CRISA Board EDAC Usage in DMC and SPU OBS

Doc. PACS-CL-TN-060, issue 1.1

Prepared by :	A. Mazy
Verified by :	
Authorised by :	
Approved by :	H. Feuchtgruber

filename : PACS-CL-TN-XXX EDAC Usage.doc

Document Change Record

Issue	Date	Comments
1.0	01 september 2010	initial issue.
1.1	03 september 2010	Updated with information from testing, old emails and source code invalidating the CRISA documentation

last saved by amazy on 03-09-2010 11:09

Table of Contents

1 Scope

This document explains how the EDAC is used in the DMC OBS. Since SPU OBS is using the same code, this document is also applicable to the SPU.

2 Support Documents

RD01	HPL-IC-1248-01-CRS_ISS5	HW/SW Interface Document

3 EDAC implementation on the CRISA boards

The EDAC is part of the DMPSC and PMPSC (Data and Program Memory Support Chip). The EDAC is placed on the bus between the memory and the CPU.

For Data memory, the bus is 40bits width (32bits of data and 8 control bits). Once the EDAC is enabled, it checks the consistency of all the data transiting on the bus. If a single bit failure is detected, it is corrected automatically. Furthermore, an interrupt is raised to signal the CPU that it must correct the memory.

The EDAC is not correcting the RAM itself; it is only modifying the data read from the memory.

The EDAC can be enabled individually for any of the 2 PM banks and 4 DM banks. On the CRISA board, only one PM bank is used for the Program RAM and 1 DM bank for the Data RAM. Other banks are used by extension boards or set of registers.

4 EDAC usage in DMC and SPU OBS

4.1 Initialisation of the EDAC at startup

Nothing is done in the HLSW OBS to enable the EDAC itself. According to the CRISA documentation, the SUSW is supposed to initialize the EDAC correctly.

However, we have been able to dump the value of the CONFBANK0 register and realized that the bit2 was not set (see section 6). Furthermore, we found old emails between CRISA and TUVIE where CRISA states that the documentation is wrong and the EDAC is not enabled by the SUSW.

The DMPSC and PMPSC interrupts are configured to forward the EDAC interrupts to the CPU. And the CPU interrupt is enabled.

```
//enable DM EDAC interrupts
*((int*)K_DMADD_DMPSC_INTMASK) &=  0xFFFFBFFE;

//enable PM EDAC interrupts
*((int*)K_PMADD_PMPSC_INTMASK) &=  0xFFFFBFFE;

// enable PMPSC interrupt
KS_IRQSetHandler(8, pmpsc_isr);
__asm(" bit clr MODE2 IRQ0E;");  // must be level sensitive interrupt
__asm("   nop;");
KS_ISREnable(8);
```

4.2 Event flow

When a single bit failure occurs, here is the chain of events:

- An external event flips the bit in the RAM
- Later, the CPU requests a read of this memory cell
- The data is transferred on the bus and passes through the EDAC
- EDAC realizes that the data is not compatible with its control bits.
- EDAC corrects that data on the bus, stores the failing address in a register and raises an interrupt.
- The corrected data arrives at the CPU that can use it normally
- The CPU receives the EDAC interrupt and enters its handler
- The interrupt handler gets the failing address and reads it. The data is once again corrected by the EDAC.
- The interrupt handler writes the corrected data at the failing address. By this way, the error disappears from the RAM.
- The interrupt handler increments a failure counter and stores the failing address in a circular buffer.
- The interrupt handler exits and the CPU continues its nominal operations

Note that, in case of double bit failures, since the EDAC is not able to correct the data, only the error counters and failing address buffer are updated.

4.3 Memory scrubbing

Additionally, the housekeeping task performs a scrubbing of the DRAM and PRAM. 32 locations of each RAM are read every 2 seconds. It takes around 9 hours to read the used memory.

Note that the mission of memory scrubbing is actually to avoid single errors to accumulate in the single word that is rarely read by the CPU.

4.4 Memory usage

The DMC OBS uses the following amount of memory:

- PRAM: 30508 words = 1.46Mbits
- DRAM: 50581 words = 1.61Mbits

Note that the DRAM contains mainly buffers that are overwritten at least every 2 seconds. Only a small part of the DRAM contains data that could affect the execution of the program.

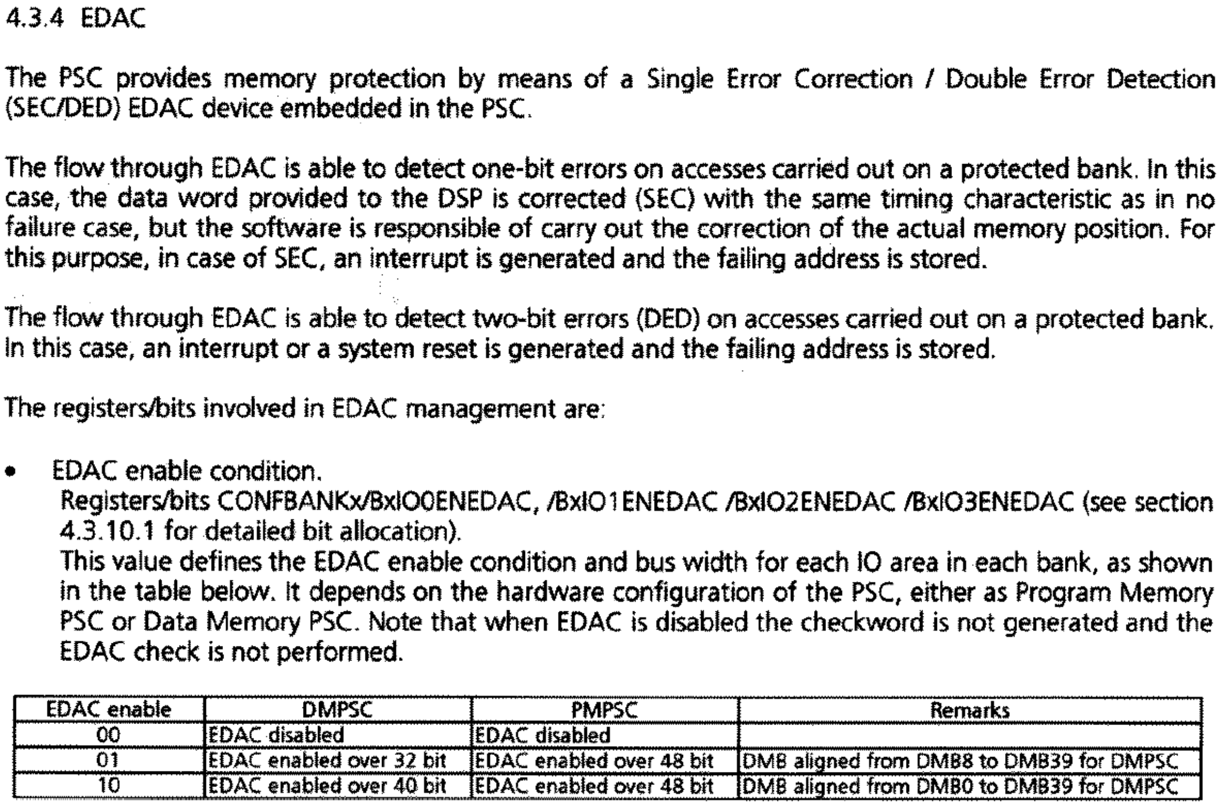
5 Corrections required in DMC OBS

The EDAC should be enabled at startup. This should be done on the DRAM and PRAM bank only. We would need to add the following line of code in the initialization of the software (in seq.c, after the configuration of the wait states for the MIM FPGA DM bank) :

```
//Enable EDAC on DRAM
dmConfBank0 = *((int*)K_DMADD_DMPSC_CONFBANK0);
SET_BIT(dmConfBank0, 0x00000004);
*((int*)K_DMADD_DMPSC_CONFBANK0) = dmConfBank0;

//Enable EDAC on PRAM
MEM_WriteBitInPM32bitMSWord(K_DMADD_PMPSC_CONFBANK1, 2, 1);
```

6 Extract from CRISA Board User Manual



4.3.4 EDAC

The PSC provides memory protection by means of a Single Error Correction / Double Error Detection (SEC/DED) EDAC device embedded in the PSC.

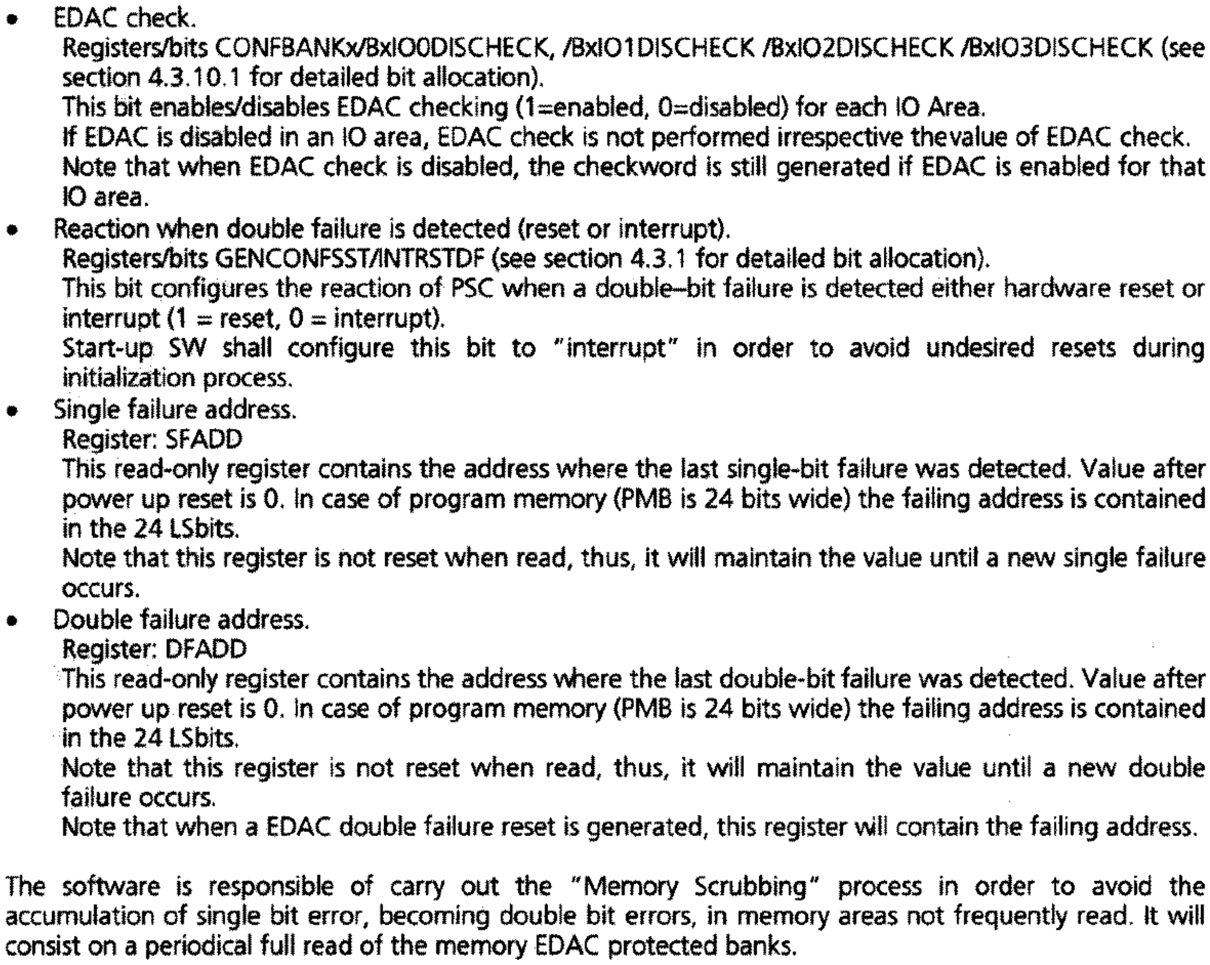
The flow through EDAC is able to detect one-bit errors on accesses carried out on a protected bank. In this case, the data word provided to the DSP is corrected (SEC) with the same timing characteristic as in no failure case, but the software is responsible of carry out the correction of the actual memory position. For this purpose, in case of SEC, an interrupt is generated and the failing address is stored.

The flow through EDAC is able to detect two-bit errors (DED) on accesses carried out on a protected bank. In this case, an interrupt or a system reset is generated and the failing address is stored.

The registers/bits involved in EDAC management are:

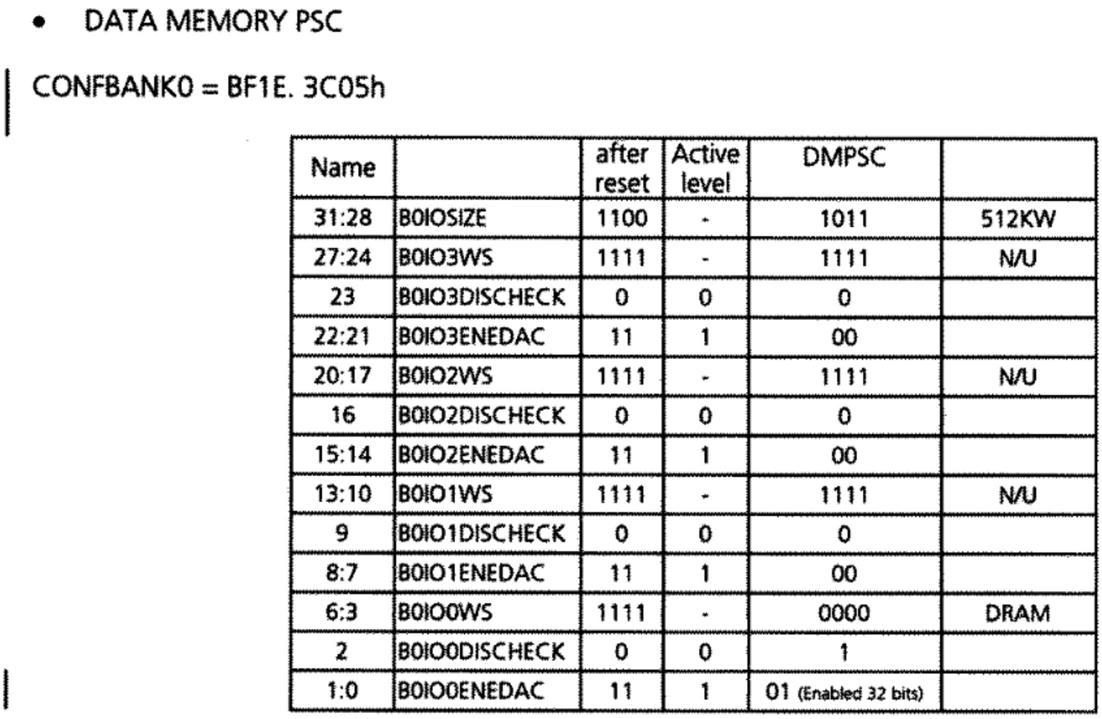
- EDAC enable condition.
  Registers/bits CONFBANKx/BxIO0ENEDAC, /BxIO1ENEDAC /BxIO2ENEDAC /BxIO3ENEDAC (see section 4.3.10.1 for detailed bit allocation).
  This value defines the EDAC enable condition and bus width for each IO area in each bank, as shown in the table below. It depends on the hardware configuration of the PSC, either as Program Memory PSC or Data Memory PSC. Note that when EDAC is disabled the checkword is not generated and the EDAC check is not performed.

EDAC enable	DMPSC	PMPSC	Remarks
00	EDAC disabled	EDAC disabled	
01	EDAC enabled over 32 bit	EDAC enabled over 48 bit	DMB aligned from DMB8 to DMB39 for DMPSC
10	EDAC enabled over 40 bit	EDAC enabled over 48 bit	DMB aligned from DMB0 to DMB39 for DMPSC



- EDAC check.
  Registers/bits CONFBANKx/BxIO0DISCHECK, /BxIO1DISCHECK /BxIO2DISCHECK /BxIO3DISCHECK (see section 4.3.10.1 for detailed bit allocation).
  This bit enables/disables EDAC checking (1=enabled, 0=disabled) for each IO Area.
  If EDAC is disabled in an IO area, EDAC check is not performed irrespective thevalue of EDAC check.
  Note that when EDAC check is disabled, the checkword is still generated if EDAC is enabled for that IO area.
- Reaction when double failure is detected (reset or interrupt).
  Registers/bits GENCONFSST/INTRSTDF (see section 4.3.1 for detailed bit allocation).
  This bit configures the reaction of PSC when a double–bit failure is detected either hardware reset or interrupt (1 = reset, 0 = interrupt).
  Start-up SW shall configure this bit to "interrupt" in order to avoid undesired resets during initialization process.
- Single failure address.
  Register: SFADD
  This read-only register contains the address where the last single-bit failure was detected. Value after power up reset is 0. In case of program memory (PMB is 24 bits wide) the failing address is contained in the 24 LSbits.
  Note that this register is not reset when read, thus, it will maintain the value until a new single failure occurs.
- Double failure address.
  Register: DFADD
  This read-only register contains the address where the last double-bit failure was detected. Value after power up reset is 0. In case of program memory (PMB is 24 bits wide) the failing address is contained in the 24 LSbits.
  Note that this register is not reset when read, thus, it will maintain the value until a new double failure occurs.
  Note that when a EDAC double failure reset is generated, this register will contain the failing address.

The software is responsible of carry out the "Memory Scrubbing" process in order to avoid the accumulation of single bit error, becoming double bit errors, in memory areas not frequently read. It will consist on a periodical full read of the memory EDAC protected banks.

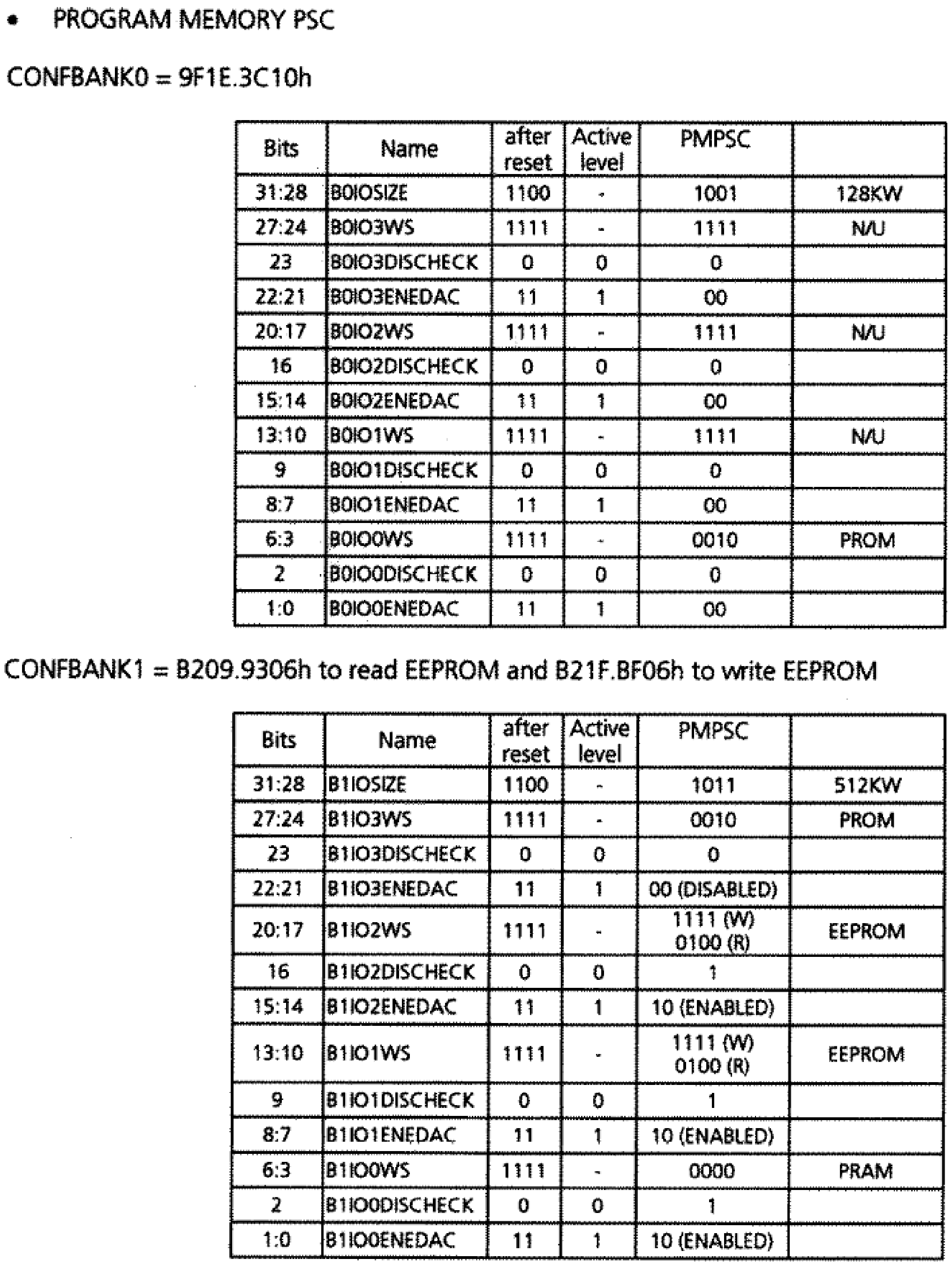


- DATA MEMORY PSC

CONFBANK0 = BF1E. 3C05h

Name		after reset	Active level	DMPSC	
31:28	B0IOSIZE	1100	-	1011	512KW
27:24	B0IO3WS	1111	-	1111	N/U
23	B0IO3DISCHECK	0	0	0	
22:21	B0IO3ENEDAC	11	1	00	
20:17	B0IO2WS	1111	-	1111	N/U
16	B0IO2DISCHECK	0	0	0	
15:14	B0IO2ENEDAC	11	1	00	
13:10	B0IO1WS	1111	-	1111	N/U
9	B0IO1DISCHECK	0	0	0	
8:7	B0IO1ENEDAC	11	1	00	
6:3	B0IO0WS	1111	-	0000	DRAM
2	B0IO0DISCHECK	0	0	1	
1:0	B0IO0ENEDAC	11	1	01 (Enabled 32 bits)	

Note that, we have performed a dump of this register while the HLSW was running and its value was: 0xBF1E3C01. This shows that the DM EDAC is disabled.



- PROGRAM MEMORY PSC

CONFBANK0 = 9F1E.3C10h

Bits	Name	after reset	Active level	PMPSC	
31:28	B0IOSIZE	1100	-	1001	128KW
27:24	B0IO3WS	1111	-	1111	N/U
23	B0IO3DISCHECK	0	0	0	
22:21	B0IO3ENEDAC	11	1	00	
20:17	B0IO2WS	1111	-	1111	N/U
16	B0IO2DISCHECK	0	0	0	
15:14	B0IO2ENEDAC	11	1	00	
13:10	B0IO1WS	1111	-	1111	N/U
9	B0IO1DISCHECK	0	0	0	
8:7	B0IO1ENEDAC	11	1	00	
6:3	B0IO0WS	1111	-	0010	PROM
2	B0IO0DISCHECK	0	0	0	
1:0	B0IO0ENEDAC	11	1	00	

CONFBANK1 = B209.9306h to read EEPROM and B21F.BF06h to write EEPROM

Bits	Name	after reset	Active level	PMPSC	
31:28	B1IOSIZE	1100	-	1011	512KW
27:24	B1IO3WS	1111	-	0010	PROM
23	B1IO3DISCHECK	0	0	0	
22:21	B1IO3ENEDAC	11	1	00 (DISABLED)	
20:17	B1IO2WS	1111	-	1111 (W) 0100 (R)	EEPROM
16	B1IO2DISCHECK	0	0	1	
15:14	B1IO2ENEDAC	11	1	10 (ENABLED)	
13:10	B1IO1WS	1111	-	1111 (W) 0100 (R)	EEPROM
9	B1IO1DISCHECK	0	0	1	
8:7	B1IO1ENEDAC	11	1	10 (ENABLED)	
6:3	B1IO0WS	1111	-	0000	PRAM
2	B1IO0DISCHECK	0	0	1	
1:0	B1IO0ENEDAC	11	1	10 (ENABLED)	

Note that we have not been able to dump the value of the CONFBANK1 register. We may assume that the PM EDAC is also disabled.