CHAPTER 3: WRITING FUNCTIONS IN ASSEMBLY

Section 3.4 (Function Return Value), page 45, Figure 3-9, right-hand column:
Replace next to last instruction LDR R2,=save8 with LDR R2,=save64

CHAPTER 8: MULTIPLICATION AND DIVISION REVISTED

Section 8.3 (Division by an Arbitrary Constant), page 171, right-hand column (code for “Divides by -7”):
Replace the instruction, “ADD R0,R1,R0,LSR 31” by “ADD R0,R1,R1,LSR 31”

Section 8.3 (Division by an Arbitrary Constant), page 171, paragraph that starts at the bottom of the page:
Replace the entire paragraph by the following:
The SMMUL instruction computes the most-significant half of a 64-bit signed product and eliminates the need to use register R2 in the previous code. The SMMLA instruction does the same, but also adds the value of another 32-bit operand to the result. This instruction can thus replace the SMULL/ADDS.N sequence in the earlier code that divides by +7. The SMMLLS instruction computes the difference of a 32-bit operand less the most-significant half of a 64-bit operand. Unfortunately, this doesn’t help simplify the code that divides by -7. There are no unsigned versions of these instructions.

Section 8.3 (Division by an Arbitrary Constant), page 172, Table 8-6:
In the third column of the table, second row, replace “Rd ← (Rn × Rm) <63..32> + Ra” by “Rd ← Ra + (Rn × Rm) <63..32>”.
In the third column of the table, third row, replace “Rd ← (Rn × Rm) <63..32> − Ra” by “Rd ← Ra − (Rn × Rm) <63..32>”.

CHAPTER 11: FIXED-POINT REALS

Section 11.5.6 (Multiplying a Q32 Value by a Fractional Value), page 250, Figure 11-7:
The wrong partial products are indicated as being equal to zero (“= 0”). Those that are zero are only those that are cross-hatched and have a large “X” on top of them.