

# CSCI E-92: Application Note 18

## Stack Contents for New Process with –Os (Optimize for Size)

*This document assumes that the printf statement has been removed from the SysTick ISR.*

Stack Contents Pushed by “ldr r0, [%shcsr]; mov r1, %[active]; orr r1, r1, %[pended]; and r0, r0, r1; push {r0}”  
 This is Top Of Stack (TOS) at lowest address; SP points to this word  
 (see AN6: QuantumInterruptInfo.txt)

Word with SVCALLACT & SVCALLPENDEDED bits

Stack Contents Pushed by “push {r4,r5,r6,r7,r8,r9,r10,r11}”  
 (see StackManipulationInAssembler/main.c; If R4 is saved by your SysTick function entry code, it does not need to be in the “push” register list above and the corresponding “pop” register list)

R4 (This value of R4 is replaced by the value of R4 saved below)
R5
R6
R7
R8
R9
R10
R11

Stack Contents Pushed by Entry Code to SysTick Handler  
 (see StackManipulationInAssembler/main.c, AN7: SysTickTimer.txt, and AN8: PendSVInterrupt.txt)

R4 (This is the value of R4 for the new process)
LR (R14) (This is the LR popped into PC by the SysTick function exit code)

Stack Frame Pushed by Processor when Acknowledging Interrupt  
 If No FP Extension (see svc.c and AN10: CauseOfDefaultISR.txt)

R0
R1
R2
R3
R12
LR (R14) (This is the LR established when the new process starts to run)
Return Address (This is the address of the first instruction to run in the new process)
xPSR (bit 9 indicates the presence of a reserved alignment word at offset +32)
Possible Reserved Word for Alignment on 8 Byte Boundary