  
SVM

3D Surround-View Monitoring Block

PI5008K

Rev. 1.2

Last update : 17. February. 2017

영통구 이의동 경기알앤디비센터 6층

443-270 경기도 수원시

전화 : 82-31-888-5300, 팩스 : 82-31-888-5398

**저작권 © 2017 (주)픽셀플러스**

**All RIGHTS RESERVED**

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision No. | Date | Description | Author(s) |
| 1.00 | Jan. 10, 2017 | * Preliminary | 최한준 |
| 1.10 | Feb. 13. 2017 | Morphing, Edge enhancement 추가 | 최한준 |
| 1.20 | Feb. 17. 2017 | Dynamic blending 추가 | 최한준 |

Contents

[1 Overview 5](#_Toc475519924)

[1.1 Operating timing 7](#_Toc475519925)

[1.2 Memory Bandwidth 7](#_Toc475519926)

[1.2.1 Compression 이용 방안 8](#_Toc475519927)

[1.2.2 입력 data의 구간별 down scaling factor 설정 8](#_Toc475519928)

[1.2.3 카메라 입력 crop 저장 9](#_Toc475519929)

[1.2.4 Write request generation timing control 9](#_Toc475519930)

[1.2.5 Invalid LUT 이용 방안 9](#_Toc475519931)

[1.2.6 Section 영역 설정 9](#_Toc475519932)

[2 Control Signals 10](#_Toc475519933)

[2.1 Control Signals 10](#_Toc475519934)

[2.2 Detail description 17](#_Toc475519935)

[2.2.1 Timing generation mode control registers 17](#_Toc475519936)

[2.2.2 Timing generation resolution control registers 18](#_Toc475519937)

[2.2.3 Write control registers 19](#_Toc475519938)

[2.2.4 Read control registers 21](#_Toc475519939)

[2.2.5 Base address control registers and LUT base address control registers 22](#_Toc475519940)

[2.2.6 LUT base address control registers 22](#_Toc475519941)

[2.2.7 Bypass 및 LUT mode, mirroring, LUT ratio control registers 23](#_Toc475519942)

[2.2.8 LUT information control registers 24](#_Toc475519943)

[2.2.9 Output resolution control registers 27](#_Toc475519944)

[2.2.10 Write request hold time control registers 27](#_Toc475519945)

[2.2.11 3D Car model blend control registers 28](#_Toc475519946)

[2.2.12 Mask color control registers 29](#_Toc475519947)

[2.2.13 Section region setting registers 29](#_Toc475519948)

[2.2.14 Crop region setting registers 30](#_Toc475519949)

[2.2.15 4 Range scale down control registers 32](#_Toc475519950)

[2.2.16 Out mode/Blending/Brightness Control/Edge enhancement setting registers 33](#_Toc475519951)

[2.2.17 Compression control registers 35](#_Toc475519952)

[2.2.18 Memory bandwidth debugging registers 36](#_Toc475519953)

[2.2.19 Brightness control statistics registers 36](#_Toc475519954)

[2.2.20 Coefficient registers for Brightness control 38](#_Toc475519955)

[2.2.21 RGB to YCbCr / YCbCr to RGB coefficient registers 40](#_Toc475519956)

[2.2.22 Dynamic Blending Coefficient 41](#_Toc475519957)

Figure of Contents

[***Figure 1‑1 3D SVM과 2D AVM 비교*** 5](#_Toc475518954)

[*Figure 1‑2 SVM Top Block Diagram* 6](#_Toc475518955)

[*Figure 1‑3 SVM Operation Timing* 7](#_Toc475518956)

[***Figure 1‑4 출력 영상 생성에 필요한 해당 입력 영상 영역 분석*** 8](#_Toc475518957)

[*Figure 2‑1 LUT sampling example* 25](#_Toc475518958)

[*Figure 2‑2 Discontinuity information examples of LUT* 26](#_Toc475518959)

# Overview

본 SVM Block은 기존 AVM과 달리 출력 이미지 전반에 걸친 3D Surround view monitoring 생성을 지원하는 Block로서 4개의 Camera로부터 sync signals과 data를 받아 저장 후 사용자가 지정한 Look-up table에 맞춰 출력 이미지를 생성하여 출력 sync signals에 맞춰 data를 출력하며 지원 해상도는 1280x720 @ 60 fps로 전반적인 목표는 AVM과 동일하나 AVM에서의 차량 주위의 좁은 영역만 출력하는 점과 출력 이미지 중 일부에 대해서만 AVM 영상을 출력하는 점 등과 비교할 때 차량 주위의 보다 넓은 영역에 대한 SVM 출력, 출력 이미지의 일부에서만이 아닌 전체에 걸친 SVM 영상 출력이 가능하도록 한다는 점에서 큰 차이를 갖는다. 이를 지원하기 위해서는 기존 구조로는 불가함에 따라 전반적으로 새로운 구조의 Block이 필요하다.



**Figure 1‑1 3D SVM과 2D AVM 비교**

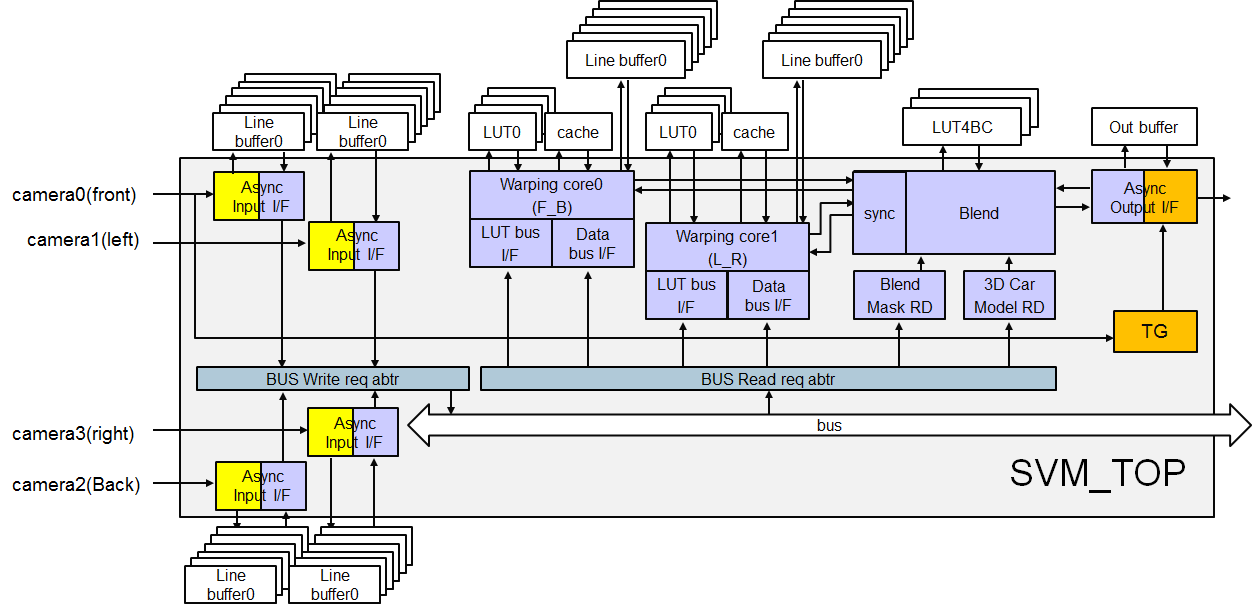


Figure 1‑2 SVM Top Block Diagram

SVM은 입력된 4개의 camera 이미지에 대한 동기화 처리와 함께 두 개의 LUT 따라 두 개의 출력 이미지를 생성하며 이 때 생성되는 두 개의 출력 이미지 중 하나는 front와 back 이미지에 대한 결과이고, 다른 하나는 left와 right 이미지에 대한 결과로서 front와 back 간, left와 right 간에는 겹치는 부분이 없으므로 각각 하나의 warping core를 통해 생성이 가능하다. 이렇게 생성된 두 개의 출력 이미지는 blend block을 통해 하나의 출력 이미지로 생성되며 view position에 맞는 3D car model도 함께 blending되어 출력된다. 기존의 2D AVM에서는 출력 이미지 중 작은 영역에 대해서만 이미지를 혼합하다 보니 입력 이미지 자체를 scale down 처리함으로써 MBW 문제를 해결할 수 있었으나 3D SVM의 경우에는 출력 이미지 전반에 걸쳐 이미지를 혼합해야 하고 더욱이 입력 이미지의 일부는 scale up, 다른 일부는 scale down 처리되어야 하는 등의 문제로 MBW 문제에 대한 해결 방안이 필수적이다.

## Operating timing

SVM는 아래 그림과 같이 tg 블록으로부터 생성된 w\_tg\_pre\_vsync를 기준으로 negedge에서부터 그 동작을 시작하게 되며 미리 5개 라인의 AVM 출력 결과를 생성해놓게 된다. 내부에 준비되어 있는 라인 data의 수는 fill\_cnt라는 신호를 통해 알 수 있으며 이를 통해 MBW 문제 여부를 분석할 수 있도록 하였다.

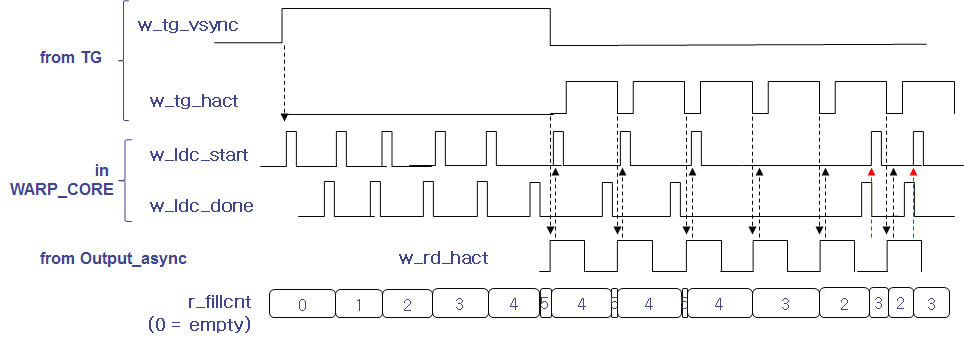


Figure 1‑3 SVM Operation Timing

## Memory Bandwidth

위와 같이 tg\_pre\_vsync가 low로 떨어지는 시점부터 SVM\_EN이 활성화되어 있는 경우 SVM은 동작을 시작하게 되고 미리 5개 라인의 출력 값을 생성해놓는다. 이후 output\_async 블록으로부터 rd\_hact를 받아 미리 생성되어 있는 출력 값을 라인 단위로 전달하고 전달과 동시에 다음 라인의 출력을 생성하기 시작한다. 이 때 한 라인의 SVM 출력을 생성하는데 걸리는 시간이 output\_async의 rd\_hact 신호 간격 대비 더 긴 경우 미리 생성해놓은 라인 버퍼의 data를 이어서 전달하면서 바로 바로 다음 라인의 SVM 출력을 생성해놓게 된다.

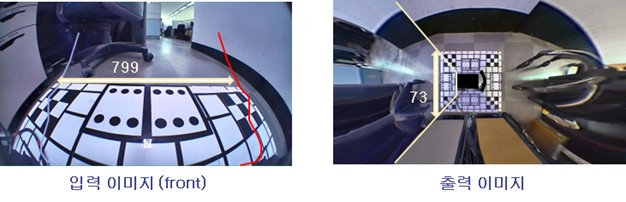
만약 한 라인의 생성 시간이 미리 저장되어 있는 5개의 라인 버퍼 data를 모두 전달할 때까지 완료되지 않는 경우 fill\_cnt 는 0이 되고 이 경우로 MBW가 부족함을 알 수 있게 된다.　이렇게　MBW가　부족한　경우의　해결　방향을　다양한　측면에서　정리해보았다．

### Compression 이용 방안

3D SVM에서 bus write 단에서 image compression을 수행하고 bus read 단에서 decompression 하는 방식으로 실제 bus access 시간을 줄이는 방식이 가능하다. Compression은 1/2 압축 모드와 1/4 압축 모드가 지원된다.

### 입력 data의 구간별 down scaling factor 설정

3D SVM의 경우 아래 그림과 같이 입력 이미지의 해상도가 그대로 사용되는 것이 아니라 일부 영역은 입력 해상도가 그대로 저장되어야 하고 다른 일부 영역은 굉장히 작게 scale down되어 사용된다.



**Figure 1‑4 출력 영상 생성에 필요한 해당 입력 영상 영역 분석**

이렇게 작게 scale down이 되는 부분에서는 cache의 hit ratio가 감소함에 따라 한 라인을 생성하는데 걸리는 시간이 늘어나게 된다. 이러한 문제의 해결과 화질 개선을 위해 입력 이미지를 부분별로 다르게 scale down 처리를 하도록 하고 scale down 전에 전처리 filter를 통과시킴으로써 aliasing 문제에 대해서도 해결하여 화질 개선도 도모하도록 하였다.

SVM Block에서 지원하는 scale down 기능은 1/2 vertical scale down과 4개의 구간을 나누어 구간별로 0, 1/2, 1/4, 1/8 의 설정이 가능한 horizontal scale down을 지원한다.

### 카메라 입력 crop 저장

SVM 출력 영상을 생성하는데 입력 이미지의 일부만 사용되는 경우가 있다. 이러한 경우 사용되지 않는 영역의 이미지는 시스템 메모리에 저장될 필요가 없으므로 실제 저장되는 영역만 설정하여 MBW를 감소시킬 수 있다.

### Write request generation timing control

또한 입력 이미지를 system memory에 write하는 방식은 한 번의 request로 한 라인을 모두 write하는 것이 아니라 burst ctrl 값에서 설정된 단위로 나누어서 여러 번 request하도록 한다. 이는 출력 라인 생성을 위한 read 동작이 write 동작으로 인해 받는 영향을 줄이기 위한 것으로 이렇게 나뉘어서 요청되는 write request 간에도 hold time을 설정할 수 있도록 하여 read 동작에 주는 영향을 최소화할 수 있다.

### Invalid LUT 이용 방안

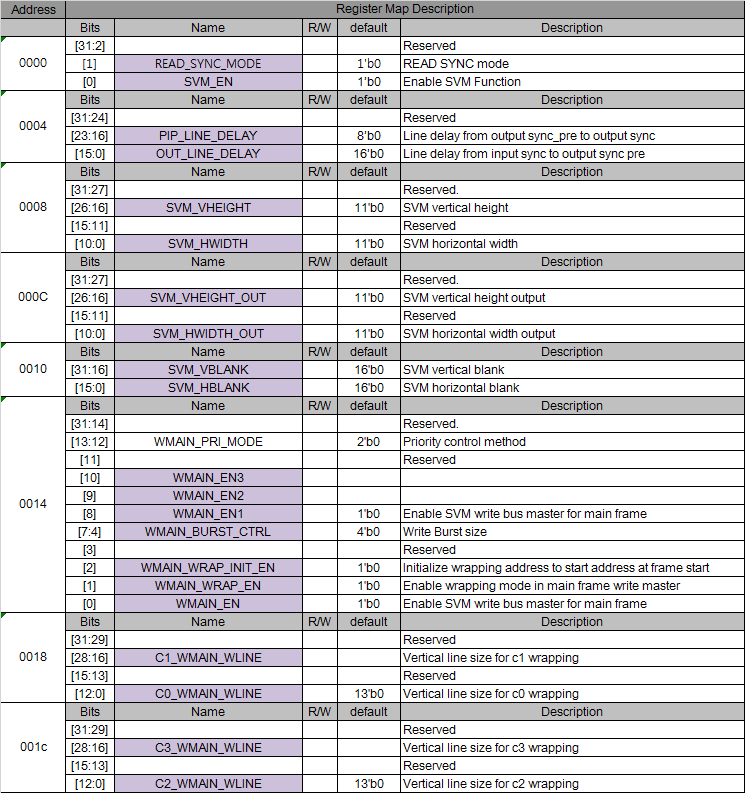
SVM 내의 warping core에서 사용하는 LUT에는 invalid LUT 값을 설정할 수 있고 그 값의 경우에는 system memory로의 read 동작없이 바로 mask color를 출력하도록 하여 MBW를 감소시킨다. 즉, front와 back 이미지에 의해 출력을 생성하는 warping core에서는 left와 right에 해당하는 부분이 mask color로 출력될 것이고 left와 right 이미지에 의해 출력을 생성하는 warping core에서는 front와 back에 해당하는 부분이 mask color로 출력된다.

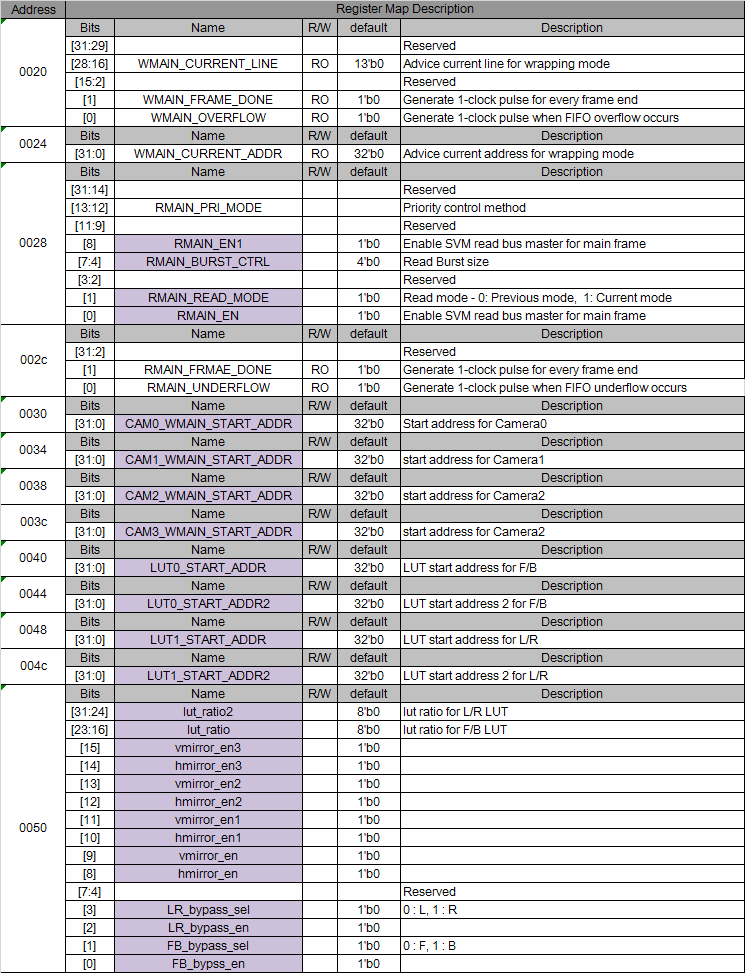
### Section 영역 설정

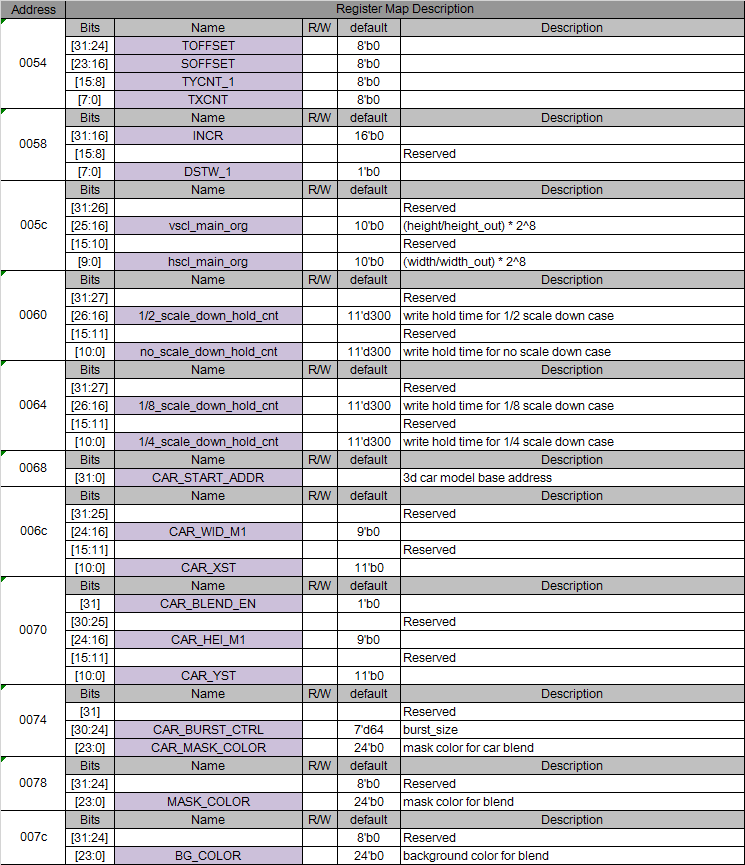
마지막으로 출력 이미지가 생성되는 활성 영역을 설정함에 따라 필요한 MBW를 감소시킬 수 있다.

# Control Signals

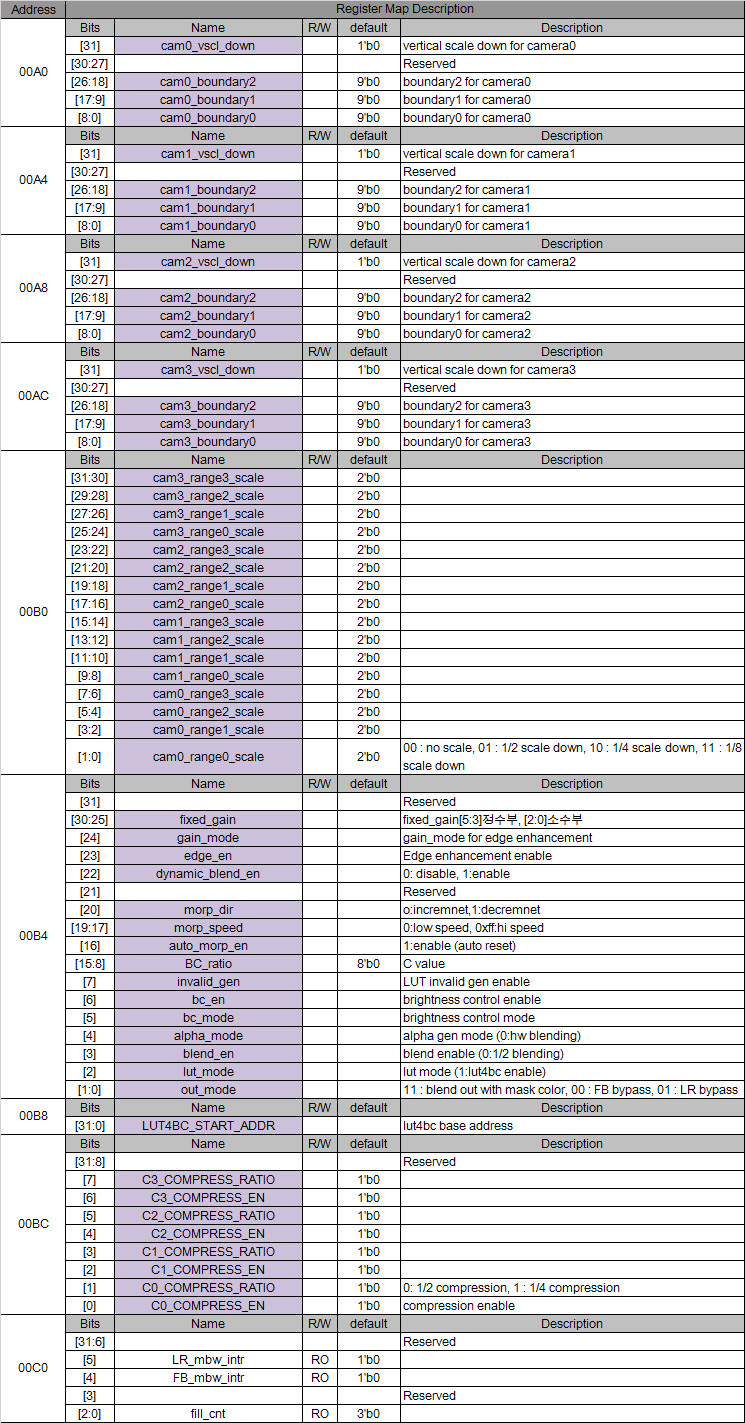
## Control Signals

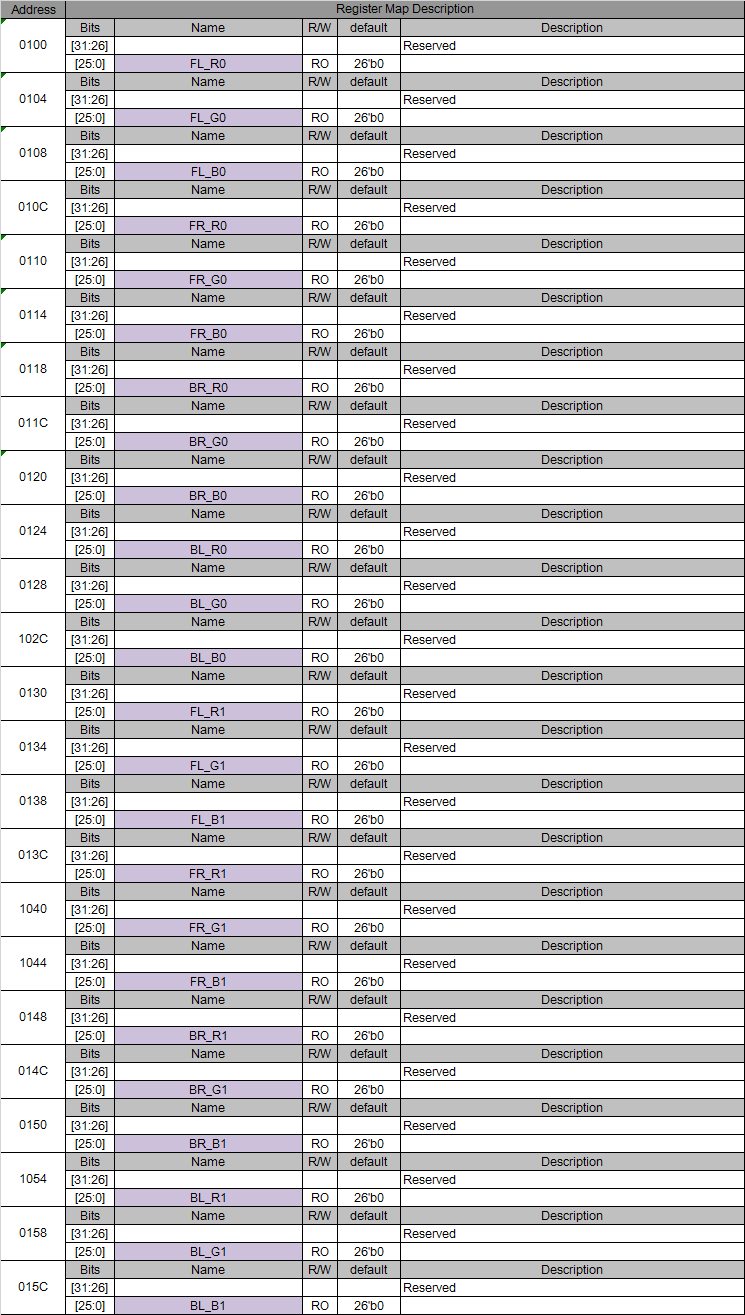


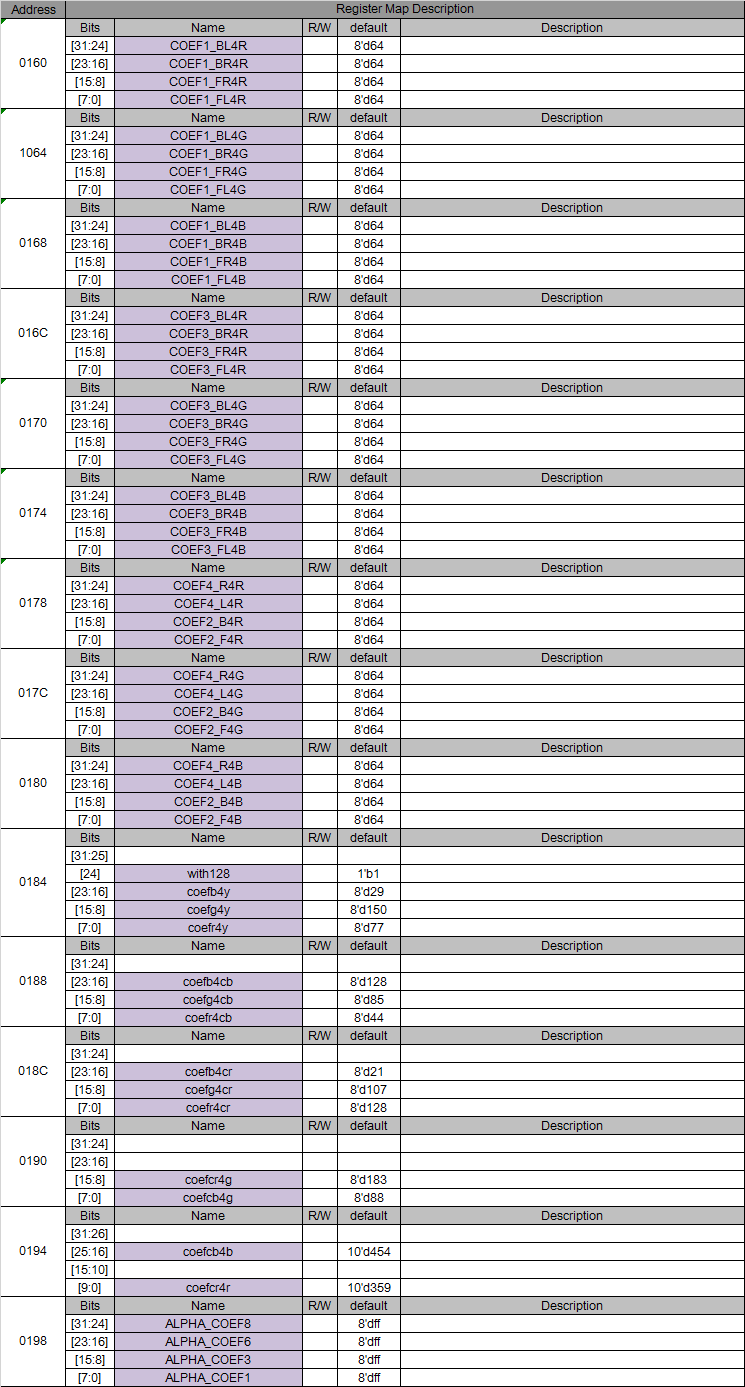








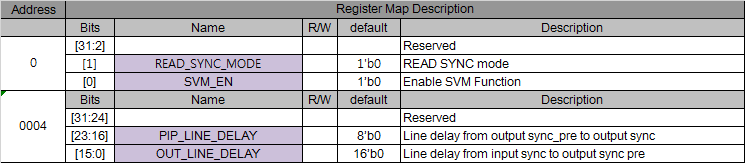




위 Control register 각각의 의미는 다음에서 설명하도록 하겠다.

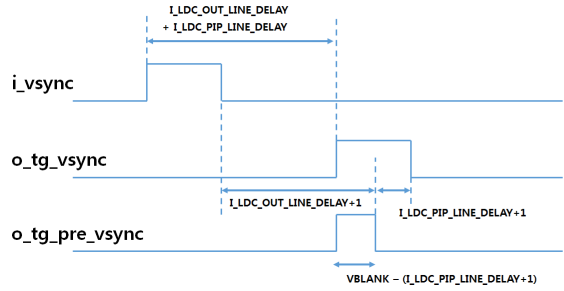
## Detail description

### Timing generation mode control registers

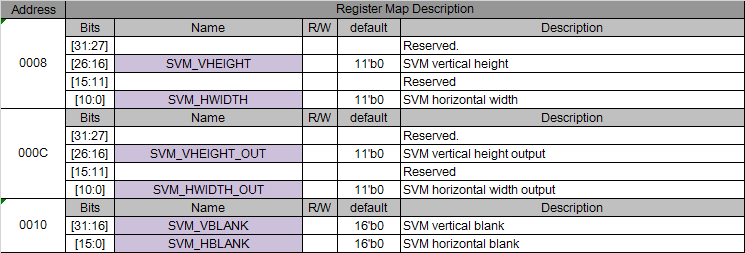


SVM에서는 두 가지 방식의 timing generation을 지원하며 그 한가지는 HDM(Horizontal sync Delay Mode : READ\_SYNC\_MODE = 0) 방식이고, 나머지는 FRC(Frame Rate Control : READ\_SYNC\_MODE = 1) 방식이다. HDM 방식에서는 입력되는 hsync를 delay 시키는 방식으로 출력 timing signal을 생성하게 되고 FRC 방식에서는 설정된 값(SVM\_HWIDTH\_OUT, SVM\_VHEIGHT\_OUT과 SVM\_HBLANK, SVM\_VBLANK)에 따라 입력과 별도 timing의 signal을 생성하게 된다.

출력 vsync의 생성은 입력 vsync negedge를 기준으로 OUT\_LINE\_DELAY 만큼의 line delay를 가지고 생성되고 내부 동작 시작에 필요한 pre\_vsync 생성을 위해 PIP\_LINE\_DELAY 신호가 사용된다.



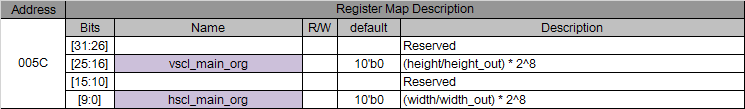
### Timing generation resolution control registers



SVM의 입력 해상도는 SVM\_HWDITH, SVM\_VHEIGHT로 설정되고 출력 해상도는 SVM\_HWIDTH\_OUT, SVM\_VHEIGHT\_OUT으로 설정된다. Frame rate 생성을 위한 horizontal, vertical blank 개수도 SVM\_HBLANK, SVM\_VBLANK로 설정된다.

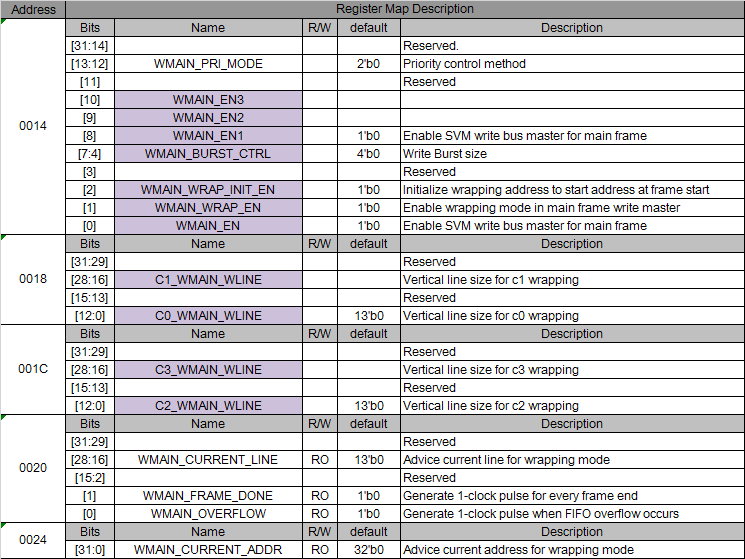
SVM 내 Timing generation 블록에서는 SVM\_HWIDTH\_OUT, SVM\_VHEIGHT\_OUT과 SVM\_HBLANK, SVM\_VBLANK에 맞춰 출력 sync 신호를 생성하게 된다.

그리고 LUT는 입력 해상도에 맞춰 생성된 상태에서 출력 해상도가 입력 해상도와 다른 경우는 다음의 control registers도 설정되어야 한다.

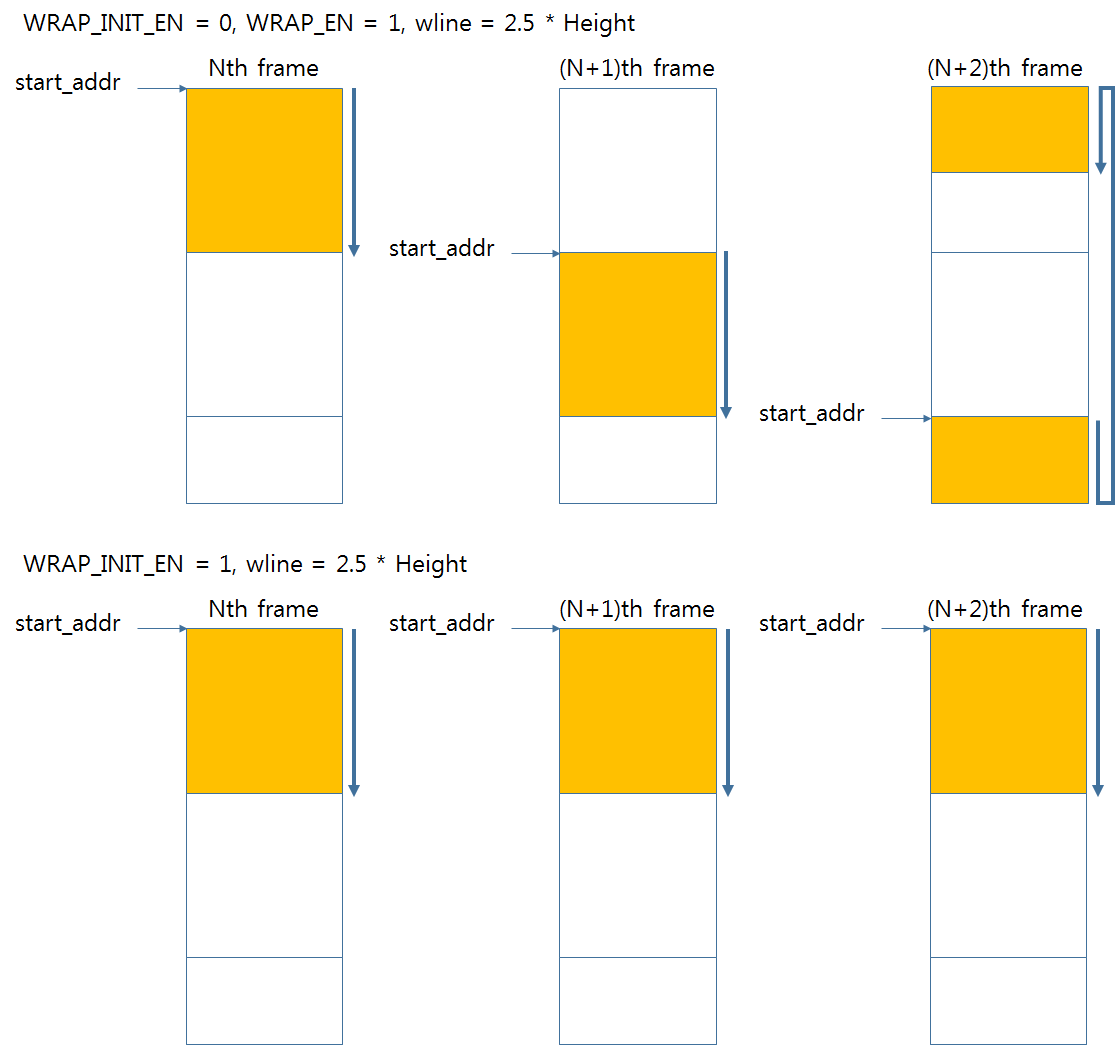


이를 통해 입력 해상도에 맞춰 생성된 LUT도 다른 해상도의 출력이 가능하다. 즉, 1280x720의 입력에 대해 720x1280의 출력 등이 가능하다. 다만 이 경우에는 변경된 해상도에 따라 새로운 sync signals의 생성이 필요하므로 READ\_SYNC\_MODE = 1로 FRC mode이어야만 한다.

### Write control registers



위 control registers는 camera 입력 신호를 system memory에 write하기위한 설정 값으로 WMAIN\_EN = 1에서부터 write하게 되고 (WMAIN\_EN 외에 WMAIN\_EN1~3을 두어 각 Camera 별로 write 여부를 선택할 수 있도록 하였다) system memory 내에 일정영역을 할당하여 그 안에서 wrapping하여 저장할 수 있도록 지원한다(WMAIN\_WRAP\_EN = 1). 단, WMAIN\_WRAP\_INIT\_EN = 1인 경우에는 base address에 저장하게 된다.



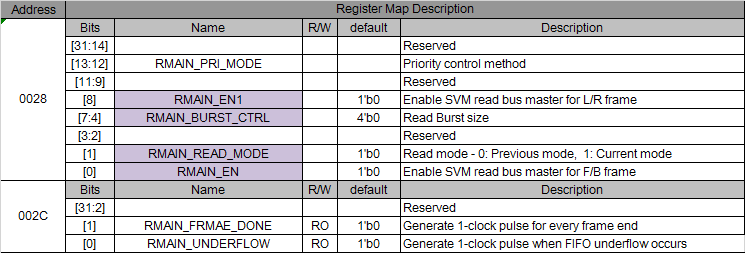
그리고 하나의 라인을 여러 번에 write request를 통해 system memory에 write하게 되는데 그 단위 설정은 WMAIN\_BURST\_CTRL 값으로 지원되며 그 의미는 아래와 같다. 만약 하나의 라인이 이 단위로 나누어 떨어지지 않는 경우에는 마지막 request에 대해서 나머지만큼만 write 하게 된다.

|  |  |
| --- | --- |
| WBURST | Burst Length |
| 5 | 16 |
| 6 | 32 |
| 7 | 64 |
| 8 | 128 |

WMAIN\_WLINE은 wrapping을 감안한 전체 라인 수를 나타낸다. WLINE은 각 카메라 별로 별도로 설정할 수 있다. 이 부분 또한 각 Camera 별로 저장 영역을 다르게 설정할 수 있도록 C0\_WMAIN\_WLINE ~ C3\_WMAIN\_WLINE이 따로 설정될 수 있다.

0x3030, 0x3034의 registers는 read only registers로서 현재 write되고 있는 라인 수 및 address 등의 정보를 제공한다.

### Read control registers



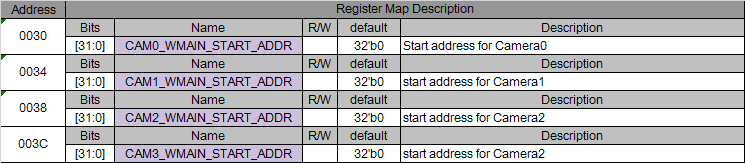
SVM에서는 write와 비슷하게 read 관련 설정 registers가 있으며 위와 같다. Write와 동일하게 RAMIN\_EN = 1 에서부터 출력이 생성되며 (RMAIN\_EN과 RMAIN\_EN1을 두어 F/B 출력 생성부와 L/R 출력 생성부를 별도록 동작 가능하도록 지원할 수 있다) RMAIN\_READ\_MODE를 보고 이전 저장된 frame으로부터 출력을 생성할지 현재 저장 중인 frame으로부터 생성할지를 결정할 수 있고 RMAIN\_BURST\_CTRL 값에 따라 아래 표와 같은 단위로 data를 read하게 된다. 일반적인 경우 4로 설정하여 사용되나 LUT가 가로 방향으로 휘어진 정도가 약한 경우 더 큰 값으로 설정하는 것이 cache hit ratio을 높일 수 있다.

|  |  |
| --- | --- |
| RBURST | Burst Length |
| 4 | 8 |
| 5 | 16 |
| 6 | 32 |
| 7 | 64 |
| 8 | 128 |

Read의 경우에는 write와 비슷하게 burst\_ctrl 값을 설정할 수 있도록 하였고 각 값에 의한 burst length값은 위 표와 같다.

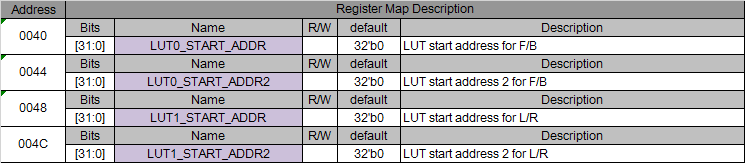
또한 출력 라인 생성 중 MBW 문제로 underflow가 발생되는 경우를 확인하기 위한 read only register로 RMAIN\_UNDERFLOW를 두었다.

### Base address control registers and LUT base address control registers



위 control registers는 4개의 camera 입력 각각을 저장할 base address 설정 registers이다.

### LUT base address control registers



F/B 의 입력을 통해 출력을 생성하는 warping core에서 사용할 LUT의 시작 주소를 LUT0\_START\_ADDR과 LUT0\_START\_ADDR2에 저장하고 L/R 의 입력을 통해 출력을 생성하는 warping core에서 사용할 LUT의 시작 주소를 LUT1\_START\_ADDR과 LUT1\_START\_ADDR2에 저장한다. 두 개의 address 중 어느 것을 사용할지는 lut\_ratio를 통해서 결정되고 이 값을 통해 morphing 지원이 가능하다.

### Bypass 및 LUT mode, mirroring, LUT ratio control registers

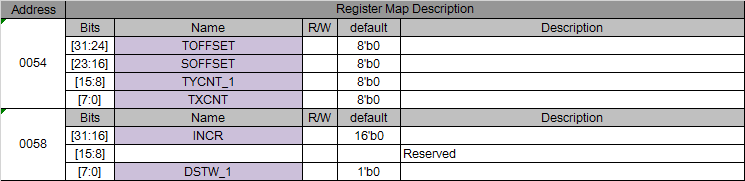


SVM에서는 테스트를 위해 입력된 이미지를 그대로 출력으로 볼 수 있다. 예를 들어 Front bypass image를 출력하고자 하는 경우 FB\_bypass\_en = 1, FB\_bypass\_sel = 0 으로 설정하고 out\_mode = 00 으로 설정함으로써 지원된다 (LR\_bypass 동일). 그리고 입력 이미지를 가로 혹은 세로로 반전시켜야 하는 경우를 위해 각 camera별로 hmirror\_en, vmirror\_en을 두었다.

|  |  |  |
| --- | --- | --- |
| FB\_bypass\_en | FB\_bypass\_sel | Operation |
| 0 | x | Output image generated by LUT |
| 1 | 0 | Front image bypass |
| 1 | 1 | Back image bypass |

FB 출력을 위한 두 개의 LUT 간의 morphing 기능 지원을 위해서 lut\_ratio (LR 출력을 위해서는 lut\_ratio2)가 사용된다. Lut\_ratio = 0 이면 LUT0\_START\_ADDR의 LUT만 사용이 되고 lut\_ratio = 0xff 이면 LUT0\_START\_ADDR2 만 사용된다. 중간 값의 경우 두 개의 LUT 간의 중간 값이 사용된다. (0x30b4 참조)

### LUT information control registers



위 control registers는 LUT에 대한 정보 설정 신호로서 i\_TXCNT는 한 라인에 포함된 LUT의 개수를 의미하고 i\_TYCNT는 LUT의 라인 개수를 의미한다(i\_TYCNT는 1을 뺀 값으로 설정한다. i\_TYCNT\_1). i\_TXCNT와 i\_TYCNT는 전체 이미지를 반으로 나눈 다음 sampling 간격인 i\_DSTW 으로 나눈 값이 되며 이 때 딱 떨어지지 않는 경우에 가로 방향에 대해서는 i\_SOFFSET, 세로 방향에 대해서는 i\_TOFFSET을 설정한다. 마지막으로 215 / i\_DSTW 로 i\_INCR 값의 설정이 필요하다.

SVM은 가로, 세로 동일한 sample 간격으로 매 픽셀마다의 LUT로부터 sampling하여 최종 LUT를 생성하여 사용한다. 이 때 sampling된 하나의 LUT 값은 32b으로 구성되며 각 bits별 의미는 아래와 같다.



[31] : Camera selection bits

[30] : 불연속 표시

[29:15] : LUT y 좌표 (정수부 11b, 소수부 4b 포함)

[14:0] : LUT x 좌표 (정수부 11b, 소수부 4b 포함)

sampling되는 포인트는 아래 그림과 같으며 따라서 가로 세로 해상도에 따른 LUT의 개수는 다음과 같이 계산될 수 있다. (여기에서 DSTW는 sampling 간격을 의미한다)

한 라인의 LUT sample 개수(TXCNT) = ceil ( IMG\_WIDTH / 2 / DSTW ) \* 2 + 1

LUT의 라인 개수(TYCNT) = ceil ( IMG\_HEIGHT / 2 / DSTW ) \* 2 + 1

(720p의 경우 1280 에 대해 dstw = 8 이면 161이 되며 따라서 TXCNT의 최대값은 161이 된다.

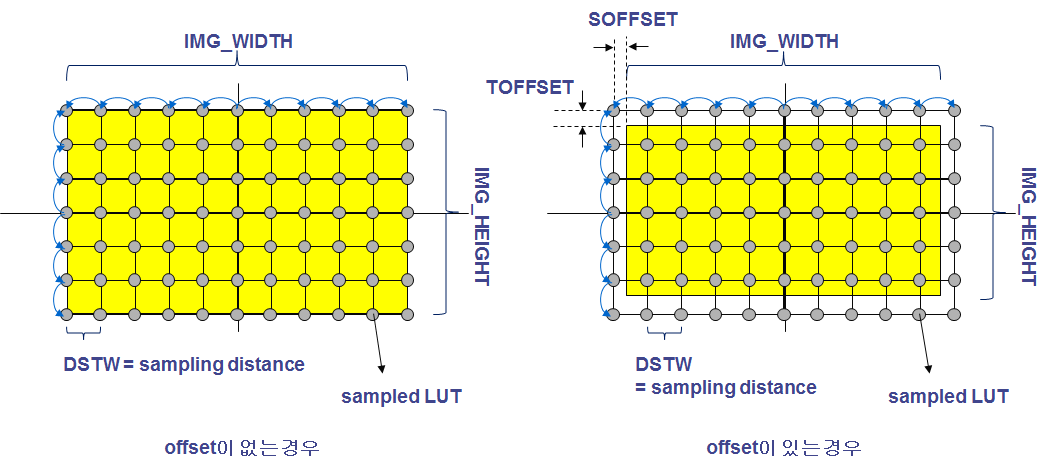


Figure 2‑1 LUT sampling example

위 그림에서 노란색 부분이 출력 이미지의 영역이 되며 임의의 sampling 간격으로 나누다 보면 오른쪽 그림과 같이 딱 떨어지지 않아 offset이 발생하는 경우가 있게 된다. 이 경우 SOFFSET, TOFFSET을 설정하여 해결할 수 있다.

출력 영상이 두 가지 이상의 종류의 LUT 처리를 포함하는 경우 서로 다른 LUT가 인접하는 부분에서는 깔끔하지 못한 영상이 출력되며 이는 서로 다른 LUT 간의 interpolation을 통해 결과 이미지 생성이 처리되기 때문이다. 또한 이러한 경우 대체로 MBW에 대해 악영향을 미치게 된다.

이러한 문제를 해결하고 경계 부분의 깔끔한 출력 영상을 얻기 위해 SVM은 LUT 내 불연속 정보 설정을 지원한다. 아래 그림 중 왼쪽에서 각 색깔 별로 서로 다른 LUT를 사용하는 경우라고 하고 동그라미가 LUT 값이라고 하면 빨간색에 해당하는 LUT 값에 대해 [30]를 1로 설정함으로써 여러 LUT가 사용되는 경우의 인접한 부분에 대해서 불연속임을 인지하여 처리하게 된다. 불연속 정보는 가로 방향에 대해서는 불연속 sample 값에, 세로 방향에 대해서는 이전 라인 sample 값에 설정된다.

아래 그림의 오른쪽은 원본 이미지에 대해서 가로 방향 2-section LUT와 세로 방향 2-section을 morphing 한 결과로서 경계 부분이 깔끔한 것을 볼 수 있다.

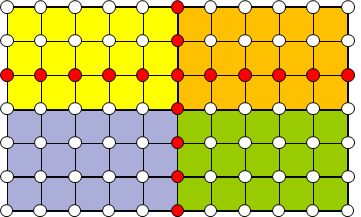
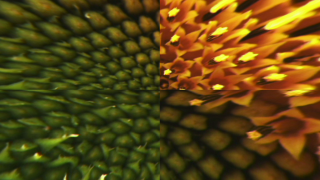
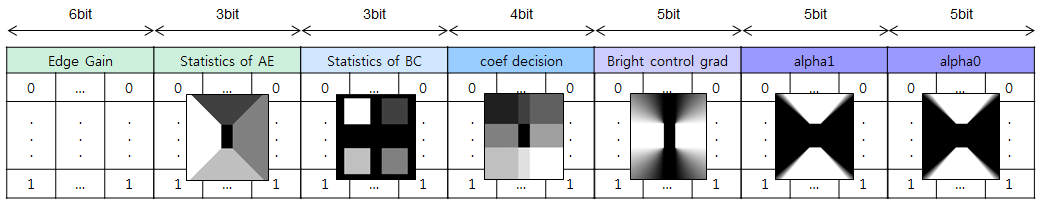
 

Figure 2‑2 Discontinuity information examples of LUT

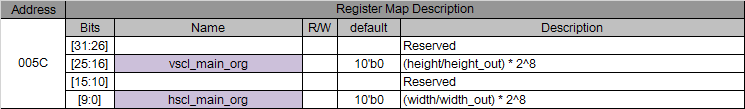
SVM에서는 4개의 camera 입력에 대해 두 개의 warping core를 통해 F/B 출력 이미지와 L/R 출력 이미지를 각각 생성하도록 함으로써 F/B을 위한 LUT와 L/R을 위한 LUT가 필요하다. 하나의 warping core로 F와 B를 처리하고 다른 하나의 warping core로 L과 R을 처리할 수 있는 이유는 F와 B간, L과 R간에는 overlap되는 부분이 없기 때문이고 각 warping core에서 생성된 이미지 간에는 blending 처리가 필요하게 된다. 이에 대한 처리는 뒤에서 설명하도록 하겠다.

그리고 Blending 시 F와 L, R, 그리고 B과 L, R간의 overlap 영역에서 밝기나 색상의 차이가 심한 경우 눈에 거슬릴 수 있어 이에 대한 해결 방안으로 Brightness control 을 위한 별도의 LUT 를 지원한다. 이 LUT는 31bits으로 구성되며 각 bits별 의미는 다음 그림과 같다.



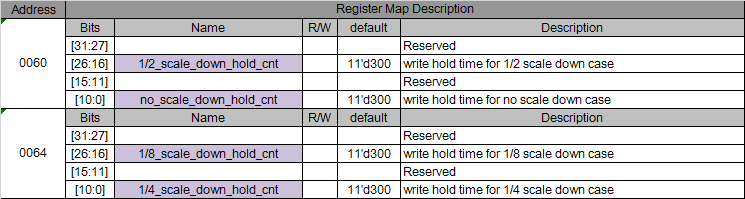
Alpha0와 alpha1은 F/B과 L/R 이미지 간의 blending을 위해 사용되며 dynamic blending을 지원하기 위해 한 두 개의 값을 가진다. 그리고 bright control gradient와 coef decision는 brightness control을 위해 사용된다. 그리고 statistics of BC는 brightness control을 위한 각 구간별 coefficient 계산을 위한 누적값 추출 영역이고, statistics of AE는 각 카메라 별 AE 조정을 위한 누적값 추출 영역을 의미한다. 마지막으로 Edge gain은 SVM 출력의 edge enhancement를 pixel 단위로 지원하기 위해 사용된다.

### Output resolution control registers



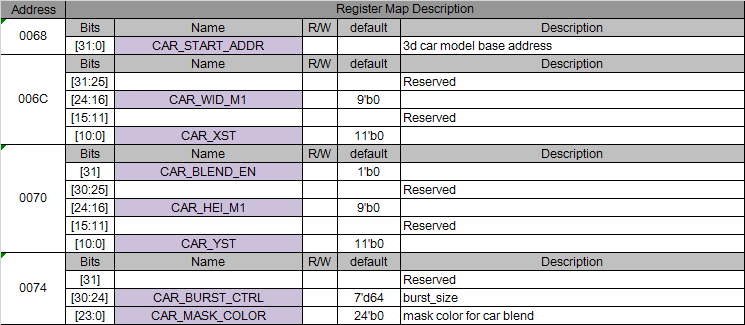
위 control registers는 입력 해상도와 출력 해상도를 다르면서 LUT가 입력 해상도에 맞춰 생성된 경우 이를 지원하기 위해 필요한 것으로 설정 수식은 위의 표와 같다.

### Write request hold time control registers

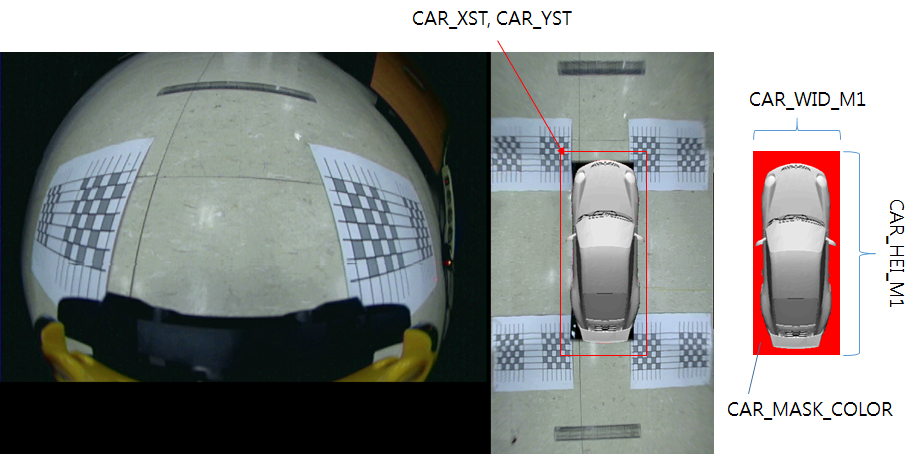


SVM에서는 write request를 일정 간격만큼 지연시켜 요청하도록 하여 read 동작의 우선권을 보장한다. 이러한 request의 한 라인당 요청 회수는 scale down 정도에 따라 다르게 되므로 scale down 정도에 따라 그 간격을 별도로 설정할 수 있도록 하였다.

### 3D Car model blend control registers

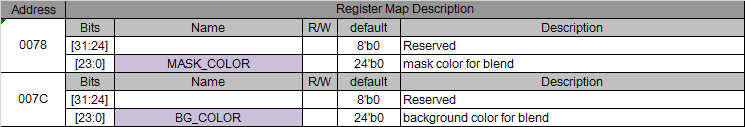


SVM에서는 view position에 따라 보이는 차량의 3D model을 blend block에서 함께 처리할 수 있도록 하고 있으며 이를 위한 차량 모델의 base address 설정과 영역 설정, 그리고 차량의 key color (mask color)를 지원함으로써 임의 모양의 차량 모델을 blending 할 수 있도록 한다. 마지막으로 CAR\_BURST\_CTRL은 data read의 burst length를 지정하는 것으로 RMAIN\_BURST 과 다르게 설정된다.



CAR\_MASK\_COLOR는 [23:16] : Y, [15:8] : Cb, [7:0] : Cr을 의미한다.

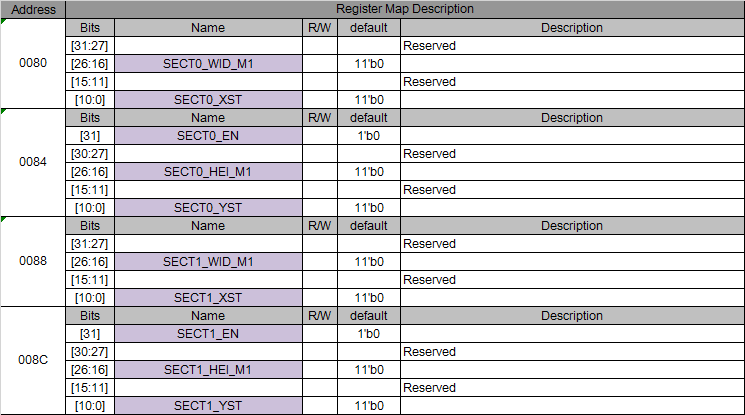
### Mask color control registers



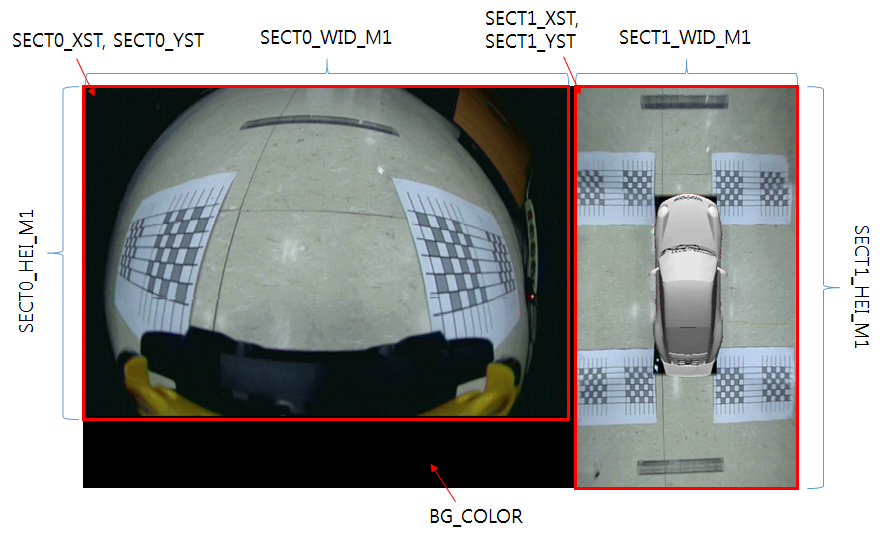
SVM에서는 front와 back 이미지에 의해 출력을 생성하는 warping core에서는 left와 right에 해당하는 부분의 invalid LUT에 대해서는 MASK\_COLOR로 대체, 출력하도록 하여 이후 blend block에서 MASK\_COLOR와 같은 색인 경우 BG\_COLOR로 대체하여 출력되도록 한다. 그리고 출력 활성 영역인 section 영역 외의 영역에 대해서도 BG\_COLOR가 출력된다.

MASK\_COLOR와 BG\_COLOR 모두 [23:16] : Y, [15:8] : Cb, [7:0] : Cr을 의미하며 MASK\_COLOR의 경우 Cb = Cr 로 설정하여야 한다.

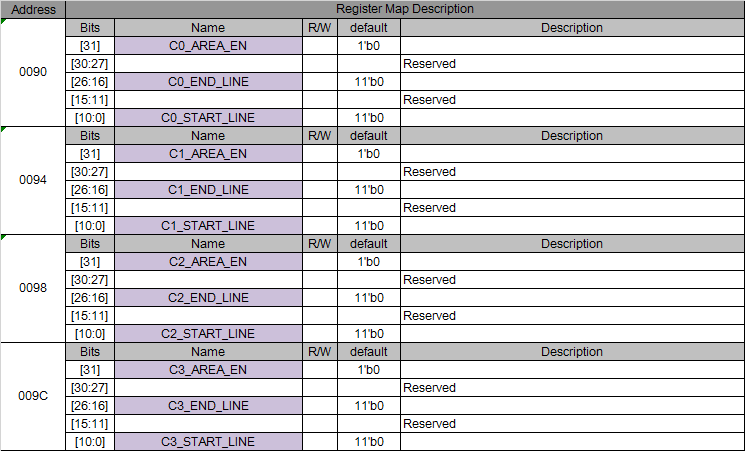
### Section region setting registers



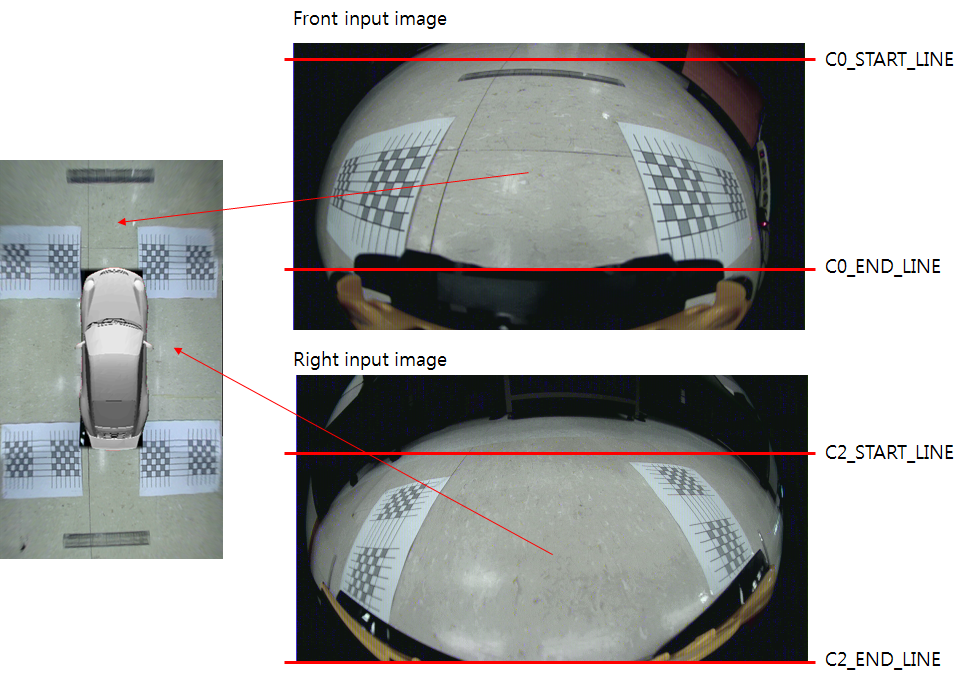
SVM에서는 두 개의 직사각형 형태를 갖는 활성 영역을 설정할 수 있다.



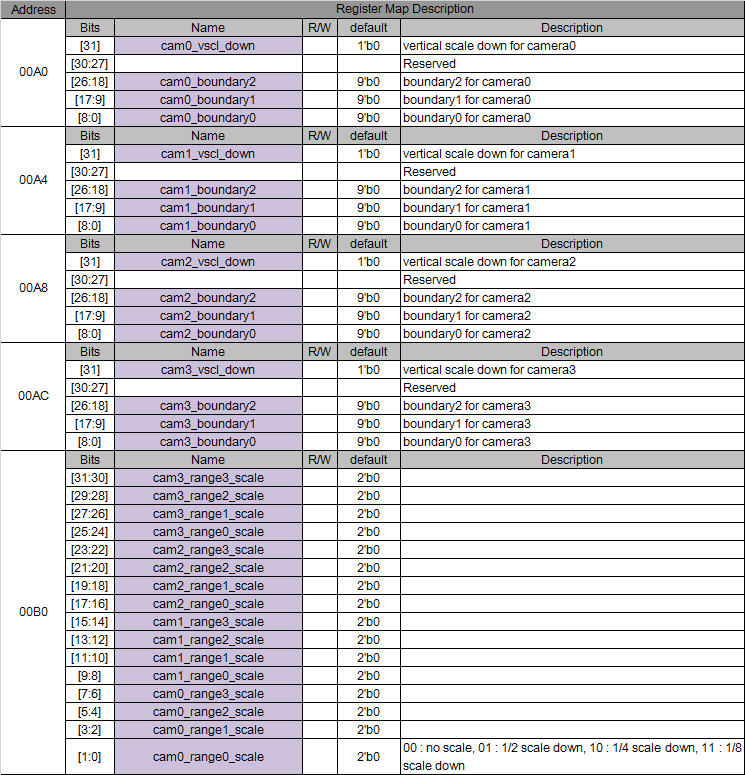
### Crop region setting registers



SVM에서는 각 카메라 별로 write하기 위한 영역을 시작 라인과 끝 라인으로 설정할 수 있다.

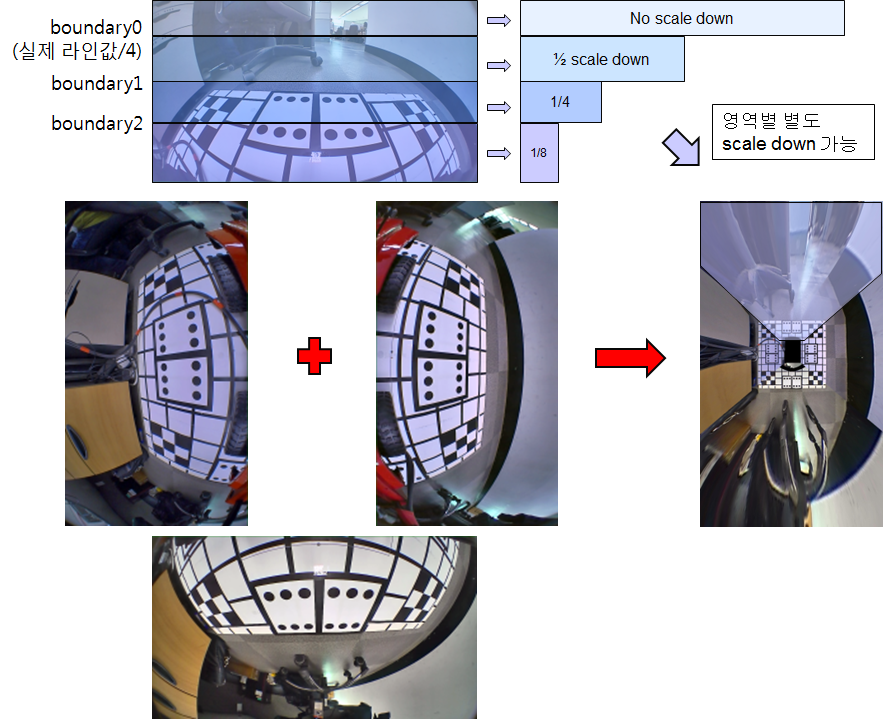


### 4 Range scale down control registers

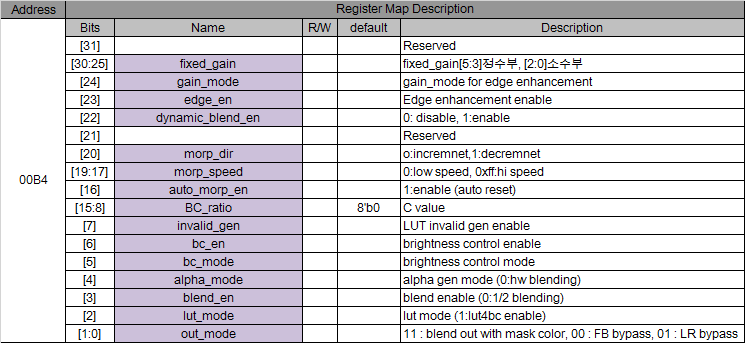


SVM에서는 하나의 카메라 입력에 대해 세로로 4개의 구간으로 나누어 각각을 서로 다른 down scaling할 수 있도록 하였다. 이를 위해 3개의 경계 라인 counter 값(이 때 설정되는 경계 라인 값은 /4 된 값이 저장된다)

과 각 구간의 scaling factor 값을 위와 같이 설정할 수 있다.

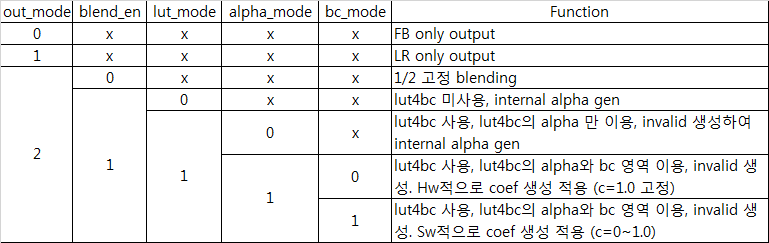


### Out mode/Blending/Brightness Control/Edge enhancement setting registers



SVM의 blend block에서는 F/B core에서 생성된 출력과 L/R core에서 생성된 출력을 입력 받아 어느 하나를 선택, 출력하거나 두 개의 입력을 blending하여 출력할 수 있다. 이를 위한 control register가 out\_mode로서 out\_mode = 2’b0 : FB 생성 이미지 출력, 2’b1 : LR 생성 이미지 출력, 2’b1x : FB와 LR 생성 이미지의 blending 이미지가 출력된다. 또한 0x3084의 [31] CAR\_BLEND\_EN = 1인 경우에는 3D 차량 모델도 해당 영역에 대해서 masking 처리되어 출력하게 된다.

그리고 Blending 과 Brightness control에 대한 각 신호 값들에 따른 동작은 아래 표와 같다.



SVM 에서는 여러 카메라가 섞이는 부분 이외에 하나의 카메라 입력에 대해서만 출력이 생성되는 영역에 대해서는 morphing 기능을 지원할 수 있고 이를 통해 한 번에 LUT가 변화하는 것보다 사용자가 직관적으로 이해할 수 있도록 도울 수 있다 (아래 그림 참조).

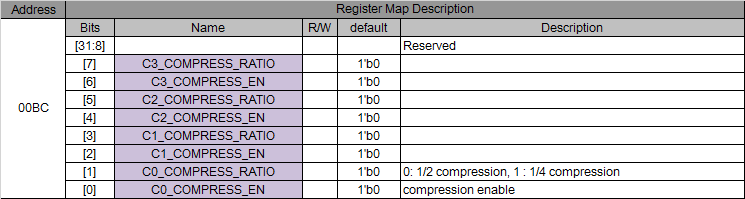
여기에서 auto\_morp\_en 값은 auto reset되며 morp\_dir 값에 따라 increment 혹은 decrement하면서 morphing이 진행된다. 단, lut\_ratio = 0x0 이면 무조건 increment되며 lut\_ratio = 8xff 이면 무조건 decrement된다. 그리고 morp\_speed 값이 클수록 빠르게 변화하게 된다.



SVM에서는 dynamic blending 기능을 지원하며 이 기능의 활성화를 위해 dynamic\_blend\_en 이 있다. 이 register가 1로 활성화되면 0x3198의 ALPHA\_COEF1, 3, 6, 8에 따라 blending 위치가 변경된다. (0xf0800198 참조)

그리고 SVM 영상의 경우 외곽으로 갈수록 많이 펴짐에 따라 출력 영상이 흐리게 생성된다. 이에 대한 해결 방안으로 SVM 영상에 대해 Edge enhancement 기능을 지원하며 이미지 전체에 대해 고정된 값을 적용하는 모드 (gain\_mode = 1)와 LUT 상에 gain 값을 저장하여 pixel 마다 다른 gain 값을 가지고 edge enhancement가 처리되는 모드 (gain\_mode = 0)를 모두 지원한다.

### Compression control registers

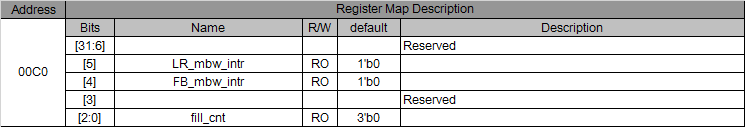


SVM에서는 MBW 감소를 위해 카메라 입력을 시스템 메모리에 저장할 때 Compression 기능을 지원하며 두 가지 모드의 압축율을 지원할 수 있다.

COMPRESSION\_RATIO = 0 : 1/2 압축

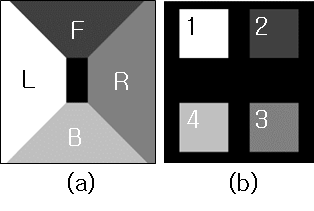
COMPRESSION\_RATIO = 1 : 1/4 압축

### Memory bandwidth debugging registers



### Brightness control statistics registers

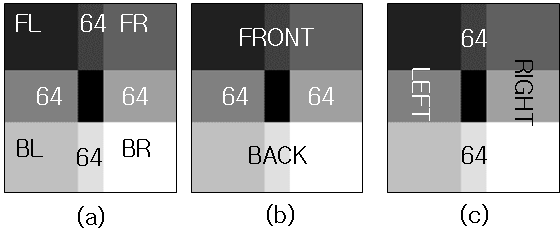
SVM에서는 Auto brightness control을 지원하며 이를 위해 FB 이미지와 LR 이미지간에 서로 섞이게 되는 영역의 색의 통계값을 RGB domain에서 구해야 한다. 예를 들어 아래 그림의 (a)와 같이 4개의 이미지가 merge된다고 할 때 (b)의 1번 사각 영역은 F와 L이 섞이는 영역이고 2는 F와 R이 섞이는 영역이 된다. 이 영역의 통계값 중 FB 이미지의 R 통계값은 FL\_R0에, LR 이미지의 R 통계값은 FL\_R1에 저장된다.





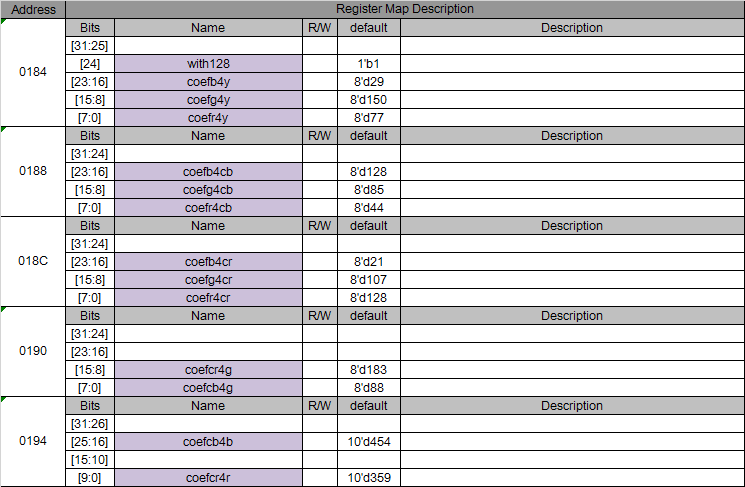
### Coefficient registers for Brightness control





앞의 통계값으로부터 Brightness control을 수행하기 위한 Coefficient 값이 저장되는 control registers로서 COEF1과 COEF3는 위 그림 (a) 영역의 값이고 COEF2는 (b)의 front와 back 영역의 값이며 COEF4는 (c)의 left와 right 영역의 값을 의미한다.

### RGB to YCbCr / YCbCr to RGB coefficient registers



Brightness control은 RGB domain에서 수행됨에 따라 YCbCr을 RGB로 변환하는 기능이 필요하고 이를 위한 변수를 위와 같이 설정할 수 있다. 의미는 다음과 같다. With\_128 은 Cb, Cr 생성에 +128 포함 여부를 나타낸다.

Y = coefr4y \* R + coefg4y \* G + coefb4y \* B

Cb = - coefr4cb \* R - coefg4cb \* G + coefb4cb \* B + 128

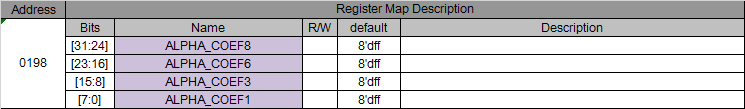
Cr = coefr4cr \* R - coefg4cr \* G - coefb4cr \* B + 128

R = Y + coefcr4r \* (Cr – 128)

G = Y – coefcb4g \* (Cb – 128) – coefcr4g \* (Cr – 128)

B = Y + coefcb4b \* (Cb – 128)

### Dynamic Blending Coefficient



SVM 에서는 각 카메라 간에 blending 되는 영역에서의 사각 문제를 해결하기 위해 dynamic blending 기능을 지원하며 이를 위해 카메라 간에 섞이는 4군데의 영역에 대해 두 개의 alpha 값 중에 interpolation하는 정도를 조정할 수 있다. 이를 위한 control registers이며 각 의미는 아래와 같다. 값이 0xff이면 alpha0, 0x0이면 alpha1을 사용하며 중간 값의 경우 interpolation된 값을 사용하게 된다. (0x30b4 참조)

