ANNEX 7-A

DFEE SOFTWARE USER'S MANUAL

	SPI-DFEE Software User Manual	Doc : SAp-SPI-MD-266-99 Ed : 2 Rev : 0 Page : 1 SPI-MU-7-2836.SAP

SPI – DFEE

SOFTWARE USER MANUAL

Ed	Rev	Date	
1	0	21.01.99	Creation
2	0	19.10.00	

Préparé par	Développement Software	M.DONATI	Date	Signature
Approuvé par				
Approuvé par	Responsable Assurance Produit	F.LOUBERE		
Autorisé par	Chef de Projet	R.DUC		

Adresse postale: SAp/GERES, Bât 709, CEA Saclay - Orme, F-91191 Gif sur Yvette Cedex France
Fax: 33 (0)1 69 08 79 96 Secrétariat: 33 (0)1 69 08 78 27

1.Introduction

This software User Manual (SUM) applies to software implemented on SPI DFEE supervisor.

2.Procedure to build software

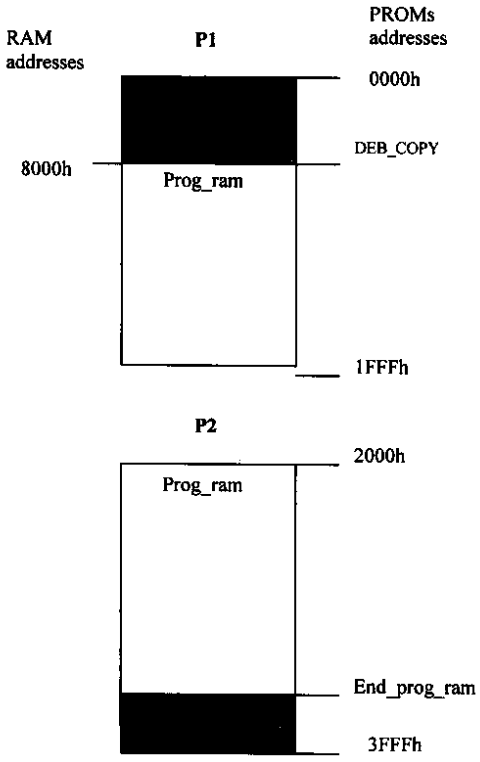
2.1 building software

Project name: on_board

To build on_board software, on_b.bat file have to be executed.
It makes compilation and link.
The addresses for link are defined in file : on_b.lin.
Programm is loaded into 2 PROMs named P1,P2.

Mapping of PROMs:

Software is in two parts : boot and prog_ram which is copied in RAM.
Prog_ram is linked with 1st address at 8000h (1st address of RAM).



Criterion on software size not to exceed PROMs capacity :

Last address of boot must be lower than : DEB_COPY.
Last address of software after link (End_prog_ram) must be lower than :
C000h - DEB_COPY.

DEB_COPY is defined in file : def.h.

In software issue b001005 DEB_COPY = E00h and End_prog_ram must not exceed B200h.

2.2 resulting files

The execution of on_b.bat produces the files:

- on_b.m51, resulting from link and giving implementation of code and data in the microcontroller memory.
- on_b.hex used to be loaded in Prom's.

Warnings after compilation :

files	Number of warnings	Type of warning
Main.c	1	'X_selftest' : unreferenced label
Maint.c	5	Pointer to different objects
It.c	4	Pointer to different objects

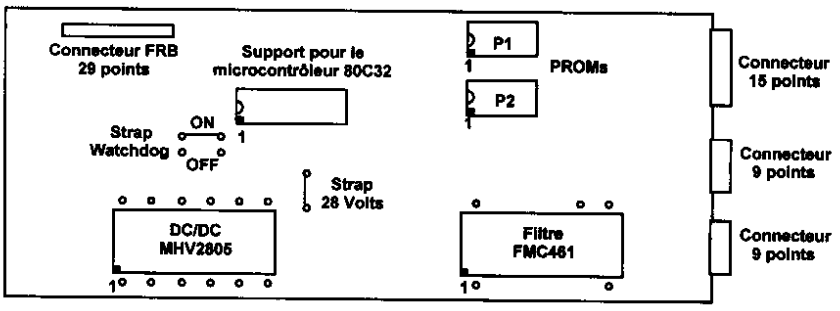
2.3 PROM programming

Software is loaded in 2 PROMs named P1 and P2.

Mapping of Supervisor memory is described in document :
'*Mapping DFEE*' SAp-SPI-MD-407-00

Procedure to programm the PROMs is described in document :
' *Programmation des PROMs du Superviseur de SPI à partir d'une EPROM 26C64* '
SEI-SPI-EZ-362-00

Location of PROMs on supervisor board :



3.Software description

3.1 Use of TC dump

Format of TC Dump:

'D'	MS byte	Mid byte	LS byte	Number of bytes	cks

3.1.1 Dump of microcontroller memory

Values of MS byte:

Value of MSB byte	Type of memory
00hex	internal memory (IDATA/BDATA)
01hex	external memory (XDATA)
FFhex	code memory (PROM)

Mid byte and LS byte contains the address value.

In case of dump of internal memory, only the Lsbyte is significant.

3.1.2 Dump of internal parameters of DFEE

3.1.2.1 Command

Internal parameters of DFEE are :

- content of ASIC registers
- Status of Software

- Result of autotest ASIC

Values of MS byte:

Value of MSB byte	Type of memory
02hex	Lowest 24 bytes of HK block
03hex	Highest 24 bytes of HK block
04hex	Soft error table

The only Lsbyte is significant. It contains the number of HK block to be dump.

3.1.2.2 Response

'D'	MS byte	Mid byte	LS byte	Number of bytes	24 bytes of data	cks

HK blocks contains the last values of ASIC registers , they are updated as described in "SPI DFEE Status Word SEI/SPI/STP/2000-0001.

In case of Soft error table, there are 16 significant bytes sent, which are the 16 first bytes sent in data area.

Description of Soft error table:

	MS bit							**LS bit**
1st byte sent	Cks_prom	Cks_ram	Cks_stby	Wdog	Not_real 8hz	Usart overrun	Not_real it receive	Not_real it transmit
	GSTAT 8hz					Bad Auto test	Bad Autotest Asic	
	Result of autotest ASIC							
	Wrong Counter VetoSat	*Wrong Counter Veto Special*	*Wrong CounterSat*	*Wrong Counter Special*	*Tst NoEnabled*	*Dereset AllExcept Acq*	*WrongIR0*	*DeviceId*
	Wrong DataRead Back	*Wrong PobjFullX orFull*	*Wrong PobjFullX orEmpty*	*Wrong PobjFullX orNot Empty*	*Wrong PobjXor Empty*	*Wrong PobjXor NotEmpty*	*Wrong PobjData Itm*	*Wrong PobjData Lbl*

	MS bit							LS bit
	SysTstStill Enabled	*DataRun Mismatch TF3or More*	*DataRun Mismatch TF2*	*DataRun Mismatch TF1*	*DataRun Mismatch TF0*	*TstHslEnv NotOff*	*TstHslEnv NotOn*	*GoNot Passed*
	Error level and autotest complete							
	Number of channels							
	Number of Issue Ccst							
	Number of Issue Soft							
	Issue Soft Year Month							
	Issue Soft Day							
	Device ID (line0 of ASIC)							
Last byte send	Checksum PROM							

The following table indicates the signification and the set and reset conditions of the above bits.

Status field	Set Conditions	Reset Conditions
Cks_prom	Checksum of PROM is not equal to checksum of code loaded inRAM	Never reset during soft execution
Cks_ram	Checksum of PROM is not equal to checksum of code loaded inRAM	Never reset during soft execution
Cks_stby	Checksum of code loaded in Ram is calculated in main loop in **stand by** mode and compared with checksum calculated at **boot** time. Set if not equal	At the end of transmission of dump Status_soft_table
Wdog	Set if watchdog is not activated by	Never reset during soft execution

	software at least since 14s	
Not_real 8hz	Last it 8hz was an it parasite	At the end of transmission of dump Status_soft_table
Usart overrun	Copy of bit set in Usart status in case of Usart Overrun	At the end of transmission of dump Status_soft_table
Not_real it receive	Last it receive character was an it parasite	At the end of transmission of dump Status_soft_table
Not_real it transmit	Last it transmit character was an it parasite	At the end of transmission of dump Status_soft_table
GSTAT 8hz	Every time microcontroller receives it 8hz	At every second boundary
Bad Auto test	At boot time if result of autotest ASIC is bad	Never reset during soft execution
Bad Autotest Asic	At boot time if result of autotest ASIC is bad	Never reset during soft execution
Number of channels	At boot time	Never reset during soft execution
Number of Issue Ccst	At boot time	Never reset during soft execution
Number of Issue Soft	At boot time	Never reset during soft execution
Issue Soft Year Month	At boot time	Never reset during soft execution
Issue Soft Day	At boot time	Never reset during soft execution
Device ID	At boot time and after restart, Value of line_0 of ASIC	Refresh in case of restart
Checksum PROM	At boot time Result of calculation of PROM checksum	Never reset during soft execution

The following table indicates the signification of the bits handled by autotest of ASIC.

Status field	Description
DeviceId	ASIC Device ID is not found to be DFEEA51C
WrongIR0	Instruction Register not on line 0 (Device ID)

DeresetAllExceptAcq	Dereset all (except Acq) did not pass
TstNoEnabled	HSL or SYS test could not be enabled
WrongCounterSpecial	AFEE Counters don't have the right value (10000…01) after increment
WrongCounterSat	AFEE Counters did not reach saturation value (11111…11) after increment
WrongCounterVetoSpecial	VETO Counters don't have the right value (10000…01) after increment
WrongCounterVetoSat	VETO Counters did not reach saturation value (11111…11) after increment
WrongPobjDataLbl	Pobj Fifo has not reached condition *Full and Not Empty* after filling Fifo with items *or* correct Event Label could not be written.
WrongPobjDataItm	Pobj Fifo has not reached condition *Empty* after flushing Fifo *or* last Item read is incorrect.
WrongPobjXorNotEmpty	Pobj Fifo, after writing 128 Labels and extracting 127 words, does not have correct checksum or has *Empty* condition active.
WrongPobjXorEmpty	Pobj Fifo, after extracting 1 more word, does not have checksum 0000…00 or has not *Empty* condition active.
WrongPobjFullXorNotEmpty	Pobj bit-shift test. Each of the 26 bits is tested for read/write values 0/1. After writing 128 Labels and extracting 127 words, checksum is incorrect or *Empty* condition active on a bit.
WrongPobjFullXorEmpty	After extracting 1 more word, checksum is incorrect or *Empty* condition inactive on a bit.
WrongPobjFullXorFull	Pobj data test. Write 128 different words. Test Register read does not give last word or Fifo *Full* condition not reached.
WrongDataReadBack	Read back operation of one of the 128 words is erroneous.
GoNotPassed	Normal test run. After normal test run condition, de-reset and start of run, *Go* did not pass.
TstHslEnvNotOn	Setting a HSL test envelope did not pass.
TstHslEnvNotOff	After HSL data extraction, HSL test envelope

	could not be closed.
DataRunMismatchTF0	Mismatch in HSL data packet for Test Run Time Frame 0
DataRunMismatchTF1	Mismatch in HSL data packet for Test Run Time Frame 1
DataRunMismatchTF2	Mismatch in HSL data packet for Test Run Time Frame 2
DataRunMismatchTF3orMore	Mismatch in HSL data packet for Test Run Time Frame 3
SysTstStillEnabled	After restoration of normal mode, Test mode still enabled.
Error level and autotest complete	

3.2 Use of TC load

'L'	MS byte	Mid byte	LS byte	Number of bytes	24 bytes of data	cks

Values of MS byte:

Value of MSB byte	Type of memory
00hex	internal memory (IDATA/BDATA)
01 hex	external memory (XDATA)
02 hex	ASIC registers

In case of memory, Mid byte and LS byte contains the address value.

Load in internal memory has to be use with care, because it contains data used by software.

In case of ASIC registers, Lsbyte contains the n° of asic register to load.

3.3 Acknowledge

Bits 3 and 4 of acknowledge which are specified as don't care are used by software :

bit n°3 is set to 1 during software is in maintenance mode.

Bitn°4 is set to 1 when a stack TC overflow occurs.

3.4 MAINTENANCE Mode

Patchable areas:

All code situated in RAM memory, that is from address 8000h to FFFFh is patchable.
List of data and code location is in file : on_bprom.m51 (resulting from link)

4 Execution Times

Boot times

Boot time is: 8,3s
time of autotest ASIC : 1,43s

TC execution times

- TC Configuration :
  Conf 1 : 12 µs
  Conf 2 : 4,5ms
  Conf 3 : 10ms

- TC of changing mode:
  CONF : 205 µs
  START : 4ms
  DIAG : 4ms
  STBY : 2ms

- TC Init : 8,3s
  TC stop maintenance : 4ms
  TC load : 6,2ms (for 24 bytes loaded)

Use of stack_TC :

- A maximum of 8 TC can be stacked up .
- A maximum of 40 TC load can be sended with minimum delay between each sending as described in Annex 1, without over run of stack.

5.Default Configuration

HK buffers and configuration buffers are initalized to 0 at boot time like the whole data memory.

Configuration values are not saved after TC stop maintenance.