8x930H*x* (8x930HD, 8x930HE) SPECIFICATION UPDATE

Release Date: March, 1997
Order Number: 272962-006

The 8x930H*x* may contain design defects or errors known as errata. Characterized errata that may cause the 8x930H*x*'s behavior to deviate from published specifications are documented in this specification update.

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature, may be obtained from:

Intel Corporation
P.O. Box 7641
Mt. Prospect IL 60056-7641

or call in North America 1-800-879-4683, Europe 44-0-1793-431-155,
France 44-0-1793-421-777, Germany 44-0-1793-421-333, other countries 708-296-9333

CONTENTS

REVISION HISTORY 1

PREFACE 2

SUMMARY TABLE OF CHANGES 4

IDENTIFICATION INFORMATION 7

ERRATA 8

SPECIFICATION CHANGES 16

SPECIFICATION CLARIFICATIONS 18

DOCUMENTATION CHANGES 22

REVISION HISTORY

Rev. Date	Version	Description
10/09/96	001	This is the new Specification Update document.
11/13/96	002	Added errata numbers 9611001, 9611002, and 9611003. Added A-1 stepping information.
12/10/96	003	Deleted Erratum number 9610001. This erratum does not exist in the 8*x*930H*x* A1 stepping. Added workaround to erratum number 9611002. Added erratum 9612001. Added specification clarification 002. Added documentation changes 001, 002, 003, 004, 005, and 006.
1/8/97	004	Added erratum 9701001. Deleted specification clarification 002. It does not apply to the 8*x*930H*x*. Added documentation changes 007 and 008. Added information for stepping A3.
2/5/97	005	Added documentation changes 009, 010, 011, and 012.
3/4/97	006	Added errata numbers 9702001 and 9702002. Added specification changes for AC and DC characteristics. Added 64-pin SDIP package marking information.

PREFACE

As of July, 1996, Intel has consolidated available historical device and documentation errata into this new document type called the Specification Update. We have endeavored to include all documented errata in the consolidation process, however, we make no representations or warranties concerning the completeness of the Specification Update.

This document is an update to the specifications contained in the Affected Documents/Related Documents table below. This document is a compilation of device and documentation errata, specification clarifications and changes. It is intended for hardware system manufacturers and software developers of applications, operating systems, or tools.

Information types defined in Nomenclature are consolidated into the specification update and are no longer published in other documents.

This document may also contain information that was not previously published.

Affected Documents/Related Documents

Title	Order
8X930Hx Universal Serial Bus Microcontroller data sheet	272928-001
8X930Hx Universal Serial Bus Microcontroller data sheet	272928-002
8X930Ax, 8X930Hx Universal Serial Bus Microcontroller User's Manual	272949-001

Nomenclature

Errata are design defects or errors. These may cause the published (component, board, system) behavior to deviate from published specifications. Hardware and software designed to be used with any component, board, and system must consider all errata documented.

Specification Changes are modifications to the current published specifications. These changes will be incorporated in any new release of the specification.

Specification Clarifications describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in any new release of the specification.

Documentation Changes include typos, errors, or omissions from the current published specifications. These will be incorporated in any new release of the specification.

NOTE:

Errata remain in the specification update throughout the product's lifecycle, or until a particular stepping is no longer commercially available. Under these circumstances, errata removed from the specification update are archived and available upon request. Specification changes, specification clarifications and documentation changes are removed from the specification update when the appropriate changes are made to the appropriate product specification or user documentation (datasheets, manuals, etc.).

SUMMARY TABLE OF CHANGES

The following table indicates the errata, specification changes, specification clarifications, or documentation changes which apply to the 8*x*930H*x* product. Intel may fix some of the errata in a future stepping of the component, and account for the other outstanding issues through documentation or specification changes as noted. This table uses the following notations:

Codes Used in Summary Table

Steps

X:	Errata exists in the stepping indicated. Specification Change or Clarification that applies to this stepping.
(No mark) or (Blank box):	This erratum is fixed in listed stepping or specification change does not apply to listed stepping.

Page

(Page):	Page location of item in this document.

Status

Doc:	Document change or update will be implemented.
Fix:	This erratum is intended to be fixed in a future step of the component.
Fixed:	This erratum has been previously fixed.
NoFix:	There are no plans to fix this erratum.
Eval:	Plans to fix this erratum are under evaluation.

Row

▌	Change bar to left of table row indicates this erratum is either new or modified from the previous version of the document.

Errata

Number	Steppings			Page	Status	ERRATA
	A1	A3				
9611001	X	X		8	NoFix	Non-Stop NoAcknowledging (NAK) with OHCI Systems
9611002	X			8	Fixed	Hub May Accidentally Be Recognized as a Low-speed Device
9611003	X	X		10	NoFix	V_{OH} on Port 1, 2, and 3 are Below Target Specification
9612001	X			11	Fixed	Receive FIFO RXFFRC Error
9701001	X			13	Fixed	Downstream J-K Signal Duty Cycle Not Symmetrical
9702001	X	X		14	No Fix	Timeout on Next IN Transaction
9702002	X	X		15	Fix	Transmit FIFO Underrun

Specification Changes

Number	Steppings			Page	Status	SPECIFICATION CHANGES
	A1	A3				
001	X	X		16	Doc	Upgrade datasheet status from “Product Preview” to “Advanced Information”.

Specification Clarifications

Number	Steppings			Page	Status	SPECIFICATION CLARIFICATIONS
	A1	A3				
001	X	X		18	Doc	T_{RHDZ1} Timing

Documentation Changes

Number	Document Revision	Page	Status	DOCUMENTATION CHANGES
001	001	22	Doc	Nonvolatile Memory Verification Port Labeled Incorrectly
002	001	22	Doc	Incorrect Address Given for TXSTAT SFR
003	001	22	Doc	Incorrect Signature Byte
004	001	22	Doc	Power-on Reset Capacitor Value Changed
005	001	22	Doc	SCON SFR’s REN Bit Description Incorrect
006	001	22	Doc	Extraneous Footnote in RXCON SFR
007	001	23	Doc	Power Off Flag Voltage Values
008	001	23	Doc	W_{CLK} Description Incorrect
009	001	23	Doc	Configuration Byte Misspelled
010	001	23	Doc	RTWCE Description Inaccurate
011	001	23	Doc	RL Instruction Misspelled
012	001	23	Doc	Footnote Incorrect in Data Instructions Table

IDENTIFICATION INFORMATION

Markings

Product	Part Number	Stepping	Marking	Package	Comment
8x930Hx Step A1	N80930HD1	A1	no marking	PLCC	Shipping media = tubes
			Q 831	PLCC	Customer sample No ROM
			SL24D	PLCC	General production, ROMless Shipping media = tape & reel
	N83930HD1		R xxxx	PLCC	8K ROM Customer ROM Code
	N83930HE1		R xxxx	PLCC	16K ROM Customer ROM Code
8x930Hx Step A3	N80930HD3	A3	Q 807	PLCC	General customer sample, ROMless
	U80930HD3		Q 880	SDIP	General customer sample, ROMless
	N80930HD3		SL26K	PLCC	General production, ROMless, Shipping media = tape & reel
	N80930HD3		no marking	PLCC	General production, ROMless, Shipping media = tubes
	U80930HD3		no marking	SDIP	General production, ROMless, Shipping media = tubes
	N83930HD3		R xxx	PLCC	General production, 8K ROM
	U83930HD3		R xxx	SDIP	General production, 8K ROM
	N83930HE3		R xxx	PLCC	General production, 16K ROM
	U83930HE3		R xxx	SDIP	General production, 16K ROM

ERRATA

9611001. Non-Stop NoAcknowledging (NAK) with OHCI Systems

PROBLEM: While in low-clock mode (LC Bit = 1), clearing the RXSETUP bit while the Transmit FIFO data register is “transmit ready” (transmit ready = data in the Transmit FIFO data register and a byte-count in the TXCNT register) could cause the part to continuously NAK in response to an IN token. This will occur until the Transmit FIFO data register is reset and reloaded. This only happens with OHCI where the bus turnaround is very fast.

IMPLICATION: If this condition occurs, the Serial Bus Interface Engine will continue to NAK even though there is data in the Transmit FIFO data register.

WORKAROUND: Software workaround is available. The user needs to make sure that the RXSETUP bit is cleared before the Transmit FIFO register becomes “transmit ready”.

STATUS: No Fix. Refer to Summary Table of Changes for affected steppings.

9611002. Hub May Accidentally Be Recognized as a Low-speed Device

PROBLEM: When a self-powered, suspended hub is reattached to the host, it could be accidentally recognized as a low-speed device. This is caused by the 8*x*930H*x* A-1 driving K signaling upstream and the host incorrectly sampling the speed of the device immediately after connect instead of after USB reset.

IMPLICATION: Enumeration will fail due to the host sending low-speed packets to a high-speed device.

WORKAROUND: To fix this problem, you must modify the reset circuitry on current 8*x*930H*x* designs. Figure 1 depicts two circuits, one for a bus-powered hub and another for a self-powered hub. For bus powered devices, the circuit is the same one that is given in Figure 14-1 on page 14-1 of the *8X930Ax, 8X930Hx Universal Serial Bus Microcontroller User’s Manual*, except the capacitor value has been changed from 1.0 µF to 0.3 µF.

For self-powered devices there are two requirements:

- The reset circuit design needs to be activated when the board is first powered-on.
- If the board is already powered-on, then a connection to the host should initiate reset. By connecting another capacitor to V_{BUS} (C2, as shown in Figure 1), a reset pulse is sent to the chip by either a power-up or by a USB connect.

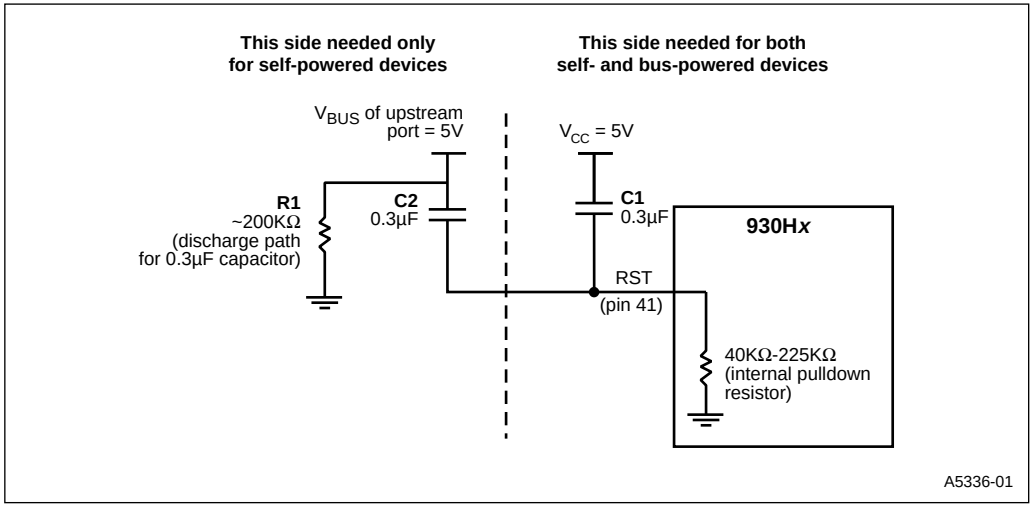

Figure 1. RST Source From V_{CC} and V_{BUS}

The circuit shown in Figure 1 uses the following components:

- C1 — Used in conjunction with the internal pulldown resistor to generate a 20 ms pulse of reset. For bus-powered devices, $V_{CC} = V_{BUS}$. This creates a 20 ms pulse upon connection to the host. For self-powered devices, a 20-ms pulse will occur regardless of the state of V_{BUS} if the capacitor ratios remain 1:1. The circuit operation and timing relationships rely on C1 = C2.
- C2 — Used only for self-powered devices. The capacitor delivers a 20-ms pulse to the RST pin when the V_{BUS} signal is detected upon device connection.
- R1 — Provides a discharge path for C2. This resistor is mandatory. If R1 is not present, C2 will not discharge the reset pulse and therefore will not allow a proper chip reset during device connection.

STATUS: Fixed. Refer to Summary Table of Changes for affected steppings.

9611003. V_{OH} on Port 1, 2, and 3 are Below Target Specification

PROBLEM: When Port 1, 2, and 3 are in quasi bidirectional mode, their V_{OH}s are below the target specification as shown:

$V_{OH} = V_{CC}$ - 0.8 V (instead of V_{CC} - 0.3 V) @10 µA
$V_{OH} = V_{CC}$ - 1.7 V (instead of V_{CC} - 0.7 V) @30 µA
$V_{OH} = V_{CC}$ - 2.7 V (instead of V_{CC} - 1.5 V) @60 µA

IMPLICATION: The fanout of port 1, 2, and 3 pins are reduced.

WORKAROUND: External buffers can be used to provide the required drive capability needed for interfaces that require more drive than the 8*x*930H*x* can support.

STATUS: No Fix. Refer to Summary Table of Changes for affected steppings.

9612001. Receive FIFO RXFFRC Error

PROBLEM: After the RXFFRC bit of the RXCON register is set, the RXFIF1:0 bits in the RXFLG register remain “11”. According to specifications, RXFIF1:0 should immediately decrement when RXFFRC is set. This problem occurs when the following conditions are simultaneously met:

1. The function interface unit (FIU) and serial bus interface engine (SIE) write a byte count to the RXCNTL SFR of endpoint 1’s receive FIFO.
2. The CPU sets the RXFFRC bit for any other endpoint’s receive FIFO (not endpoint 1).
3. The receive FIFO for the endpoint in (2) has RXFIF1:0 bits = “11”.

This problem is most likely to occur in low-clock mode when the device has a high data receive rate (bulk mode) on endpoint 1 and one or more of the other receive FIFO endpoints.

IMPLICATION: When the problem occurs, having RXFIF1:0 remain as “11” will cause firmware to incorrectly assume that there are two packets left in the receive FIFO (in dual packet mode), when in reality there is only one packet left. If firmware attempts to read the non-existent second packet, hardware will set the RXURF bit. When the RXURF bit gets set, the 8*x*930H*x* continues to NAK all OUT packets on the affected FIFO. At this point, the FIFO will be in an unknown state, requiring firmware to reset/clear that FIFO.

WORKAROUND: Additional code must be added to the firmware location(s) where a non-endpoint 1 receive FIFO is released. This code must determine if the RXFIF1:0 bits are “11” before and after setting the RXFFRC bit. If this is true, and if the RXSEQ bit has not been toggled by hardware during this time, then the error has occurred. At this point, firmware must attempt to release the receive FIFO again by re-setting the RXFFRC bit. This process must be repeated until the receive FIFO is successfully released.

Insert a firmware routine similar to the example shown in Figure 2 in your code at the point where you release the receive FIFO for all endpoints except endpoint 1. The code must be duplicated for each endpoint (except endpoint 1), replacing the “x” in the example’s code labels with the endpoint number.

STATUS: Fixed. Refer to Summary Table of Changes for affected steppings.

```
;*****************************************************
; RXFFRC Firmware Workaround
;*****************************************************
; Note: Registers 11 through 14 are utilized in this example.
; Please be sure to save and restore these registers as needed.

RELEASE_FIFO_x:
          mov     A,      RXFLG
          mov     R12,    RXFLG           ; before
          mov     R13,    RXSTAT

          setb    RXFFRC

          ; if (RXFLG_BEFORE = RXFLG_AFTER)
          ;  then { continue and check if RXFIF="11" }
          ;  else { setting RXFFRC was successful }
          cjne    A,    RXFLG, REL_FIFOx_OK

          ; if we get here R11 has RXFLG before and after - no
          ; change
          mov     R14,    RXSTAT

          ; if (RXFLG_AFTER = "11")
          ;  then { continue and check RXSEQ data toggle }
          ;  else { jump to REL_FIFOx_OK}
          anl     R11, #11000000b         ; check RXFIF after
          cjne    A,   #11000000b,  REL_FIFOx_OK

          ; RXFIF bits are "11"    RXFLG="C0"
          ; if (RXSEQ_BEFORE = RXSEQ_AFTER)
          ;  then
          ;   { no data toggle - set RXFFRC again }
          ;  else
          ;   { data toggle - OK jump to REL_FIFOx_OK}
          anl     R13, #10000000b
          anl     R14, #10000000b
          cmp     R13,   R14
          jne     REL_FIFOx_OK

          ; FIFO errata condition: RXFIF was "11" before & after; &
          ; RXSTAT didn't change
RELEASE_FIFO_x_AGAIN:
          ljmp    RELEASE_FIFO_x

REL_FIFOx_OK:
```

Figure 2. RXFFRC Firmware Workaround

9701001 Downstream J-K Signal Duty Cycle Not Symmetrical

PROBLEM: Customers cannot cascade 8*x*930H*x* hubs up to the maximum allowable five tiers as specified by the *Universal Serial Bus Specification*, version 1.0.

This problem occurs because the J-K duty cycle of the 8*x*930H*x* USB differential signal is not symmetrical. Therefore, the signaling marginality worsens when more 8*x*930H*x* hubs are cascaded. Timeouts may occur on the host and/or on the device.

If an end function device using an A3-stepping 8*x*930A*x* is connected to the 8*x*930H*x* hubs, it works well only up to four tiers of 8*x*930H*x* hubs. If an end function device using an A2-stepping 8*x*930A*x* is connected to the 8*x*930H*x* hubs, it works will only up to two tiers of 8*x*930H*x* hubs.

if an end function device using an A3-stepping 8*x*930A*x* (or any other USB device with a symmetrical duty cycle for the J-K signaling) is connected to the hubs, it works well up to four tiers of 8*x*930H*x* cascaded with another tier using a hub with a symmetrical duty cycle for the J-K signaling.

The maximum allowable cascading of hubs specified by the *Universal Serial Bus Specification*, Rev. 1.0, is:

Host/root hub-->1st tier hub-->2nd tier hub-->3rd tier hub-->4th tier hub-->5th tier hub-->end function device.

IMPLICATION: If 8*x*930H*x* hubs are cascaded more than the number of tiers described above, the USB device attached to the last tier hub will not function correctly.

WORKAROUND: No workaround.

STATUS: Fixed. Refer to Summary Table of Changes for affected steppings.

9702001. Timeout on Next IN Transaction

PROBLEM: When an OUT data packet with CRC16 error is sent to a Receive Endpoint that is not ready, the timeout will occur for next IN transaction on any Transmit Endpoints (see Case 1 below). This will happen when all of the following conditions are met:

A. A Receive Endpoint is not ready
B. An OUT data packet is sent to that endpoint
C. The OUT data packet has CRC16 error

The Receive Endpoint not ready could be due to any one of the following conditions:

1. The RXFIF.1:0 bits = 11
2. The Receive FIFO underrun or overflow or write pointer is not equal to the write marker.
3. The Receive Input Enable bit (RXIE) is not set.

When timeout on the next IN transaction occurs, the host will send another IN token to that endpoint. If this IN token is not preceded by another OUT transaction from the host that meets conditions A-C above, this IN transaction will not be timed out and the device will respond correctly to the host (see Case 2 below). All subsequent transactions on any endpoint will also be fine.

When the OUT data packet with CRC16 error is sent to a Receive Endpoint that is not ready, a next OUT transaction on any endpoints which does not meet conditions A-C above will still be fine. So will all other subsequent transactions on any endpoint.

Case 1: Problematic* OUT transaction --> IN transaction (timeout)

Case 2: Problematic* OUT transaction --> IN transaction (timeout) --> IN transaction (fine)

Case 3: Problematic* OUT transaction --> OUT transaction (fine)

** A problematic OUT transaction that meets all of conditions A - C above.*

IMPLICATION: No impact to the applications.

WORKAROUND: No workaround is needed.

STATUS: No Fix. Refer to Summary Table of Change for affected steppings.

9702002. Transmit FIFO Underrun

PROBLEM: When setting the TXCLR coincides with an IN token received, the Transmit FIFO (TXFIFO) will be underrun. This will happen when all the following conditions are met:

A. The RXSETUP bit is set
B. An IN token is received
C. Setting the TXCLR bit

IMPLICATION: When the problem occurs, the TxFIFO will be in underrun error condition (TXURF bit set). If the TXURF bit is not checked by the firmware and is not cleared properly, the 8*x*930H*x* will continue to NAK all the IN tokens on the affected TXFIFO.

WORKAROUND: Additional code must be added to the firmware wherever it is applicable. This code must determine if the RXSETUP bit is set, then set the TXCLR bit twice or more if necessary to make sure the TXFIFO is cleared. When the TXFIFO is successfully cleared, all the flags (especially TXFIFO underrun flag, TXURF) in the TXFLG register are cleared except for empty flag, TXEMP is set.

Insert a firmware routine similar to the example shown in Figure 3 in your code wherever it is necessary. The code must be duplicated for each transmit endpoint, replacing the “x” in the example code labels and the EPINDEX register with the proper endpoint number for hub or for embedded function.

STATUS: Fix. Refer to Summary Table of Change for affected steppings.

```
Clear_TXFIFO_xx:

     mov EPINDEX, #x0h        ; hub or embedded function endpoint 0
     mov A, RXSTAT
     jb ACC.6 Clear_Again_xx  ; jump if RXSETUP bit is set
     mov EPINDEX, #xxh        ; related hub or embedded function endpoint number
     orl TXCON, #10000000b    ; set TXCLR bit
     sjmp End_Clear_TXFIFO_xx:

   Clear_Again_xx:
     mov EPINDEX, #xxh        ;  related hub or embedded function endpoint number
     orl TXCON, #10000000b    ; set TXCLR bit
     mov A, TXFLG
     jb ACC.1, Clear_Again_xx ; jump if TXURF is set
     sjmp End_Clear_TXFIFO_xx

   End_Clear_TXFIFO_xx:
```

Figure 3. Transmit FIFO Underrun Firmware Workaround

SPECIFICATION CHANGES

The 8*x*930H*x* Product Preview datasheet (order number 272928-001) has been upgraded to Advanced Information datasheet (order number 272928-002). Please refer to the Summary Table of Changes for affected steppings.

The changed specifications are listed in the tables below:

Table 1. DC Characteristics

Symbol	Parameter	Min	Typical (1)	Max	Units	Test Conditions
I_{PD}	Powerdown Current Normal powerdown USB suspend		 25 145	 75 (was 50) 175	µA	
I_{DL}	Idle Mode I_{CC}			60 (was 40)	mA	Full speed (in low clock mode) PLLSEL2:0 = 110 F_{CLK} = 3 MHz
				110 (was 100)		Full speed (not in low clock mode) PLLSEL2:0 = 110 F_{CLK} = 12 MHz
I_{CC}	Active I_{CC}			75 (was 70)	mA	Full speed (in low clock mode) PLLSEL2:0 = 110 F_{CLK} = 3 MHz
				170		Full speed (not in low clock mode) PLLSEL2:0 = 110 F_{CLK} = 12 MHz

NOTE:
1. Typical values are obtained using V_{CC} = 5.0V, T_A = 25°C and are not guaranteed.

Table 2. AC Characteristics

Symbol	Parameter	CPU Frequency @ 12 MHz (M, N = 0)	CPU Frequency (F_{CLK}) Variable		Units
			Min	Max	
T_{AVLL}	Address Valid to ALE Low	21.66 (was 28.66)	(0.5+M)T_{CLK} – 20 (was -13)		ns
T_{LLAX}	Address Hold after ALE Low	4 (was 10)	4 (was 10)		ns
T_{WLWH}	WR# Pulse Width	71.33 (was 73.33)	(1+N)T_{CLK} – 12 (was -10)		ns
T_{LLRL}	ALE Low to RD# or PSEN# Low	4 (was 5)	4 (was 5)		ns
T_{LHAX}	ALE High to Address Hold	40.33 (was 56.33)	(1+M)T_{CLK} – 43 (was -27)		ns
T_{RHDZ2}	Data Float After RD# or PSEN# High	83.33 (was 93.33)		T_{CLK} (was + 10)	ns
T_{RHLH2}	RD# or PSEN# High to ALE High (data)	83.33 (was 93.33)	T_{CLK} (was + 10)		ns
T_{WHLH}	WR# High to ALE High	88.33 (was 93.33)	T_{CLK} + 5 (was +10)		ns
T_{AVDV1}	Address (Port 0) Valid to Valid Data/Instruction In	98.66 (was 106.66)		(2+M+N)T_{CLK} – 68 (was -60)	ns
T_{AVDV3}	Address (Port 2) Valid to Valid Instruction In	23.33 (was 35.33)		(1+N)T_{CLK} – 60 (was -48)	ns
T_{AVRL}	Address Valid to RD# or PSEN# Low	37.33 (was 56.33)	(1+M)T_{CLK} – 46 (was -27)		ns
T_{AVWL1}	Address (Port 0) Valid to WR# Low	37.33 (was 56.33)	(1+M)T_{CLK} – 46 (was -27)		ns
T_{AVWL2}	Address (Port 2) Valid to WR# Low	66.33 (was 83.33)	(1+M)T_{CLK} –17 (was -0)		ns

SPECIFICATION CLARIFICATIONS

001. T_{RHDZ1} Timing

PROBLEM: The TRHDZ1 (Instruction Float After RD#/PSEN# High) specification on the 8*x*930H*x* when operating at 12 MHz is 10 ns. However, a slow memory device such as an EPROM typically takes approximately 30 – 90 ns to float its output. (This specification, generically called TPHZ, may vary depending on the memory type and manufacturer.) Figures 4 and 5 illustrate the TRHDZ1 timing.

The difference between the TRHDZ1 and TPHZ specifications causes contention on the data bus (P0 in nonpage page, P2 in page mode). The 8X930H*x* begins to drive the address for the next bus cycle while the memory device is still driving data from the previous bus cycle.

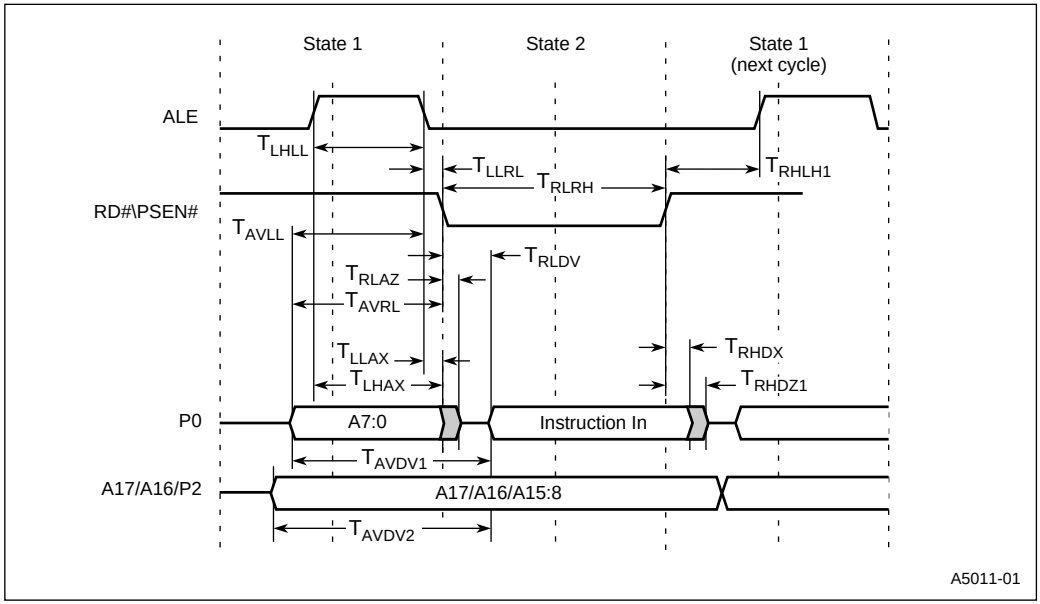

Figure 4. External Code Fetch, Nonpage Mode

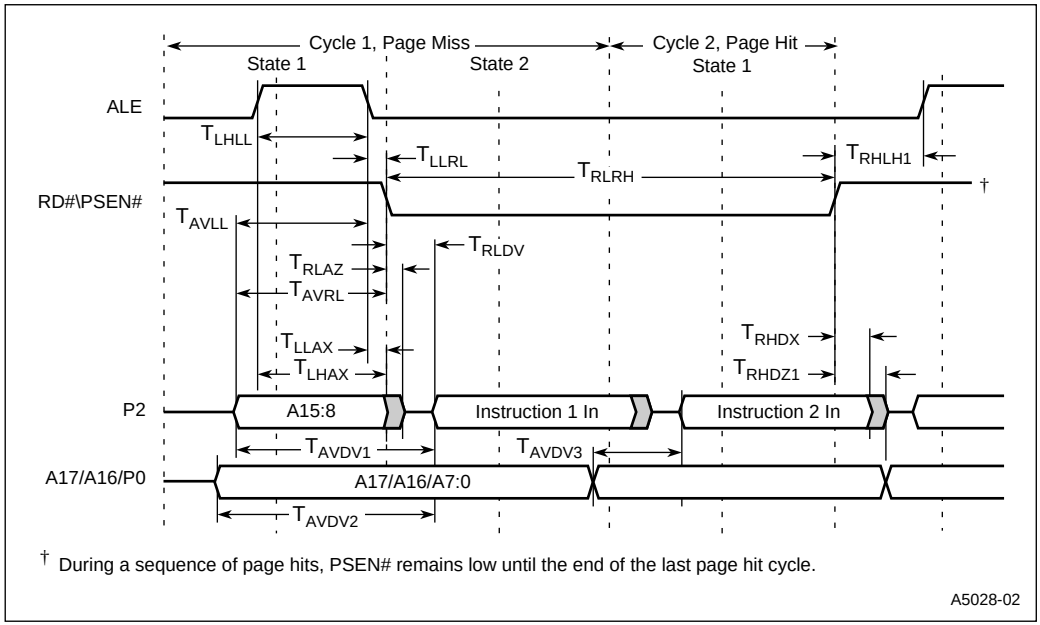

Figure 5. External Code Fetch, Page Mode

To prevent this contention, designers can use a buffer to isolate the output of the memory device from the data bus (port 0 or port 2) of the 8*x*930H*x*. This will prevent the memory device from driving the data bus during the critical period after T_{RHDZ1} expires. We suggest a buffer such as the 74F541 octal, three-state line driver shown in Figure 6.

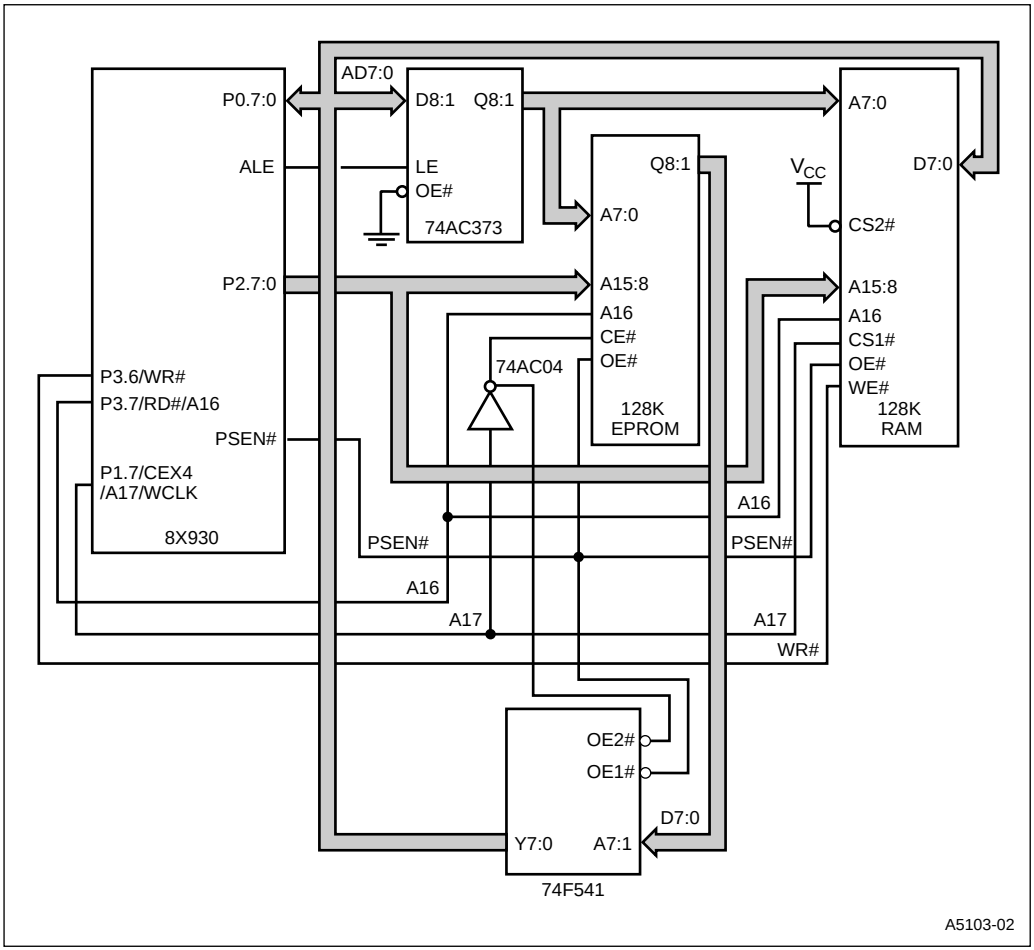

Figure 6. Example Bus Contention Solution for 8X930Hx (Nonpage Mode)

Figure 4 illustrates the connections of an 8*x*930H*x* configured for nonpage mode with EPROM and RAM in external memory. If your system uses a different configuration, your circuit will be different from the example. The 74F541 is enabled to pass data from the EPROM to the 8*x*930H*x* (port 0) when OE1# and OE2# are active. Table 3 is a truth table for the 74F541.

Table 3. Truth Table for 74F541

Inputs			Outputs
OE1#	**OE2#**	**A7:1**	**Y7:1**
L	L	L	L
L	L	H	H
X	H	X	Z
H	X	X	Z

During a read, PSEN# turns the buffer on (OE1# and OE2# are active), connecting the EPROM's output to the 8*x*930H*x*'s port 0.

When PSEN# goes high after a read, OE1# and OE2# are deasserted, and the buffer's output switches to the high-impedance state in approximately 9.5 ns (T_{PHZ} for a typical 74F541; the value may vary from one manufacturer to another). Thus, contention on the data bus is prevented. This or a similar hardware solution is recommended for 8X930Hx designs in which memory devices do not meet the $T_{RHDZ}1$ timing specification.

DOCUMENTATION CHANGES

001. Nonvolatile Memory Verification Port Labeled Incorrectly

ITEM: The illustration “Setup for Verifying Nonvolatile Memory” (Figure 17-1 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User’s Manual*) incorrectly depicts P1 as the Verify Modes port. The correct port for Verify Modes is P0.

002. Incorrect Address Given for TXSTAT SFR

ITEM: The USB Function SFR tables (Tables 3-11 and C-7) in the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User’s Manual* give an incorrect address for the TXSTAT SFR. The correct address is S:F2H.

003. Incorrect Signature Byte

ITEM: Section 17-6 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User’s Manual* mentions a signature byte at 61H. There is no signature byte at 61H.

004. Power-on Reset Capacitor Value Changed

ITEM: Figure 14-1 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User’s Manual* depicts a 1 µF power-on reset capacitor. The correct value for this capacitor is 0.3 µF.

005. SCON SFR’s REN Bit Description Incorrect

ITEM: The description for the Serial Port Control SFR’s REN bit (SCON.4), as given in Figure 13-2 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User’s Manual*, is incorrect. The text should say ”To enable reception, set this bit. To disable reception, clear this bit.”

The SCON SFR also appears in Appendix C of the same manual.

006. Extraneous Footnote in RXCON SFR

ITEM: The dagger footnote (†) does not apply to the RXFFRC and RXISO bits in the RXCON SFR, as shown in Figure 7-15 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User’s Manual*. The RXCON SFR also appears in Appendix C. Note that the dagger footnote *does* apply to the SFR’s ADVWM and REVWP bits.

007. Power Off Flag Voltage Values

ITEM: Section 15.2.2 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User's Manual* states that "the hardware sets the Power Off Flag (POF) in PCON when V_{CC} rises from < 3 V to > 3 V to indicate that on-chip volatile memory is indeterminate...since for V_{CC} < 3 V data may have been lost or some logic may have malfunctioned." The voltage value should be 3.5 V for all references, not 3 V.

008. W_{CLK} Description Incorrect

ITEM: The description for the wait clock output (W_{CLK}) given in Table 16-1 and Table B-2 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User's Manual* is incorrect. Instead of "When enabled, the W_{CLK} output produces a square wave signal with a period of one-half the oscillator frequency," the final sentence should read "When enabled, the W_{CLK} output produces a square wave signal with a period of T_{CLK}."

009. Configuration Byte Misspelled

ITEM: On page 16-11 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User's Manual*, in the third paragraph of the note, UNCONFIG0 should be UCONFIG0.

010. RTWCE Description Inaccurate

ITEM: On page 16-11 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User's Manual*, in Figure 16-11, the description of RTWCE should say "with RTWE set, setting RTWCE will enable the WAIT clock....." In other words, setting RTWCE alone will not enable the wait clock, both bits must be set.

011. RL Instruction Misspelled

ITEM: On page A-4 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User's Manual*, the 'A' should be moved to the second line for RLA and RLCA. The instruction is RL A not RLA.

012. Footnote Incorrect in Data Instructions Table

ITEM: On page A-6 of the *8x930Ax, 8x930Hx Universal Serial Bus Microcontroller User's Manual*, Note 2 for Table A-9 is not correct. ORL, ANL, and XRL all have one instruction that uses DRk (see page A-38 for an example).