



制御用 Nチャネル E/D MOS LSI

LM8051, 8031 — 8ビット1チップマイクロコンピュータ

概要

LM8051 4K×8ビット マスクROM、128×8ビットRAM内蔵

LM8031 外部ROM専用で、128×8ビットRAM内蔵

LM8051は、内部にプログラムROM、データRAM、ALU、I/Oポート、シリアルポート、タイマ/カウンタ、クロックジェネレータなどを1チップに集積したNチャネルE/D MOSプロセスの8ビット並列処理マイクロコンピュータである。内蔵メモリの容量は、プログラムメモリ(マスクROM)4096バイト、データメモリ(RAM)128バイトであり、I/Oポートは32本、2系統の16ビットタイマ/カウンタを有し、ビット操作の強力な制御指向の高性能マイクロコンピュータである。

LM8031は、LM8051よりマスクROMのみを除いた機能を有し、外部プログラムメモリ専用機種です。

特長

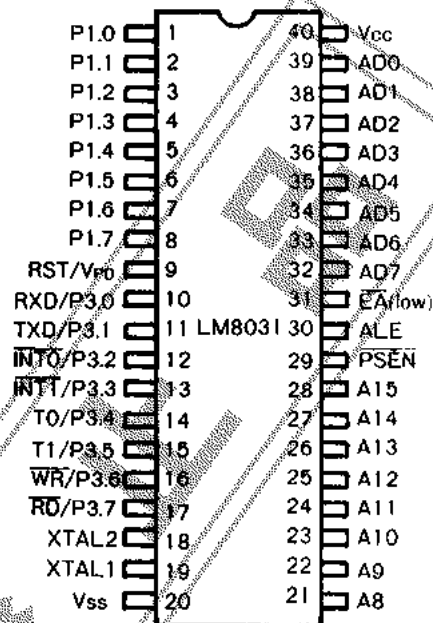
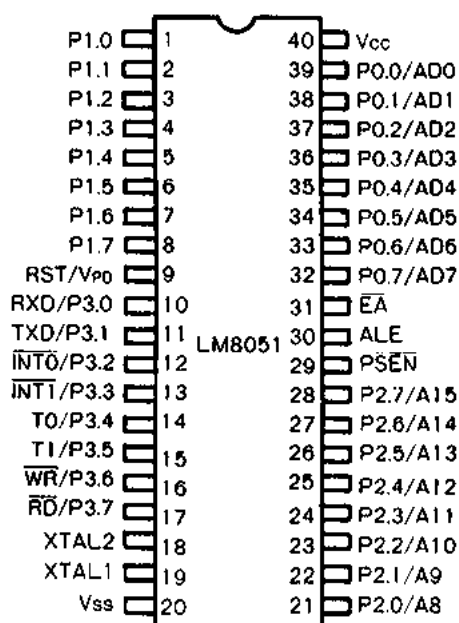
- ・4つの8ビットI/Oポート、計32本のI/O端子(LM8031は、計16本のI/O端子)。
- ・フレキシブルな4モード動作の2本の16ビットタイマ/カウンタ。
- ・フレキシブルな高性能全2重方式のシリアルI/Oポート。
- ・I/Oポートの拡張が可能。
- ・プログラムROM 4Kバイトを内蔵し、外部に60Kバイト拡張可能(LM8051)。
- LM8031は、外部メモリ 64Kバイト、直接アクセス可能。
- ・データメモリ用のRAM 128バイトを内蔵し、外部に独立に64Kバイト拡張可能。
- ・アキュムレータ等の21個の特殊機能レジスタ(SFR)。
- ・命令セット数 111種類。
- ・最小命令実行時間 1 μ sec. (全命令の58%)。乗除算命令 4 μ sec. (但し、12MHzクロック使用の場合)。
- ・豊富なメモリアドレスリングモード。
 - 1) 内部RAM、SFRの直接アドレッシング。
 - 2) 内部RAMの間接アドレッシング(@Ri, SP)。
 - 3) 内部RAM、SFRのビット単位の直接アドレッシング。
 - 4) 外部データメモリの間接アドレッシング @DPTR, @Ri)。
 - 5) プログラムメモリの直接アドレッシング(16, 11ビット指定)、相対アドレッシング(−128〜+127番地)、及び間接アドレッシング @A+DPTR, @A+PC)。
- ・アキュムレータでのパリティチェック。
- ・割込み要因は外部2、内部3の5レベル有り。2レベルの多重割込みが可能(優先順位はプログラム可能)。
- ・スタックは内部RAM 128バイト使用で64レベル直接可能。また、POP、PUSH命令を有し、スタック操作が可能。
- ・ビット単位のアクセス及びビット処理が可能。ブール演算命令が豊富である。
- ・豊富な条件付分岐命令。分岐は相対アドレスによる。
- ・インテル8048ファミリと上位コンパチブルのアーキテクチャ。
- ・インテル8051/31とコンパチブル。

※これらの仕様は、改良などのため変更することがあります。

www.DataSheet4U.com

LM8051,8031

ピン配置図

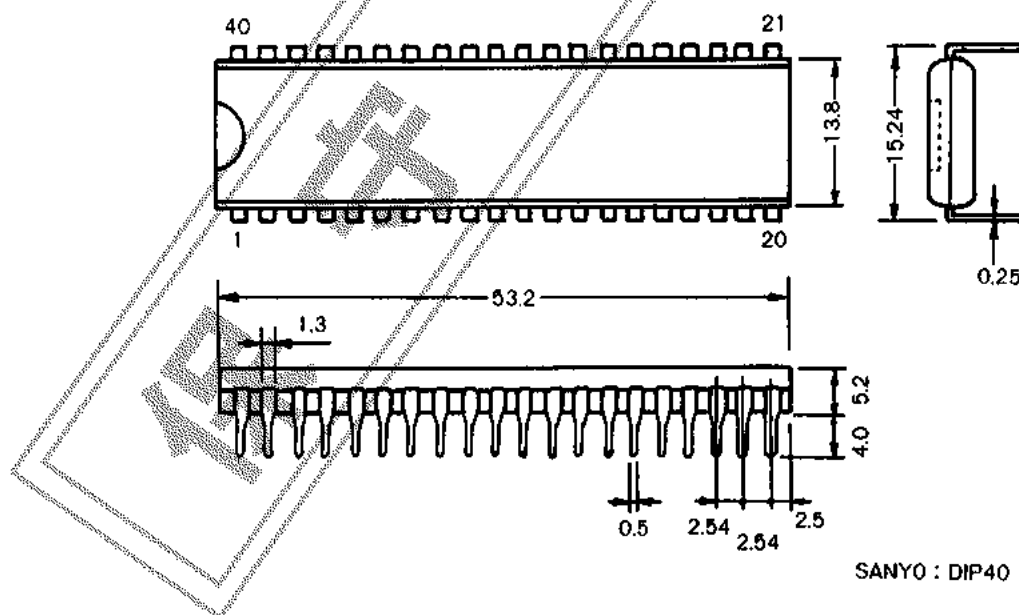


et4U.com

外形図3013 (unit : mm)

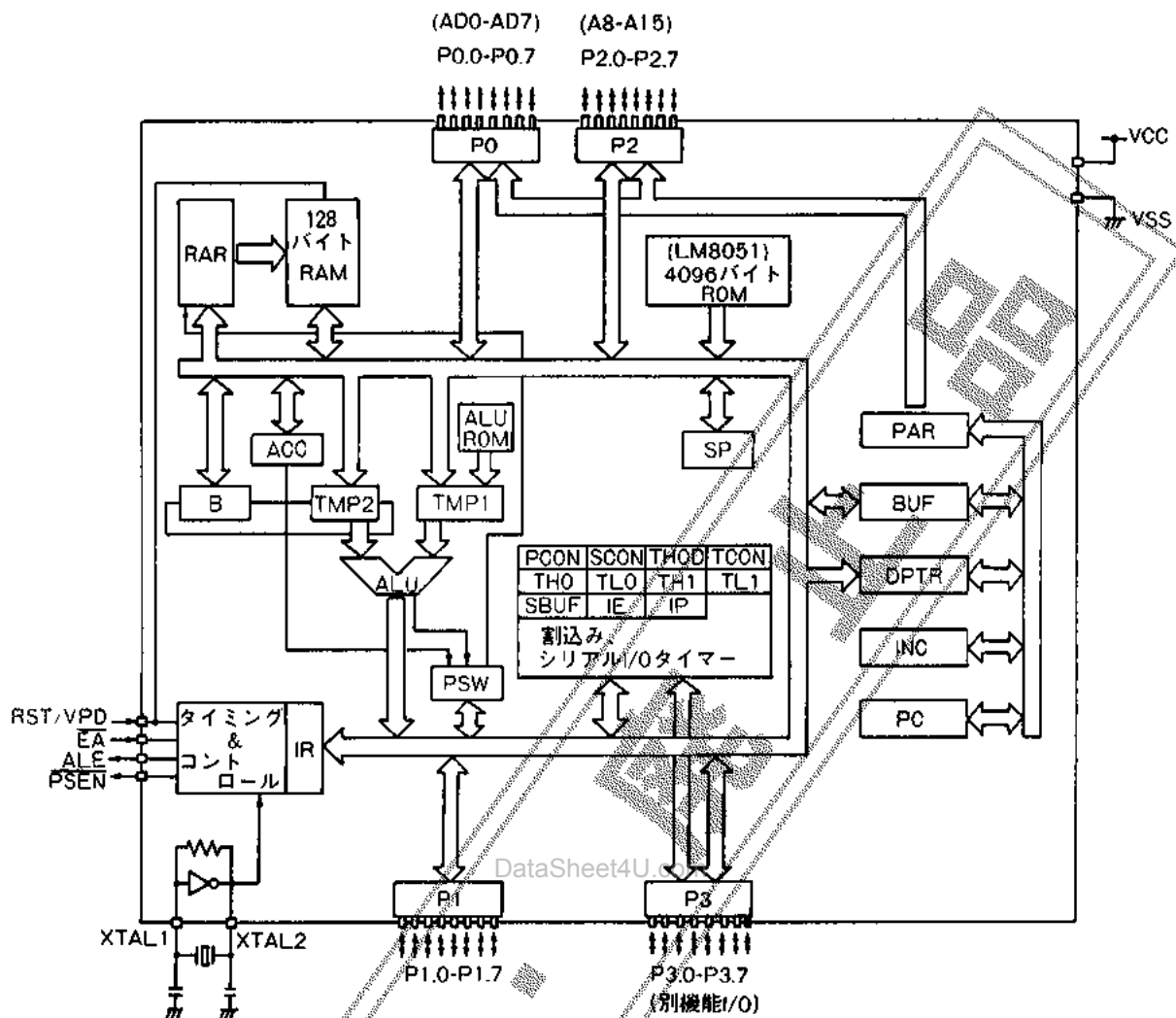
DataSheet4U.com

DataSheet



LM8051, 8031

システムブロック図



* ACC	ACCUMULATOR
* B	B REGISTER
* PSW	PROGRAM STATUS WORD
SP	STACK POINTER
DPTR	DATA POINTER (DPH, DPL)
* P0	PORT0
* P1	PORT1
* P2	PORT2
* P3	PORT3
* IP	INTERRUPT PRIORITY CONTROL
* IE	INTERRUPT ENABLE CONTROL
TMOD	TIMER / COUNTER MODE CONTROL
* TCON	TIMER / COUNTER CONTROL
TH0	TIMER / COUNTER 0 (HIGH BYTE)
TL0	TIMER / COUNTER 0 (LOW BYTE)
TH1	TIMER / COUNTER 1 (HIGH BYTE)
TL1	TIMER / COUNTER 1 (LOW BYTE)

* SCON	SERIAL CONTROL
SBUF	SERIAL DATA BUFFER
PCON	POWER CONTROL
PAR	PROGRAM ADDRESS REGISTER
BUF	ADDRESS BUFFER
INC	ADDRESS INCREMENTER
PC	PROGRAM COUNTER
IR	INSTRUCTION REGISTER
TMP1	TEMPORARY REGISTER1
TMP2	TEMPORARY REGISTER2
RAR	RAM ADDRESS REGISTER

* : ビットアクセス可能

LM8051,8031

端子機能

端 子 名	入出力	機 能 説 明
Vss	—	電源のGNDに接続。
Vcc	入 力	電源端子、通常は+5Vに接続。
P0.0~P0.7 / AD0~AD7	入出力	オープンドレイン出力の8ビットI/Oポート。 外部メモリアクセス時は、下位アドレスとデータの時分割バスとなる(I/Oポートのラッチは“H”となり前データは破壊される)。 8LS TTLが(シンク/ソース共に)ドライブ可能。
P1.0~P1.7	入出力	プルアップ抵抗内蔵の8ビットI/Oポート。 4LS TTLがドライブ可能。
P2.0~P2.7 / A8~A15	入出力	プルアップ抵抗内蔵の8ビットI/Oポート。 外部メモリアクセス時は、上位アドレスを出力。 4LS TTLがドライブ可能。
P3.0~P3.7	入出力	プルアップ抵抗内蔵の8ビットI/Oポート。 4LS TTLがドライブ可能。 また、下記の別機能の入出力端子と共用している(別機能として使用する時、I/Oポートのラッチは“H”に保持すること)。
P3.0 / RXD	入出力	シリアルポート入力端子(モード0で出力端子にもなる)。
P3.1 / TXD	出 力	シリアルポート出力端子。
P3.2 / INT0	入 力	外部割込み要求入力端子。
P3.3 / INT1	入 力	外部割込み要求入力端子。
P3.4 / T0	入 力	タイマ/カウンタ0の外部入力端子。
P3.5 / T1	入 力	タイマ/カウンタ1の外部入力端子。
P3.6 / WR	出 力	外部データメモリ書き込みストローブ信号出力端子。
P3.7 / RD	出 力	外部データメモリ読み込みストローブ信号出力端子。
RST / VPD	入 力	プルダウン抵抗を内蔵したリセット端子(クロック発振時有効)。 少なくとも、2最小命令サイクル(24クロック分)の間、“H”に保持しなければならない。 また、Vccがダウンしても規格内のVPDを供給すれば、RAMの内容は保持される。通常、この端子が“L”レベルの場合は、RAMはVccによって保持される。
ALE Address Latch Enable)	出 力	外部メモリアクセス時、下位アドレスをラッチするタイミングを出力。 周波数は、発振クロックの1/6分周となる。但し、外部データメモリをアクセスする場合のみ1周 期分出力されない。 8LS TTLがドライブ可能。
PSEN Program Store ENable)	出 力	外部プログラムメモリをフェッチする時、メモリの出力イネーブルを制御する負論理の出力。 出力周波数は、発振周波数の1/6分周となる。 内部ROM使用時は、“H”レベルとなる。 8LS TTLがドライブ可能。
EA	入 力	“H”TTLレベルに保持すると、4095番地まで内部ROMをアクセスする。 “L”TTLレベルでは、0番地から外部プログラムメモリのみをアクセスする(LM8031)。
XTAL1	入 力	発振子等による発振動作を行なうための入力端子。 外部クロック使用時は、“L”TTLレベルに固定。
XTAL2	入出力	発振動作を行なうための出力端子であり、内部クロック発生器の入力端子でもある。外部クロック 使用時の入力端子である(但し、ドライバのシンク電流に能力があること)。

LM8051, 8031

レジスタ群の機能
特殊機能レジスタ (SFR)

シンボル	アドレス	ビット操作	機 能 説 明	リセット初期値																
P 0	80H	○	ポート0ラッチ (MSB) <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>P0.7</td><td>P0.6</td><td>P0.5</td><td>P0.4</td><td>P0.3</td><td>P0.2</td><td>P0.1</td><td>P0.0</td> </tr> <tr> <td>87H</td><td>86H</td><td>85H</td><td>84H</td><td>83H</td><td>82H</td><td>81H</td><td>80H</td> </tr> </table> (LSB) 外部メモリアクセス時、強制的にFFHにセットされるため、内容がこわれる。	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	87H	86H	85H	84H	83H	82H	81H	80H	FFH
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0													
87H	86H	85H	84H	83H	82H	81H	80H													
S P	81H		スタックポインタ PUSH, CALL及びPOP, RET命令でインCREMENT, ディCREMENTされる。スタックは、内部RAMに64レベルの深さまで設定可。	07H																
DPTR			データポインタ データメモリ用の16ビットアドレスを保持する。	0000H																
OPL	82H		下位バイト	00H																
DPH	83H		上位バイト	00H																
PCON	87H		パワーコントロールレジスタ (MSB) <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>SMOD</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td> </tr> </table> (LSB) SMOD: シリアルポート用ダブルボーレート制御ビット 0=ダブルレート, 1=1:1のレート	SMOD	-	-	-	-	-	-	-	0XXXXXXX								
SMOD	-	-	-	-	-	-	-													
TCON	88H	○	タイマ/カウンタコントロールレジスタ (MSB) <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>TF1</td><td>TR1</td><td>TF0</td><td>TR0</td><td>IE1</td><td>IT1</td><td>IE0</td><td>IT0</td> </tr> <tr> <td>8FH</td><td>8EH</td><td>8DH</td><td>8CH</td><td>8BH</td><td>8AH</td><td>89H</td><td>88H</td> </tr> </table> (LSB) タイマ/カウンタ 外部割込み TF1 / 0: タイマ/カウンタ1, 0のオーバーフローでセットされ、割込み処理にはいるとリセットされる。 TR1 / 0: タイマ/カウンタ1, 0のスタート制御ビット 0=ストップ, 1=スタート(カウントアップする) IE1 / 0: 外部割込み要求(INT1, 0)の検出でセットされ、割込み処理にはいると立下りエッジ検出モードの時のみリセットされる。 IT1 / 0: 外部割込み要求(INT1, 0)の検出モード制御ビット。 割込み要求信号は、毎サイクルSSP2でストロープされる。 0="L"レベル検出 割込み源は受け付けられるまで"L"レベルを保持し、割込み処理が終わるまでに解除すること。また、IE1/0フラグもクリアしておくこと。 1=立下りエッジ検出 割込み前の"H"レベル及び割込み信号の"L"レベルは、1マシンサイクル(12発振クロック分)以上保持すること。	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	8FH	8EH	8DH	8CH	8BH	8AH	89H	88H	00H
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0													
8FH	8EH	8DH	8CH	8BH	8AH	89H	88H													
TMOD	89H		タイマ/カウンタモードコントロールレジスタ (MSB) <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>GATE</td><td>C/〒</td><td>M1</td><td>M0</td><td>GATE</td><td>C/〒</td><td>M1</td><td>M0</td> </tr> </table> (LSB) タイマ/カウンタ1 タイマ/カウンタ0 GATE: T1, 0入力のゲート制御ビット 0=T1, 0入力パルスが無条件にカウンタにはいる (カウント開始はTCONのTR1, 0を"1"にセット)。 1=T1, 0の入力パルスは、INT1, 0が"H"レベルで、カウンタに入力される。"L"レベルでは入力されない。 C/〒: タイマ、カウンタの切替え制御ビット。 0=タイマ($f_{osc}/12$ クロック入力)。 1マシンサイクルで1カウントアップする。 1=イベントカウンタ(T1, 0入力)。	GATE	C/〒	M1	M0	GATE	C/〒	M1	M0	00H								
GATE	C/〒	M1	M0	GATE	C/〒	M1	M0													

次ページへ続く

LM8051,8031

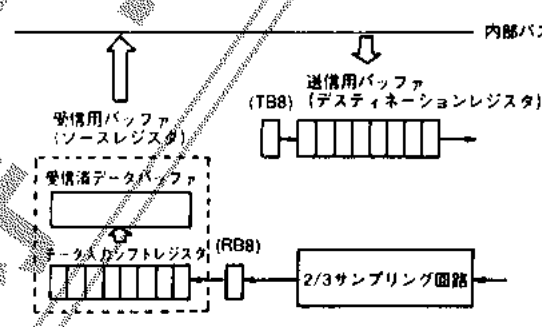
前ページから続く

シンボル	アドレス	ビット操作	機 能 説 明	リセット初期値																									
			M1, M0: タイマ / カウンタモードの設定。 <table border="1"> <thead> <tr> <th>モード</th><th>M1</th><th>M0</th><th>タイマ/カウンタ1</th><th>タイマ / カウンタ 0</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td><td>→</td><td>5ビットプリスケアラ+8ビットタイマ/カウンタ</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>→</td><td>16ビットタイマ/カウンタ</td></tr> <tr> <td>2</td><td>1</td><td>0</td><td>→</td><td>自動初期設定8ビットタイマ/カウンタ</td></tr> <tr> <td>3</td><td>1</td><td>1</td><td>停止モード</td><td>8ビット×2タイマ/カウンタ</td></tr> </tbody> </table>	モード	M1	M0	タイマ/カウンタ1	タイマ / カウンタ 0	0	0	0	→	5ビットプリスケアラ+8ビットタイマ/カウンタ	1	0	1	→	16ビットタイマ/カウンタ	2	1	0	→	自動初期設定8ビットタイマ/カウンタ	3	1	1	停止モード	8ビット×2タイマ/カウンタ	
モード	M1	M0	タイマ/カウンタ1	タイマ / カウンタ 0																									
0	0	0	→	5ビットプリスケアラ+8ビットタイマ/カウンタ																									
1	0	1	→	16ビットタイマ/カウンタ																									
2	1	0	→	自動初期設定8ビットタイマ/カウンタ																									
3	1	1	停止モード	8ビット×2タイマ/カウンタ																									
TL0 TL1 TH0 TH1	8AH 8BH 8CH 8DH		タイマ / カウンタ0 } 下位バイト タイマ / カウンタ1 } タイマ / カウンタ0 } 上位バイト タイマ / カウンタ1 } 動作モード(TMODのM1, 0で設定される)。 ①モード0 (MSB) (LSB) TH1.0 12 11 10 9 8 7 6 5 8ビット タイマ / カウンタ TL1.0 - - - 4 3 2 1 0 5ビット プリスケアラ(1/32) ②モード1 (MSB) (LSB) TH1.0 15 14 13 12 11 10 9 8 16ビット タイマ / カウンタ TL1.0 7 6 5 4 3 2 1 0 ③モード2 (MSB) (LSB) TH1.0 8ビット 初期値 TL1.0 7 6 5 4 3 2 1 0 8ビット タイマ / カウンタ オーバーフロー毎に初期値が自動設定される。 ④モード3 タイマ / カウンタ0 : 2系統の8ビットタイマ / カウンタ (MSB) (LSB) オーバーフロー スタート / ストップ TH0 7 6 5 4 3 2 1 0 TF1 TR1 TL0 7 6 5 4 3 2 1 0 TF0 TR0 タイマ / カウンタ1 : ストップモード(TCONのTR1 = "0"と同じ)。 タイマ / カウンタ1は、オーバーフロー毎にシリアルポートへ1パルス出力する。	00H 00H 00H 00H																									
P 1	90H	0	ポート1 ラッチ (MSB) (LSB) P1.7 P1.6 P1.5 P1.4 P1.3 P1.2 P1.1 P1.0 97H 96H 95H 94H 93H 92H 91H 90H	FFH																									
SCON	98H	0	シリアルI/Oコントロールレジスタ (MSB) (LSB) SM0 SM1 SM2 REN TB8 RB8 TI RI 9FH 9EH 9DH 9CH 9BH 9AH 99H 98H SM0, 1: シリアルポートモードの設定。 <table border="1"> <thead> <tr> <th>モード</th><th>SM0</th><th>SM1</th><th>機 能</th><th>ボーレイト</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td><td>シフトレジスタI/O拡張モード</td><td>fosc/12</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>8ビット非同期R/T(UART)</td><td>可 変</td></tr> <tr> <td>2</td><td>1</td><td>0</td><td>9ビット非同期R/T(UART)</td><td>fosc/64, 32</td></tr> <tr> <td>3</td><td>1</td><td>1</td><td>〃</td><td>可 変</td></tr> </tbody> </table> 可変ボーレイトは、例えばタイマ1のモード2で$(1/2)^{1-5} \times (1/16) \times [1/(256 - (TH1)_{10})] \times (fosc/12)$となる。但し、SはSMODの値、(TH1)₁₀は初期設定デシマル値、foscはXTAL2でのクロック周波数である。	モード	SM0	SM1	機 能	ボーレイト	0	0	0	シフトレジスタI/O拡張モード	fosc/12	1	0	1	8ビット非同期R/T(UART)	可 変	2	1	0	9ビット非同期R/T(UART)	fosc/64, 32	3	1	1	〃	可 変	00H
モード	SM0	SM1	機 能	ボーレイト																									
0	0	0	シフトレジスタI/O拡張モード	fosc/12																									
1	0	1	8ビット非同期R/T(UART)	可 変																									
2	1	0	9ビット非同期R/T(UART)	fosc/64, 32																									
3	1	1	〃	可 変																									

次ページへ続く

LM8051, 8031

前ページから続く

シンボル	アドレス	ビット操作	機 能 説 明	リセット初状態																
			SM2：条件付受信イネーブル制御ビット。 モード2, 3で使用する(他のモードでは"0"とする)。 0＝通常の9ビットデータUART。 1＝RB8が0の場合、RIはセットされず割り込みは生じない。RB8＝1の場合、RIはセットされ、割り込みで受信データが処理できる。 即ち、9ビットのうち、MSB 1ビットが受信制御ビットで、真のデータは8ビットとなる(マルチプロセッサ通信プロトコル用)。 REN：受信イネーブル制御ビット。 1/0＝受信イネーブル/ディセーブル (送信側のイネーブルは、SBUFをデスティネーションレジスタにする命令によってセットされる。) TB8：モード2, 3の9番目の送信ビットを設定する。 RB8：モード2, 3の9番目の受信ビットが設定される。 モード1(必ずSM2＝"0")では、ストップビットがはいり、モード0では変化しない。 TI：送信完了割り込みフラグ。 モード0では、データの8ビット目の送信後、モード1, 2, 3ではストップビット送信開始でセットされる。 次の送信を行なう前に必ずリセットしなければならない(但し、"1"でも送信は行なう)。 RI：受信完了割り込みフラグ モード0ではデータの8ビット目の受信後、モード1, 2, 3では、ストップビット、9ビット目受信中にセットされる。 次の受信を行なう前に必ずリセットしなければならない(但し、"1"でも受信は行なう)。																	
SBUF	99H		シリアルポートバッファ 送信、受信部が独立しており、それぞれデスティネーション、ソースレジスタとしてアクセスされる。また、受信部は、データ入力用シフトレジスタと受信済データ用バッファの2重構造となっており、すぐに割り込み処理でデータを読み出さなくても、次のフレームを受信できる。 	不 定																
P 2	A0H	○	ポート2、ラッチ (MSB) (LSB) <table><tr><td>P2.7</td><td>P2.6</td><td>P2.5</td><td>P2.4</td><td>P2.3</td><td>P2.2</td><td>P2.1</td><td>P2.0</td></tr><tr><td>A7H</td><td>A6H</td><td>A5H</td><td>A4H</td><td>A3H</td><td>A2H</td><td>A1H</td><td>A0H</td></tr></table>	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	A7H	A6H	A5H	A4H	A3H	A2H	A1H	A0H	FFH
P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0													
A7H	A6H	A5H	A4H	A3H	A2H	A1H	A0H													
I E	A8H	○	割り込みイネーブルコントロールレジスタ (MSB) (LSB) <table><tr><td>E A</td><td>—</td><td>—</td><td>E S</td><td>ET1</td><td>EX1</td><td>ET0</td><td>EX0</td></tr><tr><td>A FH</td><td></td><td></td><td>A CH</td><td>A BH</td><td>A AH</td><td>A 9H</td><td>A 8H</td></tr></table> EA：全割り込みイネーブル/ディセーブル制御ビット。 0＝全割り込みディセーブル。 1＝各割り込みイネーブル/ディセーブルビットで制御可。 ES：シリアルポート割り込みイネーブル制御ビット。 1/0＝イネーブル/ディセーブル ET1：タイマ/カウンタ1割り込みイネーブル制御ビット。 1/0＝イネーブル/ディセーブル	E A	—	—	E S	ET1	EX1	ET0	EX0	A FH			A CH	A BH	A AH	A 9H	A 8H	0XX00000
E A	—	—	E S	ET1	EX1	ET0	EX0													
A FH			A CH	A BH	A AH	A 9H	A 8H													

www.Da

次ページへ続く

LM8051, 8031

前ページから続く

シンボル	アドレス	ビット操作	機 能 説 明	リセット初期値																															
			EX1: 外部割込み1 イネーブル制御ビット。 1/0=イネーブル/ディセーブル ET0: タイマ/カウンタ0 割込みイネーブル制御ビット。 1/0=イネーブル/ディセーブル EX0: 外部割込み0 イネーブル制御ビット。 1/0=イネーブル/ディセーブル 各割込みイネーブルの場合、各割込みフラグがセットされてから少なくとも1サイクル(最大5サイクル)後、それぞれのベクタアドレスのLCALL命令を実行し(2サイクル消費)、割込み処理にはいる。 外部割込み0 (IE0) 0003H ★ 割込み処理実行時 タイマ/カウンタ0 (TF0) 000BH (1)自動的にクリアされる割込みフラグ 外部割込み1 (IE1) 0013H • IE1/0(立下りエッジ検出モード時) タイマ/カウンタ1 (TF1) 001BH (2)自動的にクリアされない割込みフラグ シリアルI/O (TI/RI) 0023H • IE1/0(高レベル検出モード時) • TI/RI																																
P 3	B0H	○	ポート3 ラッチ (MSB) (LSB) <table border="1"> <tr> <td>P3.7</td><td>P3.6</td><td>P3.5</td><td>P3.4</td><td>P3.3</td><td>P3.2</td><td>P3.1</td><td>P3.0</td> </tr> <tr> <td>B7H</td><td>B6H</td><td>B5H</td><td>B4H</td><td>B3H</td><td>B2H</td><td>B1H</td><td>B0H</td> </tr> </table>	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0	B7H	B6H	B5H	B4H	B3H	B2H	B1H	B0H	FFH															
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0																												
B7H	B6H	B5H	B4H	B3H	B2H	B1H	B0H																												
IP	B8H	○	割込み優先コントロールレジスタ (MSB) (LSB) <table border="1"> <tr> <td>-</td><td>-</td><td>-</td><td>PS</td><td>PT1</td><td>PX1</td><td>PT0</td><td>PX0</td> </tr> <tr> <td>-</td><td>-</td><td>-</td><td>B8H</td><td>B9H</td><td>BAH</td><td>B9H</td><td>B8H</td> </tr> </table> PS: シリアルI/O 割込み優先レベル制御ビット 1/0=高優先レベル/低優先レベル PT1: タイマ/カウンタ1 割込み優先レベル制御ビット 1/0=高優先レベル/低優先レベル PX1: 外部割込み1 割込み優先レベル制御ビット 1/0=高優先レベル/低優先レベル PT0: タイマ/カウンタ0 割込み優先レベル制御ビット 1/0=高優先レベル/低優先レベル PX0: 外部割込み0 割込み優先レベル制御ビット 1/0=高優先レベル/低優先レベル 但し、同一レベル内での優先度は 外部割込み0 (高) タイマ/カウンタ0 外部割込み1 タイマ/カウンタ1 シリアルI/O (低) の順になる。	-	-	-	PS	PT1	PX1	PT0	PX0	-	-	-	B8H	B9H	BAH	B9H	B8H	xxx00000															
-	-	-	PS	PT1	PX1	PT0	PX0																												
-	-	-	B8H	B9H	BAH	B9H	B8H																												
PSW	D0H	○	プログラムステータスワード (MSB) (LSB) <table border="1"> <tr> <td>CY</td><td>AC</td><td>F0</td><td>RS1</td><td>RS0</td><td>OV</td><td>-</td><td>P</td> </tr> <tr> <td>D7H</td><td>D6H</td><td>D5H</td><td>D4H</td><td>D3H</td><td>D2H</td><td>-</td><td>D0H</td> </tr> </table> CY: キャリーフラグ AC: 補助キャリーフラグ F0: ユーザフラグ RS1, 0: レジスタバンク選択制御ビット <table border="1"> <tr> <th>RS1</th><th>RS0</th><th>ワーキング レジスタ バンク</th> </tr> <tr> <td>0</td><td>0</td><td>バンク0 (00H-07H)</td> </tr> <tr> <td>0</td><td>1</td><td>バンク1 (08H-0FH)</td> </tr> <tr> <td>1</td><td>0</td><td>バンク2 (10H-17H)</td> </tr> <tr> <td>1</td><td>1</td><td>バンク3 (18H-1FH)</td> </tr> </table> OV: オーバーフローフラグ P: パリティフラグ アキュムレータ(ACC)のデータによって更新される。 データ内で"1"が偶数個の場合、セットされる(偶数パリティ)。	CY	AC	F0	RS1	RS0	OV	-	P	D7H	D6H	D5H	D4H	D3H	D2H	-	D0H	RS1	RS0	ワーキング レジスタ バンク	0	0	バンク0 (00H-07H)	0	1	バンク1 (08H-0FH)	1	0	バンク2 (10H-17H)	1	1	バンク3 (18H-1FH)	000000xc
CY	AC	F0	RS1	RS0	OV	-	P																												
D7H	D6H	D5H	D4H	D3H	D2H	-	D0H																												
RS1	RS0	ワーキング レジスタ バンク																																	
0	0	バンク0 (00H-07H)																																	
0	1	バンク1 (08H-0FH)																																	
1	0	バンク2 (10H-17H)																																	
1	1	バンク3 (18H-1FH)																																	

LM8051, 8031

前ページから続く

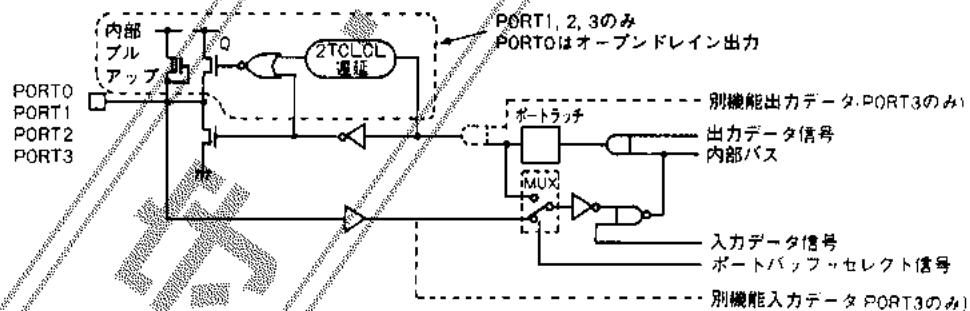
シンボル	アドレス	ビット操作	機 能 説 明	リセット初期値								
ACC	E0H	○	アキュムレータ (MSB) <table border="1"><tr><td>E7H</td><td>E6H</td><td>E5H</td><td>E4H</td><td>E3H</td><td>E2H</td><td>E1H</td><td>E0H</td></tr></table> (LSB)	E7H	E6H	E5H	E4H	E3H	E2H	E1H	E0H	00H
E7H	E6H	E5H	E4H	E3H	E2H	E1H	E0H					
B	F0H	○	Bレジスタ (MSB) <table border="1"><tr><td>F7H</td><td>F6H</td><td>F5H</td><td>F4H</td><td>F3H</td><td>F2H</td><td>F1H</td><td>F0H</td></tr></table> (LSB) 乗除算のとき、アキュムレータと対となり使われるレジスタである。 乗除算以外では、汎用レジスタとして使用可。	F7H	F6H	F5H	F4H	F3H	F2H	F1H	F0H	00H
F7H	F6H	F5H	F4H	F3H	F2H	F1H	F0H					

その他の内部レジスタとして、

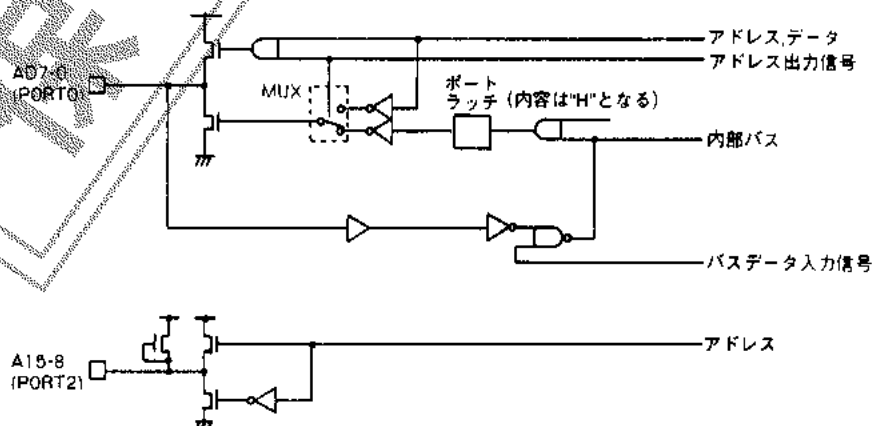
PAR	プログラムアドレスレジスタ	
BUF	アドレスバッファ	
INC	アドレスインクレメンタ	
PC	プログラムカウンタ	リセット初期値 0000H)
IR	インストラクションレジスタ	リセット初期値 00H, NOP)
TMP1	テンポラリーレジスタ1	リセット初期値 00H)
TMP2	2	リセット初期値 FFH)
RAR	RAMアドレスレジスタ	

があるが、ユーザは直接利用することはできない。

I/Oポート形式

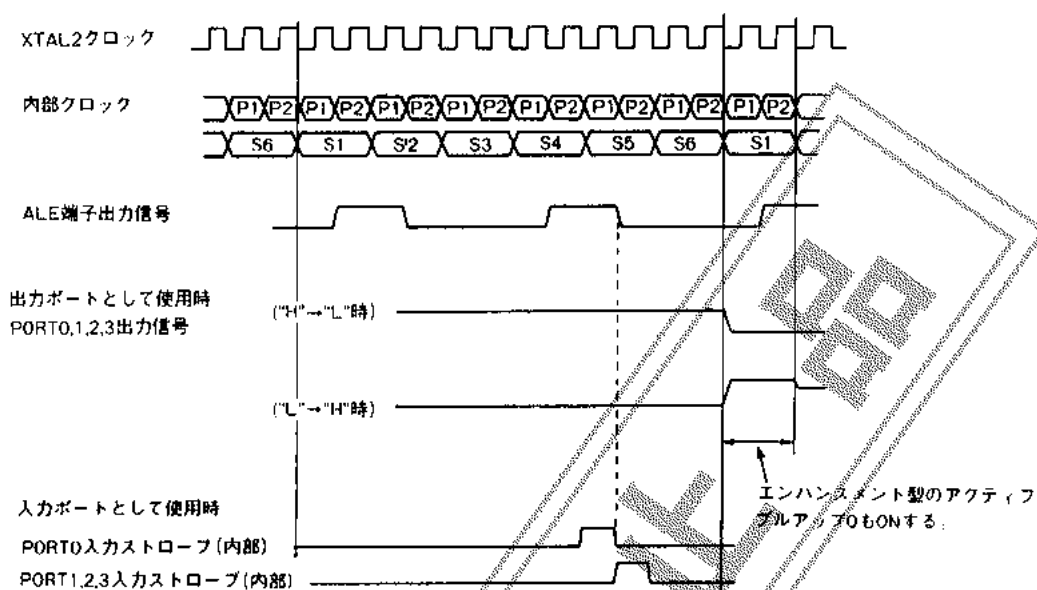


但し、外部メモリアクセス時 PORT0, PORT2は下記の形式になる。

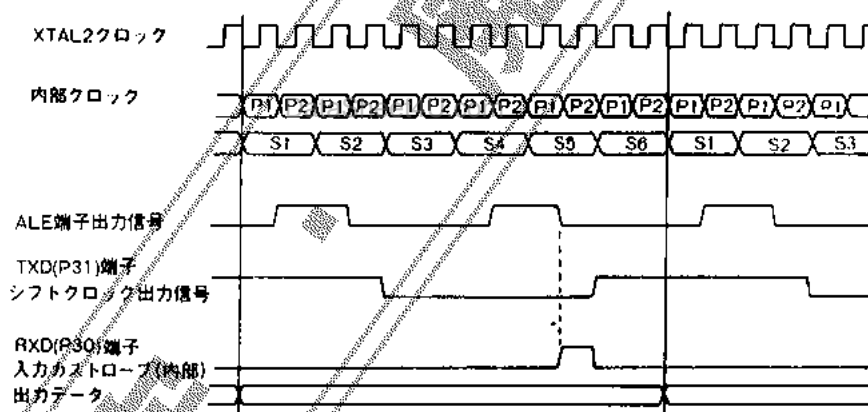


LM8051,8031

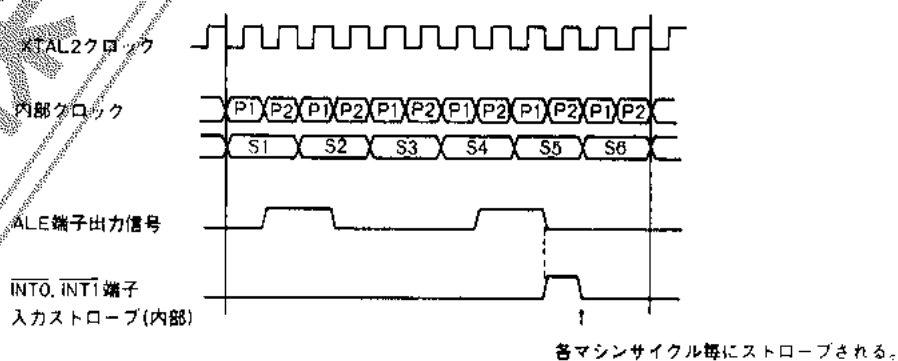
I/Oポートの入出力タイミングチャート



シリアルポート(モード0、同期式の場合)の入出力タイミングチャート



外部割込み要求信号の入カタイミングチャート



LM8051, 8031

絶対最大定格 / $T_a = 25^\circ\text{C}$, $V_{SS} = 0\text{V}$

			unit
最大電源電圧	$V_{CC\text{ max}}$	$-0.5 \sim +7.0$	V
最大入力電圧	$V_{IN\text{ max}}$	$-0.5 \sim +7.0$	V
最大出力電圧	$V_{OUT\text{ max}}$	$-0.5 \sim +7.0$	V
動作周囲温度	T_{opg}	$0 \sim +70$	$^\circ\text{C}$
保存周囲温度	T_{stg}	$-65 \sim +150$	$^\circ\text{C}$
許容消費電力	$P_d\text{ max}$ $T_a = 0^\circ\text{C}$	2	W

許容動作範囲 / $T_a = 0 \sim +70^\circ\text{C}$, $V_{SS} = 0\text{V}$

			min	typ	max	unit
電源電圧	V_{CC}		4.75	5.0	5.25	V
セラミック発振						
外付け容量	C1	XTAL1側, 12MHz発振, 図1 参照		47		pF
	C2	XTAL2側, 12MHz発振, 図1 参照		47		pF

電気的特性

直流特性 / $T_a = 0 \sim +70^\circ\text{C}$, $V_{CC} = 4.75 \sim 5.25\text{V}$, $V_{SS} = 0\text{V}$

			min	typ	max	unit
入力"L"レベル電圧	V_{IL}		-0.5		0.8	V
入力"H"レベル電圧	V_{IH}	RST/VPD, XTAL2以外の入力端子	2.0	$V_{CC} + 0.5$		V
	V_{IH1}	RST/VPD (リセット時), XTAL2端子	2.5	$V_{CC} + 0.5$		V
パワーダウン電圧	V_{PD}	RST/VPD端子 (パワーダウン時) $V_{CC} = 0\text{V}$	4.5		5.5	V
出力"L"レベル電圧	V_{OL}	ポート1, 2, 3端子, $I_{OL} = 1.8\text{mA}$			0.45	V
(注1)	$V_{OL'}$	ポート0, ALE, $\overline{\text{PSEN}}$ 端子, $I_{OL} = 3.2\text{mA}$			0.45	V
出力"H"レベル電圧	V_{OH}	ポート1, 2, 3端子, $I_{OH} = -80\mu\text{A}$	2.4			V
	V_{OH1}	ポート0, ALE, $\overline{\text{PSEN}}$ 端子, $I_{OH} = -400\mu\text{A}$	2.4			V
入力"L"レベル電流	I_{IL}	ポート1, 2, 3端子, $V_{IN} = 0.45\text{V}$	-0.8			mA
	I_{IL2}	XTAL2端子, XTAL1= V_{SS} , $V_{IN} = 0.45\text{V}$	-2.5			mA
入力"H"レベル電流	I_{IH1}	RST/VPD端子 (リセット時) $V_{IN} = V_{CC} - 1.5\text{V}$			500	μA
入力リーク電流	I_L	ポート0, $\overline{\text{EA}}$ 端子, $0.45\text{V} < V_{IN} < V_{CC}$	-10		+10	μA
消費電流	I_{CC}	全出力開放, 12MHz発振		125	160	mA
パワーダウン電流	I_{PD}	$V_{CC} = 0\text{V}$		10	20	mA
I/O端子容量	C_{IO}	$f_c = 1\text{MHz}$, $T_a = 25^\circ\text{C}$			10	pF

(注1)

アドレス、データ出力時、出力"L"レベル電圧 V_{OL} には、過渡的なシンク電流によるACノイズが生じやすい。外部メモリを使用する場合、アドレスラッチやデータバッファはできる限り近くに配置し、負荷容量を軽減すること。

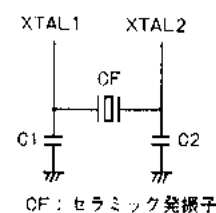


図1. セラミック発振回路

LM8051, 8031

外部クロック駆動特性 / $T_a = 0 \sim +70^\circ\text{C}$, $V_{CC} = 4.75 \sim 5.25\text{V}$, $V_{SS} = 0\text{V}$ (XTAL2入力, XTAL1 = V_{SS})

		min	max	unit
クロック周期 (注2)	TCLCL	83.3	833.3	ns
"H"レベルクロックパルス幅	TCHCX	20		ns
"L"レベルクロックパルス幅	TCLCX	20		ns
立上り時間	TCLCH		20	ns
立下り時間	TCHCL		20	ns

(注2) クロック周波数で1.2MHz~12MHzである。

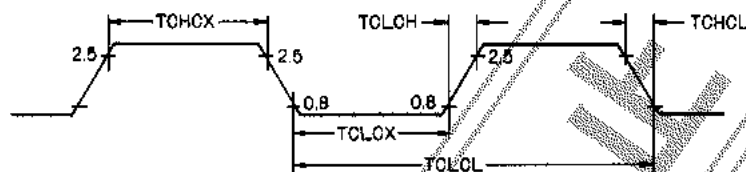


図2. XTAL2入力波形

交流特性 / $T_a = 0 \sim +70^\circ\text{C}$, $V_{CC} = 4.75 \sim 5.25\text{V}$, $V_{SS} = 0\text{V}$ ポート0, ALE, $\overline{\text{PSEN}}$ に対する負荷容量 $C_L = 100\text{pF}$ ポート0, ALE, $\overline{\text{PSEN}}$ 以外 $C_L = 80\text{pF}$

外部プログラムメモリタイミング

		12MHzクロック		1.2MHz~12MHzクロック (1/TCLCL = 1.2M~12MHz)		unit
		min	max	min	max	
ALEパルス幅	TLHLL	125		2TCLCL-40		ns
アドレスセットアップ時間	TAVLL	48		TCLCL-40		ns
アドレスホールド時間	TLLAX	48		TCLCL-35		ns
データ遅延時間(ALEより)	TLLIV		233		4TCLCL-100	ns
メモリ出力イネーブル待ち時間	TLLPL	58		TCLCL-25		ns
$\overline{\text{PSEN}}$ パルス幅	TPLPH	215		3TCLCL-35		ns
データ遅延時間($\overline{\text{PSEN}}$ より)	TPLIV		125		3TCLCL-125	ns
データホールド時間	TPXIX	0		0		ns
データフロート時間	TPXIZ		63		TCLCL-20	ns
メモリ出力ディセーブル待ち時間	TPXAV	75		TCLCL-8		ns
メモリフェッチ時間	TAVIV		302		5TCLCL-115	ns
アドレスフロート時間	TAZPL	0		0		ns

LM8051,8031

外部データメモリタイミング

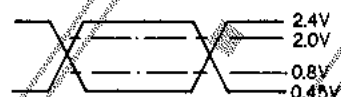
12MHzクロック			(1/TCLCL=1.2M~12MHz)		
		min max	min	max	unit
$\overline{RD}$ パルス幅	TRLRH	400	6TCLCL-100		ns
$\overline{WR}$ パルス幅	TWLWH	400	6TCLCL-100		ns
アドレスホールド時間	TLLAX	132	2TCLCL-35		ns
データ遅延時間($\overline{RD}$ より)	TRLDV	250	5TCLCL-165		ns
データホールド時間($\overline{RD}$ より)	TRHDX	0	0		ns
データフロート時間($\overline{RD}$ より)	TRHDZ	97	2TCLCL-70		ns
データ遅延時間(ALEより)	TLLDV	517	8TCLCL-150		ns
データ遅延時間(アドレスより)	TAVDV	585	9TCLCL-165		ns
メモリ出力イネーブル待ち時間	TLLWL	200 300	3TCLCL-50	3TCLCL+50	ns
アドレスセットアップ時間($\overline{RD}$, $\overline{WR}$ まで)	TAVWL	203	4TCLCL-130		ns
メモリ出力ディセーブル待ち時間	TWHLH	43 123	TCLCL-40	TCLCL+40	ns
データセットアップ時間($\overline{WR}$ まで)	TQVWX	23	TCLCL-60		ns
データセットアップ時間($\overline{WR}$ より)	TQVWH	433	7TCLCL-150		ns
データホールド時間($\overline{WR}$ より)	TWHQX	33	TCLCL-50		ns
アドレスフロート時間($\overline{RD}$ より)	TRLAZ	0	0		ns

et4U.com

(注) ACタイミングの入出力波形のテストポイントについて

DataSheet4U.com

DataShee



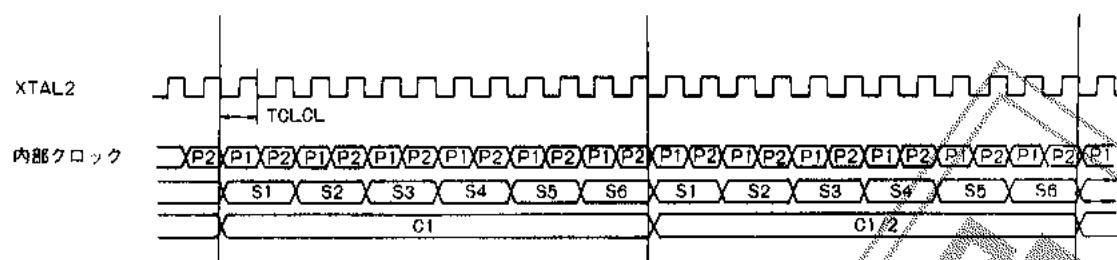
入力波形は、“H”レベルが2.4V、“L”レベルが0.45Vである。

入出力波形のテストポイントは、“H”レベルが2.0V、“L”レベルが0.8Vである。

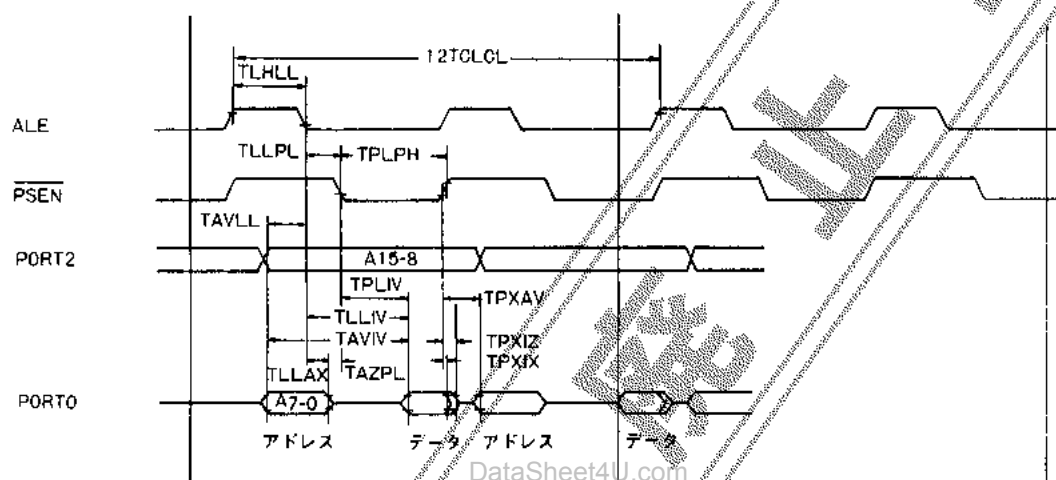
各テストポイントでのソース/シンク電流(P0)は、-400 μ A/2.4mAである。

LM8051, 8031

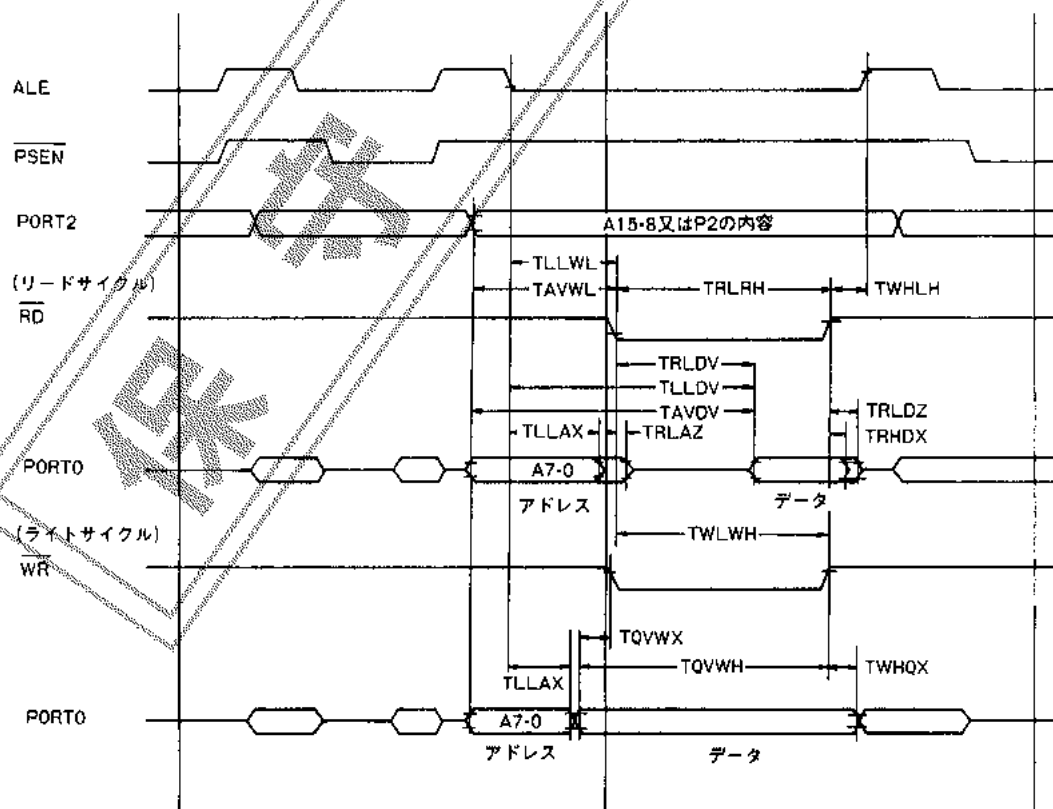
ACタイミングチャート



外部プログラムメモリタイミング



外部データメモリタイミング



LM8051, 8031

LM8051 31命令マップ

命令セット数 111種類

命令サイクル数 実行時間@12MHz発振)

命令バイト数

1サイクル 1usec)命令	64種類(58%)
2サイクル 2usec)命令	45 % (40%)
4サイクル 4usec)命令	2 % (2%)

1バイト命令	49種類(44%)
2バイト命令	46 % (42%)
3バイト命令	16 % (14%)

OPCODE	0	1	2 i=0	3 i=1	4	5	6 i=0	7 i=1	8 n=0	9 n=1	A n=2	B n=3	C n=4	D n=5	E n=6	F n=7
0	NOP	*1 AJMP a11	LJMP a16	RR A	INC A	INC d	INC @Ri								INC Rn	
1	JBC b,r	*2 ACALL a11	LCALL a16	RRC A	DEC A	DEC d	DEC @Ri								DEC Rn	
2	JB b,r	*1	RET	RL A	ADD A,#	ADD A,d	ADD A,@Ri								ADD A,Rn	
3	JNB b,r	*2	RETI	RLC A	ADDC A,#	ADDC A,d	ADDC A,@Ri								ADDC A,Rn	
4	JC r	*1	ORL d,A	ORL d,#	ORL A,#	ORL A,d	ORL A,@Ri								ORL A,Rn	
5	JNC r	*2	ANL d,A	ANL d,#	ANL A,#	ANL A,d	ANL A,@Ri								ANL A,Rn	
6	JZ r	*1	XRL d,A	XRL d,#	XRL A,#	XRL A,d	XRL A,@Ri								XRL A,Rn	
7	JNZ r	*2	ORL C,b	TMP @A+PC	MOV A,#	MOV d,#	MOV @Ri,#								MOV Rn,#	
8	SJMP r	*1	ANL C,b	MOVC A,@A+PC	DIV AB	MOV d,d	MOV d,@Ri								MOV d,Rn	
9	MOV DP,#16	*2	MOVB b,C	MOVC A,@A+DP	SUBB A,#	SUBB A,d	SUBB A,@Ri								SUBB A,Rn	
A	ORL C/b	*1	MOV C,b	INC DP	MUL AB	(NOP)	MOV @Ri,d								MOV Rn,d	
B	ANL C/b	*2	CPL b	CPL C	CJNE A,#,r	CJNE A,d,r	CJNE @Ri,#,r								CJNE Rn,#,r	
C	PUSH d	*1	CLR b	CLR C	SWAP A	XCH A,d	XCH A,@Ri								XCH A,Rn	
D	POP d	*2	SETB b	SETB C	DA A	DJNZ d,r	XCHD A,@Ri								DJNZ Rn,r	
E	MOVX A,@DP	*1		MOVX A,@Ri	CLR A	MOV A,d	MOV A,@Ri								MOV A,Rn	
F	MOVX @DP,A	*2		MOVX @Ri,A	CPL A	MOV d,A	MOV @Ri,A								MOV Rn,A	

注

Rn カレントバンク内のワーキングレジスタ

Ri, r

d 内部RAM及びSFRを指す直接アドレス。

@Ri 内部RAMを指す(カレントバンク内の)

Ri, rの示す間接アドレス。

#, #n イミューデータ(8, 16ビット)。

b, /b 内部RAM及びSFRをビット単位に指す直接アドレス。

/は補数をとる。

a11, a16 プログラムメモリ直接アドレス(11, 16ビット)。

r 相対アドレス(+127, -128バイト)

DP, PC データポインタ(DPTR), プログラムカウンタ

A アキュムレータ

LM8051, 8031

LM8051/31 命令セット一覧表

命令群	ニーモニック	命令コード	バイト数	サイクル数	動作	動作説明	影響を受けるフラグ			
							CY	AC	OV	P
演算命令	ADD A, Rn	0 0 1 0 1 n ₂ n ₁ n ₀	1	1	$A \leftarrow (A) + (Rn), \quad n=0 \sim 7$	Rnの内容をA(Acc)に加算する。	○	○	○	○
	ADD A, d	0 0 1 0 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$A \leftarrow (A) + (d)$	dの示すデータメモリ、SFRの内容をAに加算する。	○	○	○	○
	ADD A, @Ri	0 0 1 0 0 1 1 i	1	1	$A \leftarrow (A) + (Ri), \quad i=0, 1$	Riの示すメモリの内容をAに加算する。	○	○	○	○
	ADD A, #i	0 0 1 0 0 i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	$A \leftarrow (A) + i$	イミディエイトデータをAに加算する。	○	○	○	○
	ADDC A, Rn	0 0 1 1 1 n ₂ n ₁ n ₀	1	1	$A \leftarrow (A) + (C) + (Rn), \quad n=0 \sim 7$	C(CY)とRnの内容をAに加算する。	○	○	○	○
	ADDC A, d	0 0 1 1 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$A \leftarrow (A) + (C) + (d)$	Cとdの示すメモリ、SFRの内容をAに加算する。	○	○	○	○
	ADDC A, @Ri	0 0 1 1 0 1 1 i	1	1	$A \leftarrow (A) + (C) + (Ri), \quad i=0, 1$	CとRiの示すメモリの内容をAに加算する。	○	○	○	○
	ADDC A, #i	0 0 1 1 0 i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	$A \leftarrow (A) + (C) + i$	CとイミディエイトデータをAに加算する。	○	○	○	○
	SUBB A, Rn	1 0 0 1 1 n ₂ n ₁ n ₀	1	1	$A \leftarrow (A) - (C) - (Rn), \quad n=0 \sim 7$	C(ボロー)とRnの内容をAから減算する。	○	○	○	○
	SUBB A, d	1 0 0 1 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$A \leftarrow (A) - (C) - (d)$	Cとdの示すメモリ、SFRの内容をAから減算する。	○	○	○	○
	SUBB A, @Ri	1 0 0 1 0 1 1 i	1	1	$A \leftarrow (A) - (C) - (Ri), \quad i=0, 1$	CとRiの示すメモリの内容をAから減算する。	○	○	○	○
	SUBB A, #i	1 0 0 1 0 i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	$A \leftarrow (A) - (C) - i$	CとイミディエイトデータをAから減算する。	○	○	○	○
命令	INC A	0 0 0 0 0 1 0 0	1	1	$A \leftarrow (A) + 1$	Aの内容をインCREMENTする。				○
	INC Rn	0 0 0 0 1 n ₂ n ₁ n ₀	1	1	$Rn \leftarrow (Rn) + 1, \quad n=0 \sim 7$	Rnの内容をインCREMENTする。				
	INC d *	0 0 0 0 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$(d) \leftarrow (d) + 1$	dの示すメモリの内容をインCREMENTする。				
	INC @Ri	0 0 0 0 0 1 1 i	1	1	$(Ri) \leftarrow (Ri) + 1, \quad i=0, 1$	Riの示すメモリの内容をインCREMENTする。				
	INC DPTR	1 0 1 0 0 0 1 1	1	2	$DPTR \leftarrow (DPTR) + 1$	データポインタの内容をインCREMENTする。				
	DEC A	0 0 0 1 0 1 0 0	1	1	$A \leftarrow (A) - 1$	Aの内容をデCREMENTする。				○
	DEC Rn	0 0 0 1 1 n ₂ n ₁ n ₀	1	1	$Rn \leftarrow (Rn) - 1, \quad n=0 \sim 7$	Rnの内容をデCREMENTする。				
	DEC d *	0 0 0 1 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$(d) \leftarrow (d) - 1$	dの示すメモリの内容をデCREMENTする。				
	DEC @Ri	0 0 0 1 0 1 1 i	1	1	$(Ri) \leftarrow (Ri) - 1, \quad i=0, 1$	Riの示すメモリの内容をデCREMENTする。				
	MUL AB	1 0 1 0 0 1 0 0	1	4	$B \leftarrow (A) \times (B)$	Bの内容をAに乗算し、積の上位はB、下位はAにはいる。	○ (0)		○	○
	DIV AB	1 0 0 0 0 1 0 0	1	4	$B, A \leftarrow (A) / (B)$	Bの内容でAを除算し、商はA、除余はBにはいる。0による除算の場合0Vがセットされる。	○ (0)		○	○
	DA A	1 1 0 0 0 1 0 0	1	1	If (AC)=1 or (A ₃₋₀)>9 then A ₃₋₀ ← (A ₃₋₀) + 6 If (C)=1 or (A ₇₋₄)>9 then A ₇₋₄ ← (A ₇₋₄) + 6	Aの10進補正を行なう。BCD演算結果をパックされたBCDコードに交換する。	○		○	
論理演算命令	ANL A, Rn	0 1 0 1 1 n ₂ n ₁ n ₀	1	1	$A \leftarrow (A) \wedge (Rn), \quad n=0 \sim 7$	Rnの内容とAの論理積をとる。				○
	ANL A, d	0 1 0 1 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$A \leftarrow (A) \wedge (d)$	dの示すメモリ、SFRの内容とAの論理積をとる。				○
	ANL A, @Ri	0 1 0 1 0 1 1 i	1	1	$A \leftarrow (A) \wedge (Ri), \quad i=0, 1$	Riの示すメモリの内容とAの論理積をとる。				○
	ANL A, #i	0 1 0 1 0 i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	$A \leftarrow (A) \wedge i$	イミディエイトデータとAの論理積をとる。				○
	ANL d, A *	0 1 0 1 0 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	$(d) \leftarrow (d) \wedge (A)$	Aとdの示すメモリ、SFRの内容の論理積をとり、メモリ、SFRに入れる。				

次ページへ続く

LM8051, 8031

前ページから続く

命令群	ニーモニック	命令コード	バイト数	サイクル数	動作	動作説明	影響を受けるフラグ			
							CY	AC	OV	P
論理演算	ANL d, #i *	0 1 0 1 0 0 1 1 d7 d6 d5 d4 d3 d2 d1 d0 i7 i6 i5 i4 i3 i2 i1 i0	3	2	(d) ← (d) ∧ i	イミディエイトデータとdの示すメモリ、SFRの内容の論理積をとり、メモリ、SFRに入れる。				
	ORL A, Rn	0 1 0 0 1 n2 n1 n0	1	1	A ← (A) ∨ (Rn), n=0~7	Rnの内容とAの論理和をとる。				○
	ORL A, d	0 1 0 0 0 1 0 1 d7 d6 d5 d4 d3 d2 d1 d0	2	1	A ← (A) ∨ (d)	dの示すメモリ、SFRの内容とAの論理和をとる。				○
	ORL A, @Ri	0 1 0 0 0 1 1 i	1	1	A ← (A) ∨ (Ri), i=0, 1	Riの示すメモリの内容とAの論理和をとる。				○
	ORL A, #i	0 1 0 0 0 1 0 0 i7 i6 i5 i4 i3 i2 i1 i0	2	1	A ← (A) ∨ i	イミディエイトデータとAの論理和をとる。				○
	ORL d, A *	0 1 0 0 0 0 1 0 d7 d6 d5 d4 d3 d2 d1 d0	2	1	(d) ← (d) ∨ (A)	Aとdの示すメモリ、SFRの内容の論理和をとり、メモリ、SFRに入れる。				
	ORL d, #i *	0 1 0 0 0 0 1 1 d7 d6 d5 d4 d3 d2 d1 d0 i7 i6 i5 i4 i3 i2 i1 i0	3	2	(d) ← (d) ∨ i	イミディエイトデータとdの示すメモリ、SFRの内容の論理和をとり、メモリ、SFRに入れる。				
	XRL A, Rn	0 1 1 0 1 n2 n1 n0	1	1	A ← (A) ⊕ (Rn), n=0~7	Rnの内容とAの排他的論理和をとる。				○
	XRL A, d	0 1 1 0 0 1 0 1 d7 d6 d5 d4 d3 d2 d1 d0	2	1	A ← (A) ⊕ (d)	dの示すメモリ、SFRの内容とAの排他的論理和をとる。				○
	XRL A, @Ri	0 1 1 0 0 1 1 i	1	1	A ← (A) ⊕ (Ri), i=0, 1	Riの示すメモリの内容とAの排他的論理和をとる。				○
命令	XRL A, #i	0 1 1 0 0 1 0 0 i7 i6 i5 i4 i3 i2 i1 i0	2	1	A ← (A) ⊕ i	イミディエイトデータとAの排他的論理和をとる。				○
	XRL d, A *	0 1 1 0 0 0 1 0 d7 d6 d5 d4 d3 d2 d1 d0	2	1	(d) ← (d) ⊕ (A)	Aとdの示すメモリ、SFRの内容の排他的論理和をとり、メモリ、SFRに入れる。				
	XRL d, #i *	0 1 1 0 0 0 1 1 d7 d6 d5 d4 d3 d2 d1 d0 i7 i6 i5 i4 i3 i2 i1 i0	3	2	(d) ← (d) ⊕ i	イミディエイトデータとdの示すメモリ、SFRの内容の排他的論理和をとり、メモリ、SFRに入れる。				
	CLR A	1 1 1 0 0 1 0 0	1	1	A ← 0	Aをクリアする。				○
	CPL A	1 1 1 1 0 1 0 0	1	1	A ← (A)	Aの補数をとる。				○
	RL A	0 0 1 0 0 0 1 1	1	1	A7 → A6 → A5 → A4 → A3 → A2 → A1 → A0	Aの内容を1ビット左へローテートする。				
	RLC A	0 0 1 1 0 0 1 1	1	1	A7 → A6 → A5 → A4 → A3 → A2 → A1 → A0 C	Cを含めてAの内容を1ビット左へローテートする。	○			○
	RR A	0 0 0 0 0 0 1 1	1	1	A7 → A6 → A5 → A4 → A3 → A2 → A1 → A0	Aの内容を1ビット右へローテートする。				
	RRC A	0 0 0 1 0 0 1 1	1	1	A7 → A6 → A5 → A4 → A3 → A2 → A1 → A0 C	Cを含めて、Aの内容を1ビット右へローテートする。	○			○
	SWAP A	1 1 0 0 0 1 0 0	1	1	A7 A6 A5 A4 A3 A2 A1 A0	Aの内容の上位ニブルと下位ニブルを交換する。				
ビット操作	CLR C	1 1 0 0 0 0 1 1	1	1	C ← 0	Cをクリアする。	○			
	CLR b *	1 1 0 0 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	1	(b) ← 0	bの示すビットをクリアする。				
	SETB C	1 1 0 1 0 0 1 1	1	1	C ← 1	Cをセットする。	○			
	SETB b *	1 1 0 1 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	1	(b) ← 1	bの示すビットをセットする。				
	CPL C	1 1 1 1 0 0 1 1	1	1	C ← (C)	Cの補数をとる。	○			
	CPL b *	1 1 1 1 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	1	(b) ← (b)	bの示すビットの補数をとる。				
	ANL C, b	1 0 0 0 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	2	C ← (C) ∧ (b)	bの示すビットとCの論理積をとる。	○			
	ANL C, /b	1 0 0 1 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	2	C ← (C) ∧ (b)	bの示すビットの補数とCの論理積をとる。	○			
	ORL C, b	0 1 1 1 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	2	C ← (C) ∨ (b)	bの示すビットとCの論理和をとる。	○			
	ORL C, /b	0 1 1 0 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	2	C ← (C) ∨ (b)	bの示すビットの補数とCの論理和をとる。	○			
命令	MOV C, b	1 0 1 0 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	1	C ← (b)	C ← bの示すビットを転送する。	○			
	MOV b, C *	1 0 0 1 0 0 1 0 b7 b6 b5 b4 b3 b2 b1 b0	2	2	(b) ← (C)	bの示すビットへCを転送する。				

次ページへ続く

LM8051, 8031

前ページから続く

命令群	ニーモニック	命令コード	バイト数	サイクル数	動作	動作説明	影響を受けるフラグ			
							CY	AC	OV	P
データ送命令	MOV A, Rn	1 1 1 0 1 n ₂ n ₁ n ₀	1	1	A ← (Rn), n=0~7	Rnの内容をAに転送する。				○
	MOV A, d	1 1 1 0 0 1 0 1 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	A ← (d)	dの示すメモリ、SFRの内容をAに転送する。				○
	MOV A, @Ri	1 1 1 0 0 1 1 i	1	1	A ← (Ri), i=0, 1	Riの示すメモリの内容をAに転送する。				○
	MOV A, #i	0 1 1 1 0 1 0 0 i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	A ← i	イミディエイトデータをAに転送する。				○
	MOV Rn, A	1 1 1 1 1 n ₂ n ₁ n ₀	1	1	Rn ← (A), n=0~7	Aの内容をRnに転送する。				
	MOV Rn, d	1 0 1 0 1 n ₂ n ₁ n ₀ d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	2	Rn ← (d), n=0~7	dの示すメモリ、SFRの内容をRnに転送する。				
	MOV Rn, #i	0 1 1 1 1 n ₂ n ₁ n ₀ i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	Rn ← i, n=0~7	イミディエイトデータをRnに転送する。				
	MOV d, A	1 1 1 1 0 1 0 1 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	1	(d) ← (A)	Aの内容をdの示すメモリ、SFRに転送する。				
	MOV d, Rn	1 0 0 0 1 n ₂ n ₁ n ₀ d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	2	(d) ← (Rn), n=0~7	Rnの内容をdの示すメモリ、SFRに転送する。				
	MOV d, e	1 0 0 0 0 1 0 1 e ₇ e ₆ e ₅ e ₄ e ₃ e ₂ e ₁ e ₀ d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	3	2	(d) ← (e)	e(直接アドレス)の示すメモリ、SFRの内容をdの示すメモリ、SFRへ転送する。				
データ送命令	MOV d, @Ri	1 0 0 0 0 1 1 i d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	2	(d) ← (Ri), i=0, 1	Riの示すメモリの内容をdの示すメモリSFRへ転送する。				
	MOV d, #i	0 1 1 1 0 1 0 1 d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀ i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	3	2	(d) ← i	イミディエイトデータをdの示すメモリ、SFRへ転送する。				
	MOV @Ri, A	1 1 1 1 0 1 1 i	1	1	(Ri) ← (A), i=0, 1	Aの内容をRiの示すメモリへ転送する。				
	MOV @Ri, d	1 0 1 0 0 1 1 i d ₇ d ₆ d ₅ d ₄ d ₃ d ₂ d ₁ d ₀	2	2	(Ri) ← (d)	dの示すメモリ、SFRの内容をRiの示すメモリへ転送する。				
	MOV @Ri, #i	0 1 1 1 0 1 1 i i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	2	1	(Ri) ← i, i=0, 1	イミディエイトデータをRiの示すメモリへ転送する。				
	MOV DPTR, #i16	1 0 0 1 0 0 0 0 i ₁₅ i ₁₄ i ₁₃ i ₁₂ i ₁₁ i ₁₀ i ₉ i ₈ i ₇ i ₆ i ₅ i ₄ i ₃ i ₂ i ₁ i ₀	3	2	DPTR ← i16	16ビットのイミディエイトデータをデータポインタに転送する。				
	MOVC A, @A+DPTR	1 0 0 1 0 0 1 1	1	2	A ← ((A)+(DPTR))	データポインタをベースとして、Aの相対値を加えた(コード)アドレスの示すプログラムメモリの内容をAに転送する。				○
	MOVC A, @A+PC	1 0 0 0 0 0 1 1	1	2	PC ← (PC)+1 A ← ((A)+(PC))	(インクレメントされたプログラムカウンタをベースとして、Aの相対値を加えた(コード)アドレスの示すプログラムメモリの内容をAに転送する。				○
	MOVX A, @Ri	1 1 1 0 0 0 1 1	1	2	A ← ((P2), (Ri))	上位8ビットはP2、下位8ビットはRiの示す外部データメモリの内容をAに転送する。RDがアクティブになる(P3.7はセットすること)。				○
	MOVX A, @DPTR	1 1 1 0 0 0 0 0	1	2	A ← ((DPTR))	データポインタの示す外部データメモリの内容をAに転送する。RDがアクティブになる(P3.7はセットすること)。				
データ送命令	MOVX @Ri, A	1 1 1 1 0 0 1 i	1	2	((P2), (Ri)) ← (A)	Aの内容を上位8ビットはP2、下位8ビットはRiの示す外部データメモリへ転送する。WRがアクティブになる(P3.6はセットすること)。				
	MOVX @DPTR, A	1 1 1 1 0 0 0 0	1	2	((DPTR)) ← (A)	Aの内容をデータポインタの示す外部データメモリへ転送する。WRがアクティブになる(P3.6はセットすること)。				

次ページから続く

LM8051, 8031

前ページから続く

命令群	ニーモニック	命令コード	バイト数	サイクル数	動作	動作説明	影響を受けるフラグ			
							CY	AC	OV	P
データ転送命令	PUSH d	$\begin{matrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & d_5 & d_4 & d_3 & d_2 & d_1 & d_0 \end{matrix}$	2	2	$SP \leftarrow (SP) + 1$ $((SP)) \leftarrow (d)$	スタックポインタをインクリメントし、dの示す(内部)メモリ、SFRの内容を、スタックポインタの示す(内部)メモリへ転送する。				
	POP d	$\begin{matrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & d_5 & d_4 & d_3 & d_2 & d_1 & d_0 \end{matrix}$	2	2	$(d) \leftarrow ((SP))$ $SP \leftarrow (SP) - 1$	スタックポインタの示す(内部)メモリの内容をdの示す(内部)メモリ、SFRへ転送する。				
	XCH A, Rn	$\begin{matrix} 1 & 1 & 0 & 0 & 1 & n_2 & n_1 & n_0 \end{matrix}$	1	1	$(A) \leftrightarrow (Rn), n=0 \sim 7$	AとRnの内容を交換する。				○
	XCH A, d	$\begin{matrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & d_5 & d_4 & d_3 & d_2 & d_1 & d_0 \end{matrix}$	2	1	$(A) \leftrightarrow (d)$	Aとdの示すメモリ、SFRの内容を交換する。				○
	XCH A, @Ri	$\begin{matrix} 1 & 1 & 0 & 0 & 0 & 1 & 1 & i \end{matrix}$	1	1	$(A) \leftrightarrow (Ri), i=0, 1$	AとRiの示すメモリの内容を交換する。				○
	XCHD A, @Ri	$\begin{matrix} 1 & 1 & 0 & 1 & 0 & 1 & 1 & i \end{matrix}$	1	1	$(A)_{3:0} \leftrightarrow (Ri)_{3:0}, i=0, 1$	AとRiの示すメモリの下位ニブルをそれぞれ交換する。				○
サブルーチン命令	ACALL a ₁₁	$\begin{matrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & a_{10} & a_9 & a_8 & a_7 & a_6 & a_5 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ $SP \leftarrow (SP) + 1$ $((SP)) \leftarrow (PC)_{7:0}$ $SP \leftarrow (SP) + 1$ $((SP)) \leftarrow (PC)_{15:8}$ $PC_{10:0} \leftarrow a_{11}$ (ページアドレス)	ページ内絶対コール。プログラムカウンタの内容(リターンアドレス)をスタックポインタの示すスタックへ退避させ、指定されたページ内のサブルーチンアドレスへとぶ。				
	LCALL a ₁₆	$\begin{matrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & a_{15} & a_{14} & a_{13} & a_{12} & a_{11} & a_{10} \end{matrix}$ $\begin{matrix} 0 & 0 & a_9 & a_8 & a_7 & a_6 & a_5 & a_4 \end{matrix}$ $\begin{matrix} 0 & 0 & a_3 & a_2 & a_1 & a_0 & 0 & 0 \end{matrix}$	3	2	$PC \leftarrow (PC) + 3$ $SP \leftarrow (SP) + 1$ $((SP)) \leftarrow (PC)_{7:0}$ $SP \leftarrow (SP) + 1$ $((SP)) \leftarrow (PC)_{15:8}$ $PC \leftarrow a_{16}$	ロングコール。プログラムカウンタの内容(リターンアドレス)をスタックポインタの示すスタックへ退避させ、指定されたサブルーチンアドレスへとぶ。				
	RET	$\begin{matrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{matrix}$	1	2	$PC_{7:0} \leftarrow ((SP))$ $SP \leftarrow (SP) - 1$ $PC_{15:8} \leftarrow ((SP))$ $SP \leftarrow (SP) - 1$	サブルーチンからリターンする。スタックポインタの示すスタックの先頭2バイトをプログラムカウンタに転送する。				
	RETI	$\begin{matrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \end{matrix}$	1	2	$PC_{15:8} \leftarrow ((SP))$ $SP \leftarrow (SP) - 1$ $PC_{7:0} \leftarrow ((SP))$ $SP \leftarrow (SP) - 1$	割り込み処理ルーチンからリターンする。スタックポインタの示すスタックの先頭2バイトをプログラムカウンタに転送し、割り込み制御フラグの内容を割り込み前に戻し、保留中の同じか低レベルの割り込み、又は、新しい同レベルの割り込みを受け付け可能とする(割り込みは、RETI命令後、少なくとも1命令実行後可能)。				
	AJMP a ₁₁	$\begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & a_{10} & a_9 & a_8 & a_7 & a_6 & a_5 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ $PC_{10:0} \leftarrow a_{11}$	ページ内無条件絶対ジャンプ。指定されたページ内のアドレスへ無条件にとぶ。				
	LJMP a ₁₆	$\begin{matrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & a_{15} & a_{14} & a_{13} & a_{12} & a_{11} & a_{10} \end{matrix}$ $\begin{matrix} 0 & 0 & a_9 & a_8 & a_7 & a_6 & a_5 & a_4 \end{matrix}$ $\begin{matrix} 0 & 0 & a_3 & a_2 & a_1 & a_0 & 0 & 0 \end{matrix}$	3	2	$PC \leftarrow a_{16}$	無条件ロングジャンプ。指定されたアドレスへ無条件にとぶ。				
条件命令	SJMP r	$\begin{matrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ r_7 & r_6 & r_5 & r_4 & r_3 & r_2 & r_1 & r_0 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ $PC \leftarrow (PC) + r$	無条件ショートジャンプ。現プログラムカウンタから相対アドレスrへ無条件にとぶ。				
	JMP @A+DPTR	$\begin{matrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{matrix}$	1	2	$PC \leftarrow (A) + (DPTR)$	データポインタをベースとして、Aの内容を相対値とするアドレスへ無条件にとぶ。				
	JZ r	$\begin{matrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ r_7 & r_6 & r_5 & r_4 & r_3 & r_2 & r_1 & r_0 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ If (A)=0, then $PC \leftarrow (PC) + r$	もし、Aが0なら、現PCより相対アドレスrへとぶ。				
	JNZ r	$\begin{matrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ r_7 & r_6 & r_5 & r_4 & r_3 & r_2 & r_1 & r_0 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ If (A)≠0, then $PC \leftarrow (PC) + r$	もし、Aが0でなければ、現PCより相対アドレスrへとぶ。				
	JC r	$\begin{matrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ r_7 & r_6 & r_5 & r_4 & r_3 & r_2 & r_1 & r_0 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ If (C)=1, then $PC \leftarrow (PC) + r$	もし、Cが1なら、現PCより相対アドレスrへとぶ。				
	JNC r	$\begin{matrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ r_7 & r_6 & r_5 & r_4 & r_3 & r_2 & r_1 & r_0 \end{matrix}$	2	2	$PC \leftarrow (PC) + 2$ If (C)≠1, then $PC \leftarrow (PC) + r$	もし、Cが1でなければ、現PCより相対アドレスrへとぶ。				

次ページへ続く

LM8051, 8031

前ページから続く

命令群	二一モニツク	命令コード	バイト数	サイクル数	動作	動作説明	影響を受けるフラグ			
							CY	AC	OV	P
	JB b, r	0 0 1 0 b7 b6 b5 b4 r7 r6 r5 r4	0 0 0 0 b3 b2 b1 b0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (b) = 1, then PC ← (PC) + r	もし、bの示すビットが1なら、現PCより相対アドレスrへとぶ。			
	JNB b, r	0 0 1 1 b7 b6 b5 b4 r7 r6 r5 r4	0 0 0 0 b3 b2 b1 b0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (b) ≠ 1, then PC ← (PC) + r	もし、bの示すビットが1でなければ、現PCより相対アドレスrまでとぶ。			
	JBC b, r *	0 0 0 1 b7 b6 b5 b4 r7 r6 r5 r4	0 0 0 0 b3 b2 b1 b0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (b) = 1, then PC ← (PC) + r (b) ← 0	もし、bの示すビットが1なら、現PCより相対アドレスrへとぶ。bの示すビットをクリアする。			
分岐	CJNE A, d, r	1 0 1 1 d7 d6 d5 d4 r7 r6 r5 r4	0 1 0 1 d3 d2 d1 d0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (A) ≠ (d), then PC ← (PC) + r If (A) < (d), then C ← 1 else C ← 0	Aの内容とdの示すメモリ、SFRの内容を比較し、等しくなければ現PCより相対アドレスrへとぶ。(A) < (d)の場合Cをセットし、(A) ≥ (d)の場合リセットする。	○		
比較	CJNE A, #i, r	1 0 1 1 i7 i6 i5 i4 r7 r6 r5 r4	0 1 0 0 i3 i2 i1 i0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (A) ≠ i, then PC ← (PC) + r If (A) < i, then C ← 1 else C ← 0	Aの内容と、イミディエイトデータを比較し、等しくなければ現PCより相対アドレスrへとぶ。(A) < iの場合Cをセットし、(A) ≥ iの場合リセットする。	○		
比較	CJNE Rn, #i, r	1 0 1 1 i7 i6 i5 i4 r7 r6 r5 r4	1 n2 n1 n0 i3 i2 i1 i0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (Rn) ≠ i, then PC ← (PC) + r If (Rn) < i, then C ← 1 else C ← 0	Rnの内容と、イミディエイトデータを比較し、等しくなければ、現PCより相対アドレスrへとぶ。(Rn) < iの場合、Cをセットし、(Rn) ≥ iの場合リセットする。	○		
比較	CJNE @Ri, #i, r	1 0 1 1 i7 i6 i5 i4 r7 r6 r5 r4	0 1 1 1 i3 i2 i1 i0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 If (Ri) ≠ i, then PC ← (PC) + r If (Ri) < i, then C ← 1 else C ← 0	Riの示すメモリの内容とイミディエイトデータを比較し、等しくなければ現PCより相対アドレスrへとぶ。(Ri) < iの場合、Cをセットし、(Ri) ≥ iの場合リセットする。	○		
	DJNZ Rn, r	1 1 0 1 r7 r6 r5 r4	1 n2 n1 n0 r3 r2 r1 r0	2	2	PC ← (PC) + 2 Rn ← (Rn) - 1 If (Rn) ≠ 0, then PC ← (PC) + r	Rnをデクレメントし、結果が0でなければ、現PCより相対アドレスrへとぶ。			
	DJNZ d, r *	1 1 0 1 d7 d6 d5 d4 r7 r6 r5 r4	0 1 0 1 d3 d2 d1 d0 r3 r2 r1 r0	3	2	PC ← (PC) + 3 d ← (d) - 1 If (d) ≠ 0, then PC ← (PC) + r	dの示すメモリ、SFRをデクレメントし、結果が0でなければ、現PCより相対アドレスrへとぶ。			
その他	NOP	0 0 0 0	0 0 0 0	1	1	No operation	なにもせず、1マシンサイクル消費する。			

(注) *印の命令のポートに関する(バイト又はビット単位の)直接アドレッシングでは、ポートラッチ、その他の命令では、ポートに印加される外部信号が選択される。これは、I/Oポート形式に示すポートバッファセレクト信号で制御される。

記号説明

A アキュムレータ

Rn, Ri(j) カレントバンク内のワーキングレジスタ(内部RAMの一部)

d, e 直接アドレスデータ

i イミディエイトデータ

b 直接ビットアドレスデータ

r 相対アドレスデータ(−128(番地) ≤ r ≤ +127(番地))

C キャリ

() 内容を示す。

例えば、(Ri)はレジスタRiの内容をアドレスとするメモリの内容を示す。

LM8051,8031

内部RAM及び特殊機能レジスタのアドレス空間

