M66230P/FP

A²RT(ADVANCED ASYNCHRONOUS RECEIVER & TRANSMITTER)

DESCRIPTION

The M66230P/FP is an integrated circuit for asynchronous serial data communications.It is used in combination with an 8-bit micro-processor and is produced using the silicon-gate CMOS technology.

FEATURES

- Baud rate generator 500kbps (max)
- 4-byte FIFO data buffer for transmission and reception
- Error detection : CRC-CCITT, parity, overrun, and framing
- Wakeup function
- Transmisson / reception data format (number of bits)
  - Start bit 1
  - Data bit 8
  - Wakeup bit 1 or nil
  - Parity bit 1 or nil
  - Stop bit 1 or 2
- Access time t_a ($\overline{RD}$-D) : 100ns (max)
- High output current
  I_{OH}=−24mA, I_{OL}=24mA T_XD, $\overline{RTS}$, P0, P1 pins
- Schmitt triggered input R_XD,$\overline{CTS}$,$\overline{RESET}$ pins

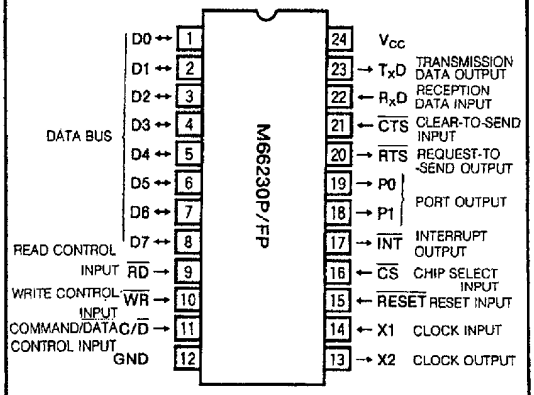
APPLICATION

Data communication control

FUNCTION

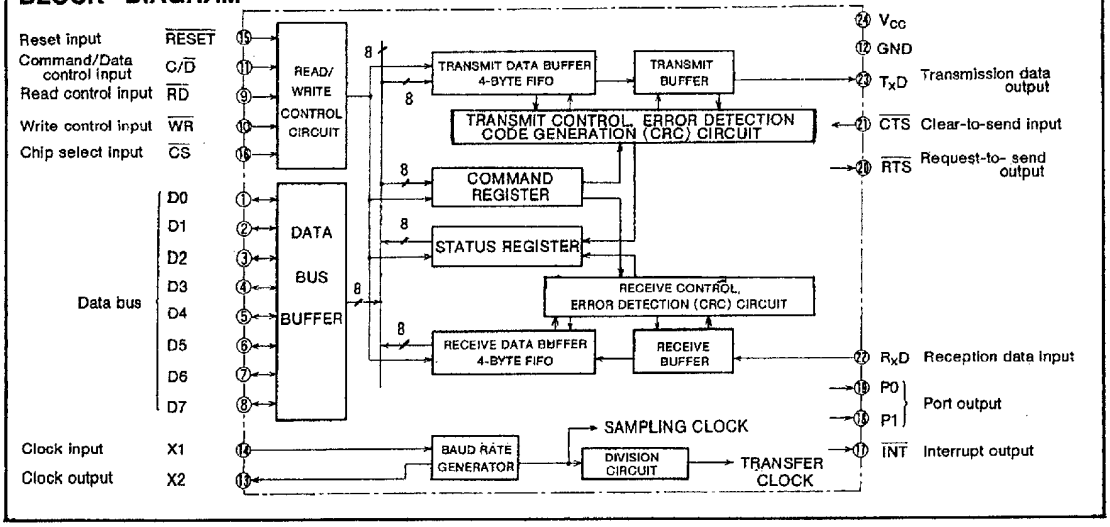
The M66230P/FP is a UART (Universal Asynchronous Receiver/Transmitter) and is used in the peripheral circuit of a MPU. The M66230 receives parallel data, converts into serial format, and then transmits the serial data via the T_XD pin. The device also receives data via the R_XD pin from external circuits and converts it into parallel format, and sends the parallel data via the data bus.

PIN CONFIGURATION (TOP VIEW)



Outline 24P4D
24P2W-A

BLOCK DIAGRAM



OPERATION

The M66230 is interfaced to a system bus and provides all functions needed for data communication.

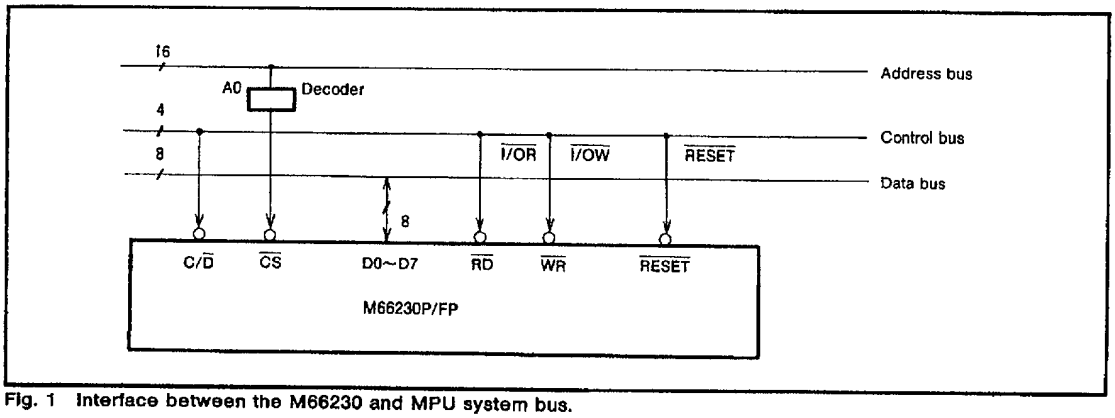


Fig. 1 Interface between the M66230 and MPU system bus.

When using the M66230, it is necessary to program the initial setting, baud rate, character length, CRC, parity, in accordance with the communication system. Once programmed, the communcation system functions are executed continuously.

When initial setting of M66230 is completed, data communication becomes possible. When the transmitter is transmit-enabled (TXEN) by a command instruction and $\overline{CTS}$ is low-level, data transfer starts up. If these conditions are not satisfied, data transmission is not executed. Reception is possible when the receiver is receive -enabled (RXEN) by a command instruction.

The MPU is able to read data when the interrupt output, $\overline{INT}$, goes low by packet end (PE) or buffer full (BF).

While receiving data, the M66230 checks for errors and provides status information. It checks for four types of errors : CRC, parity, overrun and framing errors. When an error occurs, M66230 continues operation. The error status is maintained until the error reset, (ER) is modified by a command instruction.

The access method of the M66230 is shown Table 1.

TABLE 1 Access method of the M66230.

C/D	RD	WR	CS	M66230 operation	MPU operation
L	L	H	L	Data bus←Receiving data buffer (FIFO)	Read receive data
L	H	L	L	Data bus→Transmit data buffer (FIFO)	Write transmit data
H	L	H	L	Data bus←Status register	Read the status
H	H	L	L	Data bus→Command register	Write the command
X	H	H	L	Data bus : high impedance	—
X	X	X	H	Data bus : high impedance	—

Note : X="L" or "H"

A²RT(ADVANCED ASYNCHRONOUS RECEIVER & TRANSMITTER)

PIN DESCRIPTIONS

Pin	Name	I/O	Function
X1	Clock input	Input	A crystal is externally connected to these pins for generating an internal clock. An external clock signal can be input to X1 instead of a crystal. Then X2 output opened.
X2	Clock output	output	
$\overline{RESET}$	Reset input	Input	This reset is a master reset, therefore commands should be loaded after the reset.
$\overline{CS}$	Chip select input	Input	A low level signal on the chip select input enables the M66230. The device can not be accessed when the signal is high-level.
C/$\overline{D}$	Command/ Data control input	Input	This signal distinguishes whether the information on the M66230 data bus is data, command or status information. When the signal is high-level, the data bus has command or status information. When the signal is low-level, the data bus has data.
$\overline{RD}$	Read control input	Input	The receiving data or status information is output to the data bus from the M66230 by a low-level signal.
$\overline{WR}$	Write control input	Input	The data or command output from the MPU is written to the M66230 by a low-level signal.
D0~D7	Data bus	Input/output	This is an 8-bit bi-directional bus buffer. Command, status information, and transfer data are transferred to/from the MPU via this data bus buffer.
$\overline{INT}$	Interrupt output	Output	This is used as an interrupt request to MPU. The interrupt request is generated when the receive FIFO is full, the transmit FIFO is empty, or the block reception is complete. D2 bit of command 6 controls the switching of low-level and high-level interrupt.
RxD	Reception data input	Input	The serial data is sent to this pin.
TxD	Transmission data output	Output	The serial data is transmitted from this pin.
P0	Port output	Output	This is an ordinary port pin. This pin is controlled by the D0 bit of command 6.
P1	Port output	Output	This pin has the same function as that of P0 pin and provides information of packet transmission's completion. The switching of this function is controlled by command 6, D1 bit.
$\overline{CTS}$	Clear-to-send input	Input	When the TXEN bit (D0) of command 4 is set to 1 and the $\overline{CTS}$ input is low-level, serial data is sent from the TxD pin. This is used as the clear-to-send signal.
$\overline{RTS}$	Request-to-send output	Output	This is used as the request-to-send signal. This pin is controlled by the D3 bit of command 4.

DISCRIPTION OF FUNCTION

● Baud rate generator

The 8-bit programmable divider (baud rate generator) generates the baud rate for transmit or receive. The division rate is (n+1) with a range of n=0~255. The baud rate is calculated by the following formula :

baud rate=
f(X1)/(prescaler division rate (2 or 32) · baud rate generator division rate (n+1) · 16).

The prescaler division rate is set by the D0 bit of command 1. The baud rate generator division rate is set by command 2. Example as follows :

9600bps=(9.8304MHz)/(2 · (31+1) · 16)

where prescaler division rate is 2 and baud rate division rate is 31.

● Block length counter

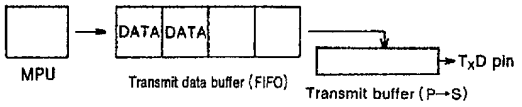
The M66230 can handle multiple-bytes of data as one block (packet).
Therefore, CRC of bytes is possible. The block length counter is a 6-bit programmable counter. The block length is (m+1) bytes with the allowed values of m=0~63.

● Transmit data buffer (FIFO)

The transmit data buffer (FIFO) consists of 4-bytes.
The transmit data buffer (FIFO) functions according to the block length.

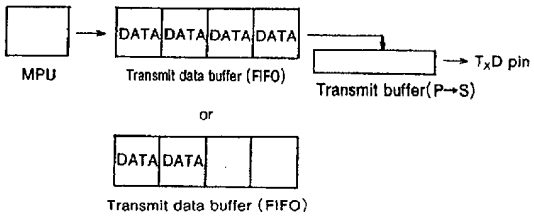
Block length=1~3
When the transmit data buffer (FIFO) becomes empty (buffer empty) and $\overline{INT}$ is set to low-active, the interrupt output $\overline{INT}$ is set to a low-level. The MPU verifies the buffer is empty when the D2 bit of the status 1 information is read. The MPU should write the block length data to the transmit data buffer (FIFO) at this moment.
When a block of data is written to the transmit data buffer (FIFO), $\overline{CTS}$ is low-level and TXEN is high-level, the data in the transmit data buffer (FIFO) is sent to the transmit buffer. If $\overline{CTS}$ is high-level while data is transmitted, all data is transmitted (including the data in the transmit data buffer (FIFO)). When the buffer becomes empty, the data in the transmit data buffer (FIFO) is not be sent to the transmit buffer until MPU writes a new block of data to the transmit data buffer (FIFO). The MPU can not write new data to the transmit data buffer (FIFO) until the buffer becomes empty.

Example : Block length=2



Block length=4 or more
When the transmit data buffer (FIFO) becomes empty and $\overline{INT}$ is set low-active, the interrupt output $\overline{INT}$ becomes low. The MPU verifies the buffer is empty by reading the D2 bit of the status 1 information.
When this happens, the MPU should write the 4-bytes of data to the transmit data buffer (FIFO). The data in the transmit data buffer (FIFO) is sent to the transmit buffer, when $\overline{CTS}$ is low-level and TXEN is high-level. When the number of bytes from the MPU becomes less than 4 at the last stage of the block transmission, the same operation should be made as the block length=1~3.
When the buffer becomes empty, the data in the transmit data buffer (FIFO) is not be sent to the transmit buffer until MPU writes data of the fixed block length to the transmit data buffer (FIFO). The MPU cannot write data to the transmit data buffer (FIFO) until the buffer becomes empty.

Example : Block length=6



● Receive data buffer (FIFO)

The receive data buffer (FIFO) consists of 4-bytes. The receive data buffer (FIFO) functions according to the block length.

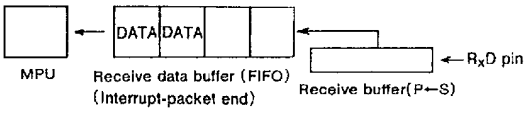
Block length=1~3

When the data of the block length is received and $\overline{INT}$ is set to low-level, the interrupt output $\overline{INT}$ becomes low-level. The MPU acknowledges the packet end by setting the D0 bit of the status 1 information.

In this case, the MPU should read all data from the receive data buffer (FIFO).

At the packet end, the data from the receive buffer cannot be transmitted to the receive data buffer (FIFO) until the MPU reads all data in the receive data buffer (FIFO). The MPU cannot read data in the receive data buffer until the packet end.

Example : Block length=2



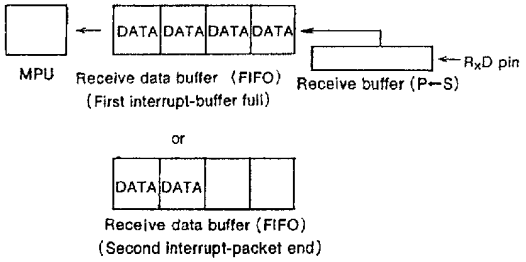
Block length=4 or more

When 4-byte data enters the receive data buffer (FIFO) (buffer full) and $\overline{INT}$ is set to low-active, the interrupt output $\overline{INT}$ becomes low-level. The MPU acknowledges the buffer full status by setting the D1 bit of the status 1 information.

In this case, the MPU should read all data in the receive data buffer (FIFO).

When the last data enters the receive data buffer (FIFO), the packet end becomes the same operation as for 1~3 byte block length. If the block length is a multiple of four, the D0 and D1 bits of the status 1 information are set when the last data enters the receive data buffer (FIFO). At packet end or buffer full, the new data cannot be transferred from the receive buffer to the receive data buffer (FIFO). The MPU cannot read data in the receive data buffer (FIFO) until packet end or buffer full occurs.

Example : Block length=6



SUPPLEMENTARY DESCRIPTION

FIFO

The major purpose is not to interrupt the MPU by each character. The MPU is interrupted when :

Transmit data buffer (FIFO) empty

Receive data buffer (FIFO) full or packet end

The MPU interruption interval is as follows :

Approximately 90μs (min) until the FIFO becomes full at 500kbps.

Approximately 36.7ms (min) until the FIFO becomes full at 1.2kbps.

Read/write operation by the MPU should be made for all data in FIFO at once.

● Wakeup

The wakeup mode of the M66230 can be set by setting the D2 bit of command 4 to "1". In wakeup mode, a 9th bit is automatically added (the wakeup bit).

Only the 9th bit of the first byte is "1", and the remainer blocks 9th bits are set to "0".

The wakeup is used when one master MPU and multiple local MPU are connected by serial I/O.

Examples of wakeup are shown below.

① Initial setting

The initial setting should be made by the input of each command.

② Wakeup mode

The wakeup mode of the M66230 is activated by setting D2 bit of the command 4 to "1". Command 5 can be input as the second byte of command 4 by setting D2 bit of the command 4 to "1" and each address is input. In the wakeup mode, the 9th bit is automatically added. Others remain the same.

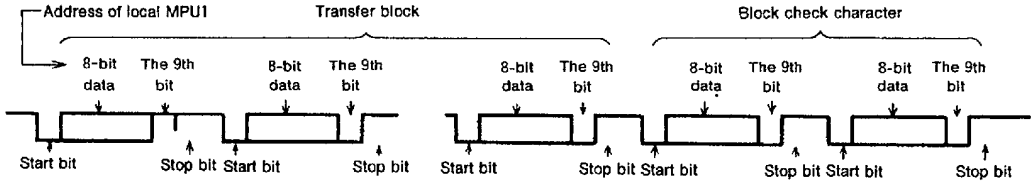
③ Wakeup and data transfer (between master MPU and local MPU1)

Data is transmitted from the master MPU to each local MPU.

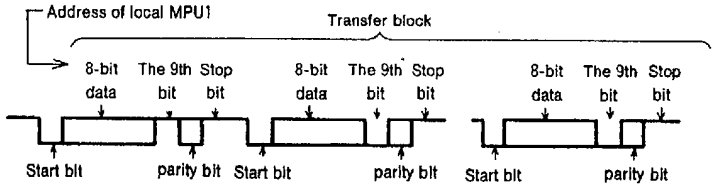
The first byte should hold the address of the local MPU. (in this case local MPU1.)

Each local M66230 checks the data (address) against command 5 (each address) when the first byte (address) is received. the M66230 which matches the address starts to accept the following data (wakeup). the M66230 which does not match the address, only accepts data, where the 9th bit is "1".

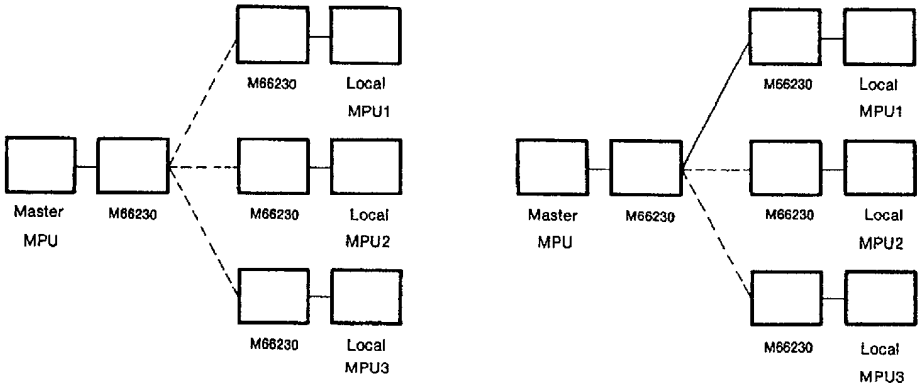
When CRC is enabled



When parity is enabled



Note : The wakeup function is automatically canceled when the transfer block data has been read by the MPU. (The wakeup mode continues.)



● Error detection

Parity error

When a parity error occurs, D5 bit of status 1 information is set. The data is send to the receive data buffer (FIFO).

Framing error

When a framing error occurs, D3 bit of the status 1 information is set. The data is sent to the receive data buffer (FIFO).

Overrun error

When data is received before all data in the receive data buffer (FIFO) has been read by MPU, D4 bit of the status 1 information is set as an overrun error.
In this case, the new data in the receive buffer are lost.

CRC error

When an error occurs after receiving block check character, D6 bit of the status 1 information is set.

The above error information is maintained until D4 bit of command 4 is set.

SUPPLEMENTARY DESCRIPTION

Comparison between parity check and CRC

Parity check

Parity check needs only one additional bit and is highly efficient. The formula is straightforward, and includes even parity and odd parity checks. In both cases, one bit is added.

CRC

The CRC polynominal expression is CRC-CCITT $X^{16}+X^{12}+X^5+1$.

CRC deals with data characters in transmitted or received blocks. (Start, stop and wakeup bits are excluded.)

When the CRC is enabled, the transmit and receive data consists of block length (1～64 bytes) +2 bytes (block check characters). The following table shows the comparision between parity check and CRC.

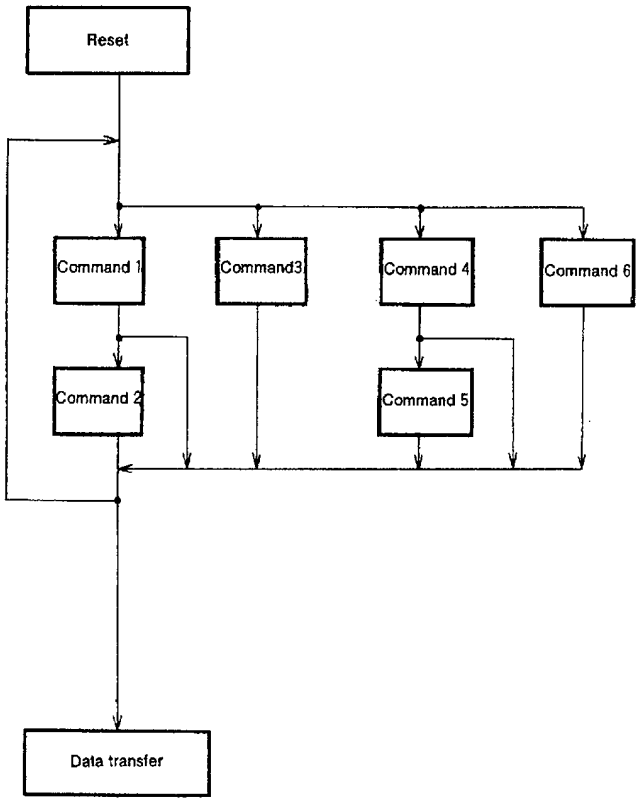
Parity check	Burst error is not detected. (50% of which can be detected.)
CRC	Burst error can be detected (Burst error detection rate is more than 99.9%.)

PROGRAMMING

The command must be loaded first to the M66230 by the MPU before data communication. M66230 has 6 command registers.

Data transfer is possible when commands have been loaded to these command registers after reset.

The flowchart of the initial setting is shown in the following diagram.

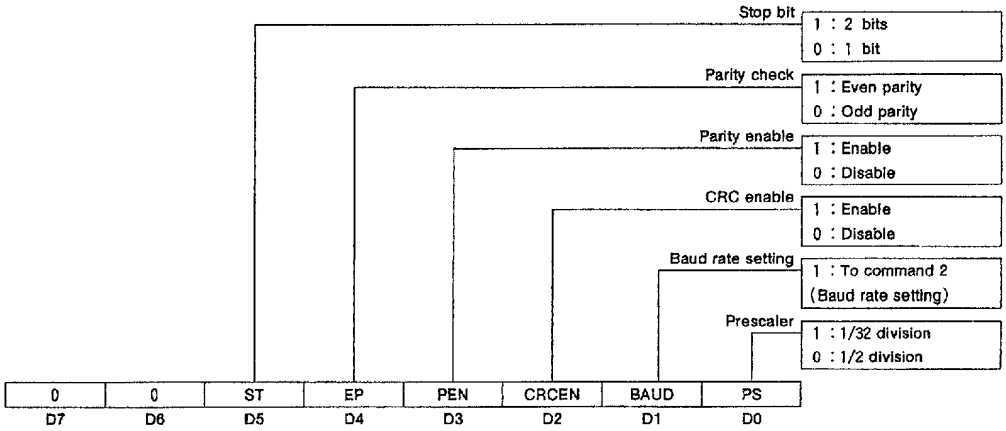


Flowchart of the M66230 initial setting.

COMMAND-INSTRUCTION FORMAT

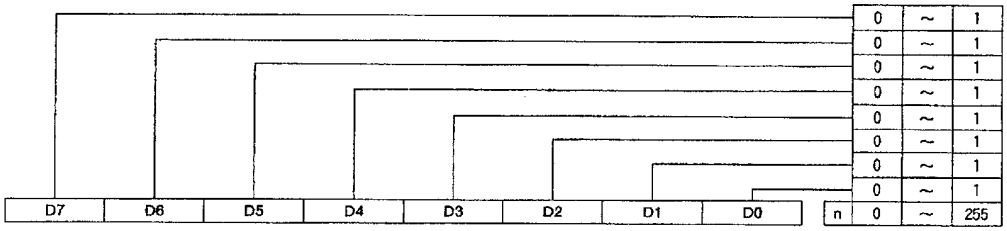
The commands are decoded by D7 and D6.

Command1

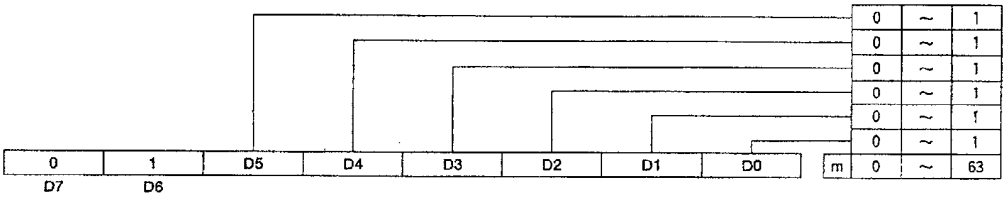


Note 1 : Priority is given to parity enable, if parity enable and CRC enable are both "1" (D3, D2=1).
Attention : TxD output wave is Stop bit (D5) setup value +1 (always)

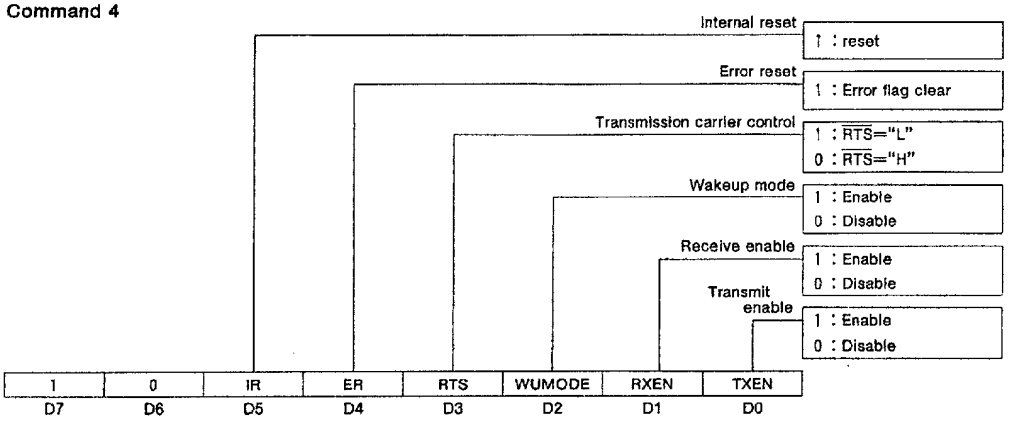
Command 2 (Baud rate setting. The second byte when D1 bit of the command 1 is set to "1")



Command 3 (Block length setting)



Command 4



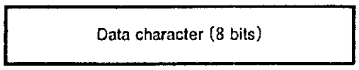
A²RT(ADVANCED ASYNCHRONOUS RECEIVER & TRANSMITTER)

TRANSMISSION FORMAT

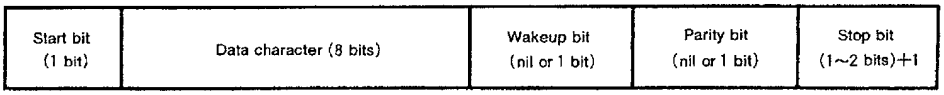
Transmit format

Parity enabled

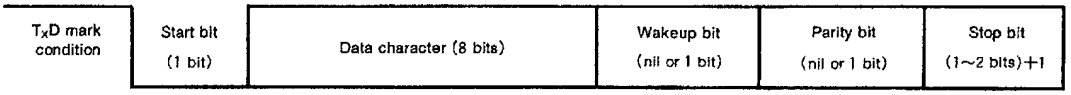
MPU→M66230



Assembled data format

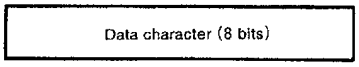


Transmitter output

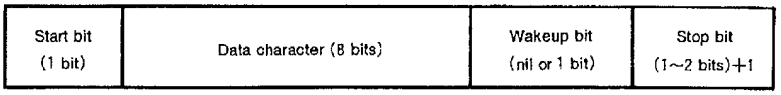


CRC enabled

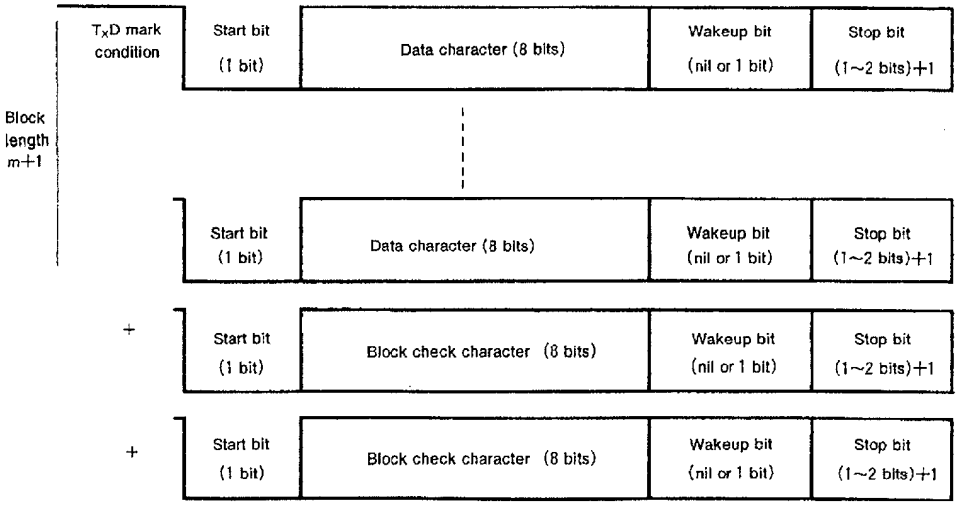
MPU→M66230



After assembly



Transmitter output



A²RT(ADVANCED ASYNCHRONOUS RECEIVER & TRANSMITTER)

ABSOLUTE MAXIMUM RATINGS (T_a=−40~+85℃, unless otherwise noted)

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage	Value using the GND pin as reference	−0.5~+7.0	V
V_I	Input voltage		−0.5~V_{CC}+0.5	V
V_O	Output voltage		−0.5~V_{CC}+0.5	V
Pd	Power dissipation	Actually mounted	500	mW
Tstg	Storage temperature		−65~+150	℃

RECOMMENDED OPERATING CONDITIONS (T_a=−40~+85℃)

Symbol	Parameter	Limits			Unit
		Min	Typ	Max	
V_{CC}	Supply voltage	4.5	5.0	5.5	V
GND	Ground		0		V
Topr	Operating temperature	−40		+85	℃

ELECTRICAL CHARACTERISTICS (T_a=−40~+85℃, V_{CC}=5V±10%, GND=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
V_{IH}	High-level input voltage	$\overline{RD}$, $\overline{WR}$, C/$\overline{D}$, $\overline{CS}$, D0~D7	2.0			V
V_{IL}	Low-level input voltage				0.8	V
V_{IH}	High-level input voltage	X1	V_{CC}×0.8			V
V_{IL}	Low-level input voltage				V_{CC}×0.2	V
V_{T+}	Positive threshold voltage	R_XD, $\overline{CTS}$, $\overline{RESET}$			2.4	V
V_{T-}	Negative threshold voltage		0.6			V
V_H	Hysteresis width		0.2			V
V_{OH}	High-level output voltage	I_{OH}=−8mA $\overline{INT}$,D0~D7	V_{CC}−0.8			V
		I_{OH}=−24mA T_XD, $\overline{RTS}$, P0, P1				
V_{OL}	Low-level output voltage	I_{OL}=8mA $\overline{INT}$,D0~D7			0.55	V
		I_{OL}=24mA T_XD, $\overline{RTS}$, P0, P1			0.55	
I_{IH}	High-level input current	V_I=V_{CC}			1.0	μA
I_{IL}	Low-level input current	V_I=GND			−1.0	μA
I_{OZH}	Off-state high-level output current	V_O=V_{CC}			5.0	μA
I_{OZL}	Off-state low-level output current	V_O=GND			−5.0	μA
I_{CC}	Static supply current	V_I=V_{CC}, GND			40	mA
C_I	Input capacitance				10	pF
$C_{I/O}$	I/O capacitance				20	pF

A²RT(ADVANCED ASYNCHRONOUS RECEIVER & TRANSMITTER)

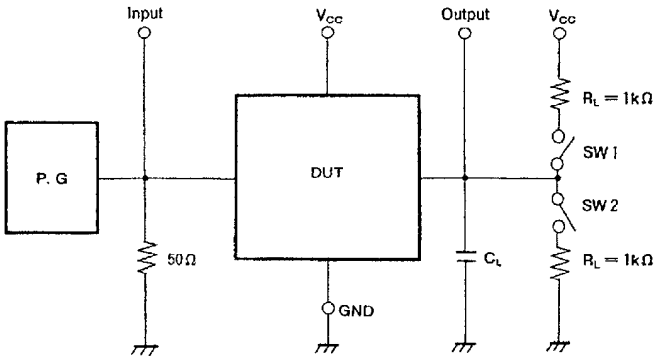
TIMING REQUIREMENTS (T_a=−40~+85℃, V_{CC}=5V±10%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{C1(X1)}$	Clock frequency	(Except Wakeup, CRC mode)		62.5			ns
$t_{WH1(X1)}$	Clock high-level pulse width			30			ns
$t_{WL1(X1)}$	Clock low-level pulse width			30			ns
$t_{C2(X1)}$	Clock frequency	(Wakeup, CRC mode)		80			ns
$t_{WH2(X1)}$	Clock high-level pulse width			38			ns
$t_{WL2(X1)}$	Clock low-level pulse width			38			ns
$t_{r(X1)}$	Clock rise time					20	ns
$t_{f(X1)}$	Clock fall time					20	ns
$t_{su(A-\overline{R})}$	Address setup time before read ($\overline{CS}$, C/$\overline{D}$)			0			ns
$t_{h(\overline{R}-A)}$	Address hold time after read ($\overline{CS}$, C/$\overline{D}$)			0			ns
$t_{W(\overline{R})}$	Read pulse width			100			ns
$t_{su(A-\overline{W})}$	Address setup time before write ($\overline{CS}$, C/$\overline{D}$)			0			ns
$t_{h(\overline{W}-A)}$	Address hold time after write ($\overline{CS}$, C/$\overline{D}$)			0			ns
$t_{W(\overline{W})}$	Write pulse width			100			ns
$t_{su(DQ-\overline{W})}$	Data setup time before write			50			ns
$t_{h(\overline{W}-DQ)}$	Data hold time after write			5			ns
$t_{rec(\overline{RESET})}$	Recoverry time between write			100			ns
$t_{W(\overline{RESET})}$	Reset pulse width			100			ns

SWITCHING CHARACTERISTICS (T_a=−40~+85℃, V_{CC}=5V±10%, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{PZH(\overline{R}-DQ)}$	Data output enable time after read			43	100	ns
$t_{PZL(\overline{R}-DQ)}$	Data output enable time after read			52	100	ns
$t_{PHZ(\overline{R}-DQ)}$	Data output disable time after read			33	85	ns
$t_{PLZ(\overline{R}-DQ)}$	Data output disable time after read			32	85	ns
$t_{PLH(\overline{R}-\overline{INT})}$	$\overline{INT}$ output propagation time after read data			62	170	ns
$t_{PHL(\overline{R}-\overline{INT})}$	$\overline{INT}$ output propagation time after read data			63	170	ns
$t_{PLH(\overline{W}-\overline{INT})}$	$\overline{INT}$ output propagation time after write data			54	150	ns
$t_{PHL(\overline{W}-\overline{INT})}$	$\overline{INT}$ output propagation time after write data			54	150	ns
$t_{PLH(\overline{W}-\overline{INT})}$	$\overline{INT}$ output propagation time after write command (command 4)			33	100	ns
$t_{PHL(\overline{W}-\overline{INT})}$	$\overline{INT}$ output propagation time after write command (command 4)			35	100	ns
$t_{PLH(\overline{W}-\overline{INT})}$	$\overline{INT}$ output propagation time after write command (command 6)			28	100	ns
$t_{PHL(\overline{W}-\overline{INT})}$	$\overline{INT}$ output propagation time after write command (command 6)			30	100	ns
$t_{PLH(\overline{W}-P0)}$	P0 output propagation time after write command			25	70	ns
$t_{PHL(\overline{W}-P0)}$	P0 output propagation time after write command			28	70	ns
$t_{PLH(\overline{W}-P1)}$	P1 output propagation time after write command			26	70	ns
$t_{PHL(\overline{W}-P1)}$	P1 output propagation time after write command			28	70	ns
$t_{PLH(\overline{W}-\overline{RTS})}$	$\overline{RTS}$ output propagation time after write command			25	70	ns
$t_{PHL(\overline{W}-\overline{RTS})}$	$\overline{RTS}$ output propagation time after write command			27	70	ns

TEST CIRCUIT

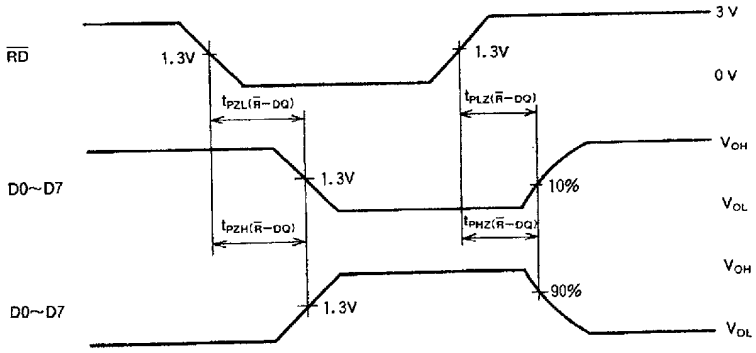


Parameter	SW1	SW2
t_{PLH}, t_{PHL}	Open	Open
t_{PLZ}	Closed	Open
t_{PHZ}	Open	Closed
t_{PZL}	Closed	Open
t_{PZH}	Open	Closed

(1) The pulse generator (PG) has the following characteristics (10%~90%) t_r=3ns, t_f=3ns
(2) The capacitance C_L = 150pF includes stray wiring capacitance and the probe input capacitance.

TIMING DIAGRAM

Input/output waveform at read data and read status



Clock Timing