

AVR242 : LED駆動と4×4キーパッドを多重化する 8ビット マイクロコントローラ

要点

- 4×4配列での16キー押釦パッド
- 点滅するコン(・)付き4桁の多重化されたLED表示器
- 生産用実時間時計/計時器
- 2つの負荷に対するON/OFF計時器制御
- 圧電スピーカ経由の(キー操作)触覚帰還
- 電源断現象を示す点滅表示
- 2重機能入出力ピン
- 最小の外部部品
- 効率的なコード
- AT90S1200用の完全なプログラムを含む
- 20ピンまたはそれ以上を持つどのAVR MCUにも適合

序説

この応用記述は2つの出力を持つ実時間時計/計時器への入力として4×4キーパッドを提供する包括的なシステムを記述します。このシステムは外部負荷と多重化された4桁のLED表示器を制御します。この応用はAVRポート構成設定の多様性と豊富な命令群の効率性を示すように設計されています。応用は20ピンまたはそれ以上のピンを持つどのAVRでも動きますが、とは言え当然スタック初期化と表配置に対して考慮されなければなりません。プログラムはAT90S1200での3段の深さのハードウェア スタックの制限内で構成され、ソフトウェア スタックを持つ他のAVRではもっと上手く構成することができます。

動作の理屈

4×4キーパッド、圧電スピーカ、2つのLED負荷、多重化された4桁LED表示器の接続は一般的に23本の入出力線が必要です。この応用は少しの工夫でこれがどう15に減らせるかを示し、より少ない20ピンAVRでの使用を可能にします。回路構成図は図1.で示され、発振器部品は完全に別物として明確化のために省略されています。

キーパッドの4つの列はポートBの下位ニブルに接続され、キーパッドの4つの行は上位ニブルに接続されます。同じ8ビットがR13～20の電流制限抵抗経由で4桁LED表示器セグメントのカソードも直接駆動します。従ってピンはLED表示器駆動時の出力としての働きと、キーパッド走査時の入出力の2つ機能を扱います。これは設定可能な性質と良い結果を齎すAVRポートの大電流駆動能力を使うことによって達成されます。

時間の大部分、ポートBはLEDセグメントを直接駆動するために9mAの電流を引き込みます。Q1～4のPNPトランジスタ経由で表示器を多重化するために、各桁は5msの時間幅で連続的に切り換えられます。LED表示器桁の共通アノードは最大72mA(9mA×8セグメント)になり得る電流がポートの処理能力の範囲を越えるため、PNPトランジスタ経由で駆動されます。

100mAまたはそれ位の駆動能力のどのPNP型(例えば、BC479)も使うことができます。これは電流を分け合うために各アノードに対して2つのポートピンを並列動作にすることによって変更できますが、要求される入出力ピン数はより大きなMCUの使用を必要とするでしょう。

各表示周回を始める前に、キーパッドを走査するため、ポートの構成設定が内部プルアップが許可された4入力とLow状態の4出力を提供するように変更されます。キーが押されている場合、変数に格納されたキー番号とでキー値を算定するためにニブルの構成設定が置き換えられます。短遅延がポートを安定させることを各ポート変更間に許します。本応用に於いてこの方法は伝統的な“snake”法よりもずっと効率的なコードです。

共通アノードの駆動は妨害を避けるためにこの時間中、禁止されます。そして多重化ルーチンの準備のためにポートの構成設定が再初期化されます。そして適切な動きを取るために主家事ルーチンがこのキー変数を使います。

実時間時計は256分周されたシステム クロックでクロック駆動されるタイマ/カウンタ0を使う割り込み駆動です。この計時器は176の数値で予め設定され、そして5ms毎の溢れで割り込み、良好な品質のクリスタルが使われた場合に高い精度を保証します。正確さのために4.096MHzのクロック用クリスタルが使われます。プログラムは少しの変更で4MHzクリスタルを使うように変更することができます。



8ビット AVR[®]
マイクロコントローラ

応用記述

本書は一般の方々の便宜のため有志により作成されたもので、Atmel社とは無関係であることを御承知ください。しおりのはじめにでの内容にご注意ください。

Rev. 1231B-05/02, 1231BJ3-03/21

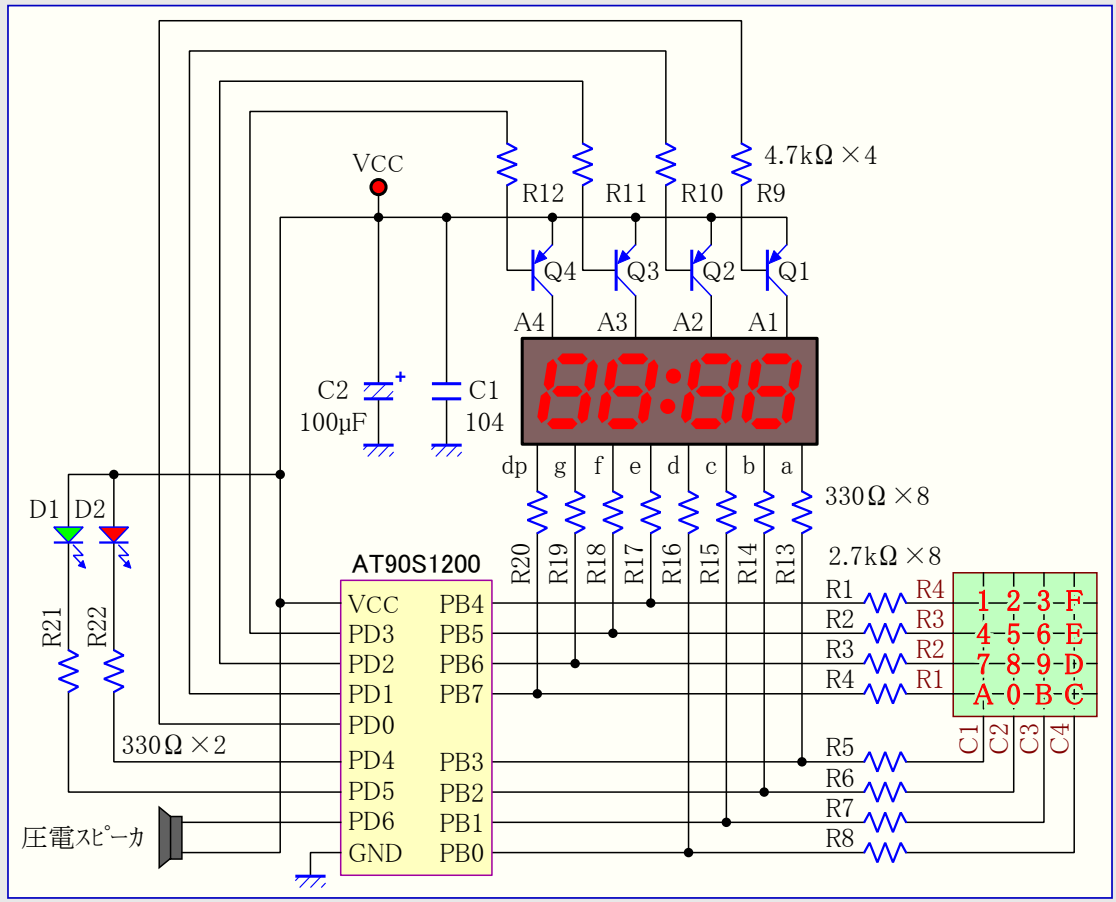
割り込み処理ルーチンは計時器を再格納し、計数器変数(`tock`(5ms計数器))、キーパッド反動(チャタリング)変数(`bouce`)、秒数を維持するための計数器(`second`)の3つの変数を増やします。これは`display`関数によって順番に表示される、分と時を更新するために主家事機能によって使われます。

主家事機能はONまたはOFFの時間に関して2つの負荷を調べ、それによってポートDの上位ニブルの出力を制御します。この応用では電流引き込み構成設定(Low活性)で駆動される赤と緑のLEDによって擬似実装されます。これらは電力負荷を駆動するための継電器駆動部または光結合トリアックによって置き換えることができます。

キーパッドは実時間(SET)と各負荷のON/OFF時間の設定、また1度に負荷をOFFする(CLEAR)ことも可能にする方法を提供します。ポートDの最上位ビットに接続された圧電スピーカはキー押下での可聴通告音を提供します。

ポートBピンの使用はいくつかの注意を要する考慮が必要です。ピンが2つの機能に対して使われるため、キーが押されている場合にそれが表示器を外で短絡しないことが重要です。これは各キーと直列に電流制限抵抗を配置することによって達成されます。入力として使われる時に内部プルアップ抵抗が使われ、外部部品を省きます。抵抗値の選択がこのようなことなので、潜在的な分割は無視できません。選択した値と5V供給に於いて、論理レベルは論理'0'に対して約0.6V、論理'1'に対して4.95Vです。R21とR22の抵抗はLEDに対する伝統的な電流制限抵抗で、供給電圧に対して適切などの値にもできます。この記述は5V供給で330Ωを使って試験されました。LEDは電流引き込み動作('0'=ON)で駆動され、指定された値で約9mAの順方向電流を提供します。

図1. キーパッド/表示部用回路構成図



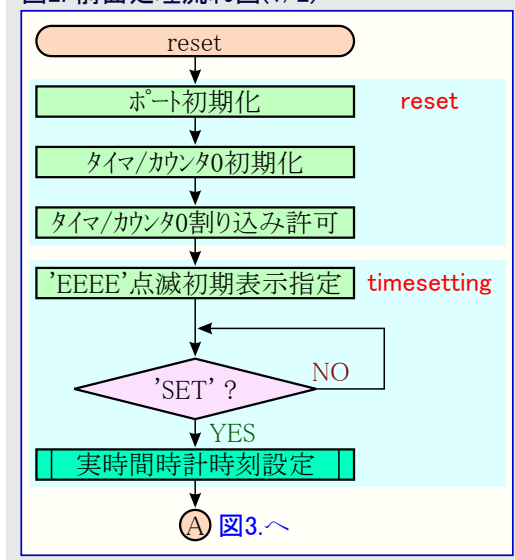
実装

ファームウェアは割り込み駆動で正確な実時間を提供する背面機能と前面処理の2つの主な部分を含みます。これらはポート、タイマ/カウンタと割り込みを設定するリセットルーチン、時間設定ルーチン、主家事機能の3つの部分から成ります。

前面処理

前面処理は殆どの時間で動いていて、実時間時計変数を更新するために5ms毎に5.127 μ s(21周期)の間だけ割り込まれます。それはreset、timesetting、housekeepingの3つの部分から成ります。流れ図は図2.で示されます。

図2. 前面処理流れ図(1/2)



reset

電源投入またはリセット条件で、システムのハードウェアを初期化するためにリセットルーチンへ移行されます。ポートはそれらの開始時方向と、どの負荷もOFFするように全てのピンをHigh設定に初期化します。始めに全てを出力として定めるには両ポートのデータ方向レジスタに\$FFの格納が必要です。方向はポートBに於いてキーパッド走査機能によって短い時間の間、変更されます。タイマ/カウンタの前置分周器は256分周クロックに設定され、タイマ/カウンタが176を設定された時に5msの割り込み周期を生成します。

割り込み周期用の式は244.1nsの命令周期時間ならば4.096MHzクロックに拘束されます。タイマ/カウンタ0(TCNT0)レジスタに格納されるべき数値nは、従って次式によって与えられます。

$$(256 - n) \times 256 \times 0.2441 \text{ } [\mu\text{s}]$$

176の値が正確に5msを提供し、高い実時間時計精度を保証します。

timesetting

LEDは時間が不正でリセットの必要があることを示すため、まさに'EEEE'点滅にされています。これはキーパッドで時刻設定(SET)キーが押されるまで続きます。これがキーパッドからの入力を処理して返還表示する'setrtc'関数を呼び出します。一旦時刻がリセットされると、主家事機能が本筋の'秒'変数からの更新と表示器の駆動を処理し、そして指示に関してキーパッドを走査します。

housekeeping

主家事機能は背面処理から配給された時刻変数を更新する作業を行い、正しい時刻でLED表示器を駆動します。指示入力を許すためにキーパッドも走査され、そして負荷に対するON/OFF計時器が調べられます。流れ図は図3.で示されます。

割り込み処理ルーチンによって増やされた秒が60と比較されます。60秒たったなら、分変数が増やされ、秒変数は0にリセットされます。同じ手順が時間に対して採用され、分変数が60と比較され、それによって時変数が増やされます。そして時変数は新しい日の開始を調べるために24と比較され、時から秒の全てが0にリセットされます。

SRAM(レジスタ)記憶の使用を節約するため、分と時は各々1バイトに制限されています。下位ニブルが下位桁を、上位ニブルが上位桁を収容します。これは(パック)BCDとして扱われ、正しい計数を保証するために適切な誤り防止が含まれなければならないことを意味します。従って分と時のバイトはニブルに分割され、そして各々の検査で大きさを調べられなければなりません。

分や時での検査のどの間でも変更全く出会わない場合、次の部分が迂回されて時刻が表示されます。時計は24時間形式で、時刻が23:59から増やされる時に必然として新しい日を開始させなければなりません。表示ルーチンはキー走査ルーチンも含む'display'と呼ばれる関数です。

display関数から戻ると、キー値と続いて負荷に対するON/OFF時間が調べられ、家事周回が繰り返される前に適切な動きが取られます。例えば、負荷1のON時間が実時間と等しければ、負荷1がONに切り換えられます。

様々な動きを示すための単一ビット群を含むのに“フラグ”語(訳補:以下で記述される'flags'変数とSREGのTフラグの総称名)が使われます。これは或る関数から別の関数へ制御を渡すために使われます。本応用に関しては1バイト内で得られるよりも1つ多い9つのフラグが必要とされます。ただ1ビットのために別のレジスタを使うことを省くため、ステータスレジスタの'T'フラグが9番目のビット用に使われています。これはプログラミングを容易にする特別な分岐命令(BRTC,BRTS)を使って検査できるために有用で、本来の'flags'検査にはSBRCとSBRSの命令が使われます。フラグはHigh活性(有効)で、それらの機能に対して表1.で示されるように割り当てられます。繰り返しを回るのに取られる時間は、実時間時計が割りこみ駆動なのでそれに影響を及ぼさず、繰り返しは1周の間に4回割り込まれるでしょう。

図3. 前面処理流れ図(2/2)

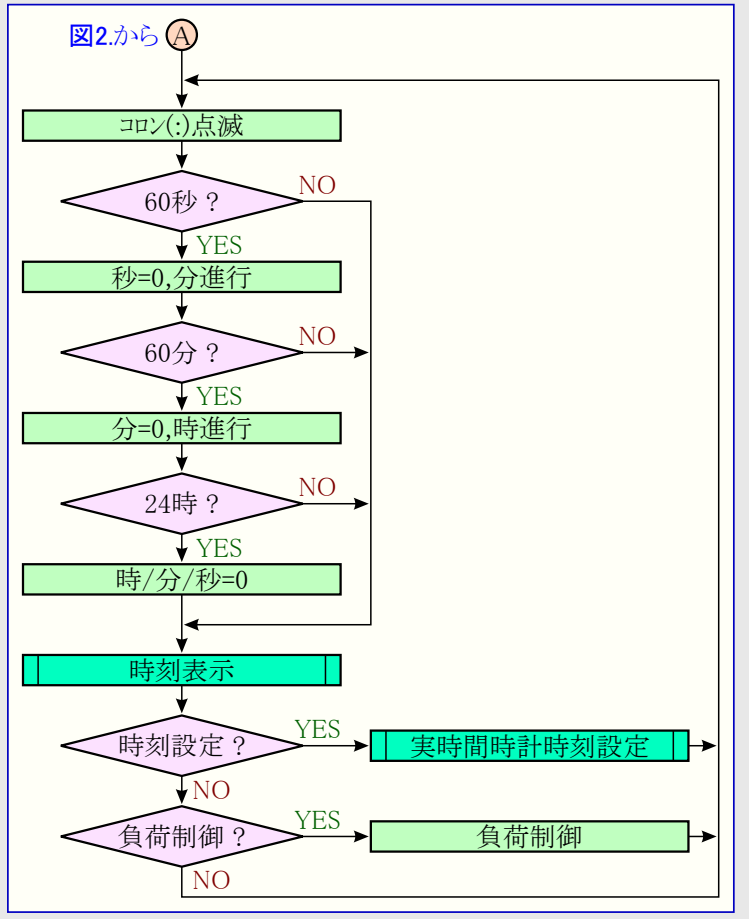


表1. フラグ語使用割り当て

'flags'ビット番号	機能
7	5ms計時パルス
6	キー押下可(反動回復)
5	負荷2 OFF
4	負荷2 ON
3	負荷1 OFF
2	負荷1 ON
1	負荷2活性(有効)
0	負荷1活性(有効)
SREGのTフラグ	時刻設定遭遇

中央コン(dp)は背面割り込み処理によって増加される'blink'変数を使って0.5秒間隔で点滅されます。これは表示機能によって遮蔽値として使われる'flash'変数を交互切り替えるのに使われます。負荷調査ルーチンは流れ図の単枠項目が示唆するよりも実際にはもっと複雑で、“フラグ”語内の様々な制御ビットを調べてそれによる動きを取ります。流れ図にこれを含めると、理解が非常に難しくなりました。

“負荷設定”指示を取り上げると、これは選択された負荷キーに対して新規のONまたはOFF時刻を設定するために'setrtc'関数を呼び出します。ここで同じ点滅方法が使われ、今入力されつつある適切な桁だけで“n”を点滅表示し、時間が入力されると、上位側から下位側へ順次移動します。従って使用者はその数値が何処になるかが分かります。

“全負荷OFF(CLEAR)”指示は以前のどのON/OFF指示も取り消して、直ちに両負荷をOFFにします。これらの処理は未だ背面に於いて正しい時刻を維持する実時間時計(RTC)に影響を及ぼしません。RTCもまた同じ処理によってどの段階でも時刻を更新するために変更することができます。

display関数

流れ図は図5.で示されます。この関数は点滅停止ルーチン、'setrtc'関数、家事(housekeeping)ルーチンによって呼び出され、キーパッドの走査と表示の多重化を扱います。より大きなAVRが使われるなら、桁駆動が関数に分割され、それを4回呼び出すことが価値を生じるでしょう。これはAT90S1200に於いて3段の深さのスタックのために行うことができません。

最初の部分は表示器のアノード駆動を禁止してキーパッドを走査します。これはポートBの構成設定を行ニブル入力と列ニブル出力に変更することによって行われます。4つの入力で内部プルアップも許可されます。4つの列ビット全てがLowを取り、行入力がPINBレジスタから読めます。これは押された行キーに依存する0,4,8,12の'key'に格納される何れかの基数、またはキーが押されていない場合の数値\$10を生じます。

そしてポートの構成設定が行ニブル出力と列ニブル入力で行ビットがLowを取るように全体を交換されます。短い安定時間後、列入力がPINBレジスタから読まれ、これは押された列キーに依存して0,1,2,または3の小変位を基数に加えるのに用いられます。最終結果はEEPROM内に格納された表内の必要とする実際のキー値を探すための指標として用いられる'key'に格納される値です。真のキー値が'key'に書き戻され、機能と呼ぶことによって使われます。これはキーが論理的な順番で整理していないために必要です。これはプログラムの書き手に対してもより大きな柔軟性を提供します。キーパッドの配列と機能は図4.で示されます。

図4. キーパッドの配列と機能

列1	列2	列3	列4	
1	2	3	F 負荷1 ON	行4
4	5	6	E 負荷1 OFF	行3
7	8	9	D 負荷2 ON	行2
A 時刻設定	0	B 全負荷OFF	C 負荷2 OFF	行1

9よりも大きな値のキーは(訳補:数値として処理されずに)捕獲され、機能と呼ぶことによって使われる“フラグ”語内の対応ビットを設定するのに使われます。\$10のキー値はキーが全く押されていないことを示します。

キーが押された場合、使用者へ感触を還元するためにポートDのビット6に接続された圧電スピーカへ短い通知音が送られます。

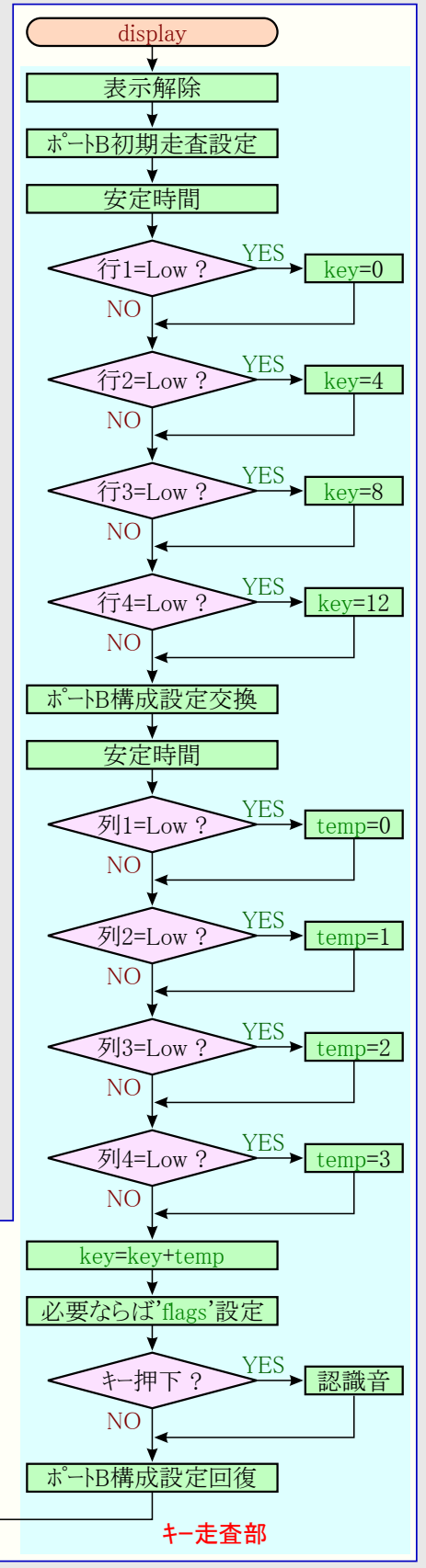
そして桁は背面処理によって設定される5msフラグによる計時の5ms時間幅での周回で多重化されます。これは(短いキー走査時間を無視して)明るくてちらつきなしの表示を生成する約50Hzの表示速度を生じます。

各桁駆動は7seg復号に対してEEPROMに格納された参照表を使い、これは'temp'レジスタ経由で指標を取り、その文字を点灯するのに必要とするバイトをアクセスするためにそれを使います。もっと意味のあるキーパッド入力にするために多数の特殊文字が使われます。例えば文字'E'は電源投入での点滅する異常表示のために定義され、文字'o'、'n'、'f'は負荷ON/OFF設定入力のために定義されています。使用者の応用に対してより大きなAVRを使う場合、これらの表をフラッシュメモリへ移してそれらを変位付きアドレス指定によってアクセスできるかもしれません。

そしてコロン(:)点滅部が0.5秒事象を調べて、前回の表示処理で使われた'flash'遮蔽値を変更し、結果として正しい時計機能を示すために中央コロンを点滅します。

そして関数は'key'に格納されたキー値と共に関数を呼んだところへ返ります。

図5. “display”関数流れ図



(訳注) 原書の図5と図6は図5として結合しました。

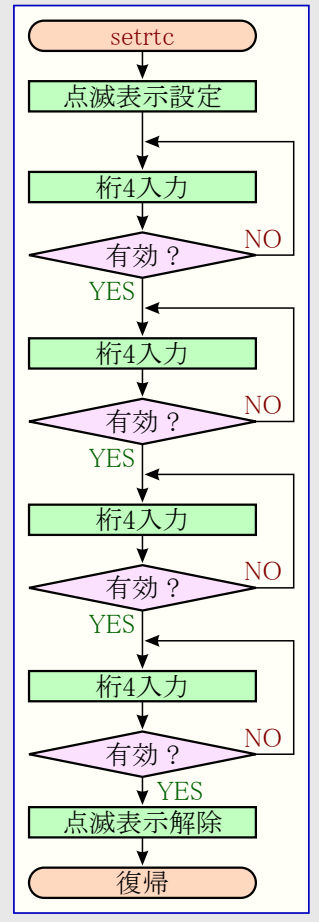
setrtc関数

流れ図は図7.で示されます。この関数は表示を設定するのにキーパッド入力が必要な全てのルーチンによって呼び出されます。これは電源投入/リセットでの実時刻入力、実時刻変更のための時刻設定(SET)キー押下、4つの負荷設定キーのどれかの押下で起きます。押下キーを得ると適切な桁を表示するためにdisplay関数を呼びます。程よいキー押下の動きを提供するために、背面割り込み関数によって5ms毎に増加される'bounce'計数器を使います。

関数は最上位桁から始めて最下位桁までの作業で4つの段階で進み、キーパッド経由で適切な値が入力されるまで各桁に於いて点滅する"n"を表示します。範囲外の値は防がれ、それが範囲内であるまで再び入力を要求します。

4つ全ての桁が正しく入力されると、関数は'hiset'変数の時と'loset'変数の分と共に(関数を)抜け出します。これらは呼ぶ機能により、家事機能によって使われる適切な変数に書き換え(置き換え)られます。

図7. "setrtc"関数流れ図

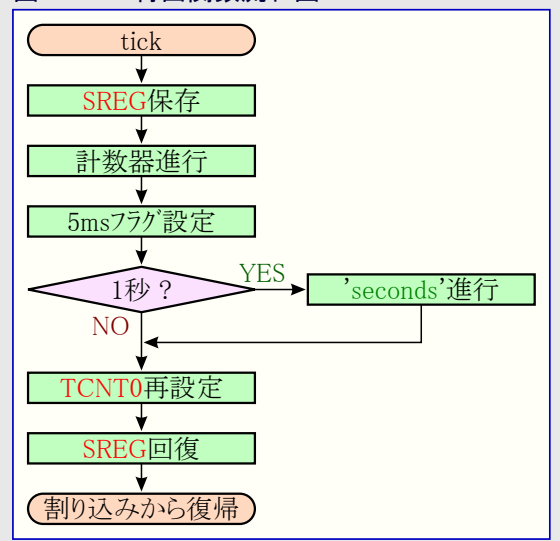


背面関数 (tick)

この関数はタイマ/カウンタ0溢れによって5ms毎に起動され、前面関数の繰り返し内のどの点に於いても割り込みます。従って前面処理を妨げることを避けるため、ルーチンは入口に於いてステータスレジスタ(SREG)を保存し、当然の事として抜け出しに於いてそれを回復します。'temp'レジスタの使用も同じ理由のために避けられます。

関数は非常に素直で、入る毎に3つの計数レジスタを増加し、displayルーチンによって使われる5msの計時フラグを設定し、タイマ/カウンタ0(TCNT0)を再設定し、そして必要ならば実時間時計(RTC)の秒計数器を増やす(進める)だけです。流れ図が図8.で示されます。

図8. "tick"背面関数流れ図



資源

表2. CPUとメモリの使用

関数	コード量(語)	周期数	使用レジスタ	割り込み	説明
reset	17	17	R16,R31	-	初期化
timesetting	9	14	R1,R2,R18,R19, R24,R25	-	実時間時計の初期設定
housekeeping	97	52 (代表値)	R1,R2,R16~R21, R24,R25,R28	-	実時刻表示維持、キーパッドに対する反応、負荷制御のための主家事繰り返し
display	158	150 (代表値)	R16,R17,R20,R21, R23~R26,R28	-	キー走査と表示機能
setrtc	47	45 (代表値)	R1,R2,R16,R20, R22,R24~R26,R28	-	時間と負荷設定入力のキーパッドを処理するための関数
tick	15	21	R0,R31	タイマ/カウンタ0溢れ	実時間の5msと1s計時を提供するための背面割り込み処理ルーチン
合計	343	-	R0~R2,R16~R26, R28,R31	タイマ/カウンタ0溢れ	

表3. 周辺機能の使用

周辺機能	説明	許可割り込み
タイマ/カウンタ0	5ms計時計数器	256前置分周でのタイマ/カウンタ0溢れ
EEPROM (32バイト)	キーに対する値割り当て、7seg復号割り当て値	-
ポートBの8つのI/Oピン	4×4キーパッド接続とLEDセグメント駆動(2重機能)	-
ポートDの3つのI/Oピン	負荷1,2(LED)と圧電スピーカ	-
ポートDの4つのI/Oピン	4桁LED表示器用アノード駆動	-

(訳注) このソフトウェアはアセンブル条件にも依りますが、おそらくアセンブル時にレジスタの2重定義を示す4つの警告が出ます。これは予め定義済みのポイント(X,Y,Z)に別の名称で再定義したことに依るもので、問題はありません。

```

;**** 応用記述 AVR242 ****
;*
;* 表題      : 多重化LED駆動と4×4キーパッド採択
;* 版       : 1.0
;* 最終更新  : 98.07.24
;* 対象     : 全AVRデバイス
;*
;* 支援Eメール : avr@atmel.com
;*
;* 説明
;* この応用記述はI/Oピンを2重機能に使用して25時間生産計時器または実時間時計を提供するプログラムを網羅します。4×4キーパッド経由での入力
;* 力で、多重化された4桁LED表示器と付加インターフェース回路経由で負荷を駆動するための2つのON/OFF出力へ出力します。この例に於いて
;* LED負荷が駆動されますが、適合する部品の追加でどの負荷も駆動することができます。キーが押された時に通知音を鳴らす圧電スピーカに
;* よってキー押下毎に感触還元が提供されます。含まれるのはキーパッド経由での時計設定、各負荷に対する24時間当たりの1つのON/OFF計時
;* 器設定、実時間時計のための機能、キー走査を可能にする主プログラムと調整ルーチンです。この例は制限されたI/O数をどう克服するかを実演す
;* るためにAT90S1200で動きますが、ヘクタ、EEPROM、スタックポイントに於ける適合する変更でどのAVRにもできます。タイミングは4.096MHzのクリ
;* タルが使われる仮定です(4MHzのクリスタルはタイマ/カウンタ設定手順で176の代わりに178を使った場合に-0.16%の誤差を生じますが、これは規則
;* 的な間隔に於いてソフトウェアで補正することができます)。時刻とON/OFF設定表示に対して提供される追加文字と共に表示データを復号するた
;* めと、キーパッド変換表にEEPROM内の参照表が使われます。使用者の応用のためにEEPROMが必要とされる場合、より大きなAVRデバイスに
;* 於いてこの表をフラッシュメモリに移すことができます。
;*****

        .INCLUDE    "1200def.inc"

;
; * 全プログラムにて使われるレジスタ *
; * 全ルーチンにて使われる全域変数 *

        .DEF        loiset      =R1          ;時刻設定での分値記憶域(ハックBCD)
        .DEF        hiset       =R2          ;時刻設定での時値記憶域(ハックBCD)
        .DEF        ld1minon     =R3          ;負荷1 ON時刻の分値
        .DEF        ld1hron      =R4          ;負荷1 ON時刻の時値
        .DEF        ld1minoff    =R5          ;負荷1 OFF時刻の分値
        .DEF        ld1hroff     =R6          ;負荷1 OFF時刻の時値
        .DEF        ld2minon     =R7          ;負荷2 ON時刻の分値
        .DEF        ld2hron      =R8          ;負荷2 ON時刻の時値
        .DEF        ld2minoff    =R9          ;負荷2 OFF時刻の分値
        .DEF        ld2hroff     =R10         ;負荷2 OFF時刻の時値
    
```



```

.DEF      temp      =R16      ;一般雑用
.DEF      second    =R17      ;実時間時計(RTC)秒計数值
.DEF      minute    =R18      ;実時間時計(RTC)分計数值(ハックBCD)
.DEF      hour      =R19      ;実時間時計(RTC)時計数值(ハックBCD)
.DEF      mask      =R20      ;桁点減用点減遮蔽値
.DEF      blink     =R21      ;コロン(:)点減間隔計数器(5ms単位)
.DEF      bounce    =R22      ;キーパッド反動(チャタリング防止)計数器(5ms単位)
.DEF      flash     =R23      ;点滅(点灯/消灯)フラグ(実際にはビット0のみ有効)
.DEF      lobyte    =R24      ;display関数用分析記憶域(ハックBCD)
.DEF      hibyte    =R25      ;display関数用時桁記憶域(ハックBCD)
.DEF      key       =R26      ;キーパッド走査からのキー番号
                                ; * 'keyscan'関数によって返される'key'値 *
                                ; 'key' キー名 機能
                                ; 0   1   1
                                ; 1   2   2
                                ; 2   3   3
                                ; 3   F   負荷1 ON
                                ; 4   4   4
                                ; 5   5   5
                                ; 6   6   6
                                ; 7   E   負荷1 OFF
                                ; 8   7   7
                                ; 9   8   8
                                ;10  9   9
                                ;11  D   負荷2 ON
                                ;12  A   時刻設定
                                ;13  0   0
                                ;14  B   全負荷OFF
                                ;15  C   負荷2 OFF
                                ;16  -   押下なし

.DEF      tock      =R27      ;5ms単位1s計数器
.DEF      flags     =R28      ;キーパッドの指示キー関係フラグ(以下のEQU参照)

.EQU      ms5       =7        ;表示時間用5ms間隔経過フラグ
.EQU      keyok     =6        ;チャタリング防止時間(0.2s)経過フラグ(確認後要解除)
.EQU      ld2off    =5        ;負荷2OFF時刻設定要求フラグ
.EQU      ld2on     =4        ;負荷2ON時刻設定要求フラグ
.EQU      ld1off    =3        ;負荷1OFF時刻設定要求フラグ
.EQU      ld1on     =2        ;負荷1ON時刻設定要求フラグ
.EQU      ld2       =1        ;負荷2ON/OFF時刻有効フラグ
.EQU      ld1       =0        ;負荷1ON/OFF時刻有効フラグ
;
;
; (ポートBピン)

.EQU      col1      =0        ;LEDセグメントa /キーパッド'列1
.EQU      col2      =1        ;LEDセグメントb /キーパッド'列2
.EQU      col3      =2        ;LEDセグメントc /キーパッド'列3
.EQU      col4      =3        ;LEDセグメントd /キーパッド'列4
.EQU      row1      =4        ;LEDセグメントe /キーパッド'行1
.EQU      row2      =5        ;LEDセグメントf /キーパッド'行2
.EQU      row3      =6        ;LEDセグメントg /キーパッド'行3
.EQU      row4      =7        ;LEDセグメントdp/キーパッド'行4
;
; (ポートDピン)

.EQU      A1        =0        ;LED桁1アノード駆動(負論理)
.EQU      A2        =1        ;LED桁2アノード駆動(負論理)
.EQU      A3        =2        ;LED桁3アノード駆動(負論理)
.EQU      A4        =3        ;LED桁4アノード駆動(負論理)
.EQU      LOAD1     =4        ;負荷(LED)1出力(負論理)
.EQU      LOAD2     =5        ;負荷(LED)2出力(負論理)
.EQU      PZ        =6        ;圧電スピーカ出力(負論理)
;
; * タイマ/カウンタ0溢れ割り込み処理ルーチンで使われるレジスタ *

.DEF      status    =R0        ;ステータスレジスタ(SREG)保存用
.DEF      timer     =R31      ;TCNT0値設定用/雑用

```



```

.ESEG                                ;EEPROM領域

.ORG      0

;      * LED表示復号用参照表 *

table1:
.DB      ~0b00111111                ; 0: '0' ($C0)
.DB      ~0b00000110                ; 1: '1' ($F9)
.DB      ~0b01011011                ; 2: '2' ($A4)
.DB      ~0b01001111                ; 3: '3' ($B0)
.DB      ~0b01100110                ; 4: '4' ($99)
.DB      ~0b01101101                ; 5: '5' ($92)
.DB      ~0b01111101                ; 6: '6' ($82)
.DB      ~0b00000111                ; 7: '7' ($F8)
.DB      ~0b01111111                ; 8: '8' ($80)
.DB      ~0b01101111                ; 9: '9' ($90)
.DB      ~0b01111001                ;10: 'E' ($86,特殊文字)
.DB      ~0b01110001                ;11: 'f' ($8E,特殊文字)
.DB      ~0b01011100                ;12: 'o' ($A3,特殊文字)
.DB      ~0b01010100                ;13: 'n' ($AB,特殊文字)
.DB      ~0b00000000                ;14: '' ($FF)
.DB      ~0b00000000                ;15: '' ($FF)

;      * キー値変換参照表 *

table2:
.DB      1                          ; 0: '1'
.DB      2                          ; 1: '2'
.DB      3                          ; 2: '3'
.DB      15                         ; 3: 'F'
.DB      4                          ; 4: '4'
.DB      5                          ; 5: '5'
.DB      6                          ; 6: '6'
.DB      14                         ; 7: 'E'
.DB      7                          ; 8: '7'
.DB      8                          ; 9: '8'
.DB      9                          ;10: '9'
.DB      13                         ;11: 'D'
.DB      10                         ;12: 'A'
.DB      0                          ;13: '0'
.DB      11                         ;14: 'B'
.DB      12                         ;15: 'C'

;***** ソースコード *****

.CSEG                                ;コード領域

.ORG      0

RJMP     reset                      ;リセットベクタ
NOP      ;[未使用] 外部割り込み0ベクタ
RJMP     tick                       ;タイマ/カウンタ0溢れ割り込みベクタ(5ms)
NOP      ;[未使用] アナログ比較器割り込みベクタ

;      * リセット/初期化 *

reset:
SER      temp                      ;(他の殆どのAVRに関してスタックポインタ初期化を忘れずに)
OUT      DDRB, temp                ;全ポートB出力設定
OUT      DDRD, temp                ;全ポートD出力設定
OUT      PORTB, temp               ;全LEDセグメントOFF(H)値出力
OUT      PORTD, temp               ;LED桁駆動と負荷(LED)を全てOFF(H)値出力
LDI      temp, 0b00000100          ;256前置分周指定値取得
OUT      TCCR0, temp               ;256前置分周でタイマ/カウンタ0始動
LDI      timer, 176                ;5ms間計数値取得[(256-n)×256×0.2441µs]
OUT      TCNT0, timer              ;5ms後溢れ値でTCNT0初期化
LDI      temp, 0b00000010          ;タイマ/カウンタ0溢れ割り込みビット値取得
OUT      TIMSK, temp               ;タイマ/カウンタ0溢れ割り込み許可
CLR      flags                     ;制御フラグ初期化
CLR      tock                       ;5ms単位1s計数器初期化
CLR      bounce                     ;5ms単位キーチャタリング防止計時器初期化
CLR      flash                       ;点滅(点灯/消灯)フラグ初期化(消灯指定)
CLR      blink                       ;5ms単位点滅間隔計数器初期化
SEI      ;全体割り込み許可

```

```

;          * 初期時刻設定 *

; 電源OFF警告及び試験として、キーパッドで時刻設定キーが押されるまでLEDを'EEEE'点滅します。

timesetting:
    LDI      hbyte, 10*16+10      ;LED時桁初期値"EE"指定
    LDI      lbyte, 10*16+10      ;LED分桁初期値"EE"指定
    SER      mask                 ;全seg点滅表示指定
notyet:   RCALL   display         ;キー走査/表示処理
          BRCC   notyet          ;時刻設定キー押下まで継続へ
;
          RCALL   setrtc         ;時刻キー入力値取得
          MOV     hour, hiset     ;時刻時計数器設定
          MOV     minute, loset   ;時刻分計数器設定
          CLT                    ;時刻設定要求フラグ解除
;
          * 主家事処理 *

housekeeping:
    CLR      mask                 ;全seg点滅要求フラグ解除
    CPI      blink, 500/5        ;点滅周期に対して0.5s経過判定
    BRNE     TMupd               ;0.5s未経過で分岐
;
          CLR      blink         ;0.5s経過で点滅間隔計時器初期化
          COM      flash         ;点滅(点灯/消灯)フラグ反転
;
          ( 時刻進行処理 )

TMupd:   CPI      second, 60     ;秒計数器1分経過判定
          BRNE     TMupd1        ;1分未経過で分岐
;
          CLR      second        ;1分経過で、秒計数器初期化
          INC      minute        ;分計数器進行
          MOV      temp, minute   ;分値複写
          ANDI     temp, 0b00001111 ;下位桁値抽出
          CPI      temp, 10       ;桁溢れ判定
          BRNE     TMupd1        ;桁溢れなしで分岐
;
          ANDI     minute, 0b11110000 ;桁溢れ有りで、上位桁値抽出/下位桁値初期化
          LDI      temp, 1*16     ;上位桁の1相当値取得
          ADD      minute, temp   ;上位桁値進行
          CPI      minute, 6*16   ;1時間経過判定
          BRNE     TMupd1        ;1時間未経過で分岐
;
          CLR      minute        ;1時間経過で、分計数器初期化
          INC      hour          ;時計数器進行
          MOV      temp, hour     ;時値複写
          ANDI     temp, 0b00001111 ;下位桁値抽出
          CPI      temp, 10       ;桁溢れ判定
          BRNE     TMupd1        ;桁溢れなしで分岐
;
          ANDI     hour, 0b11110000 ;桁溢れ有りで、上位桁値抽出/下位桁値初期化
          LDI      temp, 1*16     ;上位桁の1相当値取得
          ADD      hour, temp     ;上位桁値進行
TMupd1:  CPI      hour, 2*16+4    ;1日経過判定
          BRNE     DspKs         ;1日未経過で分岐
;
          CLR      hour          ;1日経過で、時計数器初期化
          CLR      minute        ;分計数器初期化
          CLR      second        ;秒計数器初期化

```

```

;      (表示/キー走査・キー機能処理)

DspKs:  MOV      lobyte, minute      ;表示用分値設定
        MOV      hibyte, hour       ;表示用時値設定
        RCALL    display            ;時刻表示(4桁×5ms=20ms間)/キー走査
        BRTC     case1              ;時刻設定要求なしで次検査へ
;
;      RCALL    setrtc              ;時刻設定要求で、時刻キー入力値取得
        MOV      hour, hiset         ;時刻時計数器更新
        MOV      minute, loset      ;時刻分計数器更新
        CLT                          ;時刻設定要求フラグ解除
;
case1:  SBRC     flags, ld1          ;負荷1ON/OFF時刻無効でスキップ
        RJMP    Ld1CK              ;有効で実処理へ
;
case2:  SBRC     flags, ld2          ;負荷2ON/OFF時刻無効でスキップ
        RJMP    Ld2CK              ;有効で実処理へ
;
case3:  SBRC     flags, ld1on       ;負荷1ON時刻設定要求なしでスキップ
        RJMP    Ld1onT            ;要求有りを実処理へ
;
case4:  SBRC     flags, ld1off      ;負荷1OFF時刻設定要求なしでスキップ
        RJMP    Ld1offT           ;要求有りを実処理へ
;
case5:  SBRC     flags, ld2on       ;負荷2ON時刻設定要求なしでスキップ
        RJMP    Ld2onT            ;要求有りを実処理へ
;
case6:  SBRC     flags, ld2off      ;負荷2OFF時刻設定要求なしでスキップ
        RJMP    Ld2offT           ;要求有りを実処理へ
;
case7:  RJMP     housekeeping       ;主家事処理継続へ
;      (負荷1 ON/OFF時刻処理)

Ld1CK:  CP       hour, ld1hroff     ;負荷1OFF時刻時判定
        BRNE    Ld1CK1            ;OFF時刻時不一致で次へ
;
;      CP       minute, ld1minoff   ;負荷1OFF時刻分判定
        BRNE    Ld1CK1            ;OFF時刻分不一致で次へ
;
;      SBI      PORTD, LOAD1        ;OFF時刻一致で、負荷1OFF設定
;
Ld1CK1: CP       hour, ld1hron      ;負荷1ON時刻時判定
        BRNE    case2            ;ON時刻時不一致で次の本判定へ
;
;      CP       minute, ld1minon    ;負荷1ON時刻分判定
        BRNE    case2            ;ON時刻分不一致で次の本判定へ
;
;      CBI      PORTD, LOAD1        ;ON時刻一致で、負荷1ON設定
        RJMP    case2            ;次の本判定へ
;      (負荷2 ON/OFF時刻処理)

Ld2CK:  CP       hour, ld2hroff     ;負荷2OFF時刻時判定
        BRNE    Ld2CK1            ;OFF時刻時不一致で次へ
;
;      CP       minute, ld2minoff   ;負荷2OFF時刻分判定
        BRNE    Ld2CK1            ;OFF時刻分不一致で次へ
;
;      SBI      PORTD, LOAD2        ;OFF時刻一致で、負荷2OFF設定
;
Ld2CK1: CP       hour, ld2hron      ;負荷2ON時刻時判定
        BRNE    case3            ;ON時刻時不一致で次の本判定へ
;
;      CP       minute, ld2minon    ;負荷2ON時刻分判定
        BRNE    case3            ;ON時刻分不一致で次の本判定へ
;
;      CBI      PORTD, LOAD2        ;ON時刻一致で、負荷2ON設定
        RJMP    case3            ;次の本判定へ

```

```

;      (( 負荷1 ON時刻設定 ))

Ld1onT:  SBR      flags, 1<<ld1      ;負荷1ON/OFF時刻有効フラグ設定
          RCALL   setrtc          ;時刻キー入力値取得
          MOV     ld1hron, hiset   ;負荷1ON時刻時値設定/更新
          MOV     ld1minon, loset  ;負荷1ON時刻分値設定/更新
          CBR     flags, 1<<ld1on  ;負荷1ON時刻設定要求フラグ解除
          RJMP   case4           ;次の本判定へ

;      (( 負荷1 OFF時刻設定 ))

Ld1ofT:  RCALL   setrtc          ;時刻キー入力値取得
          MOV     ld1hroff, hiset  ;負荷1OFF時刻時値設定/更新
          MOV     ld1minoff, loset ;負荷1OFF時刻分値設定/更新
          CBR     flags, 1<<ld1off ;負荷1OFF時刻設定要求フラグ解除
          RJMP   case5           ;次の本判定へ

;      (( 負荷2 ON時刻設定 ))

Ld2onT:  SBR      flags, 1<<ld2      ;負荷2ON/OFF時刻有効フラグ設定
          RCALL   setrtc          ;時刻キー入力値取得
          MOV     ld2hron, hiset   ;負荷2ON時刻時値設定/更新
          MOV     ld2minon, loset  ;負荷2ON時刻分値設定/更新
          CBR     flags, 1<<ld2on  ;負荷2ON時刻設定要求フラグ解除
          RJMP   case6           ;次の本判定へ

;      (( 負荷2 OFF時刻設定 ))

Ld2ofT:  RCALL   setrtc          ;時刻キー入力値取得
          MOV     ld2hroff, hiset  ;負荷2OFF時刻時値設定/更新
          MOV     ld2minoff, loset ;負荷2OFF時刻分値設定/更新
          CBR     flags, 1<<ld2off ;負荷2OFF時刻設定要求フラグ解除
          RJMP   case7           ;次の本判定へ

```

```

;      * キーパッド走査・LED動的点灯 *

; 毎回、キーパッドを走査し、各桁で5ms間、hibyteとlobyte位置の桁を取って時刻を表示
; するための、全てのルーチンで使われる多重化ルーチン。

display:  SER      temp      ;全1値取得
          OUT      PORTB, temp ;LED表示禁止

;      ( キーパッド走査 )

keyscan:  CBR      flags, 1<<keyok ;チャタリング防止時間(0.2s)経過フラグ初期化
          LDI      key, 16        ;キー番号初期化(押下キーなし)
          SER      temp          ;全1値取得
          OUT      PORTB, temp    ;LED表示禁止(全seg=OFF)
          IN       temp, PORTD    ;ポートD出力値取得
          ORI      temp, 0b00001111 ;LED桁アノードのビット値取得
          OUT      PORTD, temp    ;LED桁アノード全OFF設定

;      (( キーパッド行走査 ))

          LDI      temp, 0b00001111 ;行入力/列出力ビット値取得
          OUT      DDRB, temp      ;行入力/列出力に設定
          LDI      temp, 0b11110000 ;行プルアップ/列0出力ビット値取得
          OUT      PORTB, temp    ;行プルアップ許可,列全Low出力設定
KSdly1:  LDI      temp, 20          ;短時間安定計数器初期化(約15μs)
          DEC      temp          ;安定時間計数器減数
          BRNE     KSdly1        ;計数完了まで継続へ

;
          SBIS     PINB, ROW1     ;行1押下なしでスキップ
          LDI      key, 0        ;行1押下有りで、基数=0設定

;
          SBIS     PINB, ROW2     ;行2押下なしでスキップ
          LDI      key, 4        ;行2押下有りで、基数=4設定

;
          SBIS     PINB, ROW3     ;行3押下なしでスキップ
          LDI      key, 8        ;行3押下有りで、基数=8設定

;
          SBIS     PINB, ROW4     ;行4押下なしでスキップ
          LDI      key, 12       ;行4押下有りで、基数=12設定

;      (( キーパッド列走査 ))

          LDI      temp, 0b11110000 ;行出力/列入力ビット値取得(行/列の入出力交換)
          OUT      DDRB, temp      ;行出力/列入力に設定
          LDI      temp, 0b00001111 ;行0出力/列プルアップビット値取得
          OUT      PORTB, temp    ;行全Low出力設定,列プルアップ許可
KSdly2:  LDI      temp, 20          ;短時間安定計数器初期化(約15μs)
          DEC      temp          ;安定時間計数器減数
          BRNE     KSdly2        ;計数完了まで継続へ

;
          CLR      temp          ;一時キー変位値初期化
          SBIS     PINB, COL1     ;列1押下なしでスキップ
          LDI      temp, 0        ;列1押下有りで、変位=0設定

;
          SBIS     PINB, COL2     ;列2押下なしでスキップ
          LDI      temp, 1        ;列2押下有りで、変位=1設定

;
          SBIS     PINB, COL3     ;列3押下なしでスキップ
          LDI      temp, 2        ;列3押下有りで、変位=2設定

;
          SBIS     PINB, COL4     ;列4押下なしでスキップ
          LDI      temp, 3        ;列4押下有りで、変位=3設定

;
          ADD      key, temp      ;キー値指標取得(基数+変位)
          CPI      key, 16       ;キー押下有無判定
          BREQ     nokey         ;キー押下なしで分岐

```

```

; ((キー押下処理))
LDI      temp, LOW(table2) ;変換表基準位置下位値取得
ADD      key, temp         ;変換表対応位置取得
OUT      EEAR, key        ;変換表対応位置でEEPROMアドレス指定
SBI      EECR, EERE       ;対応キー番号EEPROM読み込み指示
IN       key, EEDR        ;EEPROMから対応キー番号取得

; (((時刻設定キー)))
FKCK:    CPI      key, 10   ;時刻設定キー判定
BRNE    FKCK1           ;時刻設定キー以外で次へ
;
SET      SREG, 0x01      ;時刻設定キーで、SREGのTフラグ設定
;
; (((全負荷OFFキー)))
FKCK1:   CPI      key, 11   ;全負荷OFFキー判定
BRNE    FKCK2           ;全負荷OFFキー以外で次へ
;
SBI      PORTD, LOAD1     ;全負荷OFFキーで、負荷1OFF設定
SBI      PORTD, LOAD2     ;負荷2OFF設定
CBR      flags, 1<<1d2|1<<1d1 ;負荷1と2のON/OFF時刻有効フラグを解除
;
; (((負荷1 ON時刻設定キー)))
FKCK2:   CPI      key, 15   ;負荷1ON設定キー判定
BRNE    FKCK3           ;負荷1ON設定キー以外で次へ
;
SBR      flags, 1<<1d1on   ;負荷1ON時刻設定要求フラグ設定
;
; (((負荷1 OFF時刻設定キー)))
FKCK3:   CPI      key, 14   ;負荷1OFF設定キー判定
BRNE    FKCK4           ;負荷1OFF設定キー以外で次へ
;
SBR      flags, 1<<1d1off  ;負荷1OFF時刻設定要求フラグ設定
;
; (((負荷2 ON時刻設定キー)))
FKCK4:   CPI      key, 13   ;負荷2ON設定キー判定
BRNE    FKCK5           ;負荷2ON設定キー以外で次へ
;
SBR      flags, 1<<1d2on   ;負荷2ON時刻設定要求フラグ設定
;
; (((負荷2 OFF時刻設定キー)))
FKCK5:   CPI      key, 12   ;負荷2OFF設定キー判定
BRNE    tactile         ;負荷2OFF設定キー以外で次へ
;
SBR      flags, 1<<1d2off  ;負荷2OFF時刻設定要求フラグ設定
;
; (通知音発生)
; 感触還元音生成ルーチン。5ms間、圧電スピーカへ約5kHzの音を供給します。
tactile:  CBR      flags, 1<<ms5 ;5ms間隔経過フラグ解除
          CBI      PORTD, PZ     ;圧電スピーカ電圧印加開始
          LDI      temp, 125     ;遅延計数器初期化(約92μs)
tact1:    DEC      temp          ;遅延計数器減数
BRNE    tact1             ;遅延完了まで継続へ
;
          SBI      PORTD, PZ     ;圧電スピーカ電圧印加停止
          LDI      temp, 125     ;遅延計数器初期化(約92μs)
tact2:    DEC      temp          ;遅延計数器減数
BRNE    tact2             ;遅延完了まで継続へ
;
          SBRS     flags, ms5    ;5ms経過でスキップ
          RJMP    tactile       ;5ms経過まで発音継続へ
;
          CPI      bounce, 200/5 ;キー押下から0.2s経過調査
          BRLO    nokey         ;0.2s未経過で分岐
;
          SBR      flags, 1<<keyok ;0.2s経過で、チャタリング防止時間(0.2s)経過フラグ設定

```

```

;          ( 4桁LED動的点灯処理 )

nokey:    SER          temp          ;全1値取得
          OUT          DDRB, temp    ;LED表示用再初期化(全ポートB出力設定)
          OUT          PORTB, temp   ;LED表示用再初期化(全seg=OFF=Low出力設定)

;          (( 桁1(最下位桁) ))

          CBI          PORTD, A1     ;LED桁1共通アノードON
          MOV          temp, lobyte  ;下位2桁(分)値取得
digit1:   CBR          flags, 1<<ms5 ;5ms間隔経過フラグ解除
          ANDI        temp, 0b00001111 ;下位桁(分1の桁)値のみ抽出
          OUT          EEAR, temp    ;変換表対応位置としてEEPROMアドレス指定
          SBI          EECR, EERE    ;対応LED模様型EEPROM読み込み指示
          IN           temp, EEDR    ;EEPROMから対応LED模様型取得
          SBRS        flash, 0      ;点滅(点灯/消灯)フラグ=点灯区間でスキップ
          OR           temp, mask    ;消灯区間で、点滅要求なら、消灯(1)値に変更

;
          OUT          PORTB, temp    ;LED桁1セグメント値出力開始
led1:     SBRS        flags, ms5     ;5ms経過でスキップ
          RJMP        led1          ;5ms経過まで待機継続へ

;
          SBI          PORTD, A1     ;LED桁1共通アノードOFF
          SER          temp          ;全1値取得
          OUT          PORTB, temp   ;LED全セグメントOFF

;          (( 桁2 ))

          CBI          PORTD, A2     ;LED桁2共通アノードON
          MOV          temp, lobyte  ;下位2桁(分)値取得
          SWAP        temp          ;上位桁値を下位側へ移動
digit2:   CBR          flags, 1<<ms5 ;5ms間隔経過フラグ解除
          ANDI        temp, 0b00001111 ;上位桁(分10の桁)値のみ抽出
          OUT          EEAR, temp    ;変換表対応位置としてEEPROMアドレス指定
          SBI          EECR, EERE    ;対応LED模様型EEPROM読み込み指示
          IN           temp, EEDR    ;EEPROMから対応LED模様型取得
          SBRS        flash, 0      ;点滅(点灯/消灯)フラグ=点灯区間でスキップ
          OR           temp, mask    ;消灯区間で、点滅要求なら、消灯(1)値に変更

;
          OUT          PORTB, temp    ;LED桁2セグメント値出力開始
led2:     SBRS        flags, ms5     ;5ms経過でスキップ
          RJMP        led2          ;5ms経過まで待機継続へ

;
          SBI          PORTD, A2     ;LED桁2共通アノードOFF
          SER          ltemp         ;全1値取得
          OUT          PORTB, temp   ;LED全セグメントOFF

;          (( 桁3 ))

          CBI          PORTD, A3     ;LED桁3共通アノードON
          MOV          temp, hibyte  ;上位2桁(時)値取得
digit3:   CBR          flags, 1<<ms5 ;5ms間隔経過フラグ解除
          ANDI        temp, 0b00001111 ;下位桁(時1の桁)値のみ抽出
          OUT          EEAR, temp    ;変換表対応位置としてEEPROMアドレス指定
          SBI          EECR, EERE    ;対応LED模様型EEPROM読み込み指示
          IN           temp, EEDR    ;EEPROMから対応LED模様型取得
          SBRS        second, 0     ;奇数秒でスキップ(2s周期dp点滅処理)
          ANDI        temp, 0b01111111 ;偶数秒で、コロン(dp)点灯指定

;
          SBRS        flash, 0      ;点滅(点灯/消灯)フラグ=点灯区間でスキップ
          OR           temp, mask    ;消灯区間で、点滅要求なら、消灯(1)値に変更

;
          OUT          PORTB, temp    ;LED桁3セグメント値出力開始
led3:     SBRS        flags, ms5     ;5ms経過でスキップ
          RJMP        led3          ;5ms経過まで待機継続へ

;
          SBI          PORTD, A3     ;LED桁3共通アノードOFF
          SER          temp          ;全1値取得
          OUT          PORTB, temp   ;LED全セグメントOFF

```

```

;      (( 桁4(最上位桁) ))

      CBI      PORTD, A4      ;LED桁4共通アノードON
      MOV      temp, hibyte   ;上位2桁(時)値取得
      SWAP    temp           ;上位桁値を下位側へ移動
      ANDI    temp, 0b00001111 ;上位桁(時10の桁)値のみ抽出
      BRNE   digit4         ;上位桁(時10の桁)が0以外で分岐
;
digit4:  LDI      temp, 0b11111111 ;上位桁(時10の桁)=0で、消灯に変更
      CBR      flags, 1<<ms5 ;5ms間隔経過フラグ解除
      ANDI    temp, 0b00001111 ;上位桁(時10の桁)値のみ抽出
      OUT     EEAR, temp      ;変換表対応位置としてEEPROMアドレス指定
      SBI     EECR, EERE     ;対応LED模様型EEPROM読み込み指示
      IN      temp, EEDR     ;EEPROMから対応LED模様型取得
      SBRS   flash, 0       ;点滅(点灯/消灯)フラグ=点灯区間でスキップ
      OR      temp, mask     ;消灯区間で、点滅要求なら、消灯(1)値に変更
;
led4:   OUT     PORTB, temp   ;LED桁4セグメント値出力開始
      SBRS   flags, ms5     ;5ms経過でスキップ
      RJMP   led4           ;5ms経過まで待機継続へ
;
      SBI     PORTD, A4     ;LED桁4共通アノードOFF
      SER     temp          ;全1値取得
      OUT     PORTB, temp   ;LED全セグメントOFF
      TST    mask          ;点滅指定有無検査
      BREQ   outled        ;点滅指定なしで分岐
;
      CPI     blink, 250/5 ;点滅指定有りで、0.25s経過判定(0.25×2=0.5s点滅周期)
      BRLO   outled        ;0.25s未経過で分岐
;
outled: CLR     blink       ;5ms単位点滅間隔計数器初期化
      COM    flash         ;点滅(点灯/消灯)フラグ反転
      RET                                ;呼び出し元へ復帰
;
      * 時刻入力 *

; キーパッドから実時間または負荷ON/OFFの時刻を設定するための関数。'hiset'内の時値と'loset'内の分値と共に返ります。
; ('xxset'はハックBCD値)

setrtc: SER     mask        ;全seg点滅表示指定
      LDI    hibyte, 13*16+15 ;上位2桁'n'表示値指定(13,15)
      SER    lobyte         ;下位2桁' '表示値指定(15,15)
;
;      ( 桁4:時10の桁入力 )

HurH:  CLR     bounce       ;5ms単位キーチャタリング防止計数器初期化
HurH1: RCALL   display      ;'n'表示,キーパッド入力
      SBRS   flags, keyok   ;チャタリング防止時間(0.2s)経過でスキップ
      RJMP   HurH1         ;チャタリング防止時間経過まで表示継続/待機へ
;
      CBR    flags, 1<<keyok ;チャタリング防止時間経過フラグ解除
      CPI    key, 2+1       ;時10の桁範囲(0~2)検査
      BRSH   HurH          ;範囲外で再入力へ
;
      SWAP   key           ;時10の桁値を上位ニブルへ移動(移動後の下位=0)
      MOV   hiset, key     ;時桁値一時保存
      LDI   hibyte, 0*16+13 ;上位2桁'0n'表示値仮指定
      ADD   hibyte, hiset  ;桁4を直前の入力値に変更

```



```

; (桁3:時1の桁入力)
HurL: CLR bounce ;5ms単位キー チャタリング防止計時器初期化
HurL1: RCALL display ;'xn'表示,キーパッド入力
SBRs flags, keyok ;チャタリング防止時間(0.2s)経過でスキップ
RJMP HurL1 ;チャタリング防止時間経過まで表示継続/待機へ
;
CBR flags, 1<<keyok ;チャタリング防止時間経過フラグ解除
MOV temp, hbyte ;現状時桁表示値取得/複写
ANDI temp, 0b11110000 ;時10の桁値のみ抽出
ADD temp, key ;入力した時1の桁値仮合成
CPI temp, $24 ;時範囲(0~23)検査
BRSH HurL ;範囲外で再入力へ
;
ADD hiset, key ;時桁値作成(時1の桁値合成)
MOV hbyte, hiset ;時桁値を表示値として再設定
LDI lbyte, 13*16+15 ;下位2桁'n'表示値指定
; (桁2:分10の桁入力)
MinH: CLR bounce ;5ms単位キー チャタリング防止計時器初期化
MinH1: RCALL display ;'xxn'表示,キーパッド入力
SBRs flags, keyok ;チャタリング防止時間(0.2s)経過でスキップ
RJMP MinH1 ;チャタリング防止時間経過まで表示継続/待機へ
;
CBR flags, 1<<keyok ;チャタリング防止時間経過フラグ解除
CPI key, 5+1 ;分10の桁範囲(0~5)検査
BRSH MinH ;範囲外で再入力へ
;
SWAP key ;分10の桁値を上位ニブルへ移動(移動後の下位=0)
MOV loiset, key ;分桁値一時保存
LDI lbyte, 0*16+13 ;下位2桁'0n'表示値仮指定
ADD lbyte, loiset ;桁2を直前の入力値に変更
; (桁1:分1の桁入力)
MinL: CLR bounce ;5ms単位キー チャタリング防止計時器初期化
MinL1: RCALL display ;'xxxn'表示,キーパッド入力
SBRs flags, keyok ;チャタリング防止時間(0.2s)経過でスキップ
RJMP MinL1 ;チャタリング防止時間経過まで表示継続/待機へ
;
CBR flags, 1<<keyok ;チャタリング防止時間経過フラグ解除
CPI key, 9+1 ;分1の桁範囲(0~9)検査
BRSH MinL ;範囲外で再入力へ
;
ADD loiset, key ;時桁値作成(時1の桁値合成)
CLR mask ;全seg点滅表示指定解除
RET ;呼び出し元へ復帰(hiset,loisetが入力値)
; * タイマ/カウンタ0溢れ割り込み処理 *
; 実時間時刻基準を提供するために、5ms毎に5msフラグを設定し、点滅やチャタリング防止用と1秒計数器の計数を更新します。
tick: IN status, SREG ;ステータスレジスタ(SREG)値一時保存
INC tock ;5ms単位1s計数器進行
INC blink ;5ms単位点滅間隔計数器進行
INC bounce ;5ms単位キー チャタリング防止計時器進行
SBR flags, 1<<ms5 ;表示時間用5ms間隔経過フラグ設定
CPI tock, 1000/5 ;1s経過判定
BREQ tick1 ;1s経過で分岐
;
NOP ;(実行時間調整)
RJMP tick2 ;1s未経過で分岐
;
tick1: INC second ;秒計数器進行(上限判定以降は主ルーチンで処理)
CLR tock ;5ms単位1s計数器初期化
tick2: LDI timer, 176 ;5ms間計数値取得[(256-n)×256×0.2441μs]
OUT TCNT0, timer ;5ms後溢れ値でTCNT0再設定
OUT SREG, status ;ステータスレジスタ(SREG)値回復
RETI ;割り込み元へ復帰

```



本社

Atmel Corporation

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 487-2600

国外営業拠点

Atmel Asia

Unit 1-5 & 16, 19/F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
Hong Kong
TEL (852) 2245-6100
FAX (852) 2722-1369

Atmel Europe

Le Krebs
8, Rue Jean-Pierre Timbaud
BP 309
78054 Saint-Quentin-en-Yvelines
Cedex
France
TEL (33) 1-30-60-70-00
FAX (33) 1-30-60-71-11

Atmel Japan

104-0033 東京都中央区
新川1-24-8
東熱新川ビル 9F
アトメル ジャパン株式会社
TEL (81) 03-3523-3551
FAX (81) 03-3523-7581

製造拠点

Memory

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 436-4314

Microcontrollers

2325 Orchard Parkway
San Jose, CA 95131, USA
TEL 1(408) 441-0311
FAX 1(408) 436-4314

La Chantrerie
BP 70602
44306 Nantes Cedex 3
France
TEL (33) 2-40-18-18-18
FAX (33) 2-40-18-19-60

ASIC/ASSP/Smart Cards

Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-42-53-60-00
FAX (33) 4-42-53-60-01

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Scottish Enterprise Technology Park
Maxwell Building
East Kilbride G75 0QR
Scotland
TEL (44) 1355-803-000
FAX (44) 1355-242-743

RF/Automotive

Theresienstrasse 2
Postfach 3535
74025 Heilbronn
Germany
TEL (49) 71-31-67-0
FAX (49) 71-31-67-2340

1150 East Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906, USA
TEL 1(719) 576-3300
FAX 1(719) 540-1759

Biometrics

Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex
France
TEL (33) 4-76-58-47-50
FAX (33) 4-76-58-47-60

文献請求

www.atmel.com/literature

© Atmel Corporation 2002.

Atmel製品は、ウェブサイト上にあるAtmelの定義、条件による標準保証で明示された内容以外の保証はありません。本製品は改良のため予告なく変更される場合があります。いかなる場合も、特許や知的技術のライセンスを与えるものではありません。Atmel製品は、生命維持装置の重要部品などのような使用を認めておりません。

本書中の®、™はAtmelの登録商標、商標です。
本書中の製品名などは、一般的に商標です。

© HERO 2021.

本応用記述はAtmelのAVR242応用記述(doc1231.pdf Rev.1231B-05/02)の翻訳日本語版です。日本語では不自然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意識されている部分もあります。必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。