

```

1  --- X:/src/SSP-Shulink-RX62N/ssp/task_manage.c 水 6 11 18:47:14 2014
2  +++ X:/src/WAIT-SSP/ssp/task_manage.c 日 6 15 08:53:40 2014
3  @@ -40,6 +40,7 @@
4  */
5  #include <stddef.h>
6  #include <limits.h>
7  + #include <setjmp.h>
8
9  /*
10 *      タスク管理機能
11 @@ -83,6 +84,7 @@ act_tsk(ID tskid)
12 {
13     ER        ercd;
14     uint_t    tskpri;
15 + uint_t    self_tskpri; // 呼び出しタスク
16
17     LOG_ACT_TSK_ENTER(tskid);
18     CHECK_TSKCTX_UNL();
19 @@ -89,11 +91,20 @@ act_tsk(ID tskid)
20     CHECK_TSKID_SELF(tskid);
21
22     tskpri = get_ipri_self(tskid);
23 -
24 + self_tskpri = get_ipri_self(TSK_SELF);
25     t_lock_cpu();
26     if (test_dormant(tskpri)) {
27         if (make_active(tskpri)) {
28 -         run_tsk(tskpri);
29 +         //----- ここから-----
30 +         // このコンテキストを登録
31 +         if (setjmp(task_ctx[self_tskpri]) == 0)
32 +         {
33 +             /* 登録した場合*/
34 +             dispatch(tskpri);
35 +         }
36 +         //----- ここまで-----
37 +
38 +         //dispatch(tskpri);
39     }
40     ercd = E_OK;
41 }
42 @@ -103,7 +114,6 @@ act_tsk(ID tskid)
43     t_unlock_cpu();
44
45     error_exit:
46 - LOG_ACT_TSK_LEAVE(ercd);
47     return(ercd);
48 }
49
50 @@ -143,4 +153,6 @@ iact_tsk(ID tskid)
51     return(ercd);
52 }
53
54 +
55 +
56 #endif /* TOPPERS_iact_tsk */
57

```