CONFIGURATION CSR (0x0058) – (Firmware Update)

[31] – (R/W) – vme program enable

[30…28] – (R/W) – Reserved

[27] – (R/W) – Reserved

[26…24] – (R/W) – OPCODE (bit 31 = 1 also required)

[23…9] – (R) – Reserved

8 – (R) – Busy (operation in progress)

[7…0] – (R) – Last Valid Data Read

CONFIGURATION ADR/DATA (R/W) (0x005C) – (Firmware Update)

[31] – Execute

[30…18] – Page address

[17…8] – Byte address

[7…0] – EPROM data to write

**Block = 8 pages**

**Page = 528 Bytes**

**Block Erase**

Configuration csr for block erase (opcode 4 = erase)

vmeWrite32(0x90380058, 0x84000000);

Erase blocks using top 10 bits of page address [30… 21] 0-1023

vmeWrite32(0x9038005C, 0x80000000);

…

vmeWrite32(0x9038005C, 0xFF700000);

Pull Execute low before asserting new configuration type

vmeWrite32(0x9038005C, 0x00000000);

**Write to Buffer**

Configuration csr for buffer write (opcode 0 = buffer write)

vmeWrite32(0x90380058, 0x80000000);

Write configuration data byte using byte addresses 0-527

vmeWrite32(0x9038005C, 0x80000000);

…

vmeWrite32(0x9038005C, 0x80020F00);

Pull Execute low before asserting new configuration type

vmeWrite32(0x9038005C, 0x00000000);

**Push buffer content to main memory (1 page/528 bytes)**

Configuration csr for buffer to main memory (opcode 3 = buffer push)

vmeWrite32(0x90380058, 0x83000000);

Push buffer contents using Page address

vmeWrite32(0x9038005C, 0x80000000);

…

vmeWrite32(0x9038005C, 0xFFFC0000);

Pull Execute low before asserting new configuration type

vmeWrite32(0x9038005C, 0x00000000);

**Read from main memory (cycle 528 Byte addresses for each page iteration)**

Configuration csr for main memory read (opcode 1 = main memory read)

vmeWrite32(0x90380058, 0x81000000);

Read main memory using full address (page & byte)

vmeWrite32(0x9038005C, 0x80000000);

…

vmeWrite32(0x9038005C, 0xFFF20F00);

Last valid word can be read from configuration csr bits [7…0]

dval = vmeRead32(0x90380058);

Pull Execute low before asserting new configuration type

vmeWrite32(0x9038005C, 0x00000000);

**Read from Buffer (for debug, buffer contents not main memory)**

Configuration csr for main memory read (opcode 2 = buffer memory read)

vmeWrite32(0x90380058, 0x82000000);

Read main memory using byte address

vmeWrite32(0x9038005C, 0x80000000);

…  
 vmeWrite32(0x9038005C, 0x80020F00);

Last valid word can be read from configuration csr bits [7…0]

dval = vmeRead32(0x90380058);

Pull Execute low before asserting new configuration type

vmeWrite32(0x9038005C, 0x00000000);