**OKI** Semiconductor

# ML60852A

**USB Device Controller**

## GENERAL DESCRIPTION

The ML60852A is a general purpose Universal Serial Bus (USB) device controller.

The ML60852A provides a USB serial interface engine, USB transceiver, FIFOs, control/status registers, application interface circuit, and oscillation circuit thereby easily realizing a USB system.  The ML60852A supports four types of data transfer such as control transfer, bulk transfer, interrupt transfer and isochronous transfer, and also supports five or six endpoints.

## FEATURES

- USB1.1 compliant
- Supports full-speed (12 Mbps).
- Supports four types of transfer; control transfer, bulk transfer, interrupt transfer, and isochronous transfer.
- Endpoints: 5 to 6 endpoints
  Control EP                  1
  Bulk/interrupt EP           3
  Isochronous/bulk/interrupt EP     1 or 2
- Built-in FIFO for data storage
- A two-layer configuration of FIFO for each of EP1, EP2, EP4, and EP5
- 8/16 bit DMA supported (EP1, EP2, EP4, and EP5)
- Intelligent Serial Interface Engine (SIE)
- Supports bus-powered device.
  The suspend condition is automatically detected and the low-power mode is activated. Normal operation is automatically restarted when the resume condition is detected.
- Built-in USB transceiver circuit
- Ta = -20 to +80 °C
- $V_{CC}$=3.0 to 3.6 V
- Interface with 5 V circuit is possible. (Input: 5 V tolerant, output: TTL)
- Built-in 12 MHz /6 MHz oscillation circuit
- Package options:
  44-pin plastic TQFP
  56-pin plastic LGA

**Endpoints and FIFOs**

By initializing appropriate registers, the ML60852A can be operated in either 5EP or 6EP mode. Although the transfer mode that can be used by EP0 is fixed, it is possible to select either the bulk transfer mode or the interrupt transfer mode for end points EP1, EP2, and EP3, and one of the modes of isochronous, bulk, or interrupt transfer can be selected for EP4 and EP5. In addition, it is possible to selectively set the direction of data transfer for EP1 to EP5.

| End point | 5EP Mode | | | 6EP Mode | | |
|---|---|---|---|---|---|---|
| | FIFO Capacity | Transfer mode | Remarks | FIFO Capacity | Transfer mode | Remarks |
| EP0 | Reception 32 Transmission 32 | Control transfer | | Reception 32 Transmission 32 | Control transfer | |
| EP1 | 64x2 | Bulk/interrupt transfer (IN/OUT) | DMA Possible | 64x2 | Bulk/interrupt transfer (IN/OUT) | DMA Possible |
| EP2 | 64x2 | Bulk/interrupt transfer (IN/OUT) | DMA Possible | 64x2 | Bulk/interrupt transfer (IN/OUT) | DMA Possible |
| EP3 | 32 | Bulk/interrupt transfer (IN/OUT) | | 32 | Bulk/interrupt transfer (IN/OUT) | |
| EP4 | 512x2 (64x2) | Isochronous/bulk/ interrupt transfer (IN/OUT) | DMA Possible | 256x2 (64x2) | Isochronous/bulk/ interrupt transfer (IN/OUT) | DMA Possible |
| EP5 | — | — | | 256x2 (64x2) | Isochronous/bulk/ interrupt transfer (IN/OUT) | DMA Possible |

      FIFO Capacity: The unit is bytes.

Note 1: The selection between the 5EP mode and the 6EP mode is made by bit D2 of the register SYSCON.

Note 2: EP3 permits rate feedback data sequence toggling.

Note 3: EP1, EP2, and EP3 are all mutually independent, and can be assigned for bulk transfer or interrupt transfer individually. It is possible to set the maximum packet size up to 64 bytes (32 bytes for EP3) during both bulk transfer and interrupt transfer.

Note 4: It is possible to set EP4 and EP5 to one of the modes of isochronous transfer, bulk transfer, and interrupt transfer. The maximum packet size can be up to 64 bytes when these end points are set to bulk transfer.
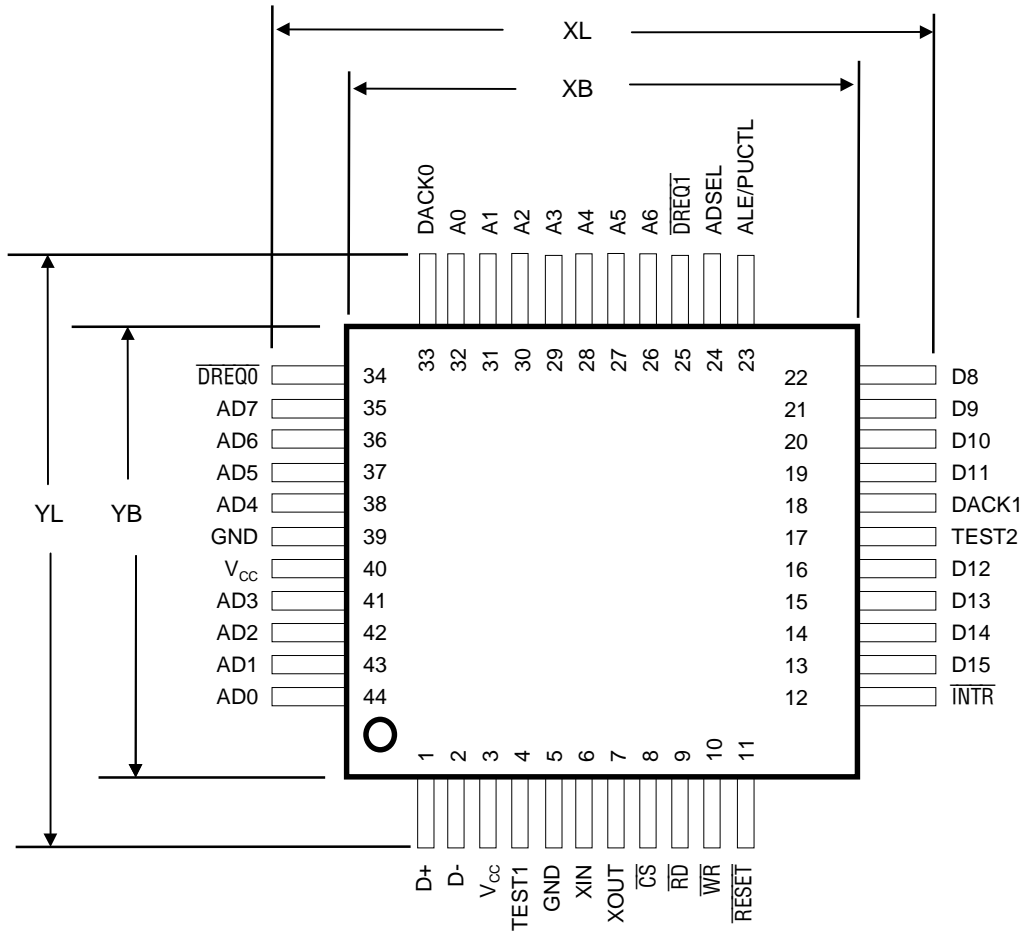
Note 5: When using EP4 and EP5 in the isochronous transfer mode:
In the 5EP mode, the maximum packet size of EP4 is 512 bytes. EP5 cannot be used.
In the 6EP mode, the maximum packet size of both EP4 and EP5 is 256 bytes.

## PIN CONFIGURATION (TOP VIEW)

**44-pin TQFP (Top View)**



Package dimensions (unit: mm)

|        | 44TQFP      |
|--------|-------------|
| XB     | 10.0 ±0.1   |
| XL     | 12.0 ±0.2   |
| YB     | 10.0 ±0.1   |
| YL     | 12.0 ±0.2   |
| Height | 1.2MAX      |
| Lead pitch | 0.8      |

**56-pin LGA (Top View)**

| NC | D8 | D9 | D11 | TEST2 | D12 | D14 | $\overline{\text{INTR}}$ | NC |
|----|----|----|-----|-------|-----|-----|------|-----|
| ALE/ PUCTL | NC | D10 | DACK1 | NC | D13 | D15 | NC | $\overline{\text{RESET}}$ |
| DREQ1 | ADSEL | | | | | | $\overline{\text{RD}}$ | $\overline{\text{WR}}$ |
| A5 | A6 | | | | | | XOUT | $\overline{\text{CS}}$ |
| A4 | NC | | | | | | NC | XIN |
| A2 | A3 | | | | | | TEST1 | GND |
| A0 | A1 | | | Pin No.1 Marking → ◎ | | | D- | $V_{CC}$ |
| DACK0 | NC | AD7 | AD5 | NC | $V_{CC}$ | AD2 | NC | D+ |
| NC | $\overline{\text{DREQ0}}$ | AD6 | AD4 | GND | AD3 | AD1 | AD0 | NC |

## PIN DESCRIPTION

| Pin name | Pin count | I/O | Description |
|---|---|---|---|
| D+, D- | 2 | I/O | USB data |
| XIN, XOUT | 2 | — | Pins for external crystal |
| AD7:AD0 | 8 | I/O | Data bus (LSB)/address inputs |
| A6:A0 | 7 | I | Address inputs |
| D15:D8 | 8 | I/O | Data bus (MSB) |
| $\overline{CS}$ | 1 | I | Chip select signal input pin. Active "L" |
| $\overline{RD}$ | 1 | I | Read signal input pin. Active "L" |
| $\overline{WR}$ | 1 | I | Write signal input pin. Active "L" |
| $\overline{INTR}$ | 1 | O | Interrupt request signal output pin |
| $\overline{DREQ0}$ | 1 | O | DMA0 request output pin |
| $\overline{DREQ1}$ | 1 | O | DMA1 request output pin |
| DACK0 | 1 | I | DMA0 reception signal input pin |
| DACK1 | 1 | I | DMA1 reception signal input pin |
| ALE/PUCTL | 1 | I,O | Address latch enable signal input pin/pull-up control pin |
| ADSEL | 1 | I | Address input format select input pin |
| $\overline{RESET}$ | 1 | I | Reset signal input pin |
| TEST1, TEST2 | 2 | I | Test pin. (Normally at "L") |
| $V_{CC}$ | 2 | | 3.3 V power supply pin |
| GND | 2 | | GND |
| | 44 | | |

## BLOCK DIAGRAM

Local
MCU

A6:A0

AD8:AD0
D15:D8

$\overline{CS}$
$\overline{WR}, \overline{RD}$
$\overline{RESET}$

$\overline{INTR}$

$\overline{DREQ0,1}$

$\overline{DACK0,1}$

Application
Interface

ML60852A

Register
Set

8-byte
Setup
Register

EP0
Transmit
FIF0

EP0
Receive
FIF0

EP1
FIF0

EP2
FIF0

EP3
FIF0

EP4
EP5
FIF0

6/12 MHz
Oscillator

Multiplication
Circuit

Protocol
Engine

USB
Transceiver

6/12 MHz XIN

XOUT

Consult with
a crystal maker
for crystal
peripheral parts

ALE/PUCTL

1.5 kΩ

22Ω

22Ω

D+

D-

**OKI** Semiconductor                                                                                                   **ML60852A**

## ABSOLUTE MAXIMUM RATINGS

| Parameter | Symbol | Condition | Rating | Unit |
|---|---|---|---|---|
| Power Supply | $V_{CC}$ | $V_{SS} = 0$ V $T_J = 25°C$ | −0.3 to + 4.6 | V |
| Input Voltage (Tolerant) | $V_{IT}$ | | −0.3 to + 6.0 | V |
| Input Voltage (Normal) | $V_I$ | | −0.3 to + $V_{CC}$ + 0.3 | V |
| Storage Temperature | $T_{STG}$ | — | −65 to + 150 | °C |

## RECOMMENDED OPERATING CONDITIONS

| Parameter | Symbol | Condition | Range | Unit |
|---|---|---|---|---|
| Power Supply | $V_{CC}$ | — | 3.0 to 3.6 | V |
| Operating Temperature | $V_{OP}$ | — | -20 to 80 | °C |
| Oscillation Frequency | $F_{OSC}$ | — | 12±0.03 or 6±0.015 | MHz |

## ELECTRICAL CHARACTERISTICS

### DC Characteristics (1)

($V_{CC}$ = 3.0 to 3.6 V, Ta = -20 to +80 °C)

| Parameter | Symbol | Condition | Min. | Typ. | Max. | Unit | Applicable pin |
|---|---|---|---|---|---|---|---|
| High-level Input Voltage | $V_{IH}$ | — | 2.1 | — | 5.5 | V | Note 1 |
| Low-level Input Voltage | $V_{IL}$ | — | −0.3 | — | −0.7 | V | |
| High-level Input Voltage | $V_{IH}$ | — | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | XIN |
| Low-level Input Voltage | $V_{IL}$ | — | −0.3 | — | $V_{CC} \times 0.2$ | V | |
| Schmitt Trigger Input Voltage | $V_{t+}$ | — | — | 1.5 | 2.0 | V | $\overline{RESET}$ |
| | $V_{t-}$ | — | 0.7 | 1.0 | — | V | |
| | $\Delta V_t$ | $(V_{t+}) - (V_{t-})$ | 0.4 | 0.5 | — | V | |
| High-level Output Voltage | $V_{OH}$ | $I_{OH}$ = −100 µA | $V_{CC} - 0.2$ | — | — | V | Note 2 |
| | | $I_{OH}$ = −4 mA | 2.4 | — | — | V | |
| Low-level Output Voltage | $V_{OL}$ | $I_{OL}$ = 100 µA | — | — | 0.2 | V | |
| | | $I_{OL}$ = 4 mA | — | — | 0.45 | V | |
| High-level Input Current | $I_{IH}$ | $V_{IH} = V_{CC}$ | — | 0.1 | 10 | µA | Note 3 |
| Low-level Input Current | $I_{IL}$ | $V_{IL}$ = 0V | −10 | −0.1 | — | µA | |
| 3-state Output Leakage Current | $I_{OZH}$ | $V_{OH} = V_{CC}$ | — | 0.1 | 10 | µA | D15:D8 |
| | $I_{OZL}$ | $V_{OL}$ = 0V | −10 | −0.1 | — | µA | AD7:AD0 |
| Power Supply Current (Operating) | $I_{CC}$ | — | — | — | 50 | mA | $V_{CC}$ |
| Power Supply Current (Standby) | $I_{CCS}$ | Note 4(~50°C) | — | — | 60 | µA | $V_{CC}$ |
| | | Note 4(~80°C) | — | — | 400 | | |

Notes: 1. Applied to D15: D8, AD7: AD0, A6: A0, $\overline{CS}$, $\overline{RD}$, $\overline{WR}$, DACK0, DACK1, ALE, and ADSEL.
      2. Applied to D15: D8, AD7: AD0, A6: A0, $\overline{INTR}$, $\overline{DREQ0}$, $\overline{DREQ1}$ and ALE/PUCTL
      3. Applied to XIN, AD7: AD0, $\overline{CS}$, $\overline{RD}$, $\overline{WR}$, DACK, ALE, and ADSEL.
      4. The XIN pin is fixed at a high level or a low level in the suspend state.
         All the output pins are open.

**DC Characteristics (2) USB Port**

($V_{CC}$ = 3.0 to 3.6 V, Ta = -20 to +80 °C)

| Parameter | Symbol | Condition | Min. | Typ. | Max. | Unit | Applicable pin |
|---|---|---|---|---|---|---|---|
| Differential Input Sensitivity | $V_{DI}$ | (D+) − (D −) | 0.2 | — | — | V | D+, D- |
| Differential Common Mode Range | $V_{CM}$ | Includes $V_{DI}$ range | 0.8 | — | 2.5 | V | |
| Single Ended Receiver Threshold | $V_{SE}$ | — | 0.8 | — | 2.0 | V | |
| High-level Output Voltage | $V_{OH}$ | RL of 15 kΩ to GND | 2.8 | — | 3.6 | V | |
| Low-level Output Voltage | $V_{OL}$ | RL of 1.5 kΩ to 3.6 V | — | — | 0.3 | V | |
| Output Leakage Current | $I_{LO}$ | 0 V <$V_{IN}$ <3.3 V | −10 | — | +10 | µA | |

**AC Characteristics USB Port**

($V_{CC}$ = 3.0 to 3.6 V, Ta = -20 to +80 °C)

| Parameter | Symbol | Condition | Min. | Typ. | Max. | Unit | Applicable pin |
|---|---|---|---|---|---|---|---|
| Rise Time | $t_R$ | CL = 50 pF | 4 | — | 20 | ns | D+, D- |
| Fall Time | $t_F$ | CL = 50 pF | 4 | — | 20 | ns | |
| Output Signal Crossover Voltage | $V_{CRS}$ | — | 1.3 | — | 2 | V | |
| Driver Output Resistance | $Z_{DRV}$ | Steady State Driver | 28 | — | 44 | Ω | |
| Data Rate | $T_{DRATE}$ | Ave. Bit Rate (12 Mbps ±0.25%) | 11.97 | — | 12.03 | Mbps | |

Notes: 1. 1.5 kΩ pull-up to 3.3 V on the D + data line.

## SIGNAL DESCRIPTIONS

### USB Interface

| Signal | Type | Assertion | Description |
|--------|------|-----------|-------------|
| D+ | I/O | — | USB data (Plus). This signal and the D- signal are the transmitted or received data from/to USB Bus. The table below shows values and results for these signals.<br><br><table><tr><th>D+</th><th>D-</th><th>Result</th></tr><tr><td>0</td><td>0</td><td>Single end 0</td></tr><tr><td>0</td><td>1</td><td>Differential "0"</td></tr><tr><td>1</td><td>0</td><td>Differential "1"</td></tr><tr><td>1</td><td>1</td><td>Undefined</td></tr></table> |
| D- | I/O | — | USB Data (Minus). This signal and the D+ signal are the transmitted or received data from/to USB Bus. The table above shows values and results for these signals. |

### Crystal Oscillator Interface

| Signal | Type | Assertion | Description |
|--------|------|-----------|-------------|
| XIN | I | — | For internal oscillation, connect a crystal to XIN and XOUT. |
| XOUT | O | — | For external oscillation, supply an external 12 MHz clock signal to XIN. Set XOUT to be open. |

**Application Interface**

| Signal | Type | Assertion | Description |
|--------|------|-----------|-------------|
| D15: D8 | I/O | — | Upper byte (MSB) of data bus. |
| AD7: AD0 | I/O | — | Lower byte (LSB) of data bus when ADSEL is LOW. Address and lower byte of data bus are multiplexed when ADSEL is HIGH. |
| A6: A0 | I | — | Address when ADSEL is LOW. |
| $\overline{CS}$ | I | LOW | Chip Select. When this signal is asserted LOW, the ML60852A is selected and ready to read or write data. This signal is invalid in single address mode during DMA transfer. |
| $\overline{RD}$ | I | LOW | Read Strobe. When this signal is asserted LOW, the Read instruction is executed. |
| $\overline{WR}$ | I | LOW | Write Strobe. When this signal is asserted LOW, the Write instruction is executed. |
| $\overline{INTR}$ | O | (Note 1) | Interrupt Request. When this signal is asserted, the ML60852A makes an interrupt request to the application. |
| $\overline{DREQ0}$ | O | (Note 1) | DMA Request. This signal requests the DMA0 to make a DMA transfer. |
| $\overline{DREQ1}$ | O | (Note 1) | DMA Request. This signal requests the DMA1 to make a DMA transfer. |
| DACK0 | I | (Note 2) | DMA Acknowledge Signal for $\overline{DREQ0}$. This signal, when asserted, enables accessing FIFOs, without address bus setting. |
| DACK1 | I | (Note 2) | DMA Acknowledge Signal for $\overline{DREQ1}$. This signal, when asserted, enables accessing FIFO, without address bus setting. |
| ALE/PUCTL | I or O | HIGH | When ADSEL is HIGH, the address and $\overline{CS}$ on AD7: AD0 are latched at the trailing edge of this signal. D+ pull-up resistor connection output when ADSEL is LOW. $V_{CC}$ potential when bit D3 of SYSCON register is "1", and high-impedance when it is "0". |
| ADSEL | I | — | When ADSEL is LOW, the address is input on A6: A0 and data is input on AD7: AD0. When ADSEL is HIGH, address and data are multiplexed on AD7: AD0. |
| $\overline{RESET}$ | I | LOW | System Reset. When this signal is asserted LOW, the ML60852A is reset. When the ML60852A is powered on, this signal must be asserted for 1 μs or more. |

Notes: 1. The assertion polarity can be modified by appropriately initializing the polarity selection register (POLSEL).
         The default is LOW.
     2. The assertion polarity can be modified by appropriately initializing the polarity selection register (POLSEL).
         The default is HIGH.

## FUNCTIONAL DESCRIPTIONS

### (1) USB Interface

The ML60852A is a USB device controller. The ML60852A provides the following functions which are bases for a USB protocol. Therefore, the application can process a lot of its own functions.
- Bit synchronization
- Encoding and decoding NRZI signals.
- Generating and detecting Sync bytes.
- Bit stuffing
- Generating and checking CRCs (CRC5, CRC16).
- Encoding and decoding PID (packet identifier).
  1. Decoding token.
  2. Encoding and decoding handshake.
- Generating and detecting SOP and EOP.
- Enpacket (packing) and depacket (unpacking)
- Comparing device addresses.
- Storing 8-byte setup data from a host into the setup register.
- Transmitting data in transmit FIFO.
- Storing receive data into receive FIFO of the corresponding endpoint.

### (2) USB Transfer Modes

The ML60852A supports four kinds of transfer modes such as control transfer mode , interrupt transfer mode, bulk transfer mode, and isochronous transfer mode, which are specified by USB Standards.
- (a) Control transfer mode is used to receive and respond to configurations and commands from a host, and to exchange status information between the host and peripherals.
- (b) Bulk transfer mode is used for transferring large amounts of data in a limited time period when sufficient USB bus resources become available.
- (c) Interrupt transfer mode is used for transferring moderate amounts of data in a specified amount of time.
- (d) The isochronous transfer mode is used to continuously transfer audio data, moving pictures data and other data.
- (e) Isochronous transfer mode is used for streaming, real time data transfers such as audio or video.

**OKI** Semiconductor                                                                                 **ML60852A**

### (3) Endpoints and FIFOs

In the ML60852A, it is possible to select, by making appropriate setting in the SYSCON register, the 5EP mode in which there are five end points or the 6EP mode in which there are six end points.  Although the transfer mode that can be used by EP0 is fixed, it is possible to select either the bulk transfer mode or the interrupt transfer mode for the end points EP1, EP2, and EP3, and one of the modes of isochronous, bulk, or interrupt transfer can be selected for EP4 and EP5.  In addition, it is possible to selectively set the direction of data transfer for EP1 to EP5.

| End point | 5EP Mode | | | 6EP Mode | | |
|---|---|---|---|---|---|---|
| | FIFO Capacity | Transfer mode | Remarks | FIFO Capacity | Transfer mode | Remarks |
| EP0 | Reception 32 Transmission 32 | C | | Reception 32 Transmission 32 | Control transfer | |
| EP1 | 64x2 | B/Int (IN/OUT) | DMA Possible | 64x2 | B/Int (IN/OUT) | DMA Possible |
| EP2 | 64x2 | B/Int (IN/OUT) | DMA Possible | 64x2 | B/Int (IN/OUT) | DMA Possible |
| EP3 | 32 | B/Int (IN/OUT) | Rate | 32 | B/Int (IN/OUT) | Rate |
| EP4 | 512x2 (64x2) | Iso/B/Int (IN/OUT) | DMA Possible | 256x2 (64x2) | Iso/B/Int (IN/OUT) | DMA Possible |
| EP5 | — | — | | 256x2 (64x2) | Iso/B/Int (IN/OUT) | DMA Possible |

          FIFO Capacity: The unit is bytes.
Note:   Transfer modes:
          C = Control transfer
          B = Bulk transfer
          Int = Interrupt transfer
          Iso = Isochronous transfer
          Rate = Compatible with data sequence toggling of rate feedback.

**(4) Operation of Control Transfer**

Control transfer is the default transfer mode for host-device communications as outlined in USB specifications. Control transfer uses structured message pipes and is composed of the following three stages:

(a)  Setup stage

In this stage, a setup token and 8 bytes of setup data are transmitted from the host. The ML60852A decodes the setup token, and automatically stores the 8 bytes of setup data in the setup register.  When this is completed normally, the ML60852A returns ACK to the host.

The 8-byte setup data is the standard request code defined in Section 9.3 of the USB Standards, or a code of the requests unique to each device class, etc.  The request is decoded on the local MCU side.

(b)  Data stage

If the request specified by the 8-byte setup data is also accompanied by transfer of parameter data from the host to the device, the transfer is a control write transfer, and the OUT token and the data packet are transmitted from the host.  When these are received normally, the ML60852A stores the parameter data in the EP0 receive FIFO and returns ACK to the host.

If the request is accompanied by transfer of parameter data from the device to the host, the transfer is a control read transfer, and when the host sends the IN token, the ML60852A sends the parameter data that was already stored beforehand in the EP0 transmit FIFO by the local MCU.  When the host receives this normally, it returns an ACK to the ML60852A.

On the other hand, in the case of requests that do not contain any parameter data that need to be transmitted or received, data stage will not be present and the processing proceeds directly to the status stage from the setup stage.

(c)  Status stage

The status stage is a stage intended for reporting the status of the result of executing a request from the device to the host.  During a control write transfer or a control transfer without data, an IN token is sent by the host, and the ML60852A returns a response to it.  During a control read transfer, an OUT token and a zero length packet (ZLP) is sent by the host, and the ML60852A returns a response to it.

During the above control transfers, the local MCU needs only to read from or write to the 8-byte setup registers mapped at 00h to 07h, the EP0 transmit FIFO mapped at 70h, and the EP0 receive FIFO mapped at 78h according to the interrupt cause, and all other operations will be carried out automatically by the ML60852A.

**(5) Data packet transmission and reception procedure during bulk transfer and interrupt transfer modes**

The ML60852A is normally used on the peripheral device side. In this method of use, the ML60852A is connected on one side to the host via the USB bus and is connected on the other side via a parallel interface to the local microcontroller (local MCU) inside the peripheral device.

Transfer of data is the major function in all types of transfer modes other than the control transfer mode. When carrying out transfer of data packets between the ML60852A and the host, the following packet communication is carried out via the USB bus for data transfer of each packet.

(a) Token packet transfer (IN token or OUT token) from the host to the ML60852A.

(b) Data packet transfer in the desired direction (from the host to the device or from the device to the host).

(c) Transfer of handshake packet in a direction opposite to that of the data packet.

When packet transfer is completed normally, an ACK packet is returned in step (c) and the operation proceeds to the next packet transfer.

The ML60852A requests the local MCU to transmit or receive a packet of data by asserting the $\overline{\text{INTR}}$ pin. The interrupt cause will be "packet ready". The transmit packet ready interrupt is one that requests that the packet of data to be transmitted be written in the transmit FIFO, and the receive packet ready interrupt is one that requests the local MCU to read out the data that has been received and stored in the receive FIFO.

The above procedures of transferring one packet of data are explained below for transmission and reception separately.

1) During transmission

The local MCU writes one packet of data that has to be transmitted in the transmit FIFO of the corresponding EP in the ML60852A, and sets the transmit packet ready bit of the corresponding EP status register of the ML60852A. When the host transmits an IN token packet to the ML60852A, the ML60852A transmits to the host the data packet stored in the above transmit FIFO. When the host receives one data packet normally, it returns an ACK packet to the ML60852A. Consequently, the ML60852A resets the transmit packet ready status, thereby completing the transfer of one data packet over the USB bus. When the transmit packet ready status is reset, the ML60852A gives a request to the local MCU in terms of a transmit packet ready interrupt thereby prompting the local MCU to write the next packet of data to be transmitted.

2) During reception

The host sends to the ML60852A an OUT token followed by a data packet. The ML60852A stores the received data packet in the receive FIFO of the corresponding EP. When it is confirmed that all the data packets have been accumulated and that there is no error, the ML60852A returns an ACK packet to the host. At the same time, the receive packet ready bit of the corresponding EP status register will also be set and a request is sent to the local MCU in terms of an interrupt. Upon receiving this interrupt, the local MCU reads out the received data from the ML60852A and resets the receive packet ready bit.

### (6) Data packet transmission and reception procedure during isochronous transfer mode

Transfer of data is the major function in the isochronous transfermode.  When carrying out isochronous transfer between the ML60852A and the host, the following packet communications are carried out via the USB bus for the data transfer of each packet.

(a)  Token packet transfer (IN token or OUT token) from the host to the ML60852A.
(b)  Data packet transfer in the desired direction (from the host to the device or from the device to the host).
     In the isochronous transfer mode, there is no handshaking that reports whether or not the packet transfer was done normally.
     The ML60852A requests the local MCU to send or receive packet data by asserting the $\overline{\text{INTR}}$ pin.  The interrupt cause is SOF.  Upon receiving this interrupt, the local MCU writes the packet data into the transmit FIFO of the EP set for transmission (ISO IN) in the isochronous transfer mode, or reads out data from the receive FIFO of the EP set for reception (ISO OUT) in the isochronous transfer mode.
     The above procedures of transferring one packet of data are explained below for transmission and reception separately.

1)  During transmission
    The EP for ISO IN has a two-layer FIFO configuration.  One FIFO is used for storing the packet data that is written in by the MCU via the local bus.  The other FIFO is used for transmitting the stored data to the USB bus when an IN token is received.  The roles of the two FIFOs are interchanged when an SOF packet is received. Upon receiving an SOF interrupt, the local MCU writes the data to be transmitted during the next frame into the corresponding transmit FIFO of the EP of the ML60852A.  When the host transmits an IN token packet, the ML60852A transmits to the host the packet data written in the transmit FIFO during the previous frame.
2)  During reception
    The EP for ISO OUT has a two-layer FIFO configuration.  One FIFO is used for storing the packet data that is output to the local bus when the MCU reads the received packet data.  The other FIFO is used for storing the packet data received from the USB bus.  The roles of the two FIFOs are interchanged when an SOF packet is received.
    Upon receiving an SOF interrupt, the local MCU reads out the data that has been received during the previous frame from the corresponding receive FIFO of the EP of the ML60852A.  When the host transmits an OUT token and a data packet to the ML60852A, the ML60852A stores that received data packet in the receive FIFO, and that data packet is read out by the local MCU during the next frame.

### (7) Packets and Packet Sizes

The ML60852A packs the transmit data into packets and unpacks (restores to the original form) the received data. The packed data that is recognized by the software client is a set of data consisting of one or more packets, and this is called an I/O request Packet (IRP).

Among the several packets in an IRP, all the packets other than the last packet are transferred with the maximum packet size. Only the last packet can be transferred as a "short packet", that is, a packet whose size is less than the maximum packet size.

I/O Request Packet (IRP)

| Packet 1 | Packet 2 | ●●● | Packet n-1 | Packet n |

1 Packet

M Bytes

Maximum packet size
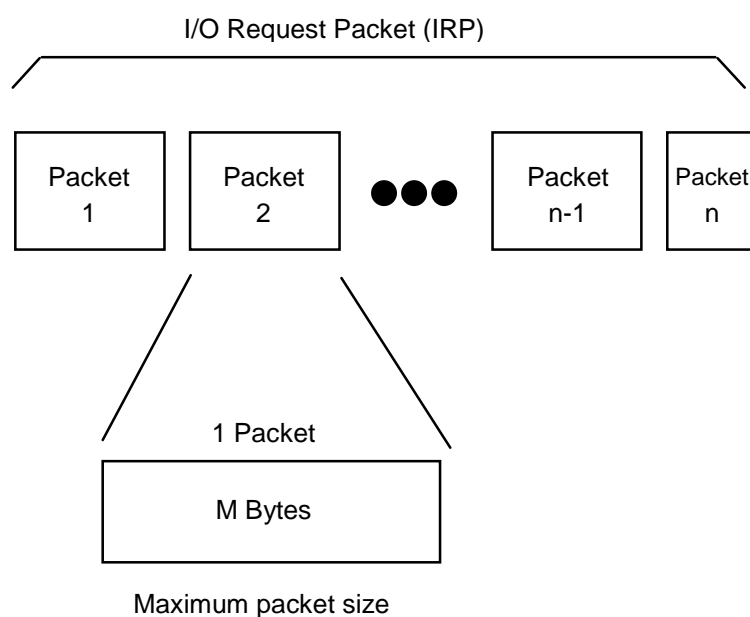
The ML60852A has payload registers corresponding to each end point, and it is possible to set the maximum packet size for each end point in these registers. The maximum packet size should be within the capacity of the corresponding FIFO, and can be set as follows:

    (1) EP0 Receive packet size can be 32 bytes or less;
    (2) EP0 Transmit packet size can be 32 bytes or less;
    (3) EP1 Transmit/receive packet size can be 64 bytes or less;
    (4) EP2 Transmit/receive packet size can be 64 bytes or less;
    (5) EP3 Transmit/receive packet size can be 32 bytes or less;
    (6) EP4 Bulk/interrupt transmit/receive packet size can be 64 bytes;
        In the 5EP mode, the EP4 isochronous packet size can be 512 bytes or less;
        In the 6EP mode, the EP4 isochronous packet size can be 256 bytes or less;
    (7) In the 6EP mode, the EP5 bulk/interrupt packet size can be 64 bytes or less;
        In the 6EP mode, the EP5 isochronous packet size can be 256 bytes or less.

On the USB bus, the separation between successive packets is distinguished by appending a special signal condition called EOP (End of Packet) at the end of each packet. The appending of EOP during transmission and the detection and removal of EOP during reception are carried out by the ML60852A automatically.

(1) At the time of transmission, the packet is deemed to have ended when the local MCU has completed writing the required number of bytes of data in the transmit FIFO and has then asserted the transmit ready status bit. (The actual addition of EOP is executed at the time of transmitting the data over the USB bus after waiting for the IN token from the host.) The packet will be a short packet if the transmit packet ready status bit is asserted after writing data with less number of bytes than the maximum packet size. In particular, by asserting the transmit packet ready status bit without writing any data, it is possible to form a null packet whose data length is zero.

(2)  At the time of reception, when an EOP is detected in the received data string, the ML60852A recognizes it as the end of the received packet and asserts the receive packet ready status bit.  The number of bytes in the received packet is counted automatically by the receive byte count register (Note 1) corresponding to that end point.
Note 1: Receive byte count register address: 58h to 5Dh and 74h to 75h.

**(8) Interrupts**

The ML60852A requests interrupts to the local MCU, etc., by asserting the -INTR pin.  The interrupt causes are the following:

    (a)  Setup ready for the 8-byte setup data
    (b)  EP0 receive packet ready
    (c)  EP0 transmit packet ready
    (d)  EP1 transmit/receive packet ready
    (e)  EP2 transmit/receive packet ready
    (f)  EP3 transmit/receive packet ready
    (g)  EP4 transmit/receive packet ready
    (h)  EP5 transmit/receive packet ready
    (i)  SOF
    (j)  USB Bus reset assert
    (k)  USB Bus reset de-assert
    (l)  Suspend
    (m) Awake

Although there is only one $\overline{\text{INTR}}$ pin, the local MCU can identify the contents of the interrupt by reading out the interrupt status register 1 (INTSTAT1) and the interrupt status register 2 (INTSTAT2).  These interrupts can also be masked dynamically by making individual settings in the interrupt enable register 1 (INTENBL1) and the interrupt enable register 2 (INTENBL2).
The causes of the interrupts, their setting and resetting conditions, and the responses to them are described below.  The functions of the setup ready bit and the packet ready bit can, in some situations, be different from those described here because of some special automatic operations done by the ML60852A.  Please see the descriptions of the registers EP0STAT to EP5STAT for more details of such functions.

(1) Setup ready interrupt

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| Setup ready interrupt generation | ML60852A | The setup ready bit (D2 of EP0STAT) is asserted when the 8-byte setup control data is received normally and has been stored in the set of setup registers. |
|  |  | An interrupt is generated at this time if D0 of INTENBL1 has been asserted. |
|  |  | $\rightarrow$ The firmware can now read the set of setup registers. |
| End of setup ready interrupt | Local MCU (firmware) | After making the firmware read the 8-byte setup data, write a "1" in bit D2 of EP0 status register (EP0STAT).  This causes the interrupt to be de-asserted. |
|  |  | The interrupt will not be de-asserted If a new 8-byte setup data is received during this period.  In this case, discard the setup data that was being read at that time and read the new 8-byte setup data. |

(2) EP0 Receive packet ready interrupt
   This is used mainly during the reception of a data packet in a control write transfer.

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| EP0 Receive packet ready interrupt generation | ML60852A | The EP0 receive packet ready bit (D0 of EP0STAT) is asserted during a control write transfer when the processing has changed from the setup stage to the data stage, and the ML60852A has detected EOP of the data packet and has stored the data without error in the EP0 receive FIFO. The end of a packet is recognized when an EOP has arrived in the cases of both full packets and short packets.<br>An interrupt is generated at this time, if the EP0 receive packet ready interrupt enable bit (D6 of INTENBL1) has been asserted.<br>(EOP: End of packet) |
| End of EP0 receive packet ready interrupt | Local MCU (firmware) | In the case of EP0 reception, after the number of bytes of the EP0 receive FIFO data indicated by the EP0 receive byte count register (EP0RXCNT) has been read, write a "1" to the EP0 receive packet ready bit (bit D0 of EP0STAT). (This status is reset when a "1" is written in this bit.) |

Note:   A short packet is a packet with a number of bytes less than the maximum packet size.

(3) EP0 Transmit packet ready interrupt
    This is used mainly during the transmission of a data packet in a control read transfer.

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| EP0 Transmit packet ready interrupt generation | ML60852A | During a control read transfer when the processing has changed from Setup stage to Data stage, it is necessary for the local MCU (firmware) to write the data to be transmitted to the transmit FIFO of EP0. The local MCU must enable EP0 transmit packet ready interrupt bit (bit D7 of INTENBL1) to facilitate write operation to EP0TXFIFO. At this time, an EP0 transmit packet ready interrupt is generated. The cause of the interrupt is EP0 transmit packet ready bit (D1 of EP0STAT) being low.<br><br>This interrupt will invoke the local MCU (firmware) to process EP0 transmit operation which will result in sending packets to the host.<br><br>For the second and subsequent packets, in addition to EP0 transmit packet ready interrupt being enabled, before the interrupt is generated, it is necessary for an ACK response to come from the host for the packet that has just been sent. |
| End of EP0 transmit packet ready interrupt | Local MCU (firmware) | In the case of EP0 transmission, after one packet of EP0 transmit data has been written in EP0TXFIFO, write a "1" into the EP0 transmit packet ready bit (bit D1 of EP0STAT). This puts the ML60852A in a state in which it can transmit the data (that is, it can transmit the data packet when an IN token arrives), and the $\overline{\text{INTR}}$ pin is de-asserted at the same time.<br><br>Even when the number of bytes in the write data is less than the maximum packet size, it is possible to transmit the data by writing a "1" into the transmit packet ready status bit. This makes it possible to transmit a short packet. |

(4)  Receive packet ready interrupts (EP1, EP2, EP3, EP4 bulk, EP5 bulk)
     These interrupts are generated when the respective EP has received an appropriate data packet from the USB
     bus and the local MCU can read that data.

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| Receive packet ready interrupt generation | ML60852A | The receive packet ready bit (D0) of the corresponding EP status register (EPnSTAT) is asserted during data reception when the EOP of the data packet has been received and the data has been stored without error in the corresponding FIFO.  The end of a packet is recognized when an EOP has arrived in the cases of both full packets and short packets.<br><br>An interrupt is generated at this time, if the corresponding receive packet ready interrupt enable bit has been asserted.<br>(EOP: End of packet) |
| End of receive packet ready interrupt | Local MCU (firmware) | After the number of bytes in the receive FIFO data (EPnFIFO) indicated by the corresponding receive byte count register (EPnRXCNT) has been read, write a "1" into the receive packet ready bit D0 of the corresponding EP status register (EPnSTAT).  (This status is reset when a "1" is written in this bit.) |

(5) Transmit packet ready interrupts (EP1, EP2, EP3, EP4 bulk, EP5 bulk)
These interrupts are generated when it is possible for the local MCU to write the data packet to be sent to the USB bus from the corresponding EP.

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| Transmit packet ready interrupt generation | ML60852A | (1) In the case of bulk transfer and interrupt transfer<br><br>When the respective EP has been set for transmission (bit D7 of EPnCONF='1'), the transmit packet ready bit of the corresponding EP (bit D1 of EPnSTAT) is de-asserted at which time it is possible to write the transmit data into the FIFO.<br><br>At this time, an interrupt is generated if the corresponding EP transmit packet ready interrupt enable bit (INTENBL1) has been asserted.<br><br>For the second and subsequent packets, in addition to this condition, before the interrupt is generated, it is necessary for an ACK response to come from the host for the packet that has just been sent. |
| End of transmit packet ready interrupt | Local MCU (firmware) | (1) In the case of bulk transfer and interrupt transfer<br><br>After the one packet of the corresponding EP transmit data has been written in EPnTXFIFO, write a "1" into the corresponding transmit packet ready bit (bit D1 of EPnSTAT).  This puts the ML60852A in a state in which it can transmit the data and the $\overline{INTR}$ pin is de-asserted at the same time.<br><br>When the number of bytes in the write data is less than the maximum packet size, a short packet can be sent by setting (write '1') the transmit packet ready status of the endpoint. |

Note) EP1,EP2,EP4,and EP5 each has 2 layers of FIFO. Settings in the transmit packet ready control register (TXPKTCONT) control the assert and de-assert conditions of transmit pactet ready interrupts. For details, see"Transmit Packet Ready Control Register in INTERNAL REGISTERS.

**OKI** Semiconductor

(6)  SOF Interrupt

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| SOF Interrupt generation | ML60852A | When SOF interrupt has been enalbed (bit D0 of INTENBL2), and an SOF packet is detected on the USB bus. |
| End of SOF interrupt | Local MCU (firmware) | When a "1" is written in the SOF interrupt status bit of the interrupt status register 2 (bit D2 of INTSTAT2). |

(7)  USB Bus reset assert interrupt

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| USB Bus reset assert interrupt generation | ML60852A | The ML60852A automatically detects the condition when the SE0 state continues for 2.5µs or longer at the D+ and D- pins. Once this condition is detected, if USB bus reset assert interrupt has been enabled (D1 of INTENBL2) an interrupt is generated.<br>→ Carry this out by firmware processing for bus reset. |
| End of USB bus reset assert interrupt | Local MCU (firmware) | When a "1" is written in the corresponding bit of the interrupt status register 2 (D1 of INTSTAT2). |

(8)  USB Bus reset de-assert interrupt

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| USB Bus reset de-assert interrupt generation | ML60852A | When USB Bus reset de-assert interrupt enable bit (D2 of INTENBL2) has been enalbed and there is a recovery to the J state from the SE0 state of 2.5µs or longer at the D+ and D- pins.<br>→ Carry this out by firmware processing for bus reset release. |
| End of USB bus reset de-assert | Local MCU (firmware) | When a "1" is written in the corresponding bit of the interrupt status register 2 (D2 of INTSTAT2). |

(9)  Suspend state interrupt

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| Suspend state interrupt generation | ML60852A | When device suspend interrupt (D3 of INTENBL2) has been enabled and idle condition persists for 3ms or more at the D+ and D- pins.<br>→ The internal oscillations in the ML60852A are stopped automatically when the idle condition continues for an additional 2ms after this interrupt has been generated. The firmware can take steps to put the device in the power save mode. |
| End of suspend state interrupt | Local MCU (firmware) | When a "1" is written in the corresponding bit of the interrupt status register 2 (INTSTAT2). |

(10) Awake interrupt

| Operation | Source of operation | Description (conditions, responses, etc.) |
|---|---|---|
| Awake interrupt generation | ML60852A | When device awake interrupt (D4 of INTENBL2) has been enabled and the EOP, which is the end of resume signal, is detected at the D+ and D- pins. |
| End of awake interrupt | Local MCU (firmware) | When a "1" is written in the corresponding bit of the interrupt status register 2 (INTSTAT2). |

20

**(9) DMA (Direct Memory Access)**

It is possible to carry out 8-bit wide or 16-bit wide DMA transfer for the bulk transfer of EP1, EP2, EP4, and EP5, and for the isochronous transfer of EP4 and EP5.  The data bus used is the following:
    During 8-bit transfer: AD7 to AD0
    During 16-bit transfer: D15 toD8, AD7 to AD0
It is possible to carry out DMA transfers over two channels, Channel 0 and Channel 1.  Both demand transfer and single transfer are supported.  The settings of the DMA transfer mode and parameters are done using the DMA control register and the DMA interval register described later in this manual.

In the demand transfer mode, the $\overline{\text{DREQ}}$ pin is asserted when the reading or writing of a data packet becomes possible.  The $\overline{\text{DREQ}}$ pin is de-asserted when the transfer of all the data of the receive packets is completed by the external DMA controller.  Therefore, other devices cannot access the local bus during DMA transfer.

On the other hand, in the single transfer mode, the $\overline{\text{DREQ}}$ pin is de-asserted at the end of transfer of the number of bytes (or words) of one transfer, and the other devices can access the local bus during this period.

**(10) Power-down**

When the ML60852A detects the suspend state on the USB bus, it automatically stops the internal oscillations and enters the power-down state.  When the resume signal is detected on the USB bus, the oscillations are restarted automatically and the power-down state is released.

**OKI** Semiconductor                                                                                 **ML60852A**

**(11) Operation of 2-layer FIFO structure during Bulk Transfer**

The FIFOs of EP1 and EP2 have a 64 bytes x 2-layer structure.  Also, when EP4 is assigned for bulk transfer, its FIFO also has a 64-bytes x 2-layer structure.  As a consequence, these FIFOs can temporarily store a maximum of 128 bytes of bulk transfer data. Please note that the double layered FIFO operation can be modified by changing the settings in the transmit packet ready control register (TXPKTCONT).

(1) 2-Layer reception (bulk-out) operation ("O" indicates the assert (set to '1') condition and "x" indicates de-assert (set to '0') condition)

| | In the case of 1→2→3→4→5a→6<br>In the case of 1→2→3→4→5b→6 | Layer A 64 bytes | Layer B 64 bytes | Layer A PKT RDY | Layer B PKT RDY | EPn receive PKT RDY | $\overline{\text{INTR}}$ |
|---|---|---|---|---|---|---|---|
| 1 | Start storing data in layer A FIFO | ▬ | | x | x | x | x |
| 2 | Data of one packet has been stored. | ▬▬ | | ○ | x | ○ | ○ |
| 3 | Start reception and storing of data in layer B. | ▬▬ | | ○ | x | ○ | ○ |
| 4 | Local MCU starts reading layer A. | ▬▬ | | ○ | x | ○ | ○ |
| 5a | When the storing of packet in layer B is completed before the completion of reading layer A. | ▬▬ | | ○ | ○ | ○ | ○ |
| 5b | When the reading of packet in layer A is completed before the completion of storing data in layer B. | | ▬ | x | x | x | x |
| 6 | From 5a: Layer A has become empty.<br>From 5b: Layer B has become full. | | ▬ | x | ○ | ○ | ○ |
| 7 | Start reading layer B. | | ▬ | x | ○ | ○ | ○ |

Note: The above illustration assumes that the local MCU (firmware) resets the receive packet ready bit of the respective EPnSTAT register immediately after completition of reading the received data in the corresponding endpoint's FIFO (EPnFIFO).

- When one packet of receive data is stored in layer A of the FIFO and EOP is received, the ML60852A asserts the packet ready bit of EPn and also asserts the $\overline{\text{INTR}}$ pin.  This makes it possible for the local MCU to read the receive data.
- Subsequently, data can be received from the host, and the ML60852A switches the FIFO for storing to layer B.
- When one packet of data described above has been read from layer A of the FIFO, make the local MCU reset the receive packet ready status of EPn (by writing a "1" into bit D0 of EPnSTAT).
- At the time the EPn receive packet ready status is reset, if the reception of layer B has not been completed, the ML60852A resets the EPn receive packet ready status and de-asserts the $\overline{\text{INTR}}$ pin.
- However, if the reception of layer B has been completed a the time the EPn receive packet ready status is reset, the ML60852A rejects the request from the local MCU to reset the EPn receive packet ready status, and continues to maintain the EPn receive packet ready status and the asserted condition of the $\overline{\text{INTR}}$ pin.

(2) 2-Layer transmission (bulk-in) operation ("O" indicates the assert (set to '1') condition and "x" indicates de-assert (set to '0') condition)

| | In the case of 1→2→3→4→5a→6<br>In the case of 1→2→3→4→5b→6 | Layer A 64 bytes | Layer B 64 bytes | Layer A PKT RDY | Layer B PKT RDY | EPn transmit PKT RDY | INTR |
|---|---|---|---|---|---|---|---|
| 1 | Layer A and layer B are both empty. | | | x | x | x | ○ |
| 2 | The local MCU starts writing into layer A. | ▬ | | x | x | x | ○ |
| 3 | Writing of one packet is completed. | ▬▬ | | ○ | x | x | ○ |
| 4 | The data of layer A is being transmitted while the next packet is being written in layer B. | ▬▬ | ▬ | ○ | x | x | ○ |
| 5a | When layer A is still being transmitted while the writing in layer B has already completed. | ▬▬ | ▬▬ | ○ | ○ | ○ | x |
| 5b | When layer B is still being written while the writing in layer A has already completed. | | ▬ | x | x | x | ○ |
| 6 | From 5a: Layer A has become empty.<br>From 5b: Layer B has become full. | | ▬▬ | x | ○ | x | ○ |
| 7 | Transmission of layer B is also started. | | ▬ | x | ○ | x | ○ |

Note: The above illustration assumes that the local MCU (firmware) asserts (write '1') the transmit packet ready bit of the corresponding endpoint in register EPnSTAT immediately after completion of writing the transmit data into the corresponding EPnFIFO.

- If the EPn transmit packet ready interrupt enable bit of INTENBL1 has been asserted, the transmit FIFO is empty, and EPn transmit packet ready bit is de-asserted, the EPn transmit packet ready interrupt is asserted. This makes it possible to write the transmit data into the EPn transmit FIFO.
- When the data of one packet is written in layer A FIFO, make the local MCU set the transmit packet ready status (bit D1 of EPnSTAT). By setting the transmit packet ready status, it becomes possible to transmit data to the host. At this time, since layer B is still empty, the INTR pin maintains the asserted condition, thereby indicating that the next packet data can be written. In this case, although bit D1 of EPnSTAT remains in the '0' condition, the ML60852A recognizes that transmission is possible from layer A and starts transmission when an IN token is received from the host.
- It is possible for the local MCU to write the next packet of transmit data in the layer B FIFO while the data in layer A is being transmitted over the USB bus.
- When the writing of the data to be transmitted in layer B has been completed, the local MCU sets the transmit packet ready bit, and the INTR pin becomes de-asserted at this time if the transmission of layer A data has not been completed (that is, the ACK message is received from the host and the transmit packet ready bit is reset). The local MCU cannot yet write the subsequent packet.
- If the layer A becomes empty before layer B goes into the transmit enable condition and transmission from layer A FIFO is carried out normally, an ACK is received from the host in response to this succussful transmission. This ACK will cause ML60852A to automatically deassert the layer A packet ready bit and hence generate an interrupt cause. The INTR pin remains asserted, and the local MCU can write data into layer A FIFO after completion of writing into layer B FIFO.
- The transmission of data in layer A is continued from state, and when layer A becomes empty and the transmission is completed normally, an ACK response is received from the host, whereupon the ML60852A asserts the INTR pin thereby prompting the local MCU to write data into layer A.

**OKI** Semiconductor

**(12) Error Processing and Retry Operation**

1) Error processing during transmission
   When an error such as a CRC error is detected in the data transmitted by the ML60852A, the host will not send the ACK packet, and hence the ML60852A does not reset the transmit packet ready status, but waits while retaining the current packet of data.  The current packet of data is transmitted again when the next IN token is received from the host.

2) Error processing during reception
   When an error is detected in the data received over the USB bus, the ML60852A does not assert the interrupt signal to the local MCU and will also not send any message to the host (leading to a timeout condition).
   When the timeout condition is generated, the host recognizes that an error has occurred, and can take measures such as re-transmitting the data, etc.  In addition, since no interrupt request is generated, the local MCU will not read the erroneous data.

## INTERNAL REGISTERS

The register file of the ML60852A includes registers to set operating conditions and registers to report the status of operating and the results of processing.
These registers are read only, write only or capable of boh read and write.

The setup register where setup data transmitted from host to device in a control pipe are stored is mapped in this register file. Also, data that are tranmitted or received by the device are transferred using transmit and receive FIFOs mapped in the addresses of the register file.
The mapping of these registers are described in the following pages.

**OKI** Semiconductor                                                                 **ML60852A**

**Address and Names of Registers (1)**

| Category | Address | Symbol | R/W | Register name | Page |
|---|---|---|---|---|---|
| FIFO | 70h | EP0TXFIFO | W | EP0 Transmit FIFO | 34 |
| | 78h | EP0RXFIFO | R | EP0 Receive FIFO | 34 |
| | 79h | EP1FIFO | R or W | EP1 Transmit/Receive FIFO | 35 |
| | 7Ah | EP2FIFO | R or W | EP2 Transmit/Receive FIFO | 35 |
| | 7Bh | EP3FIFO | R or W | EP3 Transmit/Receive FIFO | 36 |
| | 7Ch | EP4FIFO | R or W | EP4 Transmit/Receive FIFO | 36 |
| | 7Dh | EP5FIFO | R or W | EP5 Transmit/Receive FIFO | 37 |
| | | | | | |
| | | | | | |
| | | | | | |
| Common | 00h | bmRequestType | R | bRequest Type Setup Register | 38 |
| | 01h | bRequest | R | bRequest Setup Register | 38 |
| | 02h | wValueLSB | R | wValueLSB Setup Register | 39 |
| | 03h | wValueMSB | R | wValueMSB Setup Register | 39 |
| | 04h | wIndexLSB | R | wIndexLSB Setup Register | 40 |
| | 05h | wIndexMSB | R | wIndexMSB Setup Register | 40 |
| | 06h | wLengthLSB | R | wLengthLSB Setup Register | 41 |
| | 07h | wLengthMSB | R | wLengthMSB Setup Register | 41 |
| | | | | | |
| | 20h | DVCADR | R/W | Device Address Register | 44 |
| | 21h | INTSTAT1 | R/Rst | Interrupt Status Register 1 | 45 |
| | 22h | INTSTAT2 | R/Rst | Interrupt Status Register 2 | 46 |
| | | | | | |
| | 24h | INTENBL1 | R/W | Interrupt Enable Register 1 | 47 |
| | 25h | INTENBL2 | R/W | Interrupt Enable Register 2 | 48 |
| | | | | | |
| | 2Dh | FRAMELSB | R | Frame Number LSB Register | 49 |
| | 2Eh | FRAMEMSB | R | Frame Number MSB Register | 49 |
| | 2Fh | SYSCON | R/W | System Control Register | 50 |
| | 30h | POLSEL | R/W | Polarity Selection Register | 50 |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| DMA | 10h | DMA0CON | R/W | DMA0 Control Register | 42 |
| | 11h | DMA0INTVL | R/W | DMA0 Interval Register | 43 |
| | 12h | DMA1CON | R/W | DMA1 Control Register | 42 |
| | 13h | DMA1INTVL | R/W | DMA1 Interval Register | 43 |
| | | | | | |
| | | | | | |

**Address and Names of Registers (2)**

| Category | Address | Symbol | R/W | Register name | Page |
|---|---|---|---|---|---|
| | 40h | EP0CONF | R/W | EP0 Configuration Register | 52 |
| | 41h | EP1CONF | R/W | EP1 Configuration Register | 53 |
| | 42h | EP2CONF | R/W | EP2 Configuration Register | 53 |
| | 43h | EP3CONF | R/W | EP3 Configuration Register | 53 |
| | 44h | EP4CONF | R/W | EP4 Configuration Register | 53 |
| | 45h | EP5CONF | R/W | EP5 Configuration Register | 53 |
| | | | | | |
| | | | | | |
| | 48h | EP0CONT | R/W | EP0 Control Register | 54 |
| | 49h | EP1CONT | R/W | EP1 Control Register | 55 |
| | 4Ah | EP2CONT | R/W | EP2 Control Register | 55 |
| | 4Bh | EP3CONT | R/W | EP3 Control Register | 55 |
| | 4Ch | EP4CONT | R/W | EP4 Control Register | 55 |
| | 4Dh | EP5CONT | R/W | EP5 Control Register | 55 |
| | | | | | |
| | | | | | |
| EP Support | 50h | EP0PLD | R/W | EP0 Payload Register | 56 |
| | 51h | EP1PLD | R/W | EP1 Payload Register | 56 |
| | 52h | EP2PLD | R/W | EP2 Payload Register | 56 |
| | 53h | EP3PLD | R/W | EP3 Payload Register | 57 |
| | 54h | EP4PLDLSB | R/W | EP4 Payload LSB Register | 57 |
| | 55h | EP5PLDLSB | R/W | EP5 Payload LSB Register | 57 |
| | | | | | |
| | | | | | |
| | 58h | EP0RXCNT | R | EP0 Receive Byte Counter Register | 58 |
| | 59h | EP1RXCNT | R | EP1 Receive Byte Counter Register | 58 |
| | 5Ah | EP2RXCNT | R | EP2 Receive Byte Counter Register | 58 |
| | 5Bh | EP3RXCNT | R | EP3 Receive Byte Counter Register | 59 |
| | 5Ch | EP4RXCNTLSB | R | EP4 Receive Byte Counter LSB Register | 59 |
| | 5Dh | EP5RXCNTLSB | R | EP5 Receive Byte Counter LSB Register | 59 |
| | | | | | |
| | | | | | |

**OKI** Semiconductor                                                   **ML60852A**

**Address and Names of Registers (3)**

| Category | Address | Symbol | R/W | Register name | Page |
|---|---|---|---|---|---|
| | 60h | EP0STAT | | EP0 Status Register | 60 |
| | 61h | EP1STAT | | EP1 Status Register | 63 |
| | 62h | EP2STAT | | EP2 Status Register | 63 |
| | 63h | EP3STAT | | EP3 Status Register | 64 |
| | 64h | EP4STAT | | EP4 Status Register | 63 |
| | 65h | EP5STAT | | EP5 Status Register | 63 |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| EP Support | 6Ch | EP4PLDMSB | R/W | EP4 Payload MSB Register | 65 |
| | 6Dh | EP5PLDMSB | R/W | EP5 Payload MSB Register | 65 |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | 74h | EP4RXCNTMSB | R | EP4 Receive Byte Counter MSB Register | 66 |
| | 75h | EP5RXCNTMSB | R | EP5 Receive Byte Counter MSB Register | 66 |
| | | | | | |
| | | | | | |
| | | | | | |
| | 3Eh | TXPKTCONT | R/W | Transmit Packet Ready Control Register | 67 |
| | | | | | |
| | | | | | |
| | | | | | |
| Option | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

**OKI** Semiconductor                                                                    **ML60852A**

## FUNCTIONS OF REGISTERS

### EP0 Transmit FIFO (EP0TXFIFO)

| Address | 0 x 70 |
|---|---|
| Type | Byte data |
| Access type | Write only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP0 Transmit data ||||||||

The EP0 transmit data can be written in by writing to the address 70h.
The transmit data to the host in the data stage during a control read transfer is stored in EP0TXFIFO. When the ML60852A issues an EP0 transmit packet ready interrupt request, the local MCU writes the transmit data to the address 70h.
It is possible to write the packet data successively by writing continuously.

The EP0TXFIFO is cleared under the following conditions.
    1. When an ACK signal is received from the host for the data transmission from EP0
    2. When a setup packet is received

### EP0 Receive FIFO (EP0RXFIFO)

| Address | 0 x 78 |
|---|---|
| Type | Byte data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP0 Receive data ||||||||

The receive data from the host computer in the data sate during a control Write transfer is stored in EP0RXFIFO. EP0 receive data can be read out by the local MCU through reading the address 78h when the ML60852A issues an EP0 receive packet ready interrupt request. It is possible to read successively the data in the packet by reading continuously.

The EP0RXFIFO is cleared under the following conditions:
    1. When the local MCU resets EP0 receive packet ready bit.
    2. When a setup packet is received.
    3. When the local MCU writes a "0" in the stall bit.

**EP1 FIFO (EP1FIFO)**

| Address | 0 x 79 |
|---|---|
| Type | Byte data |
| Access type | Write only or Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP1 Transmit data or EP1 receive data ||||||||

It is possible to specify the direction of transfer of EP1 by setting the EP1 configuration register EP1CONF. The FIFO address of EP1 is the same in both the transmit direction and the receive direction.

When EP1CONF (D7) = 0, EP1 is in the receive direction and EP1FIFO is in the read-only state.

When EP1CONF (D7) = 1, EP1 is in the transmit direction and EP1FIFO is in the write-only state.

When set for transmission, all bytes of EP1FIFO can be cleared by clearing EP1FIFO (writing a "1" into EP1CONT (D2)).

**EP2 FIFO (EP2FIFO)**

| Address | 0 x 7A |
|---|---|
| Type | Byte data |
| Access type | Write only or Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP2 Transmit data or EP2 receive data ||||||||

It is possible to specify the direction of transfer of EP2 by setting the EP2 configuration register EP2CONF. The FIFO address of EP2 is the same in both the transmit direction and the receive direction.

When EP2CONF (D7) = 0, EP2 is in the receive direction and EP2FIFO is in the read-only state.

When EP2CONF (D7) = 1, EP2 is in the transmit direction and EP2FIFO is in the write-only state.

When set for transmission, all bytes of EP2FIFO can be cleared by clearing EP2FIFO (writing a "1" into EP2CONT (D2)).

**EP3 FIFO (EP3FIFO)**

| Address | 0 x 7B |
|---|---|
| Type | Byte data |
| Access type | Write only<br>or<br>Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP3 Transmit data or EP3 receive data | | | | | | | |

It is possible to specify the direction of transfer of EP3 by setting the EP3 configuration register EP3CONF. The FIFO address of EP3 is the same in both the transmit direction and the receive direction.

When EP3CONF (D7) = 0, EP3 is in the receive direction and EP3FIFO is in the read-only state.

When EP3CONF (D7) = 1, EP3 is in the transmit direction and EP3FIFO is in the write-only state.

When set for transmission, all bytes of EP3FIFO can be cleared by clearing EP3FIFO (writing a "1" into EP3CONT (D2)).

**EP4 FIFO (EP4FIFO)**

| Address | 0 x 7C |
|---|---|
| Type | Byte data |
| Access type | Write only<br>or<br>Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP4 Transmit data or EP4 receive data | | | | | | | |

It is possible to specify the direction of transfer of EP4 by setting the EP4 configuration register EP4CONF. The FIFO address of EP4 is the same in both the transmit direction and the receive direction.

When EP4CONF (D7) = 0, EP4 is in the receive direction and EP4FIFO is in the read-only state.

When EP4CONF (D7) = 1, EP4 is in the transmit direction and EP4FIFO is in the write-only state.

When set for transmission, all bytes of EP4FIFO can be cleared by clearing EP4FIFO (writing a "1" into EP4CONT (D2)).

**EP5 FIFO (EP5FIFO)**

| Address | 0 x 7D |
|---|---|
| Type | Byte data |
| Access type | Write only<br>or<br>Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | EP5 Transmit data or EP5 receive data | | | | | | | |

In the ML60852A, by making a setting of the system control register, it is possible to select either the 5EP mode with the number of EPs being 5 or the 6EP mode with the number of EPs being 6. In the 5EP mode, only EP0 to EP4 are present and EP5 will not be present. In the EP6 mode, all end points EP0 to EP5 will be valid.

It is possible to specify the direction of transfer of EP5 by setting the EP5 configuration register EP5CONF. The FIFO address of EP5 is the same in both the transmit direction and the receive direction.

    When EP5CONF (D7) = 0, EP5 is in the receive direction and EP5FIFO is in the read-only state.
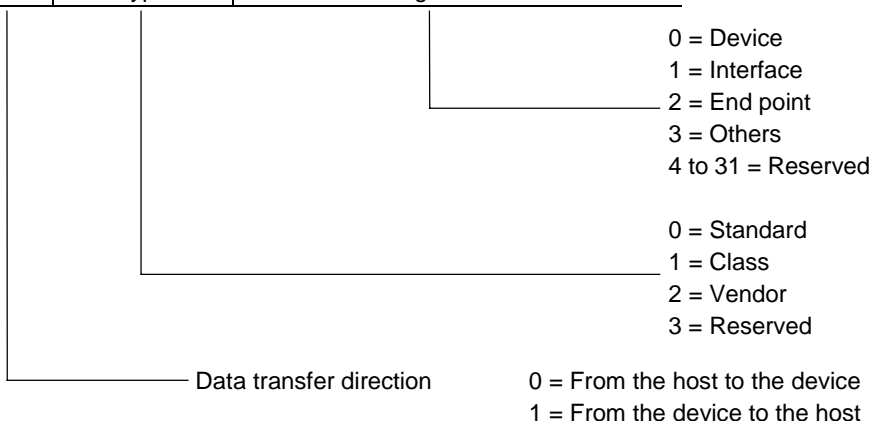
    When EP5CONF (D7) = 1, EP5 is in the transmit direction and EP5FIFO is in the write-only state.

When set for transmission, all bytes of EP5FIFO can be cleared by clearing EP5FIFO (writing a "1" into EP5CONT (D2)).

**bmRequestType Setup Register (bmRequestType)**

| Address | 0 x 00 |
|---|---|
| Type | Bit map |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | | Type | | Receiving side definitions | | | | |

0 = Device
1 = Interface
2 = End point
3 = Others
4 to 31 = Reserved

0 = Standard
1 = Class
2 = Vendor
3 = Reserved

Data transfer direction      0 = From the host to the device
1 = From the device to the host

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and is stored in the 8 setup registers including this register. The formats of these data items are defined in Section 9.3 of the USB Standards.

**bRequest Setup Register (bRequest)**

| Address | 0 x 01 |
|---|---|
| Type | Byte data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | x | x | x | x | x | x | x | x |
| After a bus reset | x | x | x | x | x | x | x | x |
| Definition | | | | Request code | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the second byte is stored in this register. The content of the request code is defined in Section 9.3 of the USB Standards and in related documents.

**wValueLSB Setup Register (wValueLSB)**

| | |
|---|---|
| Address | 0 x 02 |
| Type | 2-Byte data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | wValueLSB | | | | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the third byte is stored in this register.  This is the lower-order byte of the two-byte data.

**wValueMSB Setup Register (wValueMSB)**

| | |
|---|---|
| Address | 0 x 03 |
| Type | 2-Byte data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | wValueMSB | | | | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the fourth byte is stored in this register.  This is the higher-order byte of the two-byte data.

**wIndexLSB Setup Register (wIndexLSB)**

| Address | 0 x 04 |
|---|---|
| Type | 2-Byte data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | wIndexLSB | | | | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the fifth byte is stored in this register.  This is the lower-order byte of the two-byte data.

**wIndexMSB Setup Register (wIndexMSB)**

| Address | 0 x 05 |
|---|---|
| Type | 2-Byte data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | wIndexMSB | | | | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the sixth byte is stored in this register.  This is the higher-order byte of the two-byte data.

**wLengthLSB Setup Register (wLengthLSB)**

| Address | 0 x 06 |
|---|---|
| Type | 2-Byte data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | wLengthLS | | | | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the seventh byte is stored in this register. This is the lower-order byte of the two-byte data.

**wLengthMSB Setup Register (wLengthMSB)**

| Address | 0 x 07 |
|---|---|
| Type | 2-Byte data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | wLengthMSB | | | | | | | |

During the setup stage of control transfer based on a request from the host, the 8-byte setup data transmitted by the host is automatically received by the ML60852A and the eighth byte is stored in this register. This is the higher-order byte of the two-byte data.

**DMA0, 1 Control Registers (DMA0, 1CON)**

| Address | 0 x 10, 0 x 12 |
|---|---|
| Type | Bit map |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | | | | The previous value is retained | | | | |
| Definition | | | | | | | | |

DMA Enable
  0 = DMA Disabled
  1 = DMA Enabled

DMA Address mode
  0 = Single address mode
  1 = Dual address mode

DMA Byte count
  0 = The number of bytes is not inserted
  1 = The data of the number of bytes is inserted in the leading byte or leading word of the transfer data. (Note 1)

DMA Transfer data width
  0 = Byte width (8 bits)
  1 = Word width (16 bits) (Note 2)

DMA Transfer mode
  0 = Single transfer mode
  1 = Demand transfer mode

EP Specification
  Specifies the target EP for the DMA transfer
  0=EP1, 1=Ep2, 2=EP4, 3=EP5 (Note 3)

DMA Interrupting (Note 4)
  0=Normal operation
  1=The $\overline{\text{DREQ}}$ pin is de-asserted

Note 1: During the 16-bit mode, the higher order byte of the leading word will be 00h.
Note 2: The higher order byte and the lower order byte are allocated in the little-endian sequence. That is, the LSB corresponds to AD0 to AD7 and the MSB corresponds to D8 to D15.
  During the 16-bit mode and when the packet size is an odd number of bytes, the higher order byte of the last word will be 00h.
Note 3: When the EP specifications for the DMA channels 0 and 1 both have the same values, DREQ0, DREQ 1 and DACK0, DACK 1 will respectively be equivalent.
Note 4: The settings of all bits other than bit D7, that is, of bits D0 to D6 should be completed at the time of initialization (at the latest, before a token packet for EP1 to Ep5 arrives), and should not be altered thereafter. Write a "1" to D7 in order to temporarily stop DMA transfer in the middle. When the transfer is restarted by writing a "0" to D7, it is possible to restart the transfer from the byte (or word) next to the one at which the transfer was interrupted.

**DMA0, 1 Interval Registers (DMA0, 1INTVL)**

| Address | 0 x 11, 0 x 13 |
|---|---|
| Type | Byte data |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | The previous value is retained | | | | | | | |
| Definition | Interval time | | | | | | | |

This specifies the interval in the single DMA transfer mode, that is, the time duration after the end of DMA transfer of the previous byte (or the previous word) until DREQ is asserted again.  The time for 1 bit is 84ns (12 MHz, period of one-cycle).

    Interval time = (DREQ enable time) + 84xn (ns)

See the description of the DMA Transfer Timings (1), (2), (5), and (6) for the DREQ enable time.

**Device Address Register (DVCADR)**

| | |
|---|---|
| Address | 0 x 20 |
| Type | 7-bit data |
| Access type | Read/Write |

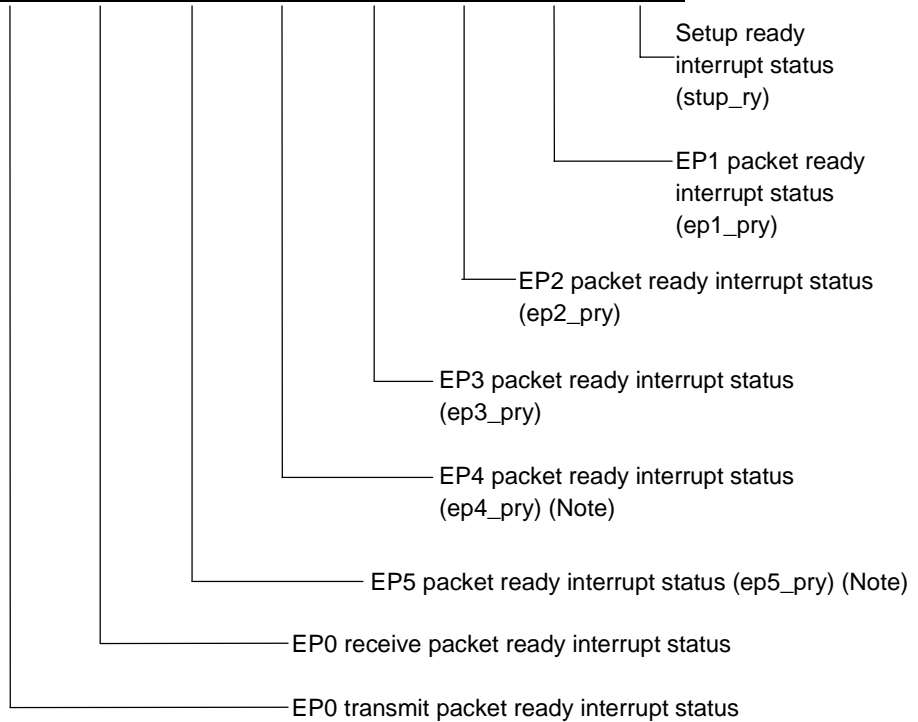| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | | | | | | | | |
| Definition | | Device address (R/W) | | | | | | |

The device address given by a SET_ADDRESS request from the host is written in this register by the local MCU. Thereafter, the ML60852A judges the specified address in the token from the host, and this device will process only the token packets sent to this device address.

Bit D7 is fixed at "0", and even if a "1" is written, it will be ignored.

**Interrupt Status Register 1 (INTSTAT1)**

| Address | 0 x 21 |
|---|---|
| Type | Bit map |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | X | x | x | x | x | x | x | 0 |
| Definition | | | | | | | | |

Setup ready interrupt status (stup_ry)

EP1 packet ready interrupt status (ep1_pry)

EP2 packet ready interrupt status (ep2_pry)

EP3 packet ready interrupt status (ep3_pry)

EP4 packet ready interrupt status (ep4_pry) (Note)

EP5 packet ready interrupt status (ep5_pry) (Note)

EP0 receive packet ready interrupt status

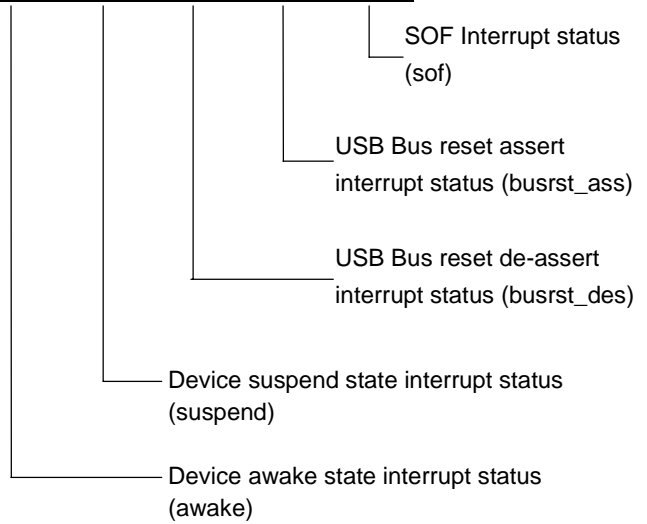EP0 transmit packet ready interrupt status

Note:    When isochronous transfer has been set in the EP4 or EP5 configuration register, the EP4 or EP5 packet ready interrupt status will always be fixed at "0".

**OKI** Semiconductor                                                                **ML60852A**

**Interrupt Status Register 2 (INTSTAT2)**

| Address | 0 x 22 |
|---|---|
| Type | Bit map |
| Access type | Read/Reset |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | note | | 0 |
| Definition | 0 | 0 | 0 | | | | | |

note ; The previous value is retained

SOF Interrupt status (sof)

USB Bus reset assert interrupt status (busrst_ass)

USB Bus reset de-assert interrupt status (busrst_des)

Device suspend state interrupt status (suspend)

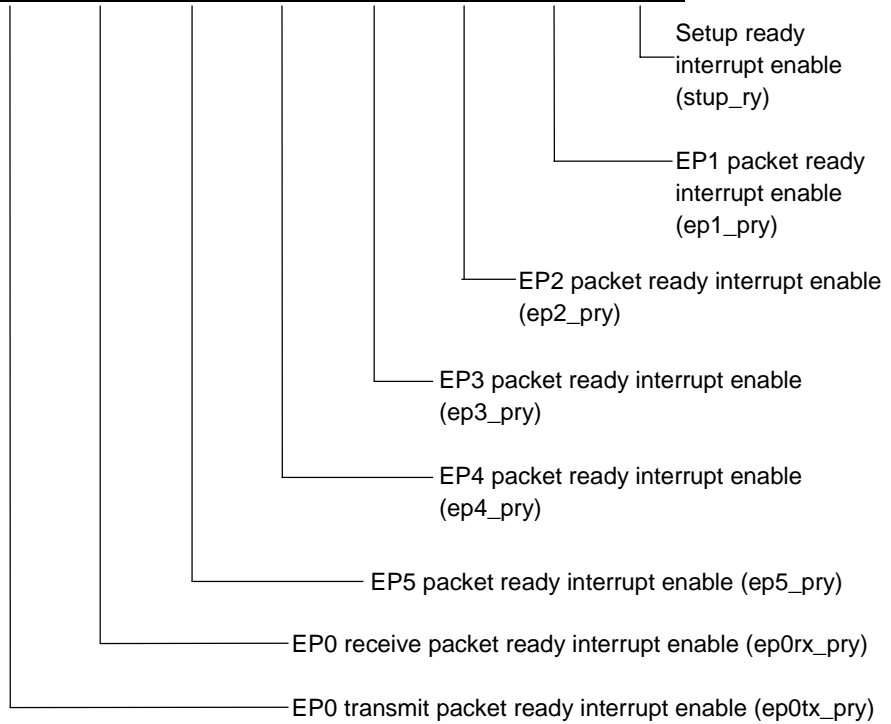Device awake state interrupt status (awake)

The status bit becomes "1" when the corresponding interrupt is generated.
The status is cleared when a "1" is written in that status bit itself.
(See Section (8) "Interrupt" for Functional Descriptions.)

**Interrupt Enable Register 1 (INTENBL1)**

| Address | 0 x 24 |
|---|---|
| Type | Bit map |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| After a bus reset | The previous value is retained | | | | | | | |
| Definition | | | | | | | | |

Setup ready interrupt enable (stup_ry)

EP1 packet ready interrupt enable (ep1_pry)

EP2 packet ready interrupt enable (ep2_pry)

EP3 packet ready interrupt enable (ep3_pry)

EP4 packet ready interrupt enable (ep4_pry)

EP5 packet ready interrupt enable (ep5_pry)

EP0 receive packet ready interrupt enable (ep0rx_pry)

EP0 transmit packet ready interrupt enable (ep0tx_pry)

**Interrupt Enable Register 2 (INTENBL2)**

| Address | 0 x 25 |
|---|---|
| Type | Bit map |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | The previous value is retained | | | | | | | |
| Definition | 0 | 0 | 0 | | | | | |

SOF Interrupt enable (sof)

USB Bus reset assert interrupt enable (busrst_ass)

USB Bus reset de-assert interrupt enable (busrst_des)

Device suspend state interrupt enable (suspend)

Device awake state interrupt enable (awake)
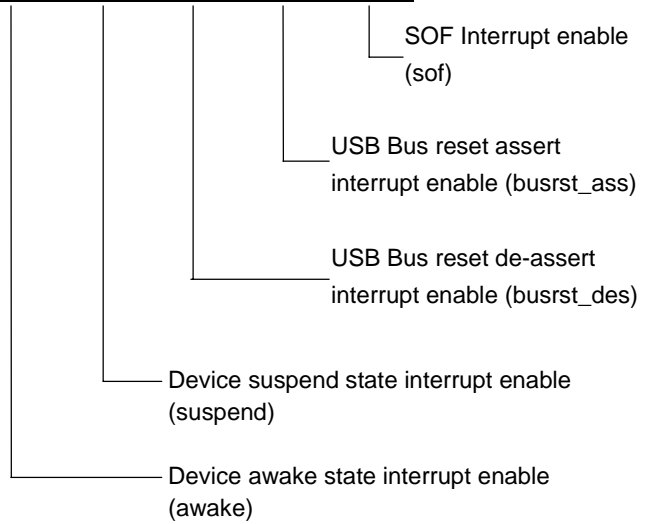
**Frame Number LSB Register (FRAMELSB)**

| Address | 0 x 2D |
|---|---|
| Type | 11-Bit data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | Frame number LSB | | | | | | | |

This is valid when containing an end point in the isochronous transfer mode. When a start of frame (SOF) packet is transmitted by the host, the ML60852A automatically writes into the FRAMELSB and FRAMEMSB registers.

**Frame Number MSB (FRAMEMSB)**

| Address | 0 x 2E |
|---|---|
| Type | 11-Bit data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | | | | | | Frame number MSB | | |

This is valid when containing an end point in the isochronous transfer mode. When a start of frame (SOF) packet is transmitted by the host, the ML60852A automatically writes into the FRAMELSB and FRAMEMSB registers.

**System Control Register (SYSCON)**

| Address | 0 x 2F |
|---|---|
| Type | Bit map |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | | The previous value is retained | | | | | | |
| Definition | 0 | | | | | | | |

Software reset
(R/W) (sereset)

Power-down mode
(R/W) (power)
0 = Power saving is not
done in the suspend
mode.
1 = Power saving is done
in the suspend mode.

EP Mode (R/W) (ep_mod)
0 = EP0 to EP5
1 = EP0 to EP4

Pull-up control (R/W) (plup)

Remote wakeup (R/W)

PLL Enable (pll_enable)

PLL Multiplication factor selection (fsel)
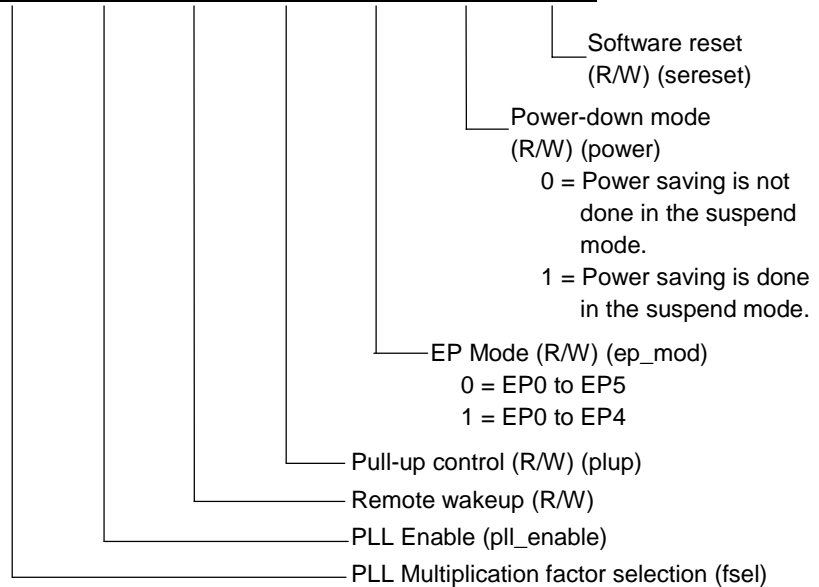
Software reset:       Write-only bit. Even when this bit is read out, it will be fixed at "0". A system reset is
                      executed when a "1" is written in this bit. This is functionally equivalent to a hardware reset.
                      However, this bit itself will always remain "0".

Power-down mode:      Read/Write bit. When this bit is "0", the oscillations will not be stopped even during the
                      suspend mode. When this bit is made "1", the oscillations will be stopped in the suspend
                      mode and the device goes into the power save mode.

EP mode:              Read/Write bit. The 6EP mode is selected when this bit is 0 and the 5EP mode is selected
                      when this bit is "1".

Pull-up control:      The content of this bit becomes valid when the ADSEL pin is "L". When the ADSEL pin is
                      "L", the internal switch becomes ON and the ALE pin is pulled up to the Vcc level if this bit is
                      "1". On the other hand, if the ADSEL pin is "L" and also this bit is "0", the internal switch is
                      made OFF and the ALE pin goes into the high impedance state.

Remote wakeup:       A remote wakeup is executed when a "1" is written in this bit. However, this bit itself will
                      always remain "0".
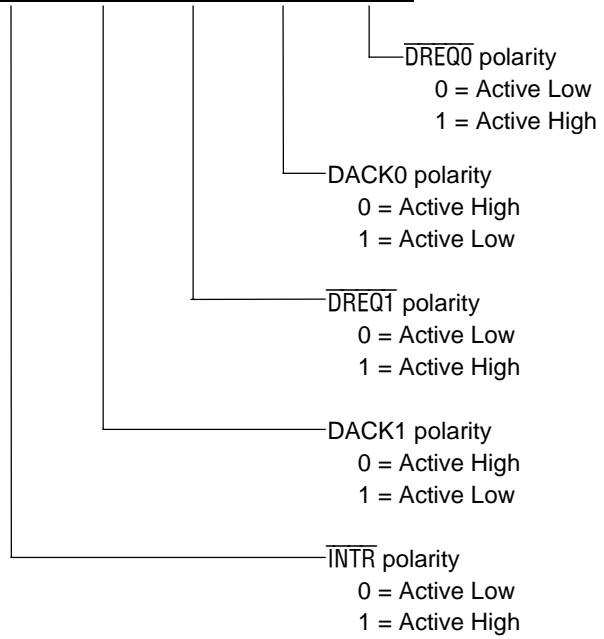
PLL enable:          This bit is for enabling the internal PLL. The internal PLL cannot be used when this bit is "0".
                      The signal input to the XIN pin becomes the source oscillations for the internal circuits.
                      The internal PLL can be used when this bit is "1".

PLL multiplication
factor selection:     The frequency multiplication factor is 4 when this bit is "0", and will be 8 when this bit is
                      "1".
                      (The source oscillator frequency will be 12 MHz in the 4x mode and will be 6 MHz in the 8x
                      mode.)

**Polarity Selection Register (POLSEL)**

| Address | 0 x 30 |
|---|---|
| Type | Bit map |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | The previous value is retained | | | | | | | |
| Definition | 0 | 0 | 0 | | | | | |

$\overline{\text{DREQ0}}$ polarity
0 = Active Low
1 = Active High

DACK0 polarity
0 = Active High
1 = Active Low

$\overline{\text{DREQ1}}$ polarity
0 = Active Low
1 = Active High

DACK1 polarity
0 = Active High
1 = Active Low

$\overline{\text{INTR}}$ polarity
0 = Active Low
1 = Active High

**EP0 Configuration Register (EP0CONF)**

| Address | 0 x 40 |
|---|---|
| Type | Bit map |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 0 |  | 0 | 0 | 0 | 0 |

Transfer type (read only)
00 = Control transfer

Configuration bit (read only)
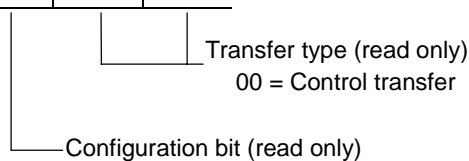
Transfer type:    Although these bits indicate the type of transfer, since EP0 has been fixed for control transfer in the ML60852A, this value is always fixed at 00h.  The local MCU cannot write into these bits.

Configuration bit:    The configuration bit of EP0 becomes "1" after a USB bus reset.
When this bit is "1", the data transmitted by the host to the end point can be received and also data can be transmitted from the end point to the host.  When this bit is "0", this LSI will not respond to any transaction targeting this end point.  This bit cannot be written in by the local MCU.

**EP1, 2, 3, 4, 5 Configuration Registers (EP1, 2, 3, 4, 5CONF)**

| Address | 0 x 41 to 45 |
|---|---|
| Type | Bit map |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Definition |  | 0 | 0 |  | 0 | 0 |  |  |

Transfer type
  10 = Bulk transfer
  11 = Interrupt transfer
  01 = Isochronous
        transfer

Configuration bit

Transfer direction
  0=Reception,
  1=transmission
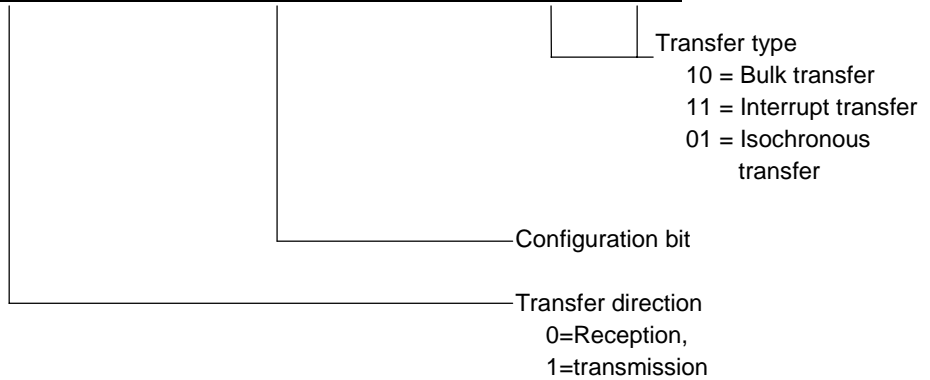
Transfer type:     These bits indicate the type of transfer.  Only EP4 and EP5 can be set to isochronous transfer.
Configuration bit:  When a Set Configuration request to make that EP active is received from the host, make sure
                  that the local MCU writes a "1" into this bit during the status stage in the control
                  transfermode.
                  Data transmission and reception can be made between the host and the EP when this bit is "1".
                  When this bit is "0", this LSI will not respond to the transactions targeted at that EP.
Transfer direction:  The local MCU sets the direction of data transfer using this bit.

**EP0 Control Register (EP0CONT)**

| Address | 0 x 48 |
| --- | --- |
| Type | Bit map |
| Access type | Read/Write |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| After a hardware reset | 0 | 0 | 0 | x | 0 | 0 | x | 0 |
| After a bus reset | 0 | 0 | 0 | x | 0 | 0 | x | 0 |
| Definition | 0 | 0 | 0 |  | 0 | 0 |  |  |
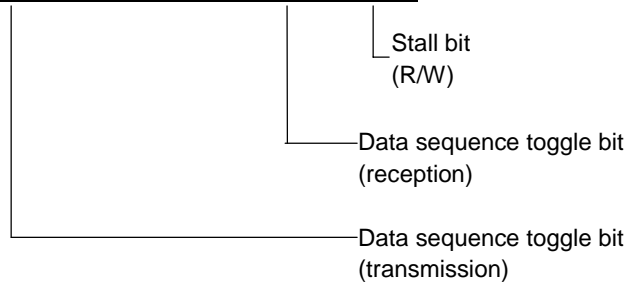
Stall bit
(R/W)

Data sequence toggle bit
(reception)

Data sequence toggle bit
(transmission)

Stall bit: During EP0 reception (data stage of a control write transfer), if a packet with a number of bytes exceeding the maximum packet size specified in EP0PLD is received (or if the EOP packet is missing), the ML60852A automatically sets this bit to "1". In order to conform to the Protocol Stall of USB Rev. 1.1, <u>this bit is reset automatically to "0" when a setup packet is received.</u>

Data sequence toggle bits: The ML60852A automatically carries out synchronization using the data sequence toggle mechanism. Further, any write operation to these bits (D4 and D1) will be invalid.

**EP1, 2, 3, 4, 5 Control Registers (EP1, 2, 3, 4, 5CONT)**

| Address | 0 x 49 to 4D |
|---|---|
| Type | Bit map |
| Access type | See below |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 0 | 0 | | | | |

D0 — Stall bit (R/W)

D1 — Data sequence toggle bit (R/Reset)

D2 — FIFO Clear (R/Reset)

D3 — Rate feedback (R/W)
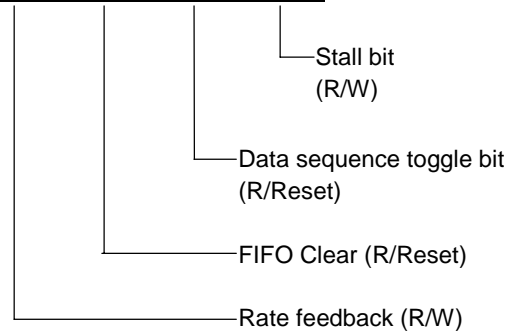
| | |
|---|---|
| Stall bit: | During EP0 reception (data stage of a control write transfer), if a packet with a number of bytes exceeding the maximum packet size specified in EP0PLD is received (or if the EOP packet is missing), the ML60852A automatically sets this bit to "1". |
| Data sequence toggle bit: | A reset will be made when a "1" is written in this bit. At the time of initializing the EP, reset the toggle bit of the data packet by writing a "1" to this bit, and specify PID of DATA0 (this bit too will become "0"). Thereafter, the synchronization operation using the data sequence toggle mechanism will be made automatically. |
| FIFO Clear: | The EP will be valid only when it has been set for transmission by the EP control register. When a "1" is written in this bit, the transmit FIFO of that EP will be cleared. (However, this bit itself will remain "0".) |
| Rate feedback: | This bit is valid only in the case of EP3. This bit will be fixed at "0" in all other EPs. |

**EP0 Payload Register (EP0PLD)**

| Address | 0 x 50 |
| --- | --- |
| Type | 6-Bit data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| After a hardware reset | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 1 | 0 |  |  |  |  |

Maximum packet size

Maximum packet size:  Since the FIFO of EP0 in the ML60852A has a size of 32 bytes, write 20h into the byte bMaxPacketSize0 of the device descriptor.  The maximum packet size is fixed at 32 bytes in this EP0PLD register.  When a packet with more than 32 bytes is received, the stall bit in the EP0 status register is asserted and the stall handshake is returned to the host.

**EP1, 2 Payload Registers (EP1, 2PLD)**

| Address | 0 x 51, 52 |
| --- | --- |
| Type | 7-Bit data |
| Access type | Read/Write |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | Maximum packet size (R/W) | | | | | | |

Maximum packet size:  Make the local MCU write in this register the value of the descriptor wMaxPacketSize of the end point selected by the Set_Configuration request from the host.  The size of all packets other than a short packet is specified here in units of a byte.

When the EP has been assigned for reception, if a data packet with a number of bytes exceeding the maximum packet size specified in this register is received, the receive packet ready status bit is not asserted, but the stall bit is set in EOP and the stall handshake is returned to the host.

On the other hand, when the EP has been assigned for transmission, the transmit packet ready bit is set automatically when writing by the DMA controller of data with the maximum packet size specified in this register is completed.  The content of this register is ignored during non-DMA transmission of data.

**EP3 Payload Register (EP3PLD)**

| Address | 0 x 53 |
|---|---|
| Type | 6-Bit data |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | | | | | | |

Maximum packet size (R/W)

Maximum packet size:  Make the local MCU write in this register the value of the descriptor wMaxPacketSize of the end point selected by the Set_Configuration request from the host. The size of all packets other than a short packet is specified here in units of a byte. Set 20h (32 bytes) or less because the FIFO size is 32 bytes.

When EP3 has been assigned for reception, if a data packet with a number of bytes exceeding the maximum packet size specified in this register is received, the receive packet ready status bit is not asserted, but the stall bit is set in EOP and the stall handshake is returned to the host.

There is no need to use this register when EP3 has been assigned for transmission.

**EP4, 5 Payload LSB Registers (EP4, 5PLDLSB)**

| Address | 0 x 54, 55 |
|---|---|
| Type | 10-Bit or 9-bit data |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | Maximum packet size LSB (R/W) | | | | | | | |

Maximum packet size LSB:  Make the local MCU write in this register the value of the descriptor wMaxPacketSize of the end point selected by the Set_Configuration request from the host. The lower 8 bits should be stored in this register and the higher-order bytes should be written in the EP4,5 payload registers MSB. The maximum packet size is specified in units of a byte.

When the EP has been assigned for reception, if a data packet with a number of bytes exceeding the maximum packet size specified in these registers is received, the receive packet ready status bit is not asserted, but the stall bit is set in EOP and the stall handshake is returned to the host.

On the other hand, when the EP has been assigned for transmission, the transmit packet ready bit is set automatically when writing by the DMA controller of data with the maximum packet size specified in this register is completed.

**EP0 Receive Byte Counter Register (EP0RXCNT)**

| Address | 0 x 58 |
| --- | --- |
| Type | 6-Bit data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | Receive byte count (R) | | | | | |

The ML60852A automatically counts the number of bytes in the packet being received. Although the counting is done only up to the number of bytes equal to the maximum packet size specified in the payload register in the case of a full packet, the count will be less than that size in the case of a short packet. The local MCU refers to this value and reads out the data of one packet from the EP0 Receive FIFO.

The EP0RXCNT register is cleared under the following conditions.
    1. When the local MCU resets the EP receive packet ready bit.
    2. When a setup packet is received.
    3. When the local MCU writes a "0" into the stall bit.

**EP1, 2 Receive Byte Counter Registers (EP1, 2RXCNT)**

| Address | 0 x 59, 5A |
| --- | --- |
| Type | 7-Bit data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | Receive byte count (R) | | | | | | |

The ML60852A automatically counts the number of bytes in the packet being received. Although the counting is done only up to the number of bytes equal to the maximum packet size specified in the payload register in the case of a full packet, the count will be less than that size in the case of a short packet. The local MCU refers to this value and reads out the data of one packet from the EP1/2 Receive FIFO.

This register will be invalid when the transfer direction of the EP is set for transmission.

The EP1,2RXCNT register is cleared under the following conditions.
    1. When an OUT token is received for the EP.
    2. When the local MCU resets the EP receive packet ready bit.
    3. When the local MCU writes a "0" into the stall bit.

**EP3 Receive Byte Counter Register (EP3RXCNT)**

| Address | 0 x 5B |
|---|---|
| Type | 6-Bit data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | Receive byte count (R) | | | | | |

The ML60852A automatically counts the number of bytes in the packet being received. Although the counting is done only up to the number of bytes equal to the maximum packet size specified in the payload register in the case of a full packet, the count will be less than that size in the case of a short packet. The local MCU refers to this value and reads out the data of one packet from the EP3 Receive FIFO.

This register will be invalid when the transfer direction of the EP is set for transmission.

The EP3RXCNT register is cleared under the following conditions.

    1. When an OUT token is received for EP3.
    2. When the local MCU resets the EP receive packet ready bit.
    3. When the local MCU writes a "0" into the stall bit.

**EP4, 5 Receive Byte LSB Counter Registers (EP4, 5RXCNTLSB)**

| Address | 0 x 5C, 5D |
|---|---|
| Type | 10-Bit or 9-bit data |
| Access type | Read only |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | Receive byte count LSB (R) | | | | | | | |

The ML60852A automatically counts the number of bytes in the packet being received. Although the counting is done only up to the number of bytes equal to the maximum packet size specified in the payload register in the case of a full packet, the count will be less than that size in the case of a short packet. The local MCU refers to this value and reads out the data of one packet from the EP4/5 Receive FIFO. The lower 8 bits of the receive byte count are stored in this register and the higher order bits are stored in the EP receive byte counter MSB.

This register will be invalid when the transfer direction of the EP is set for transmission.
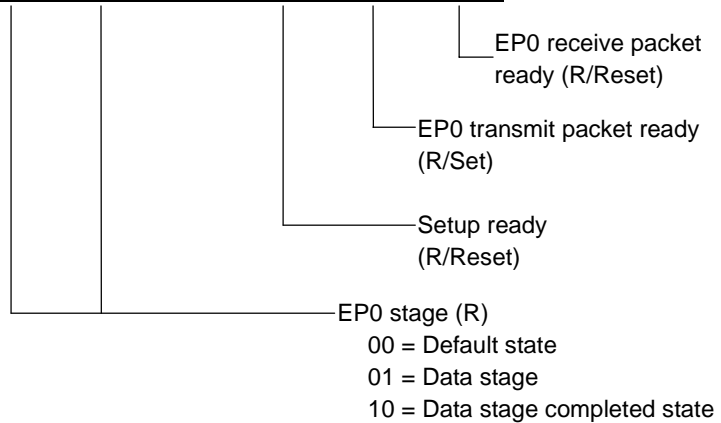
The EP4,5RXCNT register is cleared under the following conditions.

    1. When an OUT token is received for the EP.
    2. When the local MCU resets the EP receive packet ready bit.
    3. When the local MCU writes a "0" into the stall bit.

**EP0 Status Register (EP0STAT)**

| Address | 0 x 60 |
|---|---|
| Type | Bit map |
| Access type | See below |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 |  |  | 0 |  |  |  |

EP0 receive packet
ready (R/Reset)

EP0 transmit packet ready
(R/Set)

Setup ready
(R/Reset)

EP0 stage (R)
00 = Default state
01 = Data stage
10 = Data stage completed state

Please note the R/Reset and R/Set notation used above. R/Reset means: the bit field can be read by the local MCU/and it is Reset (to '0') when a "1" is written to it. The R/Set means: the bit field can be read by the local MCU/and it is Set (to '1') when a 1 is written to it.

Setup ready:     This bit is set automatically when a setup packet normally arrives in the 8-byte setup register, and the EP0 Receive FIFO is locked.

If INTENBL1(0) is asserted, the $\overline{\text{INTR}}$ pin is also asserted automatically when this bit is set. The local MCU should write a "1" into this bit after reading out the 8-byte setup data.

When this is performed, the setup ready bit is reset and the $\overline{\text{INTR}}$ pin also is de-asserted. During a control write transfer, the packet ready bit of EP0 is reset simultaneously and the lock condition is released, and it becomes possible to receive packets by EP0 during the data stage. The register will not change even if a "0" is written in this bit.

EP0 transmit packet
ready bit (D1):     The local MCU can read this bit. Writing when D1=1 sets this bit to "1". The asserting and de-asserting conditions are described below.

| Bit name | Asserting condition | Operation when asserted |
|---|---|---|
| EP0 transmit packet ready (D1) | When the local MCU has set this bit | Data can be transmitted from EP0. |

| Bit name | De-asserting condition | Operation when de-asserted |
|---|---|---|
| EP0 transmit packet ready (D1) | 1. When an ACK is received from the host for data transmission<br>2. When a setup packet is received | EP0 is locked. That is, an NAK is automatically returned when an IN token is sent from the host. |

EP0 receive packet ready bit (D0):          The local MCU can read this bit. Writing a "1" when D0=1 resets this bit to "0". The asserting and de-asserting conditions are described below.

| Bit name | Asserting condition | Operation when asserted |
|---|---|---|
| EP0 receive packet ready (D0) | 1. When data is received by EP0 and stored in FIFO<br><br>2. When a setup packet is received during control Read transfer or control Write transfer | EP0 is locked (an NAK is automatically returned when a data packet is received from the host). Data can be read from EP0RXFIFO by local MCU. |

| Bit name | De-asserting condition | Operation when de-asserted |
|---|---|---|
| EP0 receive packet ready (D0) | 1. When the local MCU has reset this bit ( a "1" is written in this bit)<br><br>2. When the local MCU resets the setup ready bit during control Write transfer | EP0 can receive data. |

<u>EP0 Stage (D5, D4):</u> These bits indicate the stage transition during control transfer.
The flowchart of the stage transition is shown below.
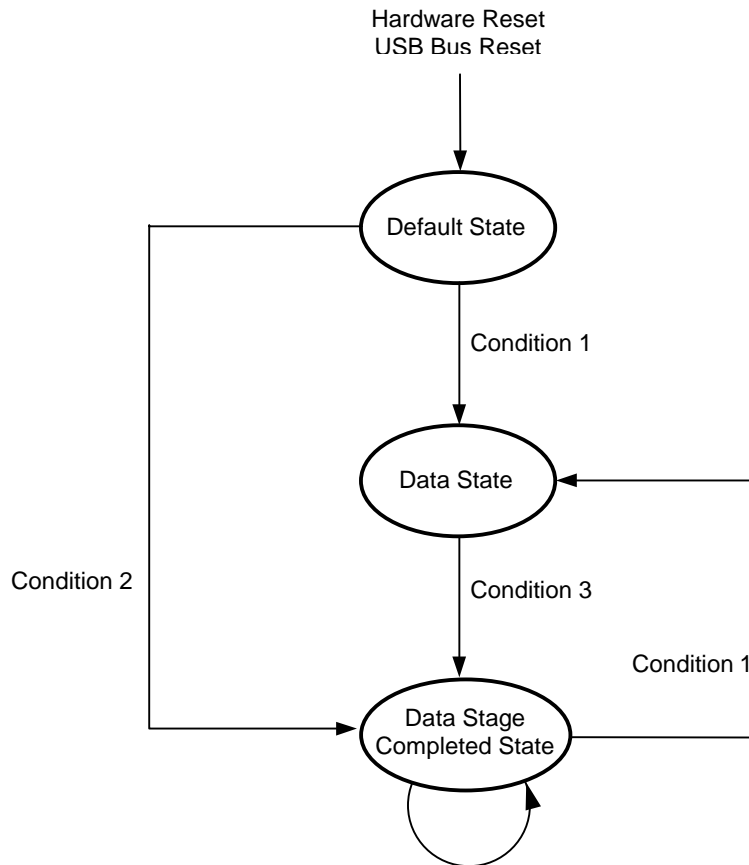


Condition 1:        Reception of a setup packet of control READ transfer or control WRITE transfer.
Condition 2:        Reception of a setup packet of control transfer without data.
Condition 3:        Reception of a token (IN/OUT) of a direction opposite to the data flow in the data
stage.

**EP1, 2, 4, 5 Status Registers (EP1, 2, 4, 5STAT)**

| Address | 0 x 61, 62, 64, 65 |
|---|---|
| Type | Bit map |
| Access type | See below |

|  | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 0 | 0 | 0 | 0 |  |  |

EP Receive packet
ready (Read/Reset)

EP Transmit packet ready
(Read/Set)

This register is valid only when the corresponding EP has been set for bulk or interrupt transfer.

EP1,2,4,5 Receive packet ready bit (D0): This bit can be read by the local MCU. Also, this bit can be made "0" by writing a "1" into bit D0. The asserting and de-asserting conditions of this bit are as given below. The FIFOs of EP1, EP2, EP4, and EP5 have a 2-layer structure and also there are independent packet ready bits for layer A and layer B. The switching between these two layers is done automatically by the ML60852A.

| Bit name | Asserting condition | Operation when asserted |
|---|---|---|
| EPn Receive packet ready (D0) | When an error-free packet is received in either layer A or layer B. | The local MCU can read the EP1 Receive FIFO. EP1 is locked in the condition in which data packets have been received by both layer A and layer B. |

| Bit name | De-asserting condition | Operation when de-asserted |
|---|---|---|
| EPn Receive packet ready (D0) | When the local MCU has reset (written a "1" in) the bits of both layer A and layer B. | Reception can be made by EP1 when the bit of either layer A or layer B has been reset. |

EP1,2,4,5 Transmit packet ready bit (D1): This bit can be read by the local MCU. Also, this bit can be made "1" by writing a "1" into bit D1. The asserting and de-asserting conditions of this bit are as given below. The FIFO of EP1 has a 2-layer structure and also there are independent packet ready bits for layer A and layer B. The switching between these two layers is done automatically by the ML60852A.

| Bit name | Asserting condition | Operation when asserted |
|---|---|---|
| EPn Transmit packet ready (D1) | (1) When the local MCU has set the bits of both layer A and layer B. (2) When the local MCU has set the bits of either layer A or layer B. | Transmission can be made from EP1 when either layer A or layer B has been asserted. |

| Bit name | De-asserting condition | Operation when de-asserted |
|---|---|---|
| EPn Transmit packet ready (D1) | When an ACK message is received from the host for the data transmission to either layer A or layer B. | EP1 is locked when transmit data has not been prepared for both layer A and layer B. |

**EP3 Status Register (EP3STAT)**

| Address | 0 x 63 |
| --- | --- |
| Type | Bit map |
| Access type | See below |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 0 | 0 | 0 | 0 | | |

EP3 Receive packet
ready (Read/Reset)

EP3 Transmit packet ready
(Read/Set)

This register is valid only when EP3 has been set for bulk or interrupt transfer.

EP3 Receive packet ready bit (D0): This bit can be read by the local MCU.  Also, this bit can be made "0" by writing a "1" into bit D0.  The asserting and de-asserting conditions of this bit are as given below.

| Bit name | Asserting condition | Operation when asserted |
| --- | --- | --- |
| EP3 Receive packet ready (D0) | When an error-free packet is received. | EP3 is locked. |

| Bit name | De-asserting condition | Operation when de-asserted |
| --- | --- | --- |
| EP3 Receive packet ready (D0) | When the local MCU has reset (written a "1" in) this bit. | Reception can be made by EP3. |

EP3 Transmit packet ready bit (D1): This bit can be read by the local MCU.  Also, this bit can be made "1" by writing a "1" into this bit.  The asserting and de-asserting conditions of this bit are as given below.

| Bit name | Asserting condition | Operation when asserted |
| --- | --- | --- |
| EP3 Transmit packet ready (D1) | When the local MCU has set (write '1') this bit. | When an IN token is received from the host, ML60852A will automatically transmit its EP3FIFO transmit data on to the USB bus. |

| Bit name | De-asserting condition | Operation when de-asserted |
| --- | --- | --- |
| EP3 Transmit packet ready (D1) | After a hardware reset, bus reset, or when an ACK is received in response to succussful data transmission. | Data can be written to the Tx FIFO of EP3. |

**OKI** Semiconductor                                                      **ML60852A**

**EP4, 5 Payload MSB Registers (EP4, 5PLDMSB)**

| | |
|---|---|
| Address | 0 x 6C, 6D |
| Type | 10-Bit or 9-bit data |
| Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 0 | 0 | 0 | 0 | | |

EP Payload MSB

The higher-order 6 bits of EP4PLDMSB are fixed at "0" and the higher-order 7 bits of EP5PLDMSB are fixed at "0".

**EP4, 5 Receive Byte MSB Counter Registers (EP4, 5RXCNTMSB)**

| | | |
|---|---|---|
| Address | 0 x 74, 75 |
| Type | 10-Bit or 9-bit data |
| Access type | Read only |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Definition | 0 | 0 | 0 | 0 | 0 | 0 | | |

EP Receive byte count MSB
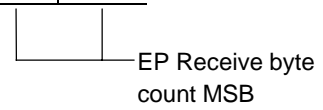
The higher-order 6 bits of EP4RXCNTMSB are fixed at "0" and the higher-order 7 bits of EP5RXCNTMSB are fixed at "0".

**Transmit Packet Ready Control Register (TXPKTCONT)**

| | Address | 0 x 3E |
|---|---|---|
| | Type | Bit map |
| | Access type | Read/Write |

| | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|
| After a hardware reset | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| After a bus reset | The previous value is retained | | | | | | | |
| Definition | 0 | 0 | | | 0 | | | 0 |

Write Disable

EP1 Transmit Packet Ready Control bit

EP2 Transmit Packet Ready Control bit

Write Disable

EP4 Transmit Packet Ready Control bit

EP5 Transmit Packet Ready Control bit

Write Disable

Write Disable
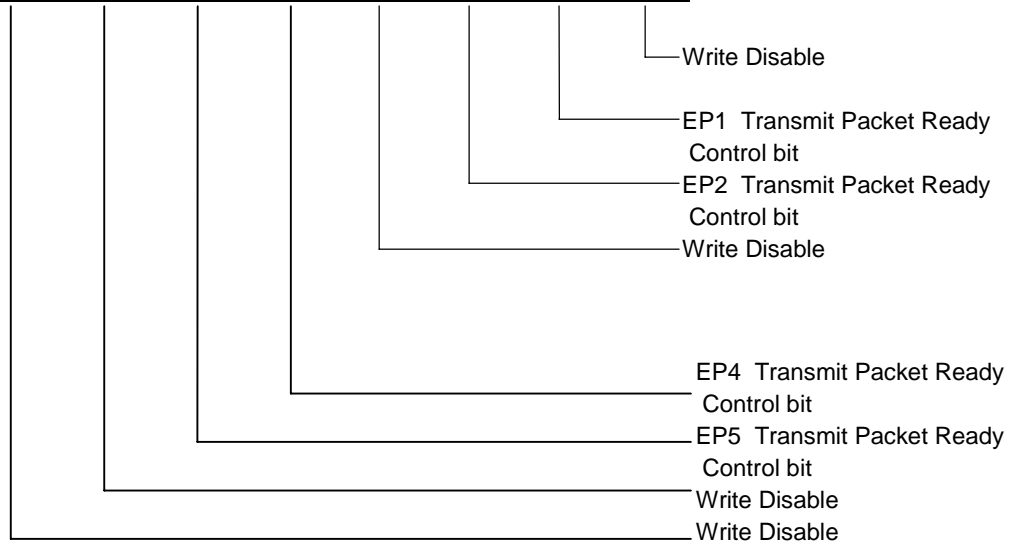
○ EPn (n=1,2,4,5) transmit packet ready control bit

This register is mainly used for controlling the double-layered FIFO structure operation of ML60852A.
Writing a "1" into this bit controls the assert and de-assert conditions of the EP1,EP2,EP4,EP5 transmit packet ready interrupts and operations of bits corresponding to EPn (n=1,2,4,5) status register.
For EPn (n=1,2,4,5) status register, see "INTERNAL REGISTERS".
This register controls the above conditions and bit operations only during transmission (from ML60852A to HOST) and does not control them during reception (from HOST to ML60852A).

**OKI** Semiconductor

• Relationship between the transmit FIFO status, register status of each EP and INTR signal

Note: The following table assumes that the transmit packet ready status bit (D1 of EPnSTAT) is set (to '1') immediately after writing transmit data to each layer of the EPnFIFO.

| Trnsmit FIFO status | Register status of each EP<br>Packet ready interrupt enable bit = 1<br>Transmit packet ready control bit = 0 | Register status of each EP<br>Packet ready interrupt enable bit = 1<br>Transmit packet ready control bit = 1 |
|---|---|---|
| Both layers are not empty | INTR signal is de-asserted | INTR signal is de-asserted |
| One layer is empty and<br>Other layer is not empty | INTR signal is asserted | INTR signal is de-saaerted |
| Both layers are empty | INTR signal is asserted | INTR signal is asserted |

\* Definition of transmit FIFO "empty"
  When one of the following two conditions is satisfied, transmit FIFO is empty.

(1) In the early transmit stage just after the corresponding endpoint configuration, the local MCU has not written a "1" into the D1 bit of the EPnSTAT register yet.
(2) The transmit data written by the local MCU is transferred to HOST and an ACK signal is returned from HOST.

\* When the packet ready interrupt enable bit of the corresponding EP is de-asserted, the INTR signal is de-asserted irrespective of the status of transmit FIFO.
  (When other interrupt factor is generated, the INTR signal is asserted.)
  The interrupt factor of the transmit packet ready interrupt can be changed by using the transmit packet ready control bit.
  For instance, if one layer of transmit FIFO is empty and other layer is not empty,

● The asserted INTR signal is de-asserted by changing the transmit packet ready control bit to "1" from "0".

● The de-asserted INTR signal is asserted by changing the transmit packet ready control bit to "0" from "1".

## TIMING DIAGRAM

**READ Timing (1)**
**(Address Separate, ADSEL =0)**

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Address Setup Time | $t_1$ | | 0 | — | ns | |
| Address ($\overline{CS}$) Hold Time | $t_2$ | | 0 | — | ns | (2) |
| Read Data Delay Time | $t_3$ | Load 20 pF | — | 46 | ns | (1) |
| Read Data Hold Time | $t_4$ | | 0 | 21 | ns | |
| Recovery Time | $t_5$ | FIFO READ | 63 | — | ns | (3) |
| FIFO Access Time | $t_6$ | FIFO READ | 63 | — | ns | (3) |

Notes: (1)  $t_3$ is defined depending upon $\overline{CS}$ or $\overline{RD}$ which becomes active last.
       (2)  $t_2$ is defined depending upon $\overline{CS}$ or $\overline{RD}$ which becomes active first.
       (3)  3-clock time of oscillation clock (clock period:21 ns). It is required for increment of FIFO.
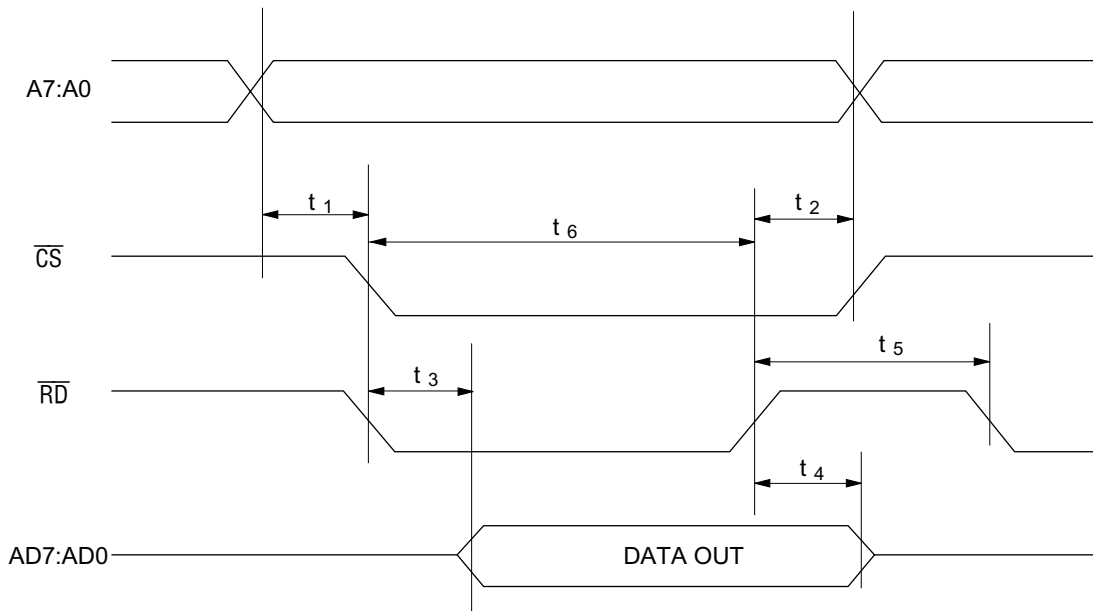
**READ Timing (2)**
**(Address/Data Multiplex, ADSEL =1)**

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Address ($\overline{CS}$) Setup Time | $t_1$ | | 10 | — | ns | |
| Address ($\overline{CS}$) Hold Time | $t_2$ | | 0 | — | ns | |
| Read Data Delay Time | $t_3$ | Load 20 pF | — | 46 | ns | |
| Read Data Hold Time | $t_4$ | | 0 | 21 | ns | |
| Recovery Time | $t_5$ | FIFO READ | 63 | — | ns | (1) |
| FIFO Access Time | $t_6$ | FIFO READ | 63 | — | ns | (1) |

Notes: (1) 3-clock time of oscillation clock (clock period:21 ns). It is required for increment of FIFO.
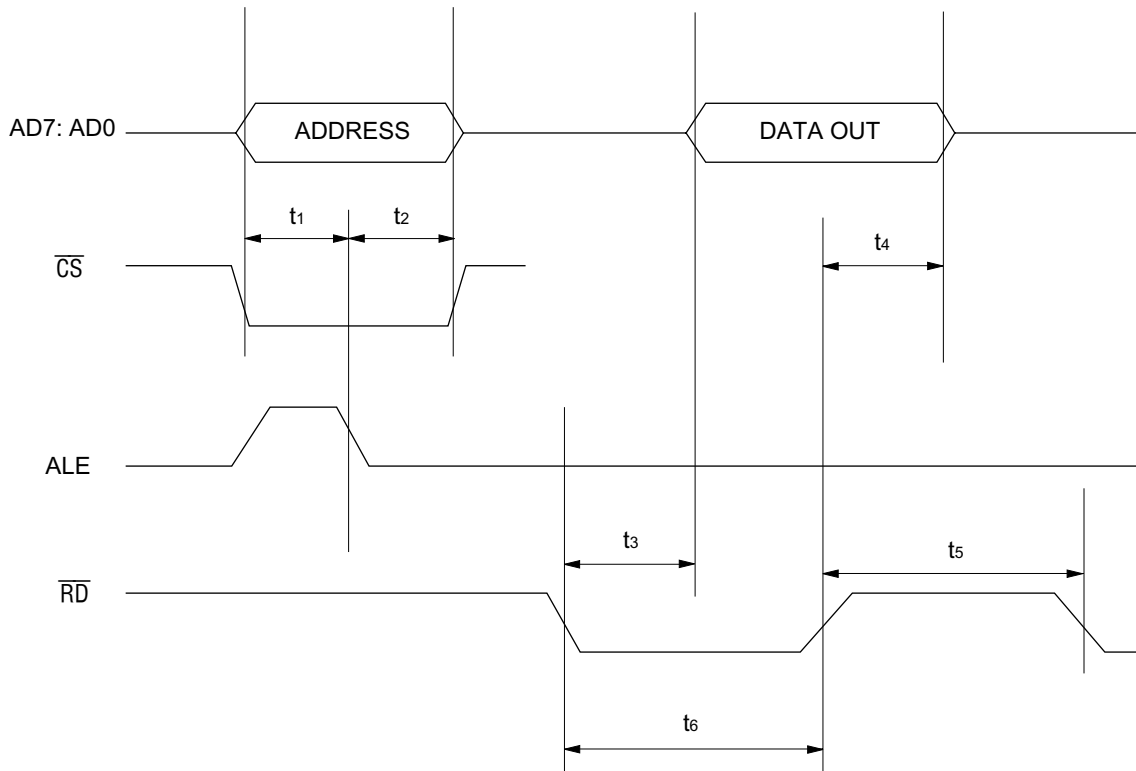
**WRITE Timing (1)**
**(Address Separate, ADSEL =0)**

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Address Setup Time | $t_1$ | | 0 | — | ns | |
| Address ($\overline{CS}$) Hold Time | $t_2$ | | 0 | — | ns | |
| Write Data Setup Time | $t_4$ | | 30 | — | ns | |
| Write Data Hold Time | $t_5$ | | 0 | — | ns | |
| Recovery Time | $t_6$ | FIFO WRITE | 63 | — | ns | (1) |
| FIFO Access Time | $t_7$ | FIFO WRITE | 63 | | ns | (1) |

Note: (1) 3-clock time of oscillation clock (clock period: 21ns). It is required for increment of FIFO.
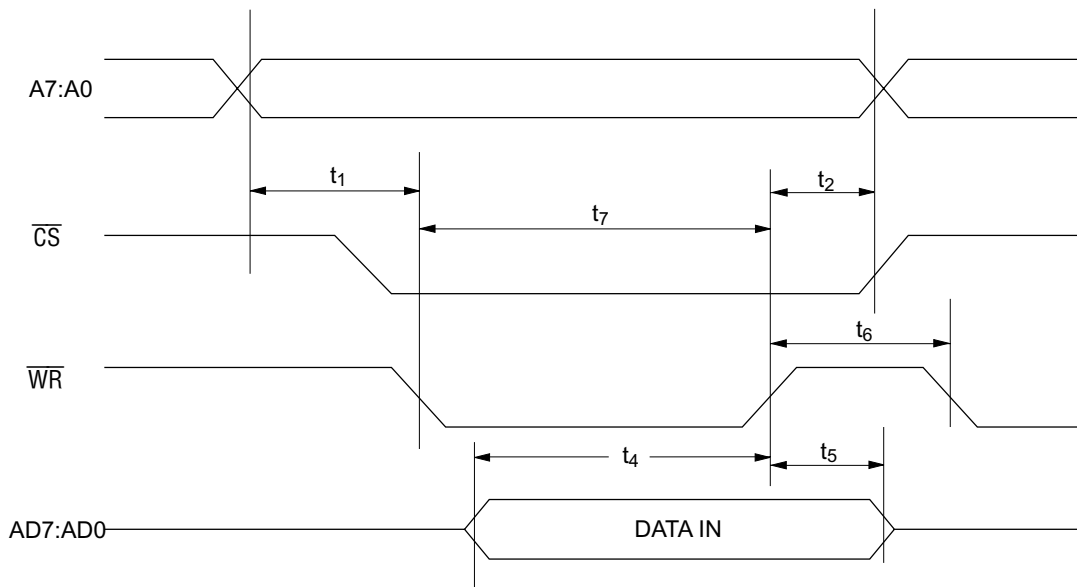
**WRITE Timing (2)**
**(Address/Data Multiplex, ADSEL =1)**

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| Address ($\overline{CS}$) Setup Time | $t_1$ | | 10 | — | ns | |
| Address ($\overline{CS}$) Hold Time | $t_2$ | | 0 | — | ns | |
| Write Data Delay Time | $t_3$ | | 30 | — | ns | |
| Write Data Hold Time | $t_4$ | | 0 | — | ns | |
| Recovery Time | $t_5$ | FIFO WRITE | 63 | — | ns | (1) |
| FIFO Access Time | $t_6$ | FIFO WRITE | 63 | — | ns | (1) |

Notes:  (1)  3-clock time of oscillation clock (clock period: 21 ns). It is required for increment of FIFO.
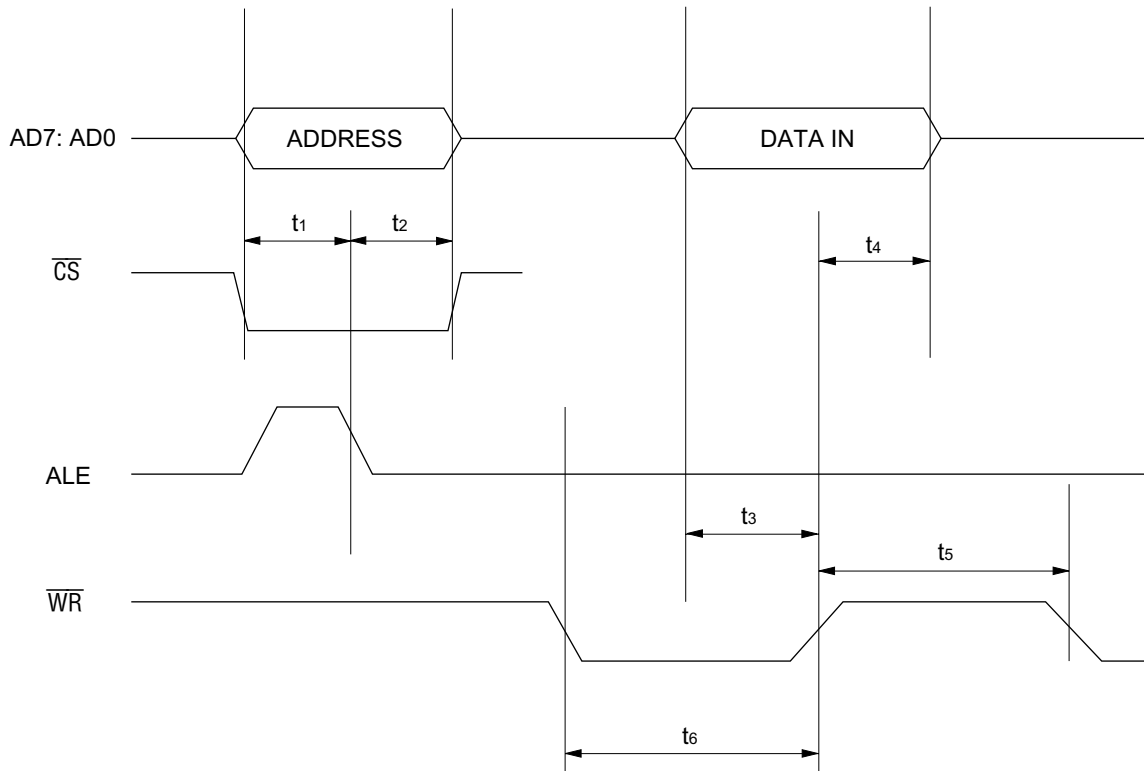
**DMA Transfer Timing (1)**

ML60852A to Memory (Single Transfer, Single Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| $\overline{\text{DREQ}}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| $\overline{\text{DREQ}}$ Enable Time | $t_2$ | | — | 63 | ns | (4) |
| DACK Hold Time | $t_3$ | | 0 | — | ns | |
| Read Data Delay Time | $t_4$ | Load 20 pF | — | 46 | ns | (1) |
| Data Hold Time | $t_5$ | | 0 | — | ns | |
| Recovery Time | $t_6$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1) When in Single Address mode, $\overline{\text{CS}}$ and A6: A0 are ignored.
$t_1$ and $t_4$ are defined depending on DACK or $\overline{\text{RD}}$ which becomes active last.
(2) 3-clock time of oscillation clock (clock period: 21 ns).
(3) 5-clock time of oscillation clock (clock period: 21 ns).
(4) It is possible to increase $t_2$ by setting the DMA interval register (DMAINTVL).
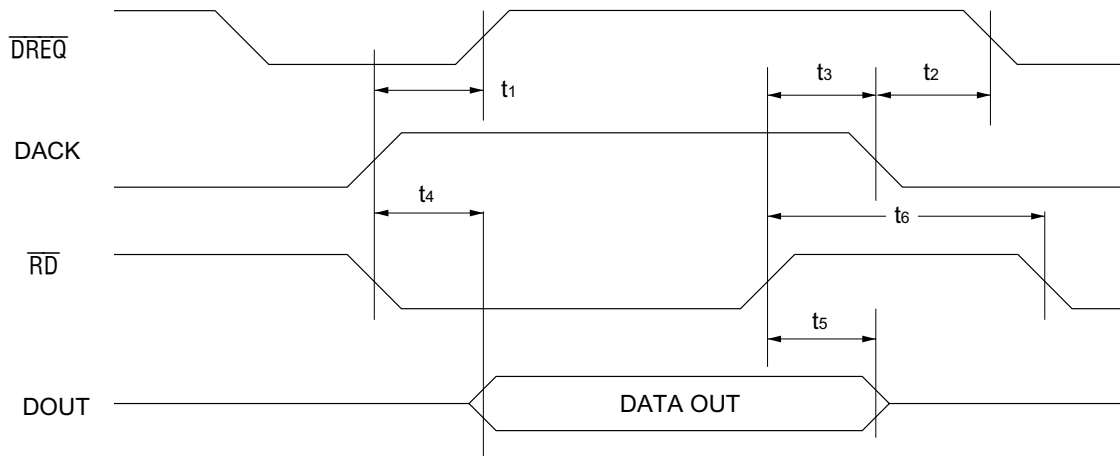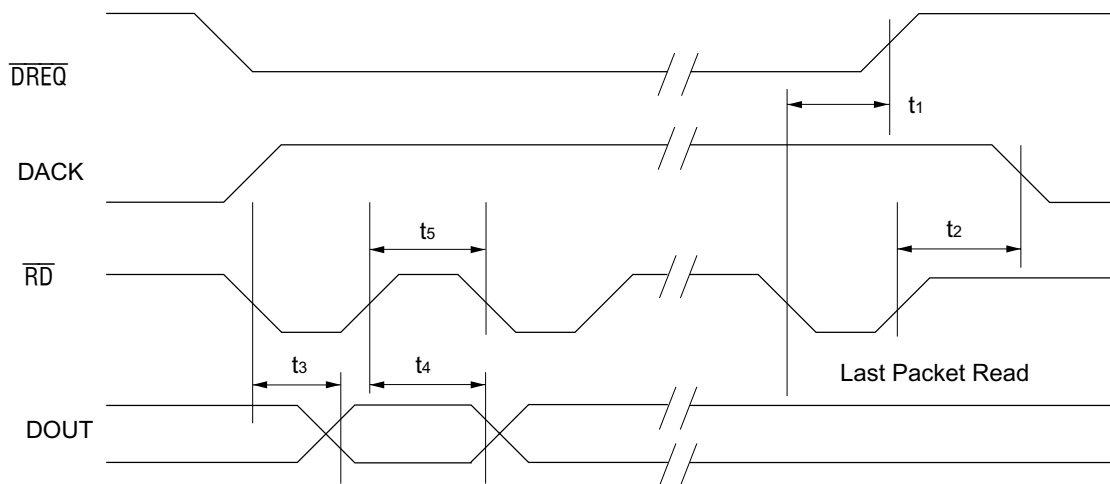
**DMA Transfer Timing (2)**

ML60852A to Memory (Single Transfer, Dual Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| $\overline{\text{DREQ}}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| $\overline{\text{DREQ}}$ Enable Time | $t_2$ | | — | 63 | ns | (4) |
| Read Data Delay Time | $t_3$ | Load 20 pF | — | 46 | ns | (1) |
| Data Hold Time | $t_4$ | | 0 | — | ns | |
| Recovery Time | $t_5$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1) When in Dual Address mode, the DACK is ignored.

$t_1$ and $t_3$ are defined depending on $\overline{\text{CS}}$ or $\overline{\text{RD}}$ which becomes active last.

A6: A0 specifies the FIFO address.

Refer to READ Timing (1) for Address Setup Time and Address Hold Time.

(2) 3-clock time of oscillation clock (clock period: 21 ns).

(3) 5-clock time of oscillation clock (clock period: 21 ns).

(4) It is possible to increase t $t_2$ by setting the DMA interval register (DMAINTVL).

**DMA Transfer Timing (3)**

ML60852A to Memory (Demand Transfer, Single Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|-----------|--------|-----------|------|------|------|------|
| $\overline{\text{DREQ}}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| DACK Hold Time | $t_2$ | | 0 | — | ns | |
| Read Data Delay Time | $t_3$ | Load 20 pF | — | 46 | ns | (1) |
| Data Hold Time | $t_4$ | | 0 | — | ns | |
| Recovery Time | $t_5$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1)  When in Single Address mode, $t_3$ is defined depending on DACK or $\overline{\text{RD}}$ which becomes active last.
A6: A0 and $\overline{\text{CS}}$ are ignored.
(2)  3-clock time of oscillation clock (clock period: 21 ns).
(3)  5-clock time of oscillation clock (clock period: 21 ns).

**DMA Transfer Timing (4)**

ML60852A to Memory (Demand Transfer, Dual Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|-----------|--------|-----------|------|------|------|------|
| $\overline{DREQ}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| $\overline{CS}$ Hold Time | $t_2$ | | 0 | — | ns | |
| Read Data Delay Time | $t_3$ | Load 20 pF | — | 46 | ns | (1) |
| Data Hold Time | $t_4$ | | 0 | — | ns | |
| Recovery Time | $t_5$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes:  (1)  When in Dual Address mode, the DACK is ignored.

   $t_3$ is defined depending on $\overline{CS}$ or $\overline{RD}$ which becomes active last.

   A6: A0 specifies the FIFO address.

   Refer to READ Timing (1) for Address Setup Time and Address Hold Time.

   (2)  3-clock time of oscillation clock (clock period: 21 ns).

   (3)  5-clock time of oscillation clock (clock period: 21 ns).
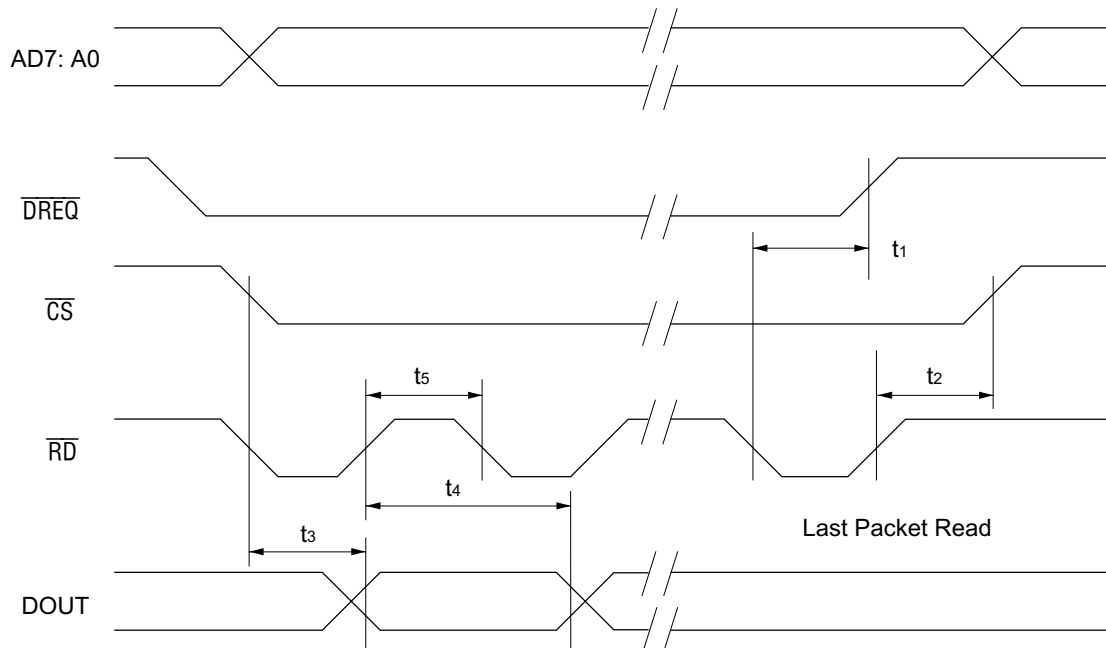
**DMA Transfer Timing (5)**

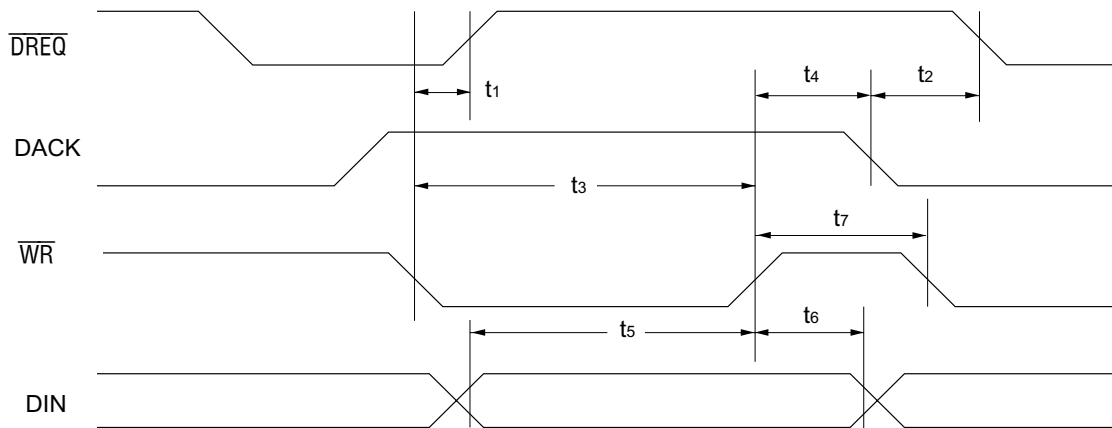Memory to ML60852A (Single Transfer, Single Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| $\overline{\text{DREQ}}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| $\overline{\text{DREQ}}$ Enable Time | $t_2$ | | — | 63 | ns | (4) |
| FIFO Access Time | $t_3$ | FIFO WRITE | 42 | — | ns | (1) |
| DACK Hold Time | $t_4$ | | 0 | — | ns | |
| Write Data Setup Time | $t_5$ | | 30 | — | ns | |
| Write Data Hold Time | $t_6$ | | 5 | — | ns | |
| Recovery Time | $t_7$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1) When in Single Address mode, $\overline{\text{CS}}$ and A6: A0 are ignored.
        (2) 3-clock time of oscillation clock (clock period: 21 ns).
        (3) 5-clock time of oscillation clock (clock period: 21 ns).
        (4) It is possible to increase $t_2$ by setting the DMA interval register (DMAINTVL).
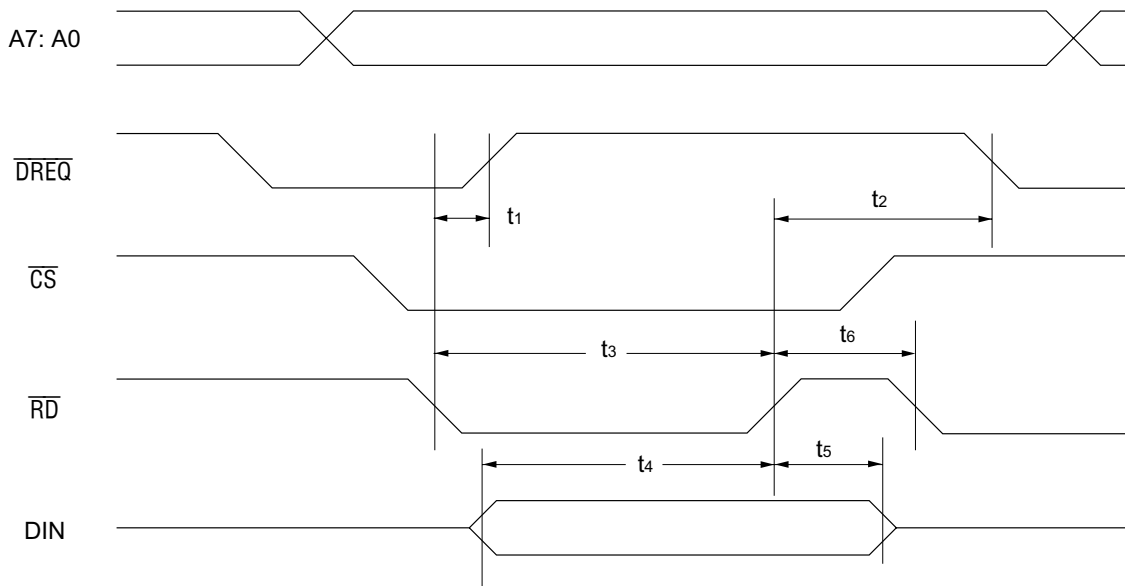
**DMA Transfer Timing (6)**

Memory to ML60852A (Single Transfer, Dual Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| $\overline{DREQ}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| $\overline{DREQ}$ Enable Time | $t_2$ | | — | 63 | ns | (4) |
| FIFO Access Time | $t_3$ | FIFO WRITE | 42 | — | ns | (1) |
| Write Data Setup Time | $t_4$ | | 30 | — | ns | |
| Write Data Hold Time | $t_5$ | | 5 | — | ns | |
| Recovery Time | $t_6$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1) When in Dual Address mode, the DACK is ignored.
$t_1$ and $t_3$ are defined depending on $\overline{CS}$ or $\overline{WR}$ which becomes active last.
Refer to WRITE Timing (1) for Address Setup Time and Address Hold Time.
(2) 3-clock time of oscillation clock (clock period: 21 ns).
(3) 5-clock time of oscillation clock (clock period: 21 ns).
(4) It is possible to increase $t_2$ by setting the DMA interval register (DMAINTVL).

**DMA Transfer Timing (7)**

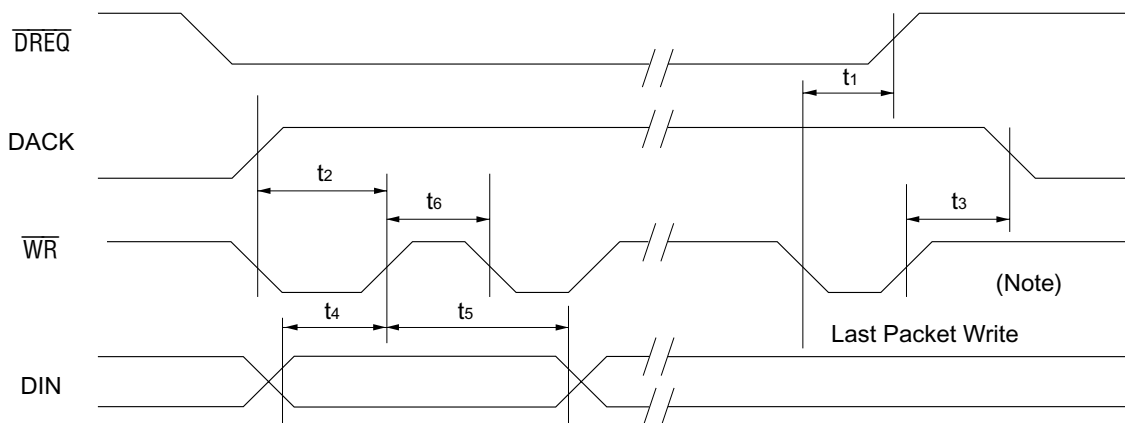Memory to ML60852A (Demand Transfer, Single Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|-----------|--------|-----------|------|------|------|------|
| $\overline{\text{DREQ}}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| FIFO Access Time | $t_2$ | FIFO WRITE | 42 | — | ns | (1) |
| DACK Hold Time | $t_3$ | | 0 | — | ns | |
| Write Data Setup Time | $t_4$ | | 30 | — | ns | |
| Write Data Hold Time | $t_5$ | | 5 | — | ns | |
| Recovery Time | $t_6$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1) When in Single Address mode, A6: A0 and $\overline{\text{CS}}$ are ignored.
$t_2$ is defined depending on DACK or $\overline{\text{WR}}$ which becomes active last.
(2) 3-clock time of oscillation clock (clock period: 21 ns).
(3) 5-clock time of oscillation clock (clock period: 21 ns).



(Note) The last Write to reach the byte size (maximum packet size) specified by the EP1 Payload Register.
To terminate DMA transfer before reaching the maximum packet size, set EP1 Packet Ready by writing "1"to the EP1 Endpoint Packet Ready bit.

**DMA Transfer Timing (8)**

Memory to ML60852A (Demand Transfer, Dual Address Mode)

| Parameter | Symbol | Condition | Min. | Max. | Unit | Note |
|---|---|---|---|---|---|---|
| $\overline{\text{DREQ}}$ Disable Time | $t_1$ | Load 20 pF | — | 20 | ns | |
| FIFO Access Time | $t_2$ | FIFO WRITE | 42 | — | ns | (1) |
| $\overline{\text{CS}}$ Hold Time | $t_3$ | | 0 | — | ns | |
| Write Data Setup Time | $t_4$ | | 30 | — | ns | |
| Write Data Hold Time | $t_5$ | | 5 | — | ns | |
| Recovery Time | $t_6$ | 8-bit DMA | 63 | — | ns | (2) |
| | | 16-bit DMA | 105 | — | ns | (3) |

Notes: (1)  When in Dual Address mode, the DACK is ignored.
A6: A0 specifies the FIFO address.
Refer to WRITE Timing (1) for Address Setup Time and Address Hold Time.
$t_2$ is defined depending on $\overline{\text{CS}}$ or $\overline{\text{WR}}$ which becomes active last.
(2)  3-clock time of oscillation clock (clock period: 21 ns).
(3)  5-clock time of oscillation clock (clock period: 21 ns).
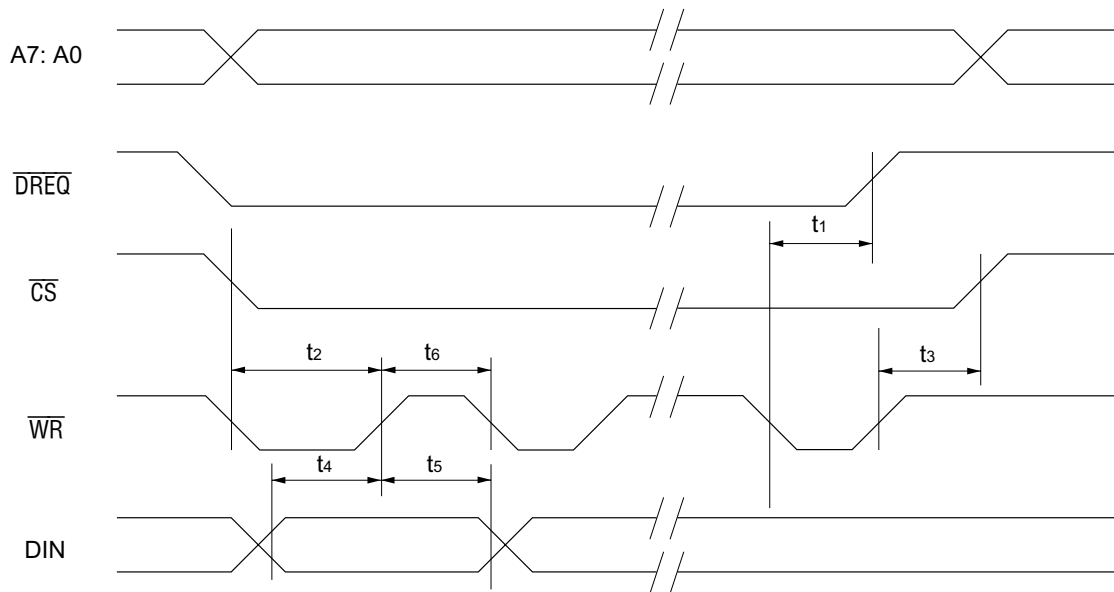


(Note)  Refer to the previous page.

**REVISION HISTORY**

| Date | Changes compared to previous version |
|------|--------------------------------------|
| Nov. 2001 | -The guaranteed operating temperature range has been changed from 0°C ~ 70°C to −20°C ~ +80°C |

NOTICE
1.   The information contained herein can change without notice owing to product and/or technical improvements. Before using the product, please make sure that the information being referred to is up-to-date.

2.   The outline of action and examples for application circuits described herein have been chosen as an explanation for the standard action and performance of the product. When planning to use the product, please ensure that the external conditions are reflected in the actual circuit, assembly, and program designs.

3.   When designing your product, please use our product below the specified maximum ratings and within the specified operating ranges including, but not limited to, operating voltage, power dissipation, and operating temperature.

4.   Oki assumes no responsibility or liability whatsoever for any failure or unusual or unexpected operation resulting from misuse, neglect, improper installation, repair, alteration or accident, improper handling, or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified operating range.

5.   Neither indemnity against nor license of a third party's industrial and intellectual property right, etc. is granted by us in connection with the use of the product and/or the information and drawings contained herein. No responsibility is assumed by us for any infringement of a third party's right which may result from the use thereof.

6.   The products listed in this document are intended for use in general electronics equipment for commercial applications (e.g., office automation, communication equipment, measurement equipment, consumer electronics, etc.). These products are not authorized for use in any system or application that requires special or enhanced quality and reliability characteristics nor in any system or application where the failure of such system or application may result in the loss or damage of property, or death or injury to humans.
Such applications include, but are not limited to, traffic and automotive equipment, safety devices, aerospace equipment, nuclear power control, medical equipment, and life-support systems.

7.   Certain products in this document may need government approval before they can be exported to particular countries. The purchaser assumes the responsibility of determining the legality of export of these products and will take appropriate and necessary steps at their own expense for these.

8.   No part of the contents contained herein may be reprinted or reproduced without our prior permission.