



---

# USB Compliance Checklist Peripheral Silicon (Excluding Hub Silicon)

For the 1.1 USB Specification  
Checklist Version 1.03  
June 18, 1999

---

## *USB Device Product Information*

Date	July 20, 1999
Vendor Name	Motorola
Vendor Street Address	
Vendor City, State, Postal Code	
Vendor Country	
Vendor Phone Number	
Vendor Contact, Title	
Product Name	
Product Model Number	MPC823
Product Revision Level	Z3
Manufacture & Model Identifier of peripheral used for testing	
Signature of Preparer	

## Table of Contents

<b>1</b>	<b>INTRODUCTION .....</b>	<b>2</b>
1.1	General Notes and Glossary .....	2
<b>2</b>	<b>SIGNALS AND TIMING .....</b>	<b>3</b>
2.1	Low-Speed Ports (applicable to any USB port which can operate at 1.5Mb/s).....	3
2.2	Full-Speed Ports (applicable to any USB port which can operate at 12Mb/s) .....	4
<b>3</b>	<b>SIGNALING PROTOCOL AND ERROR HANDLING .....</b>	<b>5</b>
3.1	Bitstreams .....	5
3.2	Fields.....	5
3.3	Packets.....	6
3.4	Transactions .....	6
3.5	Transfers .....	7
<b>4</b>	<b>RECOMMENDED QUESTIONS .....</b>	<b>7</b>
4.1	Device Robustness .....	7
4.1.1	Bitstreams .....	7
4.1.2	Fields .....	8
4.1.3	Packets.....	8
4.1.4	Transactions.....	8
4.1.5	Transfers.....	9
<b>5</b>	<b>EXPLANATIONS.....</b>	<b>10</b>

## Revision History

<u>version</u>	<u>changes</u>	<u>date</u>
1.03	revised ST13, added LS6, FS7, FS8	1999.6.18
1.02	revised introduction, fixed $\mu$ , $\Omega$ , typos	1999.2.5
1.01	added preparer's signature and changed checklist contact info	1999.1.4
1.00	initial release: added contact info and some minor clarifications	1998.11.20
.95	first public review draft, released for Taipei USB Plugfest	1998.10.26

# 1 Introduction

This checklist helps designers of USB microcontrollers, transceivers, cores, or other ASICs for USB peripherals to assess their products' compliance with the Universal Serial Bus Specification, Revision 1.1. Unless explicitly stated otherwise, all references to the USB Specification refer to Revision 1.1.

This checklist is also used, in part, to qualify USB interface silicon for the USB-IF Integrators List. This document and other USB compliance tools, including USB Check, are available in the developers section of the USB-IF's website, <http://www.usb.org/developers/>. The compliance checklists are updated periodically, so developers should check for updates when starting new projects.

Section 5, Recommended Questions, contains questions covering areas not required by the USB Specification. Answering these questions is not a requirement for compliance with the Specification or acceptance to the Integrators List. However, vendors are strongly encouraged to take these questions into consideration when designing their products.

Questions or comments regarding the Integrators List, Compliance Workshop testing results, or checklist submissions should be sent to [admin@usb.org](mailto:admin@usb.org). If you have questions regarding the checklist itself, feel it fails to adequately cover an aspect of the USB specification, have found an error, or would like to propose a question, please contact the USB-IF at [checklists@usb.org](mailto:checklists@usb.org).

## 1.1 General Notes and Glossary

- All voltages are referenced to the USB ground of the device being tested.
- Bit order is from left to right.

bitstream	A set of bits represented as NIBs or NZBs.
field	A particular group of bits represented as NIBs or NZBs. <i>E.g.</i> a SYNC field is KJKJKJKK in NIBs or 00000001 in NZBs.
NIB	NRZI bit. Represented as J, K and 0 (single ended 0).
NZB	NRZ bit. Represented as 1, 0 and S (single ended 0).
packet	The most frequently referenced data structure, built out of fields.
phase	of a transaction is made up of 0 or 1 packet e.g. token phase
sense	The relationship between D+ and D- voltages for J state on a USB connection. For full-speed, $D+ > D-$ , for low-speed link $D+ < D-$ .
stage	A set of one or more unidirectional transactions.
timeout period	The amount of time that an agent awaiting a response waits before invalidating the transaction.
target	Destination of a transaction., which could be a pipe in the host or an endpoint in a device.
transaction	The basic unit of unidirectional data transfer and a set of packets. For unidirectional endpoints (bulk, interrupt, isochronous) this is the final level of communication with protocol implications.
transfer	A unit of bi-directional data transfer built out of stages. It is used by only by control endpoints.
turnaround time	The time between an agent seeing the EOP for the previous packet and starting to drive the bus for a new packet. <i>I.e.</i> , the J period after the EOP as measured at the agent. This time also applies to the period between packets when the host is driving both packets.

## 2 Signals and Timing

ID	question	response	sections in spec
ST1	Is the data line crossover voltage between 1.3 and 2.0V?	yes no <b>See note 1</b>	7.1.2
ST2	Do all single ended receivers recognize 0.8V or below as a logic low?	yes no <b>See note 1</b>	7.1.4
ST3	Do all single-ended receivers recognize 2.0V or more as a logic high?	yes no <b>See note 1</b>	7.1.4
ST4	Do all differential receivers have an input sensitivity of at least 200mV between 0.8 and 2.5 volts common mode?	yes no <b>See note 1</b>	7.1.4
ST5	Is the device's pull up active only when $V_{BUS}$ is 1.17V or more?	yes no <b>See note 1</b>	7.1.5
ST6	Is the input impedance of D+ and D-, without termination and pull up resistors, more than 300k $\Omega$ ?	yes no <b>See note 1</b>	7.1.6
ST7	Does the device respond to a reset no sooner than 2.5 $\mu$ s and no later than 10ms after the SE0 begins, regardless of the SE0's position in a bitstream?	yes no <b>See note 2</b>	7.1.7.3
ST8	Is the device's reset recovery time less than 10ms?	yes no <b>See note 2</b>	7.1.7.3
ST9	At the end of reset is the device in the default state?	yes <u>x</u>	7.1.7.3 9.1.1
ST10	Does the device enter suspend if the bus is idle for 3ms or more?	yes no <b>See note 3</b>	7.1.7.4
ST11	Has the device's power consumption dropped to its suspended value after the hub's upstream bus segment has been idle for 10ms?	<u>x</u> no <b>See note 10</b>	7.1.7.4
ST12	When suspended, does the device recognize any non-idle state on its upstream port, including a reset, as a resume signal?	yes <u>x</u>	7.1.7.5
ST13	Does the device recognize the end of resume signaling and return to the state it was in prior to suspend?	yes no <b>See note 2</b>	7.1.7.5
ST14	Is the device able to accept a SetAddress() request 10ms after resume is signaled?	yes no <b>See note 2</b>	7.1.7.5
ST15	Does the device complete its wakeup within 20ms?	yes no <b>See note 2</b>	7.1.7.5
ST16	Do active data line outputs drive to 2.8–3.6V with a 14.25k $\Omega$ load to ground?	yes no <b>See note 1</b>	7.3.2
ST17	Do active data line outputs drive to 0–0.3V with a 1.425k $\Omega$ load to 3.6V?	yes no <b>See note 1</b>	7.3.2

Note: the 1.17V threshold voltage for pull up activation is derived from a device using a resistor connected directly to  $V_{BUS}$ , assuming worst-case values for  $V_{BUS}$  (5.25V), pull up size (6.53k $\Omega$ , which holds Dx at 3.6V with a 5.25V  $V_{BUS}$ ), pull down size (15.75k $\Omega$ ), and single-ended receiver sensitivity (0.8V).

### 2.1 Low-Speed Ports

(applicable to any USB port which can operate at 1.5Mb/s)

LS1	Are data line rise times between 75ns and 300ns when driving into any single ended, capacitive load between 200 and 450pF?	yes no <b>See note 1</b>	7.1.2
LS2	Are data line fall times between 75ns and 300ns when driving into any single ended, capacitive load between 200 and 450pF?	yes no <b>See note 1</b>	7.1.2
LS3	Are the rise and fall times matched to within 20% for J→K transitions?	yes no <b>See note 1</b>	7.1.2

LS4	Are the rise and fall times matched to within 20% for K→J transitions?	yes no <b>See note 1</b>	7.1.2
LS5	Is an SE0 less than 210ns long rejected as an EOP?	yes <u>x</u>	7.1.4
LS6	Does the device drive the J state at the end of an EOP for a full low-speed bit time?	yes <u>x</u>	7.1.7
LS7	Is the transmission data rate between 1.4775 and 1.5225Mb/s?	yes <u>x</u>	7.1.11
LS8	Is the differential driver jitter for consecutive transitions less than ±25ns?	yes <u>x</u>	7.1.13.1
LS9	Is the differential driver jitter for paired transitions less than ±10ns?	yes <u>x</u>	7.1.13.1
LS10	Is the EOP width between 1.25µs and 1.5µs at the transmitter?	yes <u>x</u>	7.1.13.2
LS11	Does the receiver accept an SE0 between 670ns and 1.76µs long, followed by a J, as an EOP?	yes <u>x</u>	7.1.13.2
LS12	Does the receiver accept a packet whose first bit has been distorted by as much as ±25ns?	yes <u>x</u>	7.1.14
LS13	Does the receiver accept a packet whose last bit has been lengthened by as much as 260ns (dribble bit)?	<u>x</u> no <b>See note 4</b>	7.1.14 7.1.9
LS14	Is the receiver data jitter tolerance at least ±141ns for consecutive transitions?	yes <u>x</u>	7.1.15
LS15	Is the receiver jitter tolerance for paired transitions at least ±184ns?	yes <u>x</u>	7.1.15
LS16	Is the device's turn-around time between two and 6.5 low-speed bit times, or 7.5 bit times if the device has a fixed cable?	yes <u>x</u>	7.1.18
LS17	Is the time-out period 16–18 low-speed bit times?	yes <u>x</u>	7.1.19
LS18	Is D- between 2.7 and 3.6V and D+ between 0.0 and 0.3V when the bus is idle?	yes no <b>See note 1</b>	7.2.3

Note: the low-speed receiver jitter tolerances listed here do not apply to hosts and hubs. Consult section 7.1.15 for host and hub jitter requirements.

## 2.2 Full-Speed Ports

(applicable to any USB port which can operate at 12Mb/s)

FS1	With series termination resistors, does the device's source impedance remain in the shaded areas of Figure 7-3?	yes no <b>See note 1</b>	7.1.1.1
FS2	Are data line rise times between 4.0 and 20ns when driving into a single-ended 50pF load?	yes no <b>See note 1</b>	7.1.2
FS3	Are data line fall times between 4.0 and 20ns when driving into a single-ended 50pF load?	yes no <b>See note 1</b>	7.1.2
FS4	Are the rise and fall times matched to within 10% for J→K transitions?	yes no <b>See note 1</b>	7.1.2
FS5	Are the rise and fall times matched to within 10% for K→J transitions?	yes no <b>See note 1</b>	7.1.2
FS6	Is an SE0 less than 14ns long rejected as an EOP?	yes <u>x</u>	7.1.4
FS7	Does the device drive the J state at the end of an EOP for a full low-speed bit time?	yes <u>x</u>	7.1.7
FS8	If the device tracks the K→low-speed EOP→J transition on its upstream port at the end of resume, does it correctly handle the low-speed EOP?	yes <u>x</u>	7.1.7.5
FS9	Is the transmission data rate between 11.97 and 12.03Mb/s?	yes <u>x</u>	7.1.11
FS10	Is the differential driver jitter for consecutive transitions less than ±2.0ns?	yes <u>x</u>	7.1.13.1
FS11	Is the differential driver jitter for paired transitions less than ±1.0ns?	yes <u>x</u>	7.1.13.1
FS12	Is the EOP width between 160ns and 175ns at the transmitter?	yes <u>x</u>	7.1.13.2
FS13	Does the device accept an SE0 between 82ns and 250ns long, followed by a J, as an EOP?	yes <u>x</u>	7.1.13.2 7.1.14

FS14	Does the receiver accept a packet whose first bit has been distorted by as much as $\pm 25\text{ns}$ ?	yes <input checked="" type="checkbox"/>	7.1.14
FS15	Does the receiver accept a packet whose last bit has been lengthened by as much as $75\text{ns}$ ?	<input checked="" type="checkbox"/> no <b>See note 4</b>	7.1.14 7.1.9
FS16	Is the receiver data jitter tolerance at least $\pm 20.0\text{ ns}$ for consecutive transitions?	yes <input checked="" type="checkbox"/>	7.1.15
FS17	Is the receiver jitter tolerance for paired transitions at least $\pm 12.0\text{ ns}$ ?	yes <input checked="" type="checkbox"/>	7.1.15
FS18	Is the device's turn-around time between two and 6.5 full-speed bit times, or 7.5 bit times if the device has a fixed cable?	yes <input checked="" type="checkbox"/>	7.1.18
FS19	Is the time-out period 16–18 full-speed bit times?	yes <input checked="" type="checkbox"/>	7.1.19
FS20	Is D+ between 2.7 and 3.6V and D- between 0.0 and 0.3V when the bus is idle?	yes no <b>See note 1</b>	7.2.3

## 3 Signaling Protocol and Error Handling

### 3.1 Bitstreams

B1	Is the possibility of both D+ and D- registering as NIB 1 during bus transitions accounted for?	yes no <b>See note 5</b>	7.1.2 7.1.13.1
B2	Is the USB signaling either full-speed or low-speed but not both?	yes <input checked="" type="checkbox"/>	7.1.5
B3	Does the sense of USB signaling correspond to the signaling speed?	yes <input checked="" type="checkbox"/>	7.1.7
B4	Is the bitstream on the bus NRZI encoded?	yes <input checked="" type="checkbox"/>	7.1.8
B5	Is bit stuffing performed on all data transmitted, including CRCs, prior to NRZI encoding?	yes <input checked="" type="checkbox"/>	7.1.9 8.3.5
B6	Is bit stuffing performed even if the stuffed bit follows the last bit of a packet?	yes <input checked="" type="checkbox"/>	7.1.9
B7	Is NRZI to NRZ decoding done before bit unstuffing?	yes <input checked="" type="checkbox"/>	7.1.9
B8	Is bit unstuffing performed on all received data, including CRCs?	yes <input checked="" type="checkbox"/>	7.1.9 8.3.5
B9	Is bit unstuffing done before the bitstream is parsed?	yes <input checked="" type="checkbox"/>	7.1.9
B10	Is all transmission on D+ and D- done LSb first?	yes <input checked="" type="checkbox"/>	8.1

### 3.2 Fields

A field is one of:

address	7 bit field
data	0 to 1023 byte field
data CRC	16 bit field
endpoint	4 bit field
EOP	3 bit field with NIB value 00J
frame number	11 bit field
PID	8 bit field, whose types are listed in section 8.3.1
SYNC	8 bit field with NZB value 00000001

token CRC	5 bit field
-----------	-------------

F1	Is the SYNC field, as measured on the bus wires, correct (NIB KJKJKJKK)?	yes	<u>x</u>	8.2
F2	Are all PIDs used among those listed in Table 8-1?	yes	<u>x</u>	8.3.1
F3	Are the PID check bits the ones complement of the packet type field?	yes	<u>x</u>	8.3.1
F4	Are the CRC generator's contents inverted and sent to the checker MSb first?	yes	<u>x</u>	8.3.5
F5	Is the token CRCs generated with the polynomial NZB 00101 on the ADDR and ENDP fields of IN, SETUP, and OUT tokens?	yes	<u>x</u>	8.3.5.1
F6	If all bits are received without error, does the CRC computation on a token or SOF leave a residual of NZB 01100 at the EOP?	yes	<u>x</u>	8.3.5.1
F7	Is the data CRC generated with the polynomial NZB 100000000000101 on the data field of a data packet?	yes	<u>x</u>	8.3.5.2
F8	If all bits are received without error, does the CRC computation on the data field leave a residual of NZB 100000000001101 at the EOP?	yes	<u>x</u>	8.3.5.2

### 3.3 Packets

A packet can be one of the following:

packet	fields comprising packet
data	SYNC PID data data CRC EOP
handshake	SYNC PID EOP
PRE	SYNC PID
SOF	SYNC PID frame number token CRC EOP
token	SYNC PID endpoint token CRC EOP

P1	Are all token packets 32 bits long and followed by an EOP?	yes	<u>x</u>	8.4.1
P2	Are all token packets of the form SYNC PID address endpoint token CRC EOP?	yes	<u>x</u>	8.4.1
P3	Are all data packets an integral number of bytes long (4 to 1027) excluding the EOP?	yes	<u>x</u>	8.4.3
P4	Is the data packet constituted as sync followed by PID followed by 0 to 1023 bytes of data followed by data CRC followed by EOP	yes	<u>x</u>	8.4.3
P5	Are all handshake packets 16 bits + EOP?	yes	<u>x</u>	8.4.4
P6	Are all handshake packets of the form SYNC PID EOP?	yes	<u>x</u>	8.4.4
P7	Is the data payload of a low-speed packet limited to a maximum of 8 bytes?	yes	no	8.6.5
		<b>See note 6</b>		
P8	Is the PRE packet 16 bits long?	yes	<u>x</u>	8.6.5
P9	Does the PRE packet consist of only a SYNC followed by a PID?	yes	<u>x</u>	8.6.5

### 3.4 Transactions

Transactions are sets of packets used for unidirectional data transfer. Transactions are discussed in detail in section 8.5 of the USB Specification.

TA1	Does an isochronous endpoint synthesize frame markers to replace SOFs	yes	no	5.10.6
-----	-----------------------------------------------------------------------	-----	----	--------

	which may be lost due to bus error?	<b>See note 6</b>	
TA2	Do handshakes conform to order of precedence described in section 8.4.5?	yes no <b>See note 6</b>	8.4.5
TA3	Does the generated packet comply with the flows show in Figure 8-9, 8-11, 8-13, or 8-14, as appropriate?	yes <u>x</u>	8.5 8.6.5
TA4	Do interrupt endpoints used in rate feedback mode toggle the sequence bit without regard to presence or type of handshake?	yes no <b>See note 6</b>	8.5.3
TA5	Is an unsuccessful (NAKed or timed-out in non-token phase) transaction retried?	yes no <b>See note 7</b>	8.6
TA6	Does the retried transaction use the same data PID as the original transaction?	yes <u>x</u>	8.6

### 3.5 Transfers

Transfers are data structures used by control endpoints. Each transfer is made up of setup and status stages, possibly with a data stage. Transfers can be one of:

*setup0*                      *out1 out0 out1 ... out0/1*                      in1  
*setup0*                      *in1 in0 in1 ... in0/1*                      out1  
*setup0*                      in1

Transactions in italics constitute the data stage. The suffix of 0 or 1 indicates the data PID used in the transaction.

TF1	Does the data stage always start with a data1 PID?	yes <u>x</u> <b>See note 11</b>	8.5.2
TF2	Are all the transactions of the data stage in the same direction?	yes <u>x</u> <b>See note 11</b>	8.5.2
TF3	Is there status stage's direction opposite that of the data stage?	yes <u>x</u> <b>See note 11</b>	8.5.2
TF4	Is the data packet used in the status stage zero bytes in length?	yes <u>x</u> <b>See note 11</b>	8.5.2

## 4 Recommended Questions

### 4.1 Device Robustness

#### 4.1.1 Bitstreams

RB1	Is a single ended NIB 1 more than one bit time long ignored?	yes no <b>See note 1</b>	
RB2	Does an agent ignore a truncated (up to 90%) first bit of the sync field without impacting the rest of the bitstream?	yes <u>x</u>	
RB3	Is the state of the differential receiver ignored during single ended signaling?	yes no <b>See note 1</b>	
RB4	Does the target reject bitstreams less than one bit time long without impacting future transactions?	yes <u>x</u>	



RB5	Does the target adjust to the difference in frequency and phase between incoming clock and its internal clock?	yes	<u>x</u>	
RB6	Is a packet with a bit-stuff error rejected by the target?	yes	<u>x</u>	
RB7	Is a bitstream, which is not part of a packet, with bit stuff error ignored by the target?	yes	<u>x</u>	
RB8	Does the target reject packets with bit stuff error at the last bit of the packet?	yes	<u>x</u>	

### 4.1.2 Fields

RF1	Is the sync field recognized as valid even if the first two bits of it are corrupted? (Only the last 3 bits actually need to be decoded.)	yes	<u>x</u>	
RF1	Is a packet with packet type not listed in Table 8-1 ignored by the target?	yes	<u>x</u>	
RF2	Is a packet with a corrupt PID (PID check error) ignored by the target?	yes	<u>x</u>	
RF3	Is a token with a bad CRC ignored by the target?	yes	<u>x</u>	
RF4	Is a CRC error on a data packet recognized by the target?	yes	<u>x</u>	

### 4.1.3 Packets

RP1	Is a token whose address field doesn't match any address in the device ignored by the device?	yes	<u>x</u>	
RP2	Is a token whose endpoint field doesn't match any endpoint in the address ignored by the device?	yes	<u>x</u>	
RP3	Is a token which doesn't match the direction of its target endpoint ignored by the device?	yes	<u>x</u>	
RP4	Is a SETUP token to a unidirectional endpoint ignored by the device?	yes	<u>x</u>	
RP5	Is every endpoint capable of handling zero length data packets in its assigned directions?	yes	<u>x</u>	
RP6	Does an ISO endpoint use a zero length data packet if fresh frame data is not available?	yes	no <b>See note 6</b>	
RP7	Is a packet whose length doesn't match the standard length for the packet type rejected by target?	yes	no <b>See note 8</b>	
RP8	Does the measurement of packet length take into account the possibility of jitter and hub repeater skews in the EOP?	yes	<u>x</u>	
RP9	Is a bitstream that does not constitute a valid packet rejected by the target?	yes	<u>x</u>	
RP10	Are low-speed packets received by full-speed upstream ports ignored?	yes	<u>x</u>	8.6.5

### 4.1.4 Transactions

RTA1	Does an isochronous pipe handle holes or bubbles in pipe which may arise during suspend/resume operation?	<u>x</u>	no <b>See note 4</b>	
RTA2	Is a packet which doesn't fit the current phase of a transaction rejected by the target?	yes	<u>x</u>	
RTA3	Does the receipt of a token always start a new transaction and end a pending transaction?	yes	<u>x</u>	
RTA4	Is a data packet with same PID as the previous data packet to an endpoint ignored, other than ACKing the data packet?	yes	no <b>See note 9</b>	
RTA5	Does a time-out or error in any phase cause the transaction to be terminated?	yes	<u>x</u>	

RTA6	Is a transaction always started with a token?	yes <input checked="" type="checkbox"/>	
RTA7	Is the data toggle implemented independently for each unidirectional endpoint?	yes    no <b>See note 6</b>	
RTA8	Does an isochronous data source ignore a handshake without impacting subsequent transactions?	yes <input checked="" type="checkbox"/>	
RTA9	Can consecutive packets in the same direction be handled , provided there are two or more bit times of interpacket gap between each packet?	yes <input checked="" type="checkbox"/>	

#### 4.1.5 Transfers

RTF1	Does the receipt of a nonzero length data packet in the status stage cause the transfer to be terminated with an error indication?	yes    no <b>See note 6</b>	
------	------------------------------------------------------------------------------------------------------------------------------------	--------------------------------	--

# 5 Explanations

This section should be used to explain any “no” answers or clarify answers on checklist items above. Please key entries to the appropriate checklist question.

Note 1 – Depends on Transceiver. \_\_\_\_\_

Note 2 – Depends on Software. \_\_\_\_\_

Note 3 – Hardware indicates idle state. Software decides what to do with it. \_\_\_\_\_

Note 4 – Regarded by hardware as an erratic packet. \_\_\_\_\_

Note 5 – Depends on Transceiver. Hardware gets rxd line which we use for data; and rp and rm which is checked for se0, only. \_\_\_\_\_

Note 6 – Handled by Software. \_\_\_\_\_

Note 7 – This mode programmed on BD; it’s not the default mode. \_\_\_\_\_

Note 8 – Hardware checks for the integer number of octets; remaining processing handled by Software. \_\_\_\_\_

Note 9 – Hardware receives the packet (put in BD) and ACK. Should be ignored by Software. \_\_\_\_\_

Note 10 – Not applicable for the MPC823. \_\_\_\_\_

Note 11 – Software must correctly program the buffer descriptors. \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_