



An Efficient Context Switch for Cooperative Multi-Tasking

*Author: Mark Sullivan
Joplin, MO
USA
msullivan@niobrara.com*

This technique has many applications. Examples include: a software UART, software PWM, software pin interrupt, background timing etc., running concurrently with other tasks.

My method has a very small task switching overhead, but is limited in that tasks must all fit in one 256 byte page; tasks cannot relinquish the MCU inside a sub-routine, and the RTOS does not preserve W, INDF, or the flags. Only the PC is saved and restored. The technique is easily extended to save other registers.

INTRODUCTION

This technique allows a number of threads to share the PICmicro™ MCU. Each can consume a fixed number of cycles between relinquishes or the RTOS can monitor the RTCC to invoke the next task according to real-time, as long as each task limits itself to a maximum number of MCU cycles before relinquishing the MCU.

Here's a skeletal example:

```
CBLOCK
ContextA          ; PC for task A
ContextB          ; PC for task B
EndC

Initialize
    movlw    TASKA
    movwf   CONTEXTA
    movlw    TASKB
    movwf   CONTEXTB
    goto    RTOS

RTOS
    ;add code here to synchronize with the real time if desired
    movf    CONTEXTA,W      ; fetch A's PC
    call    CALLTASK        ; invoke task A
    movwf   CONTEXTA       ; save context

    movf    CONTEXTB,W
    call    CALLTASK
    movwf   CONTEXTB
    goto    RTOS

CALLTASK
    movwf   PC,F
    macro   RELINQUISH
    retlw   $ + 1          ; return PC of next instruction
    EndM

TASKA
    RELINQUISH          ; initialize task A stuff

LOOPA
    RELINQUISH          ; task A does some processing

    RELINQUISH          ; etcetra
    goto    LOOPA

TASKB
    ; similar to above
```

Microchip Technology Incorporated, has been granted a nonexclusive, worldwide license to reproduce, publish and distribute all submitted materials, in either original or edited form. The author has affirmed that this work is an original, unpublished work and that he/she owns all rights to such work. All property rights, such as patents, copyrights and trademarks remain with the author.

