Contents

clawson_FreeRTOS_SAVE_CONTEXT.png

clawson   Posted : Wed. May 26, 2010 - 11:24 AM   ★★★★★ Total votes: 0   #2

SPL and SPH are IO registers, not CPU addressable registers like R0..R13 so you need to use IN/OUT to access the contents. As about the only reason a C programmer ever needs to access SP is for an RTOS context switch here's the context switch from FreeRTOS as an example:

Level: Moderator
Joined: Mon. Jul 18, 2005
Posts: 98779 View posts
Location: (using avr-gcc in)
Finchingfield, Essex, England

```
#define portSAVE_CONTEXT()                                                  \
    asm volatile (  "push   r0                              \n\t"   \
                    "in         r0, __SREG__         \n\t"   \
                    "cli                                     \n\t"   \
                    "push   r0                               \n\t"   \
                    "push   r1                               \n\t"   \
                    "clr    r1                               \n\t"   \
                    "push   r2                               \n\t"   \
                    "push   r3                               \n\t"   \
                    "push   r4                               \n\t"   \
                    "push   r5                               \n\t"   \
                    "push   r6                               \n\t"   \
                    "push   r7                               \n\t"   \
                    "push   r8                               \n\t"   \
                    "push   r9                               \n\t"   \
                    "push   r10                              \n\t"   \
                    "push   r11                              \n\t"   \
                    "push   r12                              \n\t"   \
                    "push   r13                              \n\t"   \
                    "push   r14                              \n\t"   \
                    "push   r15                              \n\t"   \
                    "push   r16                              \n\t"   \
                    "push   r17                              \n\t"   \
                    "push   r18                              \n\t"   \
                    "push   r19                              \n\t"   \
                    "push   r20                              \n\t"   \
                    "push   r21                              \n\t"   \
                    "push   r22                              \n\t"   \
                    "push   r23                              \n\t"   \
                    "push   r24                              \n\t"   \
                    "push   r25                              \n\t"   \
                    "push   r26                              \n\t"   \
                    "push   r27                              \n\t"   \
                    "push   r28                              \n\t"   \
                    "push   r29                              \n\t"   \
                    "push   r30                              \n\t"   \
                    "push   r31                              \n\t"   \
                    "lds    r26, pxCurrentTCB        \n\t"   \
                    "lds    r27, pxCurrentTCB + 1    \n\t"   \
                    "in         r0, 0x3d                     \n\t"   \
                    "st         x+, r0                       \n\t"   \
                    "in         r0, 0x3e                     \n\t"   \
                    "st         x+, r0                       \n\t"   \
                );
```

3D and 3E in this are the SP register halves