chart, we can gain the total 10

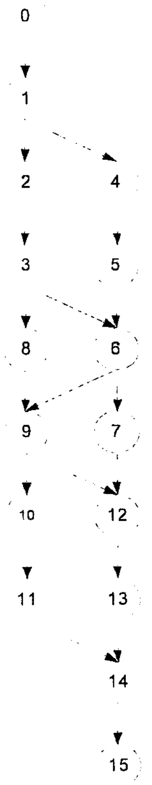


그림 4. 부호화 처리 순서
Fig. 4. Flow steps of encoding processing.

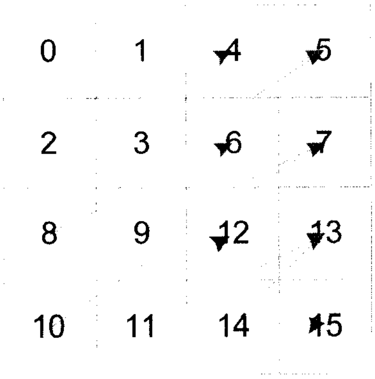


그림 5. 16 4x4 블록의 순서 변경
Fig. 5. Changed order of 16 4x4 block.

steps. Because this is a parallel pipelined method, the left step of 8 steps is less than the right 10 pipelined step, we just need to calculate the 10 steps instead of the all steps. But in this proposed method there are 3 blocks also need to wait for the reconstruction results of the previous blocks. The three blocks are block 1, block 14 and block 15 and they have to wait for previous block 0, block 13 and block 14, respectively. Form the figure 2, we can obtain block 1 and block 15. The two blocks need to get the left pixel of I to L, and block 14 need to get the upper right pixel of E to H. Figure 6 shows the parallel pipelined encoding process of intra 4x4 prediction. Therefore the total required processing time is defended as below:

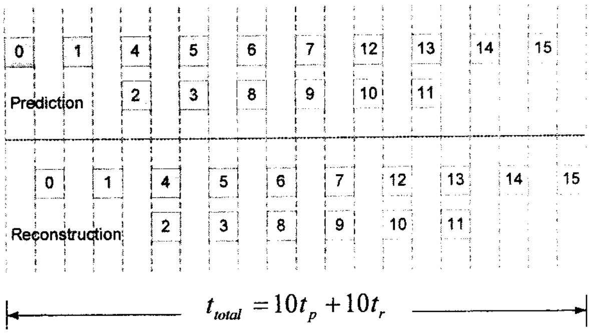


그림 6. 인트라 4x4 예측의 병렬 파이프라인 처리
Fig. 6. Parallel pipelined encoding process of intra 4x4 prediction.

표 3. 16 4x4 블록의 참조 블록
Table 3. Reference blocks for 16 4x4 blocks.

4x4block Index	Reference 4x4block index	4x4block Index	Reference 4x4block index
0	None	8	2,3
1	0	9	2,3,6,8
2	0,1	10	8,9
3	0,1,2	11	8,9,10
4	1	12	3,6,7,9
5	4	13	6,7,12
6	1,3,4,5	14	9,11,12,13
7	4,5,6	15	12,13,14

$$t_{total} = 10t_p + 10t_r \qquad (6)$$

From table 3, the required pixels for each intra 4x4 prediction modes, we divide the three blocks which are block 1, block 14 and block 15 into two parts. One part is block 1 and block 15, which has to wait for the left pixels of reconstruction block predict the mode 1, 2, 4, 5, 6 and 8. And another part is block 14 which has to wait for the upper right pixels of reconstruction block predict the mode 3 and 7. Figure 7 shows the proposed parallel pipelined process of 4x4 intra prediction with prediction mode scheduling. Therefore the proposed total required processing time is defined as below:

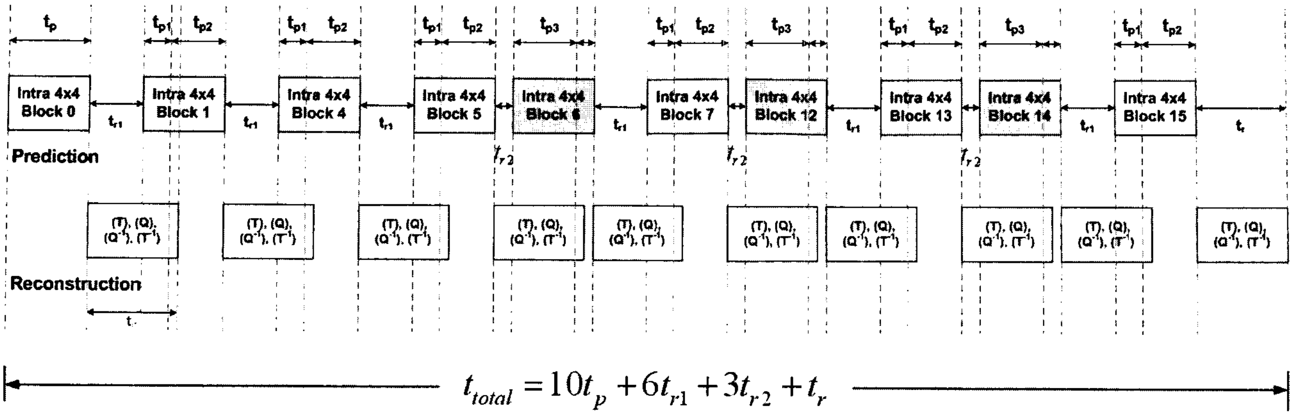


그림 7. 제안한 인트라 4x4 예측의 평행한 파이프라인 실행 방법

Fig. 7. Proposed parallel and pipelined process of 4x4 intra prediction with prediction mode scheduling.

$$t_{total} = 10t_p + 6t_r' + 3t_r'' + t_r \tag{7}$$

$$\text{Where} \begin{cases} t_r' = t_r - t_{p1} & t_r > t_{p1} \\ t_r'' = t_r - t_{p3} & t_r > t_{p3} \end{cases} \tag{8}$$

Here, t_r denotes the reduced time for block 1, 4, 5, 7, 13 and 15, and t_r denotes the reduced time for block 6, 12 and 14.

Ⅳ. Analysis results

In this paper, we proposed a parallel intra prediction algorithm. It can reduce the processing time about 20% to 30% compare with the pipelined intra prediction algorithm in [9]. Because a fast mode decision method is used that can save the processing time about 25% to 35%.

Analyze the perdition time (tp) and the reconstruction time as below:

$$t_p = A \quad t_r \tag{9}$$

Where the parameter A denotes the relationship between t_p and t_r. Using this order, the total standard processing time can be defined as below:

$$t_{standard} = 16(A+1)t_r \tag{10}$$

And the proposed total processing time also can be defined as follow.

$$t_{proposed} = \left(A\frac{17}{3} + 10\right)t_r \tag{11}$$

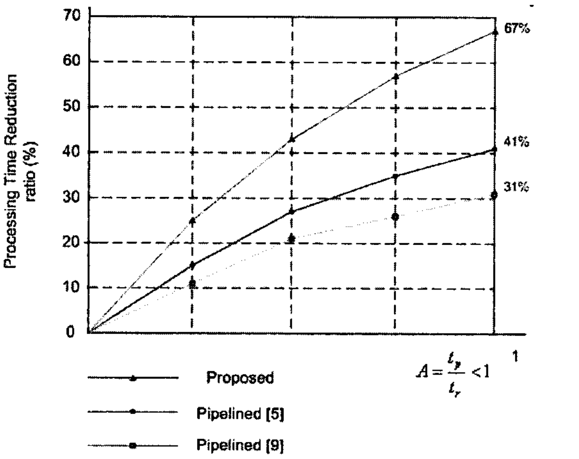


그림 8. 제안 처리 시간의 비교

Fig. 8. Processing time reduction ratio depending on A.

The reduction ratio $r_{proposed}$ between $t_{standard}$ and $t_{proposed}$ can be defined as below:

$$r_{proposed} = \frac{t_{standard} - t_{proposed}}{t_{proposed}} \tag{12}$$

Put the expressions (11) and (12) into expressions (13), the $r_{proposed}$ can be account as follow:

$$r_{proposed} = \frac{31A + 18}{17A + 30} \qquad A < 1 \tag{13}$$

Ⅴ. Conclusion

In this paper, a parallel pipeline execution algorithm for H.264/AVC intra prediction was proposed. By using the parallel of two unit and rearranged encoding order, the hardware utilization was increased and the processing time of encoding could be reduced about 67% compared with the standard process.

REFERENCES

[1] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, No. 7, July 2003.

[2] Mehdi Jafari, Shohreh Kasaei, "Fast Intra-and Inter-Prediction Mode decision in H.264 Advanced Video Coding," Communication systems, 2006. ICCS 2006. 10th IEEE Singapore International Conference. Oct. 2006.

[3] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision algorithm for intra prediction in H.264/AVC video coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 7, pp. 813-822, July 2005.

[4] Yu-Wen Huang, Bing-Yu Hsieh, Tung-Chien Chen, and L.G. Chen, "Analysis, Fast Algorithm, and VLSI Architecture Design for H.264/AVC Intra Frame Coder," IEEE Trans. Circuit and Systems for Video Technology, vol.15, no.3, pp. 378-401, Mar. 2005.

[5] Wonjae Lee, Seongjoo Lee, Jaeseok Kim, "High Speed Intra Prediction Scheme for H.264/AVC," IEEE Transactions on Consumer Electronics, Vol. 53, no. 4, November 2007.

[6] Cheng Fei, "Fast Intra Mode Selection for H.264 Video Coding," http://www.paper.edu.cn.

[7] Genhua Jin, Hyuk-Jae Lee, "A Parallel and Pipelined Execution of H.264/AVC Intra Prediction," the Sixth IEEE International Conference on Computer and Information Technology, pp. 246-246, Sept. 2006.

[8] Kibum Suh, Seongmo Park, and Hanjin Cho, "An Efficient Hardware Architecture of Intra Prediction and TQ/IQIT Module for H.264 Encoder," ETRI Journal, vol. 27, no. 5, pp. 511-524, Oct. 2005.

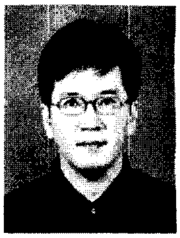
[9] Wonjae Lee, Seongjoo Lee, Jaeseok Kim, "Pipelined intra prediction using shuffled encoding order," TENCON 2006, pp. 1-4, Nov. 2006.

[10] Genhua Jin, Jin-Su Jung, and Hyuk-Jae Lee, "An Efficient Pipelined Architecture for H.264/AVC Intra Frame Processing," IEEE International Symposium on Circuits and Systems(ISCAS), pp. 1605-1608, May. 2007.

저자소개

허 가 열(학생회원)
2006년 동양대학교 전기전자과 공학사 졸업.
2006년 울산대학교 전기전자 공학부 석사과정 입학.
<주관심분야 : 영상처리 회로 설계 및 제작, SoC 설계>

조 상 복(평생회원)-교신저자
1979년 한양대학교 전자공학과 학사 졸업.
1981년 한양대학교 전자공학과 석사 졸업.
1985년 한양대학교 전자공학과 박사 졸업.
1994년~1995년 Univ. of Texas, Austin 교환교수
2003년~2004년 Univ. of California, San Diego 교환교수
2000년~2001년 울산대학교 자동차전자연구센터장
2006년~현재 울산대학교 e-Vehicle 연구 인력양성사업단장 (2단계 BK21 정보기술사업단)
<주관심분야 : SoC/VLSI 설계 및 테스트, 자동차 전장시스템 설계, 영상처리 회로 설계 및 제작, 머신비전 시스템 개발, 초고집적 메모리 설계>



조 효 문(정회원)
1990년 울산대학교 전자공학과 공학사 졸업.
1992년 울산대학교 대학원 전자공학과 공학석사 졸업.
2006년 울산대학교 대학원 전기 전자공학부 박사과정.
<주관심분야 : CMOS VLSI 및 SoC 설계, 영상 압축 및 처리>