MAXIM

ERRATA SHEET
DS2780

_____ *Revision A2 Errata*

The errata listed below describe situations where DS2780 revision A2 components perform differently than expected or differently than described in the data sheet. Maxim Integrated Products, Inc., intends to correct these errata in subsequent die revisions.

*This errata sheet only applies to DS2780 revision A2 components. Revision A2 components are branded on the topside of the package with a six-digit code in the form yywwA2, where yy and ww are two-digit numbers representing the year and work week of manufacture, respectively. To obtain an errata sheet on another DS2780 die revision, visit our website at **www.maxim-ic.com/errata**.*

1) WRITING THE ACR REGISTER

Description:

When writing the ACR register, if the DS2780 Accumulation Bias Register contains a negative value, and the correct value will not be stored. The MSB of the ACR is decremented by 1 bit, and the LSB of the ACR remains the same. For example, if the user attempts to write 0x5000h to the ACR under these conditions, the ACR actually contains the value of 0x4F00h.

The full ACR register is 4 bytes long, which allows the DS2780 to accumulate very small measured currents. However, only the upper 2 bytes are writable by the user. When the ACR is written by software, the upper 2 bytes are loaded with the desired value and the lower 2 bytes are cleared. The Current Register is also cleared until the next current conversion is completed. Any value contained in the Accumulation Bias Register is added to the zero current value and, thus, it is accumulated into the lower bits of the ACR.

If the Accumulation Bias Register is positive the ACR is incremented, but none of the upper bytes change, (0x50000000h changes to 0x50000001h) so there are no adverse effects. However, if the Accumulation Bias Register is negative, the ACR needs to be decremented, which requires the upper bytes to change (0x50000000h changes to 0x4FFFFFFF). However, because of the race condition described below, the LSB of the Current Register is held at its present state of 0x00h, while the MSB decrements to 0x4Fh, so the full ACR actually reads 0x4F00FFFFh.

Immediately after any restart event (including a POR, wake from sleep, or Write ACR), the internal circuitry begins a current gain/tempco adjustment cycle. This initial cycle is not based on the result of an A/D conversion (decimation), and so it is not intended to subsequently start the current accumulation machine (ACR). The *DONECURR* signal is held low until the falling edge of the first *ldcurr* (Load Current Register) pulse. This should have been sufficient handshaking to guarantee that the *DONECURR* signal was held low throughout the entire *ldcurr* pulse.

The signal path for handshaking was not optimal, especially when combined with extreme loading of the *ldcurr* signal (3/0.5, 1/0.5 buffer driving fan out of 51/0.5, 51/0.5); there exists a race condition where some portion of the *DONECURR* pulse is allowed to pass. The *DONECURR* glitch starts the current accumulation machine (and the current average machine). If the restart event was an ACR Write, it is only a matter of timing that determines if one of the ACR bytes has a write collision (Interface vs. Accumulation) that may cause data corruption.

Workaround:

If the Accumulation Bias Register is not used (set to zero), the ACR does not become corrupt. However, if the user wishes to use a nonzero accumulation bias, instead of just writing the ACR, the user should use the following sequence:

```
// Write the Accumulation Bias Register to 0
RESET
0xCC
0x6C
0x61
0x00

// Write the ACR
RESET
0xCC
0x6C
0x10
Write desired MSB
Write desired LSB

//Recall the original Accumulation Bias Register value from EEPROM
RESET
0xCC
0xB8
0x60
RESET
```

An evaluation program targeted for this issue was created to verify that the ACR could be properly written and read back using the Software Patch Solution.

MAXIM