DALLAS SEMICONDUCTOR

ERRATA SHEET **DS89C420** **Revision A1** **10/20/00**

Revision A1 may be identified by the date/revision brand yywwA1, where yy and ww are the year and workweek of manufacture, respectively. This errata sheet is valid only when used in conjunction with the most current version of the data sheet available from Dallas Semiconductor via the Internet.

This document contains the following types of information:

Errata: These are design errors that deviate from published specifications. Errata are intended to be fixed in subsequent revisions of the device.

Specification Modifications: These are changes to the published specifications and will be reflected in the next update of the documentation and apply to all subsequent revisions of the device.

Documentation Changes: This information includes typographical mistakes, errors, omissions or clarifications of device operation. Items listed in this section will be reflected in the next update of the documentation.

ERRATA

1. The execution of instructions with a cycle count that exceeds the byte count can cause incorrect device operation in the following conditions:
   a. The device is executing from internal (flash) memory, and
   b. an instruction opcode falls on a 256-byte page boundary (xxFFh), and
   c. the previous instruction is one of the following:

ADDC A, @Ri	DEC @Ri
MUL AB	DIV AB
DA A	ORL A, @Ri
ANL A, @Ri	XRL A, @Ri
MOVC (all forms)	XCH A, Rn
XCH A, direct	XCHD A, @Ri
INC direct*	DEC direct*
ANL direct*, A	ORL direct*, A
XRL direct*, A	MOV direct*, A
MOV direct*, Rn	MOV direct1*, direct2
MOV direct*, @Ri	POP direct*
CLR bit	SETB bit
CPL bit	

   *These instructions will cause a failure only when when the SFR is one of these:
   PSW, SP, DPS, IE, EIE, IP0, IP1, EIP0, or EIP1.

Work Around: If programming in assembly language, write the software so that the code does not allow any of the above instructions to occur such that the next opcode falls on the last byte of 256-byte page (xxFFh) boundary. If writing in C, make all routines and functions less than 256 bytes in length and use byte constants placed at xxFFh intervals in the software. This condition will be corrected on revision A2 coming shortly.

2. The auto-baud routine of the bootstrap loader does not always function reliably (observed as garbled information instead of loader banner). Functionality is better at lower baud rates. Experiments show that the maximum reliable baud rate for any given crystal frequency is given by the equation: $f_{OSC}/1152$ = Max Baud

   Work Around: Use lower baud rates. This condition will be corrected on revision A2 coming shortly.

3. The signal /PSEN is driven by a strong pull-up internally, and the component used to pull it down to enter loader mode must be capable of sinking 100mA of current to ground.

   Work Around: None. This condition will be corrected on revision A2 coming shortly.

4. To enter loader mode, /PSEN is sampled after Reset is pulled high and /EA is pulled low. This dictates that the transition of /PSEN from high to low should be delayed from the onset of Reset and /EA by eight clock cycles. This timing relationship will be eliminated in future revisions (i.e., signals can be set simultaneously).

   Work Around: Use individual manual switches, passive circuitry (RC combinations), or active counting circuits (driven off of one of the processor's available clocks: ALE preferred) to delay the falling edge of /PSEN so it will be sampled correctly (i.e., 8 oscillator clocks or more after Reset and /EA are set). This condition will be corrected on revision A2 coming shortly.

5. If the oscillator is stopped or fails for any reason, the oscillator fail detect circuitry will cause a long delay in processor restart even though the oscillator fail detect reset function is disabled.

   Work Around: None. This condition will be corrected on revision A2 coming shortly.

6. Presently available sample devices (labeled ES) were tested successfully at 5.0 volts and at room temperature at a clock frequency of 45.0 MHz.

   Work Around: None. Operate these devices as close to these conditions as is reasonably possible.

7. The In-Application Programming feature is not functional. A design error causes bank 0 to read zeros while bank 1 is being erased or programmed.

   Work Around: None. This condition will be corrected on revision A2 coming shortly.

8. The In-System Programming feature appears to have some voltage, temperature and/or frequency dependencies. No specifics are available at this time.

   Work Around: Repeating the load or erase functions after a complete power down of the part seems to produce better results.

SPECIFICATION MODIFICATIONS

1. NONE

DOCUMENTATION CHANGES

1. NONE