

# DESIGNER'S NOTEBOOK



## 'C3x Block Repeat

Contributed by Alex Tessarolo

### Design Problem

The setup time for a repeat block is four cycles. How can I move most of the setup to an initialization phase and reduce the overhead during algorithm execution?

### Solution

The repeat block requires that: (1) the RM bit in the status register be set, and (2) the RE, RS, and RC registers be loaded. All but the RM can be pre-initialized. During program execution, the RM bit can be set with an OR instruction.

Figure 1 shows examples for comparison. The algorithm in Figure 1.b, which contains the pre-initialized RE and RS registers, will execute faster. If executed repeatedly, the cycle savings could be significant.

a. Standard RPTB Initialization	b. Faster RPTB Execution
<pre> .text . . LDI    N-1,RC RPTB   InnerLoop      (1) .       ; first loop inst . . InnerLoop: .       ; last loop inst . . </pre> <p>Note 1: RPTB InnerLoop is a 4-cycle instruction. Note 2: OR 0100h,ST is a 1-cycle instruction.</p>	<pre> ; initialize pointers to RS and RE .data RPTBEndAddr .word RPTBEnd RPTBStartAddr .word RPTBStart ; initialize RE and RS .text LDP    RPTBEndAddr LDI    @RPTBEndAddr,RE LDP    RPTBStartAddr LDI    @RPTBStartAddr,RS . . .text . . LDI    N-1,RC OR     0100,ST ;RM =1 (2) RPTBStart .       ; first loop inst . . RPTBEnd .       ; last inst . . </pre>

Figure 1. Comparison of RPTB algorithms