

## LFR-FSW - Task #3199

### Analyse couverture de tests 3.2.0.24

13/11/2018 10:46 AM - Veronique bouzid

<b>Status:</b>	Closed	<b>Start date:</b>	13/11/2018
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Veronique bouzid	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Spent time:</b>	0.00 hour
<b>revision:</b>	r3.2.0.24		
<b>Description</b>			
<p>Tout se passe sur la machine rangiroa rep = /home/bouzid/LFR/GCOV_3.2.0.24 ensuite les répertoires de test /home/bouzid/LFR/GCOV_3.2.0.24/SVS-xxxx.</p> <p>Les répertoires SVS comprenant plusieurs scripts de test sont traités indépendamment, par exemple SVS-0012 SVS-0012-step2 SVS-0012-burst SVS-0012-normal SVS-0012-sbm1 SVS-0012-sbm2</p> <p>Les variables d environnement necessaires sont export DROOT=\${HOME}/LFR export SGCOV=\${DROOT}/LFR_FSW/libgcov/build_gcov_files.py export DBUILD=\${DROOT}/build-LFR_FSW-Desktop_Qt_5_11_2_GCC_64bit-Debug/ export DTEST=\${DROOT}/GCOV_3.2.0.24</p> <p>Chaque répertoire de test a son analyse gcov unitaire. On se met au debut de l arborescence de test cd \${DTEST} \${SGCOV} -r \${DBUILD} -o /home/bouzid/LFR/GCOV_3.2.0.24/SVS-0053/ /home/bouzid/LFR/GCOV_3.2.0.24/SVS-0053/gcov_out_2018-11-13\ 08\:49\:49.309256.txt puis on execute le script /home/bouzid/LFR/lance_rapport SVS-xxxx.</p> <p>Les fichiers*.html seront créés dans le repertoire de test .</p> <p>Ensuite pour obtenir l analyse complète, on se met dans le répertoire /home/bouzid/LFR/GCOV_3.2.0.24 et on lance la commande ../LFR_FSW/libgcov/gcovr.py -s ../LFR_FSW -o . -g /opt/rtems-4.10/bin/sparc-rtems-gcov . ou bien le script /home/bouzid/LFR/lance_final</p> <p>Liste des tests joués SVS-0002 à SVS-0014 SVS-0018 à SVS-0022 SVS-0026 à SVS-0034 SVS-0038 SVS-0040 à SVS-0044 SVS-0053 à SVS-0057 SVS-0059 SVS-0064 à SVS-0065 SVS-0067 SVS-0069 SVS-0070 SVS-0071 SVS-0073 SVS-0074 SVS-0076 à SVS-0079 SVS-0081 à SVS-0083 SVS-0086 SVS-0089 à SVS-0091 SVS-0095 SVS-0096</p>			
<b>Related issues:</b>			
Related to Task #3168: Livraison de 3.2.0.24 avec gcov		<b>Closed</b>	<b>30/10/2018</b>

### History

#### #1 - 13/11/2018 11:18 AM - Veronique bouzid

- Description updated

#### #2 - 13/11/2018 11:24 AM - Veronique bouzid

- File gcov.html added

#### Fichier fsw\_misc.c.

##### Les fonctions non atteintes

```
void timer_stop(unsigned char timer) ligne 87
rtems_isr watchdog_isr( rtems_vector_number vector ) ligne 116
void watchdog_stop(void) ligne 145
rtems_task scrubbing_task( rtems_task_argument unused ) ligne 569
void set_hk_lfr_ahb_correctable() // CRITICITY L ligne 1075
```

les lignes de code suivantes ne sont pas atteintes

isSynchronized = true; ligne 314  
spare\_status = rtems\_event\_send( Task\_id[TASKID\_DUMB], RTEMS\_EVENT\_6 ); ligne 332

dans la fonction void set\_hk\_lfr\_sc\_potential\_flag( bool state )  
ligne 840/841 non atteintes

dans la fonction void set\_sy\_lfr\_watchdog\_enabled( bool state )  
ligne 868/869 non atteintes

### #3 - 14/11/2018 10:07 AM - Veronique bouzid

- Status changed from New to In Progress

#### Fichier fsw\_processing.h

Dans la fonction ASM\_reorganize\_and\_divide (ligne 265)  
l instruction n est jamais atteinte (ligne 289)  
averaged\_spec\_mat\_reorganized[offsetASMRreorganized ] = INIT\_FLOAT;

### #4 - 14/11/2018 10:56 AM - Veronique bouzid

#### Dans le fichier basic\_parameters.h

dans la fonction BP1\_set (ligne 65)  
les instructions suivantes ne sont pas atteintes  
lignes 129-130  
exponent = expmin;  
significand = 0.5; // min value that can be recorded  
et  
lignes 133-134  
exponent = expmax;  
significand = 1.0; // max value that can be recorded  
et  
lignes 180-181  
exponent = expmin;  
significand = 0.5; // min value that can be recorded  
et  
lignes 184-185  
exponent = expmax;  
significand = 1.0; // max value that can be recorded  
et  
lignes 358-359  
exponent = expmax;  
significand = 1.0; // max value that can be recorded

Dans la fonction BP2\_set (ligne 534)  
les instructions suivantes ne sont pas atteintes  
lignes 780-781  
exponent = expmin;  
significand = 0.5; // min value that can be recorded

lignes 784-785  
exponent = expmax;  
significand = 1.0; // max value that can be recorded

lignes 870-871  
lignes 874-875  
lignes 916-917  
lignes 920-921  
lignes 961-962  
lignes 965-966

### #5 - 14/11/2018 11:02 AM - Veronique bouzid

#### fichier fsw\_init.c

Dans la fonction Init (ligne 110)

ligne 302  
status = rtems\_event\_send( Task\_id[TASKID\_SPIQ], SPW\_LINKERR\_EVENT );

Dans la fonction create\_message\_queues (ligne 796)  
les lignes non atteintes ( erreur sur la creation de la queue de message  
ret = status\_rcv; ligne 856  
ret = status\_send; ligne 860  
ret = status\_q\_p0; ligne 864  
ret = status\_q\_p1; ligne 868

## #6 - 14/11/2018 12:01 PM - Veronique bouzid

- File *gcov.src\_fsw\_spacewire.c.html* added

### Fichier *fsw\_spacewire.c*

lignes non atteintes

Dans la fonction *send\_task* (236)

lignes 355 356

```
spw_ioctl_send = (spw_ioctl_pkt_send*) incomingData;  
status = ioctl( fdSPW, SPACEWIRE_IOCTL_SEND, spw_ioctl_send );
```

Dans la fonction *spacewire\_several\_connect\_attempts* (ligne 536)

ligne 570

```
i = SY_LFR_DPU_CONNECT_ATTEMPT;  
SY_LFR_DPU_CONNECT_ATTEMPT = 3
```

Dans la fonction *spacewire\_set\_NP* (ligne 577)

ligne 594

```
*spwptr = *spwptr & SPW_BIT_NP_MASK;
```

Dans la fonction *spacewire\_set\_RE* (ligne 598)

ligne 617

```
*spwptr = *spwptr & SPW_BIT_RE_MASK;
```

Dans la fonction *spacewire\_get\_last\_error* (ligne 672)

tous les tests liés aux erreurs (CODE\_HEADER\_CRC, CODE\_DATA\_CRC, CODE\_EEP, CODE\_CREDIT, CODE\_WRITE\_SYNC, CODE\_EARLY\_EOP\_EEP)

Dans la fonction *check\_timecode\_and\_previous\_timecode\_coherency* (ligne 854)

ligne 881

```
ret = LFR_DEFAULT;
```

pour info, l'analyse du code retour de cette fonction incrémentera le champ *HK\_LFR\_TIMECODE\_ERRONEOUS*.

Dans la fonction *timecode\_irq\_handler* (ligne 923)

ligne 977

```
rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_14 );
```

Dans la fonction *timecode\_timer\_routine* (ligne 981)

les lignes 1016 à 1020 ne sont pas atteintes.

Explication:

En fait ces lignes sont utilisées dans la phase de boot donc on ne le voit pas. La majorité des tests montrent que la première HK recue après la séquence de boot montre que le champ *HK\_LFR\_TIMECODE\_MISSING=1* et que la dernière erreur tracée est décrite avec *HK\_LFR\_LAST\_ER\_RID=* et *42129 HK\_LFR\_LAST\_ER\_CODE=21*

Dans la fonction *spw\_send\_waveform\_CWF* (ligne 1118)

ligne 1196

```
ret = LFR_DEFAULT;
```

Dans la fonction *spw\_send\_waveform\_SWF* (ligne 1203)

ligne 1285

```
ret = LFR_DEFAULT;
```

dans la fonction *spw\_send\_waveform\_CWF3\_light*

ligne 1365

```
ret = LFR_DEFAULT;
```

## #7 - 14/11/2018 12:18 PM - Veronique bouzid

- File *gcov.LFR\_basic\_parameters\_basic\_parameters.c.html* added

- File *gcov.src\_...header\_GscMemoryLPP.hpp.html* added

- File *gcov.src\_lfr\_cpu\_usage\_report.c.html* added

- File *gcov.src\_processing\_avf0\_prc0.c.html* added

- File *gcov.src\_processing\_avf1\_prc1.c.html* added

- File *gcov.src\_processing\_avf2\_prc2.c.html* added

- File *gcov.src\_processing\_fsw\_processing.c.html* added

- File *gcov.src\_tc\_acceptance.c.html* added

- File *gcov.src\_tc\_handler.c.html* added

- File *gcov.src\_tc\_load\_dump\_parameters.c.html* added

**#8 - 14/11/2018 12:22 PM - Veronique bouzid**

- File `gcv.src_.._LFR_basic-parameters_basic_parameters.h.html` added
- File `gcv.src_fsw_init.c.html` added
- File `gcv.src_fsw_misc.c.html` added
- File `gcv.src_tm_lfr_tc_exe.c.html` added
- File `gcv.src_wf_handler.c.html` added

**#9 - 14/11/2018 12:23 PM - Veronique bouzid**

- File `gcv.src_.._header_GscMemoryLPP.hpp.html` added

**Pour le fichier GscMemoryLPP.hpp**  
on ne peut faire mieux.

**#10 - 14/11/2018 12:40 PM - Veronique bouzid**

**Fichier fsw\_processing.c**

dans la fonction `spectral_matrices_isr_f0` (ligne 88)  
lignes non atteintes  
lignes 102-104  
`spectral_matrix_regs->status = BIT_READY_0_1; // [0011]`  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_11 );`  
`break;`

ligne 118  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_3 );`

ligne 137  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_3 );`

Dans la fonction `spectral_matrices_isr_f1` (ligne 148)  
lignes 162-164  
`spectral_matrix_regs->status = BITS_STATUS_F1; // [1100]`  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_11 );`  
`break;`  
ligne 178  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_3 );`  
ligne 197  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_3 );`

Dans la fonction `spectral_matrices_isr_f2` (ligne 208)  
lignes 221-223  
`spectral_matrix_regs->status = BITS_STATUS_F2; // [0011 0000]`  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_11 );`  
`break;`  
ligne 233  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_3 );`  
ligne 245  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_3 );`

Dans la fonction `spectral_matrix_isr_error_handler` (ligne 253)  
ligne 270  
`status_code = rtems_event_send( Task_id[TASKID_DUMB], RTEMS_EVENT_8 );`

Dans la fonction `getSID` (ligne 571)  
ligne 604  
`sid = 0;`

**#11 - 14/11/2018 01:21 PM - Veronique bouzid**

**Fichier tc\_acceptance.c**

Dans la fonction `tc_parser` (ligne 120)  
ligne 179  
`status = WRONG_LEN_PKT;`  
Impossible à atteindre car le spacewire rejete les packet > 228 octets en amont de ce test.  
CCSDS\_TC\_PKT\_MAX\_SIZE = 232

Dans la fonction `tc_check_type_subtype` (ligne 229)  
ligne 277  
`status = ILL_SUBTYPE;`

Impossible d'atteindre car cette ligne qui correspond a un packet type qui vaut ni TC\_TYPE\_GEN ni TC\_TYPE\_TIME.  
Comme la validation du type est faite avant la validation du sub-type, on ne rentre jamais dans ce cas.

```
Extrait de la fonction tc_parser()
if (status == CCSDS_TM_VALID) // CHECK THE TYPE {
status = tc_check_type( packetType );
}
if (status == CCSDS_TM_VALID) // CHECK THE SUBTYPE {
status = tc_check_type_subtype( packetType, packetSubtype );
```

Dans la fonction tc\_check\_length ligne (314)  
ligne 460  
status = ILL\_SUBTYPE;

Impossible à atteindre car quand on entre dans cette fonction, le sub-type est valide.

**#12 - 14/11/2018 01:48 PM - Veronique bouzid**

**Dans le fichier tc\_handler.c**

Dans la fonction actn\_task (ligne 42)  
ligne 100-101  
close\_action( &TC, result, queue\_snd\_id );  
break;  
Impossible à atteindre car TC\_LFR\_RESET est utilisée pour dumper la couverture GCOV.

Dans la fonction action\_enter\_mode (ligne 197)  
ligne 269  
status = LFR\_EXE\_ERROR;

Dans la fonction check\_mode\_transition (448)  
ligne 506  
status = LFR\_DEFAULT;  
Quand la fonction check\_mode\_transition est appelée, on est certain que le requestedmode est valide.

Dans la fonction enter\_mode\_normal (700)  
ligne 761  
status = RTEMS\_UNSATISFIED;  
Impossible de provoquer des erreurs RTEMS

Dans la fonction enter\_mode\_burst (767)  
ligne 801  
status = RTEMS\_UNSATISFIED;  
Impossible de provoquer des erreurs RTEMS

idem pour les fonctions enter\_mode\_sbm1 (ligne 807) et enter\_mode\_sbm2 (ligne 875)

Dans la fonction restart\_science\_tasks (ligne 943)  
ligne 1828  
status = RTEMS\_UNSATISFIED;  
Impossible de provoquer des erreurs RTEMS

Dans la fonction restart\_asm\_tasks (ligne 1034)  
ligne 1093  
status = RTEMS\_UNSATISFIED;  
Impossible de provoquer des erreurs RTEMS

Dans la fonction set\_sm\_irq\_onError (ligne 1382)  
ligne 1386  
spectral\_matrix\_regs->config = spectral\_matrix\_regs->config | BIT\_IRQ\_ON\_ERROR;  
Le seul appel à cette fonction est fait par la fonction reset\_spectral\_matrix\_regs  
set\_sm\_irq\_onError(0);  
avec value =0 .  
Donc normal que le code associé au test de value==1 ne soit pas activé.

Dans la fonction close\_action (ligne 1623)  
ligne 1655  
send\_tm\_lfr\_tc\_exe\_error( TC, queue\_id );  
Impossible de generer une LFR\_EXE\_ERROR

**#13 - 14/11/2018 01:56 PM - Veronique bouzid**

**Fichier tm\_lfr\_tc\_exe.c**

La fonction send\_tm\_lfr\_tc\_exe\_error (ligne 311) n'est pas atteinte.  
Impossible de generer cette TM\_LFR\_TC\_EXE\_ERROR.

**#14 - 14/11/2018 02:15 PM - Veronique bouzid**

**Fichier wf\_handler.c**

La fonction getRingNodeToSendSWF (ligne 101) n a pas été atteinte  
Aucun appel à cette fonction dans le code

La fonction set\_wfp\_delta\_f1 (ligne 1238) n a pas été atteinte  
Aucun appel à cette fonction dans le code

Dans la fonction waveforms\_isr\_f3 (ligne 133)  
ligne 158  
spare\_status = rtems\_event\_send( Task\_id[TASKID\_DUMB], RTEMS\_EVENT\_0 );

Dans la fonction waveforms\_isr\_burst (ligne 164)  
ligne 181  
spare\_status = rtems\_event\_send( Task\_id[TASKID\_DUMB], RTEMS\_EVENT\_0 );  
ligne 193  
spare\_status = rtems\_event\_send( Task\_id[TASKID\_DUMB], RTEMS\_EVENT\_0 );

Dans la fonction waveforms\_isr (ligne 284)  
ligne 314  
spare\_status = rtems\_event\_send( Task\_id[TASKID\_DUMB], RTEMS\_EVENT\_0 );

Dans la fonction send\_waveform\_CWF3\_light (ligne 681)  
ligne 724  
ret = LFR\_DEFAULT;  
Impossible de generer une erreur RTEMS

Dans la fonction compute\_acquisition\_time (ligne 730)  
ligne 789  
deltaT = 0.;

Dans la fonction build\_snapshot\_from\_ring (ligne 804)  
lignes 857-860  
acquisitionTime\_asLong = centerTime\_asLong;  
nb\_ring\_nodes = 0;  
frequency\_asLong = FREQ\_F2;  
nbTicksPerSample\_asLong = TICKS\_PER\_T2;  
break;

ligne 908  
nbSamplesPart1\_asLong = 0;

Dans la fonction set\_wfp\_burst\_enable\_register (ligne 1175)  
ligne 1199  
waveform\_picker\_regs->run\_burst\_enable = INIT\_CHAR; // [0000 0000] no burst enabled, no waveform enabled

Dans la fonction increment\_seq\_counter\_source\_id (ligne 1286)  
ligne 1337  
sequence\_cnt = (unsigned short \*) NULL;  
Aucune idée pour tester cela, cela signifierait que le source\_id d'une TM soit erronée....

**#15 - 21/11/2018 10:06 AM - Veronique bouzid**

- Related to Task #3168: Livraison de 3.2.0.24 avec gcov added

**#16 - 13/12/2018 07:54 AM - Veronique bouzid**

Ce rapport est celui inclus dans le verification report (Chapitre 6).

**#17 - 01/04/2019 11:41 AM - bruno katra**

- Status changed from In Progress to Closed

Analyse GCOV livrée dans DP 3.2

**Files**

gcov.html	20.5 KB	13/11/2018	Veronique bouzid
gcov.src_fsw_spacewire.c.html	466 KB	14/11/2018	Veronique bouzid
gcov.LFR_basic-parameters_basic_parameters.c.html	67.2 KB	14/11/2018	Veronique bouzid
gcov.src..._header_GscMemoryLPP.hpp.html	65.8 KB	14/11/2018	Veronique bouzid

gcov.src_lfr_cpu_usage_report.c.html	25.4 KB	14/11/2018	Veronique bouzid
gcov.src_processing_avf0_prc0.c.html	135 KB	14/11/2018	Veronique bouzid
gcov.src_processing_avf1_prc1.c.html	130 KB	14/11/2018	Veronique bouzid
gcov.src_processing_avf2_prc2.c.html	106 KB	14/11/2018	Veronique bouzid
gcov.src_processing_fsw_processing.c.html	259 KB	14/11/2018	Veronique bouzid
gcov.src_tc_acceptance.c.html	147 KB	14/11/2018	Veronique bouzid
gcov.src_tc_handler.c.html	475 KB	14/11/2018	Veronique bouzid
gcov.src_tc_load_dump_parameters.c.html	598 KB	14/11/2018	Veronique bouzid
gcov.src..._LFR_basic-parameters_basic_parameters.h.html	315 KB	14/11/2018	Veronique bouzid
gcov.src_fsw_init.c.html	289 KB	14/11/2018	Veronique bouzid
gcov.src_fsw_misc.c.html	313 KB	14/11/2018	Veronique bouzid
gcov.src_wf_handler.c.html	392 KB	14/11/2018	Veronique bouzid
gcov.src_tm_lfr_tc_exe.c.html	158 KB	14/11/2018	Veronique bouzid
gcov.src..._header_GscMemoryLPP.hpp.html	65.8 KB	14/11/2018	Veronique bouzid