Reference

[1] MPEG-4 Video Group. ”Generic Coding of Audio-Visual Objects:Part 2 – Visual 14496-2,” ISO/IEC JTC1/SC29/WG11 M9477,Pattaya, March 2003.

[2] ARM Ltd. “Integrator/AP User Guide”, Sep. 1999.

[3] ARM Ltd. “Integrator/CM940T, CM920T, CM740T, and CM720T User Guide”, Apr. 2001.

[4] ARM Ltd. “Integrator/LM-XCV600E+ and Integrator/LM-EP20K600E+ User Guide” Nov. 2000.

[5] L.De Vos and M.Stegherr, “Parameterizable VLSI architectures for the full-search block-matching algorithm”, *IEEE Transaction On Circuit and Systems*, Vol. 36, pp.1309-1316, Oct. 1989.

[6] T. Komarek and P. Pirsh, “Array architectures for block matching algorithms,” *IEEE Transaction On Circuits and Systems*, Vol. 36, pp.1301-1308, Oct. 1989.

[7] K. M. Yang, M. T. Sun, and L. Wu, “A family of VLSI designs for the motion estimation algorithm,” *IEEE Transaction On Circuits and Systems Video Technol.*, Vol. 2, pp. 169-175, June 1992.

[8] Jae Hun Lee, Kyoung Won Lim, Byung Cheol Song, and Jong Becom Ra, “ A Fast Multi-resolution Block Matching Algorithm and its LSI Architecture for Low Bit-Rate Video Coding”, *IEEE Transaction. On Circuits and Systems for Video Technology*, Vol. 11, No. 12, Dec. 2001

[9] Kun-Bin Lee; Hao-Yun Chin; Hui-Cheng Hsu; Chein-Wei Jen ,“QME: an efficient subsampling-based block matching algorithm for motion estimation” **,** *Proceedings of the 2004 International Symposium on Circuits and Systems*, Vol. 2, Page(s):II - 305-8, 23-26 May 2004

[10] Seongsoo Lee; Jeong-Min Kim; Suo-Ik Chae; “New motion estimation algorithm using adaptively quantized low bit-resolution image and its VLSI architecture for MPEG2 video encoding”, *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 8, Issue 6, Page(s):734 – 744 Oct. 1998

[11] ARM Ltd. “AMBA Specification (Rev 2.0)”, May 1999.

[12] AMPHION CS6701 is available on http://www.amphion.com

[13] Nakayama, H.; Yoshitake, T.; Komazaki, H.; Watanabe, Y.; Araki, H.; Morioka, K.; Li, J.; Peilin, L.; Lee, S.; Kubosawa, H.; Otobe, Y.; “An MPEG-4 video LSI with an error-resilient codec core based on a fast motion estimation algorithm”, *IEEE Solid-State Circuits Conference, 2002. Digest of Technical Papers*. Vol. 1 Page(s):368 - 474, 3-7 Feb. 2002

[14] Y.-W. Huang, S.-Y. Chien, B.-Y. Hsieh, and L.-G. Chen, “An efficient and low power architecture design for motion estimation design for motion estimation using global elimination algorithm”, *in Proc. Acoustics, Speech, and Signal Processing (ISCASSP’02)*, May 2002, vol. 3, pp. 3120-3123.

Appendix A I/O definition

Fig. A-1 depicts the input/output port of the MPEG-4 AHB wrapper and Table A-1 describes the definition of each input/output port on the MPEG4_wrapper (Unless stated, all signals are active HIGH).

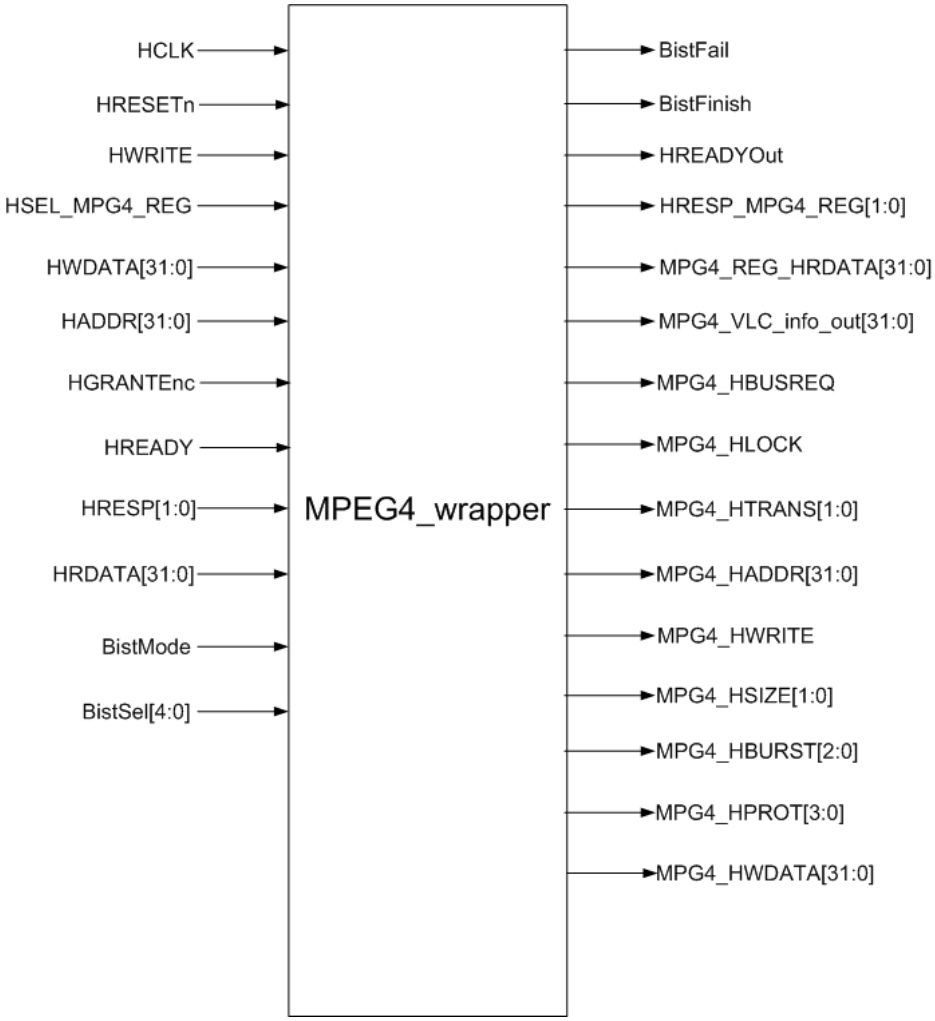

Figure A- 1 MPEG-4 AHB wrapper I/O

Table A- 1 MPEG-4 AHB wrapper I/O description

Port name	Direction	Description
HCLK	Input	System clock
HRESETn	Input	System reset (active low)
HWRITE	Input	AHB register bank slave write control
HSEL_MPG4_REG	Input	AHB register bank slave select signal
HWDATA[31:0]	Input	AHB register bank slave write in data
HADDR[31:0]	Input	AHB register bank slave address in
HGRANTEnc	Input	AHB Encoder IP master grant signal
HREADY	Input	AHB Encoder IP master ready input
HRESP[1:0]	Input	AHB Encoder IP master response input
HRDATA[31:0]	Input	AHB Encoder IP master read in data
BistMode	Input	BIST mode enable signal
BistSel[4:0]	Input	BIST ram module select
BistFail	Output	BIST fail indicate signal
BistFinish	Output	BIST finish indicate signal
HREADYOut	Output	AHB register bank slave ready output
HRESP_MPG4_REG[1:0]	Output	AHB register bank slave response output
MPG4_REG_HRDATA[31:0]	Output	AHB register bank slave output data
MPG4_VLC_info_out[31:0]	Output	MPEG-4 VLC information, connect to DMAC
MPG4_HBUSREQ	Output	AHB Encoder IP master bus request signal
MPG4_HLOCK	Output	AHB Encoder IP master bus lock control
MPG4_HTRANS[1:0]	Output	AHB Encoder IP master transfer mode control
MPG4_HADDR[31:0]	Output	AHB Encoder IP master read address control
MPG4_HWRITE	Output	AHB Encoder IP master write control
MPG4_HSIZE[1:0]	Output	AHB Encoder IP master data bus size control
MPG4_HBURST[2:0]	Output	AHB Encoder IP master burst mode control
MPG4_HPROT[3:0]	Output	AHB Encoder IP master protection

		control
MPG4_HWDATA[31:0]	Output	AHB Encoder IP master write data

Fig. A-2 depicts the input/output port of the MPEG-4 Encoder IP, and users could design a specific bus wrapper to wrap this MPEG-4 encoder IP in other specific bus architecture. Table A-2 describes definition of each input/output port on the MPEG-4 Encoder IP (Unless stated, all signals are active HIGH).

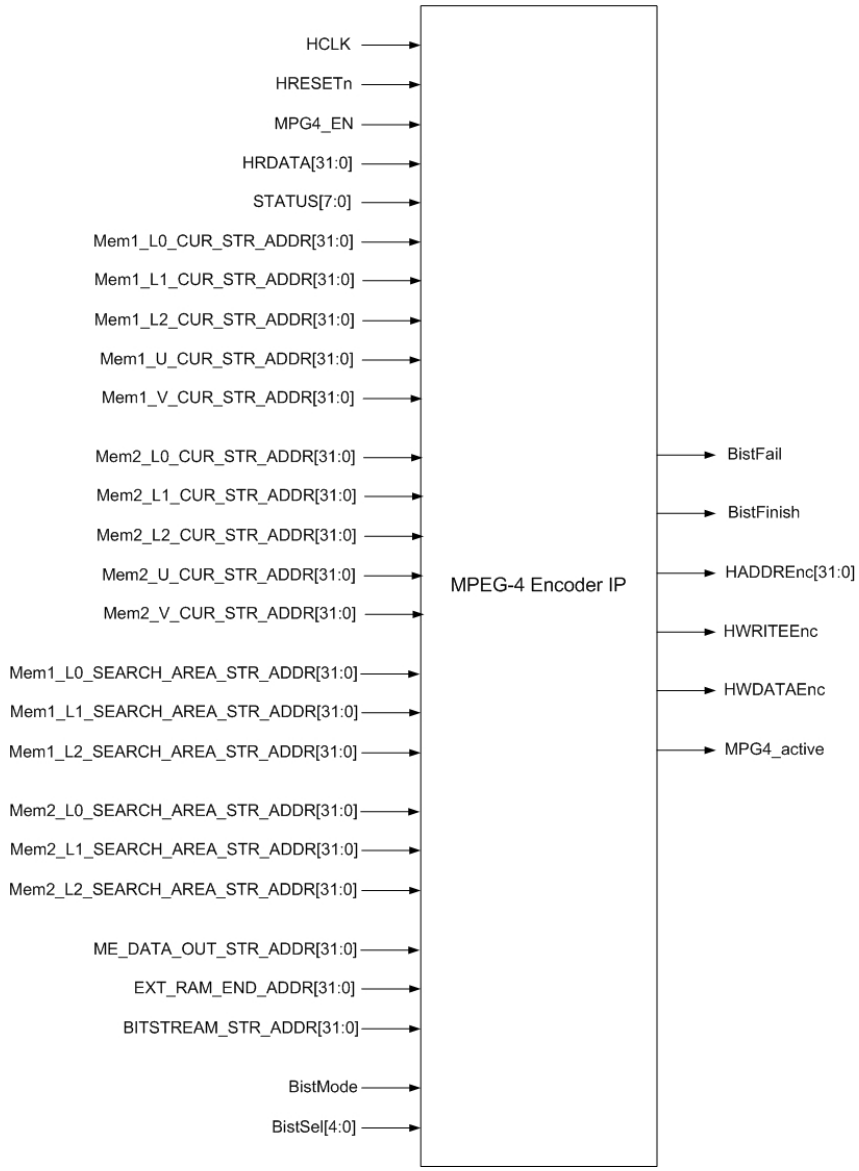

Figure A- 2 MPEG-4 Encoder IP I/O

Table A- 2 MPEG-4 Encoder IP I/O description

Port Name	Direction	Description
HCLK	Input	System clock
HRESETn	Input	System reset(active low)
MPG4_EN	Input	Chip Enable
HRDATA[31:0]	Input	Read in data
STATUS[7:0]	Input	Chip status control
Mem1_L0_CUR_STR_ADDR[31:0]	Input	Mem1 Level0 current image start address, this value is derived from the register (please refer to section 3.2 and Appendix B)
Mem1_L1_CUR_STR_ADDR[31:0]	Input	Mem1 Level1 current image start address, this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem1_L2_CUR_STR_ADDR[31:0]	Input	Mem1 Level2 current image start address, this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem1_U_CUR_STR_ADDR[31:0]	Input	Mem1 U image start address, this value is derived from the register (please refer to section 3.2 and Appendix B)
Mem1_V_CUR_STR_ADDR[31:0]	Input	Mem1 V image start address , this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem2_L0_CUR_STR_ADDR[31:0]	Input	Mem2 Level0 current image start address, this value is derived from the register (please refer to section 3.2 and Appendix B)
Mem2_L1_CUR_STR_ADDR[31:0]	Input	Mem2 Level1 current image start address, this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)

Mem2_L2_CUR_STR_ADDR[31:0]	Input	Mem2 Level2 current image start address, this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem2_U_CUR_STR_ADDR[31:0]	Input	Mem2 U image start address, this value is derived from the register (please refer to section 3.2 and Appendix B)
Mem2_V_CUR_STR_ADDR[31:0]	Input	Mem2 V image start address, this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem1_L0_SEARCH_AREA_STR_ADDR[31:0]	Input	Mem1 Level0 search area start address (this is the position Mem1_L0_CUR_STR_ADDR left 2 pixels and up 2 pixels) , this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem1_L1_SEARCH_AREA_STR_ADDR[31:0]	Input	Mem1 Level1 search area start address (this is the position Mem1_L1_CUR_STR_ADDR left 2 pixels and up 2 pixels) , this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem1_L2_SEARCH_AREA_STR_ADDR[31:0]	Input	Mem1 Level2 search area start address (this is the position Mem1_L2_CUR_STR_ADDR left 4 pixels and up 4 pixels) , this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem2_L0_SEARCH_AREA_STR_ADDR[31:0]	Input	Mem2 Level0 search area start address (this is the position Mem2_L0_CUR_STR_ADDR left 2 pixels and up 2 pixels) , this value

		is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem2_L1_SEARCH_AREA_STR_ADDR[31:0]	Input	Mem2 Level1 search area start address (this is the position Mem2_L1_CUR_STR_ADDR left 2 pixels and up 2 pixels) , this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
Mem2_L2_SEARCH_AREA_STR_ADDR[31:0]	Input	Mem2 Level2 search area start address (this is the position Mem2_L2_CUR_STR_ADDR left 4 pixels and up 4 pixels) , this value is derived from the circuit in the MPEG4_wrapper (about its position please refer to Appendix B)
ME_DATA_OUT_STR_ADDR[31:0]	Input	ME output data start address (This value is derived from the register value, please refer to section 3.2)
EXT_RAM_END_ADDR[31:0]	Input	External memory end address (This value is derived from the register value, please refer to section 3.2)
BITSTREAM_STR_ADDR[31:0]	Input	Bitstream data start address (This value is derived from the register value, please refer to section 3.2)
BistMode	Input	On-chip memory bist enable
BistSel[4:0]	Input	Bist memory select(25 memory modules totally, range : 00h~18h)
BistFail	Output	On-chip memory fail signal(active high)
BistFinish	Output	Bist finish signal
HADDREnc[31:0]	Output	Output address
HWRITEEnc	Output	Write signal(write=1 ,read=0)
HWDATAEnc[31:0]	Output	Output data
MPG4_active	Output	‘1’ : MPEG-4 IP is busy ‘0’ :MPEG-4 IP is in idle state

Appendix B Memory mapping on the ARM Integrator

We build a FPGA prototyping system on the ARM Integrator, and Fig. B-1, Fig. B-2 show the memory mapping of this FPGA prototyping system.

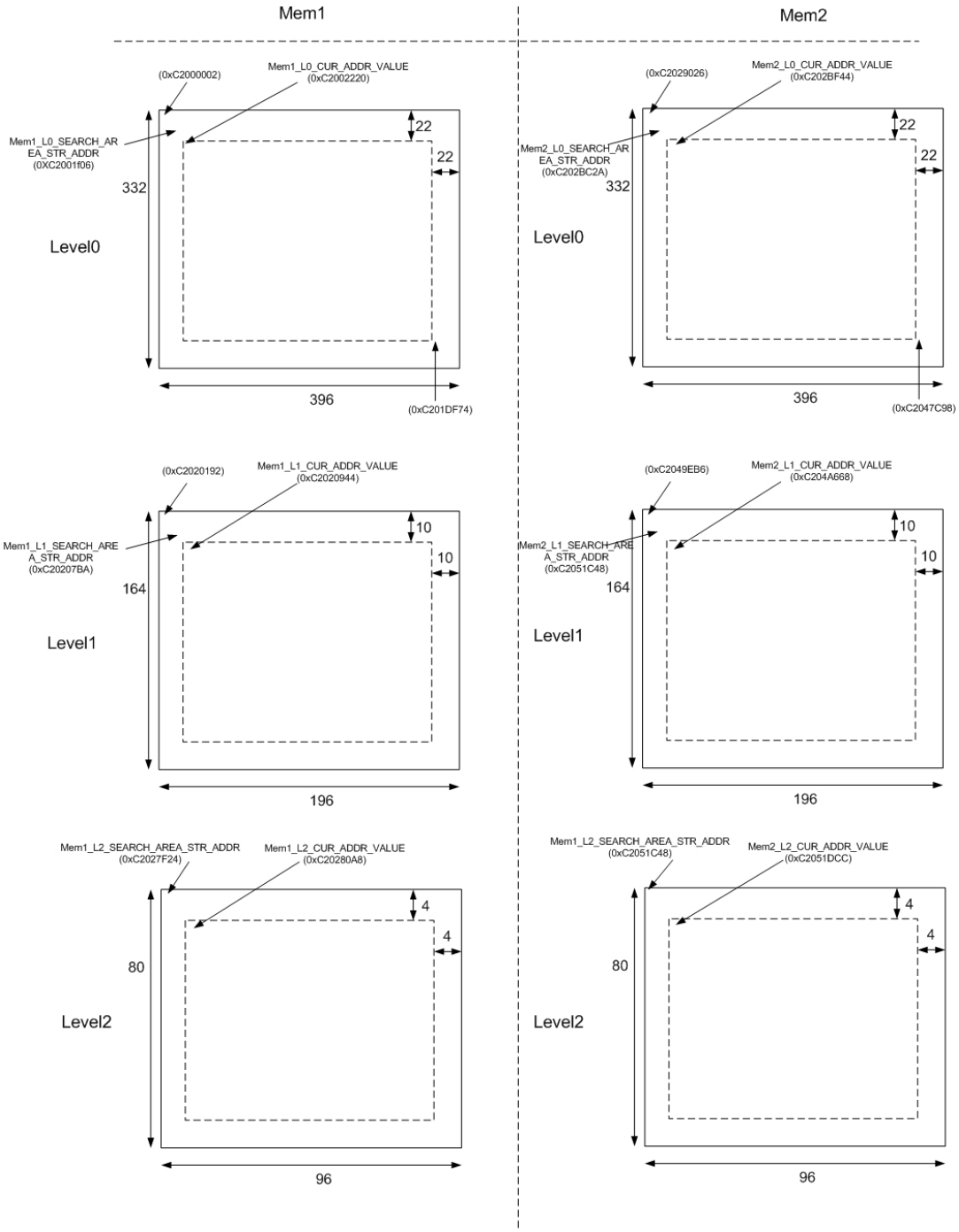

Figure B- 1 Y data memory mapping

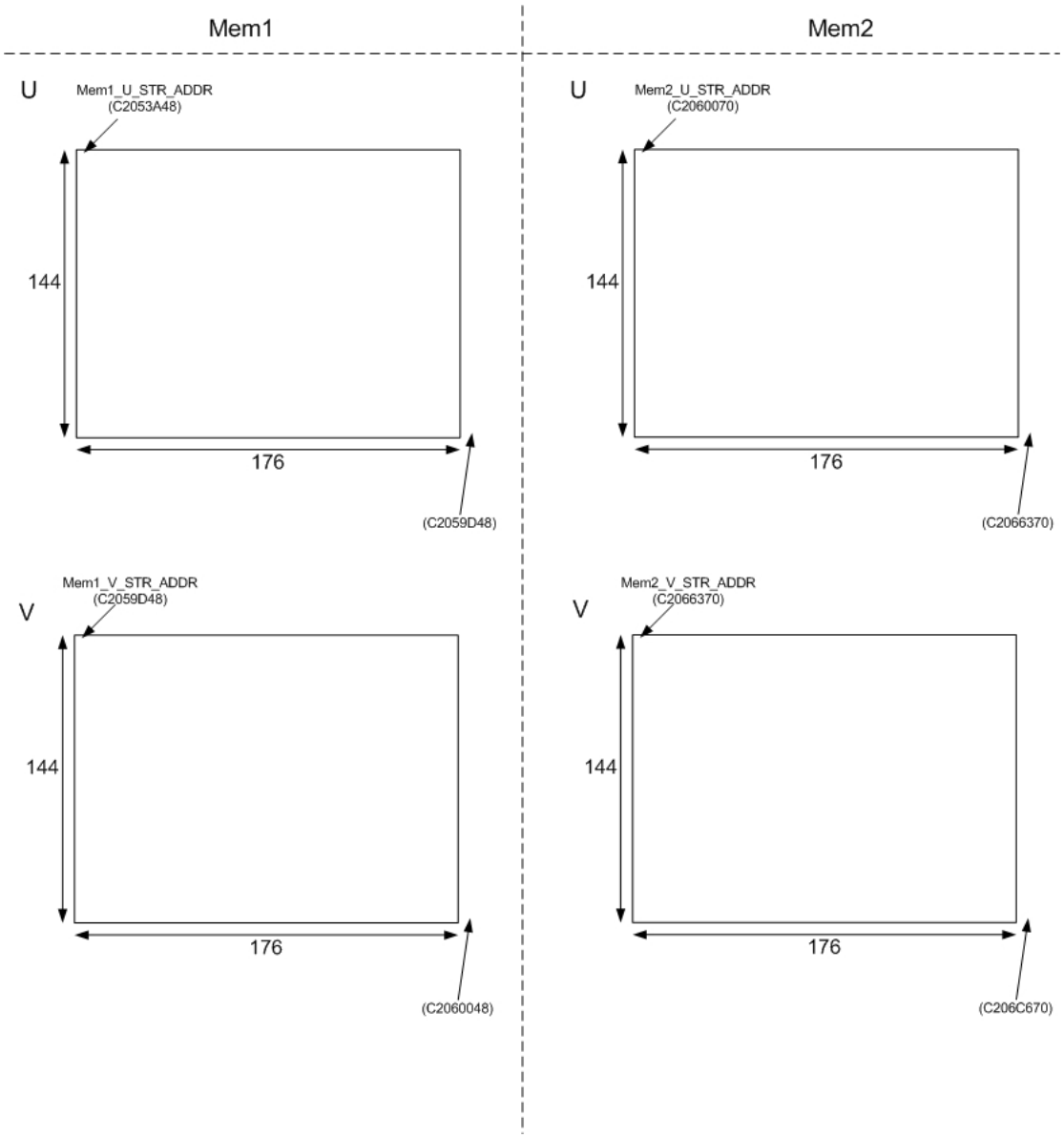

Figure B- 2 UV data memory mapping

0xC206C700 ~ 0xC206CD30 = ME output data address

0xC206CD50 ~ 0xC20FFFF8 = bitstream address