

AVR補助情報



目次

はじめに	3
割り込み	4
内部同期 (システム クロックへの同期化)	5
認識 (割り込み制御部)	5
命令同期 (命令実行待機)	6
応答処理 (割り込み応答実処理)	6
タイマ/カウンタ	7
同期動作タイマ/カウンタの計数開始/停止タイミング	7
同期動作タイマ/カウンタの計数中のタイマ/カウンタ値変更	8
同期動作タイマ/カウンタの前置分周器リセット	8
第1世代プロセス製品と第2世代プロセス製品の違い	8
前置分周クロックとの同期	8

■ はじめに

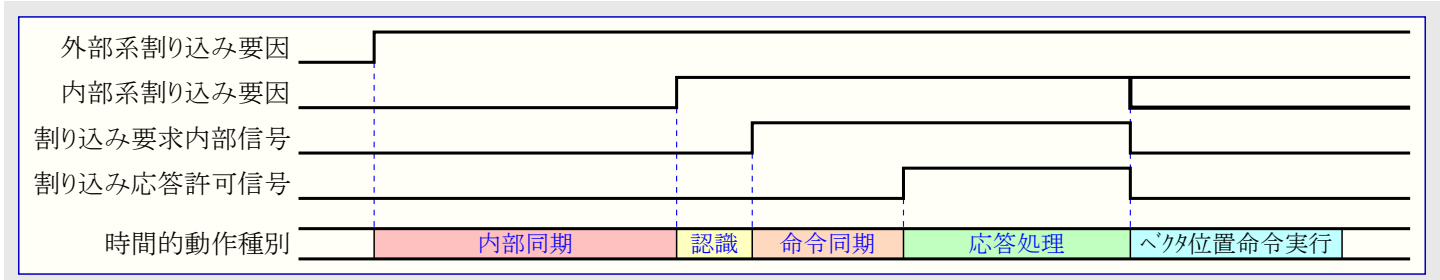
本書は基本的に公開されているAVR関連資料を基に記述していますが、不明確な部分に関して部分的な推測が含まれています。これらは本文中でその旨を記載しています。従って明確でない部分に関してはあくまでも参考補助情報として扱ってください。

■ 割り込み

割り込みには外部割り込みと内部割り込みがあります。外部割り込みはデバイス外の要因によって起こされる割り込みのことで、対して内部割り込みはデバイス内蔵の周辺機能部によって起こされる割り込みのことです。

ここでは主にタイミングに関して考察する関係で、これを外部系割り込みと内部系割り込みとに大別して考えます。これは考え方にもよりますが、前記の外部/内部割り込みの区分と必ずしも一致しません。外部系割り込みは根本の要因がデバイス外で、従ってその要因信号がデバイス内部回路との調停(同期化)を必要とされるものと定義します。これに属するのはINTnの外部割り込み、PCINTnのピン変化割り込み(一部デバイスのLowレベル割り込みを含む)、ICPnのタイマ/カウンタ捕獲割り込み、アナログ比較器割り込みなどです。内部系割り込みは根本の要因がデバイス内で、従ってその要因信号が前記の調停(同期化)を必要としないものと定義します。これは前記以外の割り込みです。

以上を前提に割り込み発生時を考察すると、次のようになります。



外部系割り込み要因の図示信号は具体的な信号レベルを表すものではありません。これはその割り込み条件に合致した要因が発生したことを正論理で示したものです。従って例えばINTnが上昇端割り込みの場合は図示信号と一致します。この外部系割り込み要因の信号をデバイス内信号として扱うため、基本的にシステムクロックによって同期化が行なわれます。この同期化の後で割り込み条件の判定が行なわれます。従って一般的に内部系割り込み要因信号に対して1.5クロック周期以上の遅延が伴います。上図の時間的動作種別の「内部同期」は、この遅延時間を表します。

内部系割り込み要因の図示信号も前記の外部系割り込み要因と同じ意味ですが、ベクタ位置命令実行時に解除(0)されることとしてみれば、基本的にデータシートなどで記載の各割り込み要求フラグ信号と一致します。これらは基本的に外部系割り込み要因信号と異なり、既にシステムクロックと同期していますので、上記の遅延はありません。けれども非同期系タイマ/カウンタをシステムクロック以外で使用する場合にはシステムクロックとの同期が必要になりますので、その場合は外部系割り込み要因として考えなければなりません。

割り込み要求内部信号は外部系及び内部系割り込み要因により起こされた各種割り込み要求信号を割り込み制御回路が認識し、その結果最終的に何らかの割り込みを行なう場合に有効となります。この割り込み制御回路での作業時間は常に1クロック周期で、上図の時間的動作種別の「認識」がこれに相当します。

割り込み応答許可信号は割り込みに対する最終的な応答処理可能を表します。割り込み応答は各命令の各々に対してその実行途中で開始できません。1クロック周期実行命令の場合は問題ありませんが、複数クロック周期命令の場合、その途中で割り込みを実行しようとする、その命令の途中経過を保存しなければならなくなります。これも単一割り込みの場合は簡単ですが、多重割り込みの場合にPC保存と同様なスタック構造が必要となり、ハードウェアの増加や応答速度の低下をもたらす、結果的に何の利点もなくなります。従って各命令の実行間でだけ割り込み応答を開始します。このために割り込み要求内部信号が有効になってから割り込み応答許可信号が有効になるまでに0~(使用最長実行クロック周期命令のクロック数-1)クロック周期の遅延があります。これを上図の時間的動作種別の「命令同期」で表します。割り込み応答許可信号の判定は各命令の先頭ではなく、最後で行なわれるとするため、多くを占める1クロック周期命令の場合は割り込み要求内部信号と同時に有効となります。AVRの最大命令実行クロック数は5クロック周期ですので最大遅延は4クロック周期になります(外部メモリアクセス命令での例外を除く)。

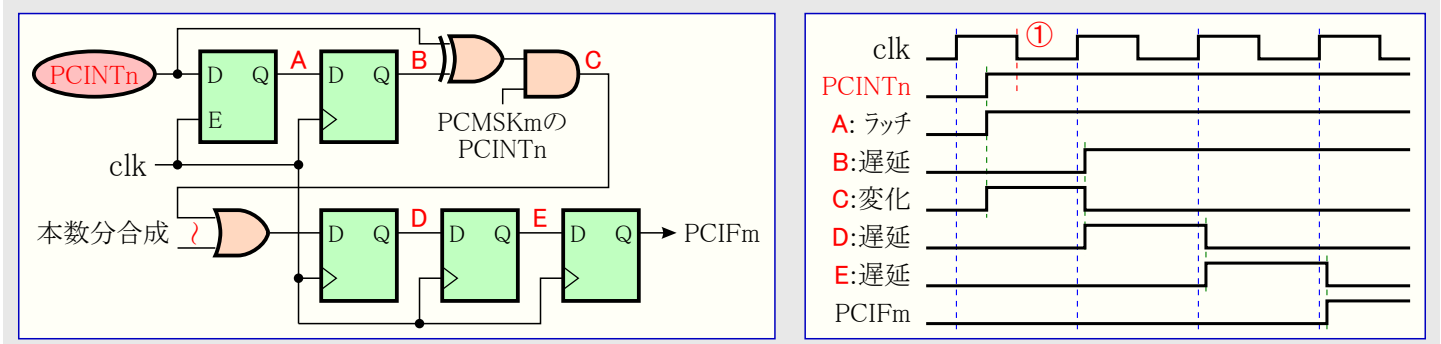
上図の時間的動作種別の「応答処理」が実際の割り込みに対する応答処理部分で、これは基本的にCALL命令と同様な動作を行います。異なるのは呼び出し先を示すPC値が割り込み制御回路によって設定されるのと、SREGの全割り込み許可(I)フラグを解除(0)することだけです。従ってこの応答処理時間はフラッシュメモリが128Kバイト以下のデバイスでは3、128Kバイトを超えるデバイスでは4クロック周期です。この応答処理時間の前後も命令が実行され、応答処理時間だけがプログラムで記述された命令以外の時間であることに注意してください。

割り込み発生時と異なり、割り込みからの復帰時は簡単です。発生時に存在した同期化などの前処理が全く存在しないため、単純にRETI命令の実行だけです。RETI命令は基本的にRET命令とほぼ同じです。異なるのはSREGの全割り込み許可(I)フラグを設定(1)することだけです。これはその割り込み処理ルーチンが実行されたと言うことは、割り込みが発生したと言うことで、直前の全割り込み許可(I)フラグは設定(1)であった、従ってその値を戻すと言う意味です。これが実際には全割り込み許可(I)フラグの保存/復帰ではなく、単純に設定(1)されるだけなことに注意してください。実際に保存/復帰を行なう場合、前述のようにスタック構造を必要とし、ハードウェアの増加と実行速度の低下をもたらしてしまいます。このため全割り込み禁止状態でその状態を保ったまま、割り込み処理ルーチンを一般のサブルーチンとして呼び出すことはできません。

■ 内部同期 (システム クロックへの同期化)

割り込み要因信号自体がシステム クロックに同期している場合を除いて、割り込み要因信号は最終的にシステム クロックに同期しなければなりません。実際には全てに先行して各信号はシステム クロックに同期化され、その後に割り込み条件判定などが行なわれます。従ってこれらの信号は同期化とその後の処理に一定の時間を費やします。

次の図はピン変化割り込みのタイミングです。この場合はPCINTnの変化からPCIFmが有効になるまでに約2.75クロック周期を要しています。純然たる同期化部分は図のB出力までのラッチとFF(フリップ フロップ)で構成された部分です。これは初段のラッチに於いてシステム クロックの周期前半(High部分)で信号値を採取し、次のFFでシステム クロックに同期化させています。ラッチでの採取確定値はシステム クロックの下降(図の①点)に対する準備時間前になり、この準備時間がほぼ0なので、この下降点以前で且つ1つ前の下降点以降の変化が今回の採取になります。従ってここでの遅延は**2.5~3.5クロック周期**になります。



上記の同期化部分は標準I/Oポートを始め、ほぼ全てのデジタル入力で使用されています。従ってこの同期化部分を含む割り込み要因信号は最低、この同期化部分による遅延を伴います。この遅延は上図で示されるように0.5~1.5クロック周期です。

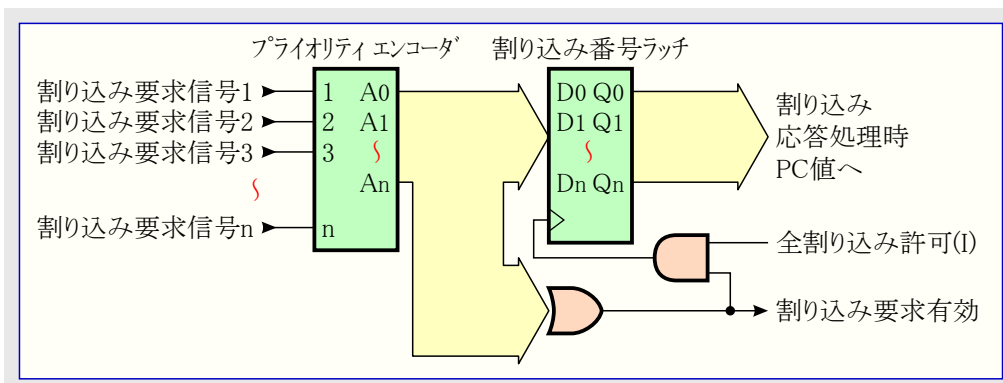
INTn外部割り込み要求に対する明確なタイミングは不明ですが、エッジ系割り込みについてはピン変化割り込みと同等と予測されます。Lowレベル割り込みには上記の同期化部分がなく、論理合わせのインバータを除いて、基本的に直接割り込み制御回路へ接続されていると考えられますので、ここでの遅延はないものと思われます。

アナログ比較器割り込みは上記の同期化部分を含めて、ACSRのアナログ比較器出力(ACO)ビットへの遅延が1~2クロック周期とされているので、その後の割り込み条件判定回路での遅延を含めると、ここでの遅延は**2~3クロック周期**と予測されます。ACOビットへの遅延規定が他と半周期異なっています。これは上記の同期化部分のラッチを省き、直接FFで受けているかもしれません。

アナログ比較器出力は16ビット タイマ/カウンタ1の捕獲起動入力としても使用可能です。この場合はACOビットへの遅延の1~2クロック周期に加えて捕獲起動入力(ICPn)用の雑音除去とエッジ検出に対する遅延を考慮しなければなりません。雑音除去が許可されると信号は4クロック周期遅延します。エッジ検出には最低1クロック周期を要すると考えられます。従ってアナログ比較器出力による16ビット タイマ/カウンタ1の捕獲割り込みは最低でも**2~3クロック周期**の遅延で、雑音除去を許可した場合は最低でも**6~7クロック周期**の遅延となることが予測されます。

■ 認識 (割り込み制御部)

外部系割り込み要因信号はシステム クロック同期化回路によってシステム クロックに同期化された後、割り込み制御部の優先順符号化回路(プライオリティ エンコーダ)に接続されます。内部系割り込み要因信号は既にシステム クロックに同期して生成されますので、システム クロック同期化回路なしで直接、優先順符号化回路に接続されます。この出力が0以外のとき、何らかの割り込み要求が有効であることを示し、その出力は割り込みベクタ アドレスの下位側を示すためにラッチされます。ラッチされた値は応答処理内で新規PC値として用いられます。このラッチ値の最下位ビット(Q0)はフラッシュ メモリが8Kバイト以下のデバイスの場合にはPC0ビット、8Kバイトを越えるデバイスの場合はPC1ビットに対応します。



ここでの遅延は常に**1クロック周期**です。割り込み要求有効信号は各命令の最後で認識されるため、1クロック周期実行命令の場合には割り込み要求有効信号が有効になった周期で認識されることになり、従って割り込み制御部に於ける1クロック周期の遅延は命令実行周期で隠されるとも言えます。そのように考えると、次項の命令同期遅延は命令実行クロック周期と同一と考えても差し支えありません。

■ 命令同期 (命令実行待機)

割り込み制御部で生成された割り込み要求有効信号は各命令の最終実行クロック周期で採取され、その命令の実行完了後、遅延なく直ちに割り込み応答処理が実行されます。4頁の図の割り込み要求内部信号はクロック数説明のために、この割り込み要求有効信号をシステムクロックに同期して生成していることに注意してください。

前項の「認識」でも記述したように、割り込み制御部での遅延が命令同期遅延内に含まれると考えると、割り込み制御部での遅延を考慮しなくてもよくなり、命令実行クロック数がそのまま遅延クロック数として扱えます。従って多くを占める1クロック周期実行命令の場合は、割り込み制御部での遅延と命令同期での遅延を併せて1クロック周期になります。AVRの最長命令実行クロック数は5ですので、この最大遅延は同じ5クロック周期になります。但し外部メモリを使用する場合、設定によっては外部メモリアクセス命令の実行クロック数が5を越える場合がありますので注意してください。また対象アプリケーションでの最大遅延は実際に使用される命令の最大実行クロック数に依存します。

■ 応答処理 (割り込み応答実処理)

各命令の最終実行クロック周期で認識された割り込み要求有効信号により、本来次に実行されるべき命令は破棄され、代わりに動作がCALL命令に類似した割り込み応答処理が実行されます。使用者に関係する部分に関して、この割り込み応答処理は次の3点を除いて、CALL命令と同じ動作です。

- ・ 呼び出し先アドレスは割り込み制御部でラッチされた割り込み番号を元に作成されます(「認識」参照)。割り込み番号で使用されないPC値ビットは0ですが、ブートローダ支援デバイスで割り込みベクタ位置をブートローダ領域に設定している場合、PC値の上位側には対応する値が設定されます。
- ・ ステータスレジスタ(SREG)の全割り込み許可(I)ビットを解除(0)します。これが行なわれなかったなら、レベル系割り込みの場合にベクタ位置の命令で新たな割り込みとして割り込み処理が行なわれ、これを繰り返す結果になってしまいます。
- ・ 割り込み制御部でラッチされた割り込み番号に対応したエッジ系割り込み要因フラグを解除(0)します。

内部系割り込み要因							
プログラムカウンタ(PC)値	\$0F00	\$0F01	\$0F01	N/A	\$0001	\$0002	\$0AAA
プログラム(フラッシュ)バス (主に事前取得)	NOP	N/A	N/A	N/A	RJMP	N/A	PUSH
命令実行	~	NOP	応答処理(CALL命令類似)			RJMP	
データバス	~	N/A	N/A	PUSH PCL(\$01)	PUSH PCH(\$0F)	N/A	N/A

上図は割り込み処理部への割り込み要求が\$0F00番地のNOP命令実行周期の先頭で発生した場合の例です。NOP命令は1クロック周期実行命令ですので、その最終クロックは実行クロック周期そのもので、このクロック周期で割り込み要求は認識され、次のクロック周期から割り込み応答処理が開始されます。この応答処理の基本手順はCALL命令と同じです。

本処理の第1周期で現在のPC値(本例では\$0F01)が一時レジスタに複写され、第2周期以降下位バイトから順にスタックへ格納されます。本例はフラッシュメモリが128Kバイト以下の場合です。128Kバイトを超える場合は3バイトの保存になりますので、割り込み応答処理全体が4クロック周期になることに注意してください。またスタックを外部メモリに配置した場合は、スタックへのバイトアクセスが外部メモリアクセス時間となるため、処理に要するクロック数が大幅に増加することにも注意してください。

本処理内でステータスレジスタ(SREG)の全割り込み許可(I)ビットの解除(0)と割り込み制御部でラッチされた割り込み番号に対応したエッジ系割り込み要因フラグの解除(0)も行なわれます。

本処理の最終クロック周期までに割り込み制御部でラッチされた割り込み番号に対応した割り込みベクタアドレス(本例では\$0001)がPCに設定されます。最終クロック周期で、このPC値の命令が事前取得されます。本例では一般的なRJMP命令で、割り込み応答処理後直ちに実行されます。

データシートで記述される最小割り込み応答時間は上記例のように1クロック実行周期命令と応答処理時間の合計を意味しています。従って対象アプリケーションでの最大割り込み応答時間は、対象アプリケーションで使用される最大実行クロック数の命令のクロック数と応答処理時間の合計になります。応答処理時間はフラッシュメモリが128Kバイト以下のデバイスが3、128Kバイトを超えるデバイスが4ですが、これはスタックを内部SRAMに配置した場合ですので、外部メモリを使用する場合には注意してください。

また外部系割り込み要因の場合は更に内部同期時間が必要で、その要因によって遅延時間が異なることにも注意してください。

尚、本記述は常時割り込み許可(ステータスレジスタ(SREG)の全割り込み許可(I)ビット=1)が前提です。そうでない場合は禁止時間も考慮してください。全割り込み禁止命令はその命令実行周期から禁止が有効になりますが、許可命令は命令の最後で許可されるため許可命令実行時には有効ではなく、次の命令から有効になることに注意してください。従って許可命令の次の命令の最後で割り込みが認識され、その命令に続いて割り込み応答処理が開始されます。換言すると許可命令後、必ず1命令は実行されます。

■ タイマ/カウンタ

タイマ/カウンタで正確な計数や微妙な時間的制御を行なうとき、対応命令に対する正確な動作タイミングが必要になる場合があります。特に計数開始と停止、計数中のタイマ/カウンタ変更(再設定)、前置分周器使用時の前置分周器用カウンタ値などの動作タイミングが問題になります。本節では一部のデバイス特有のタイマ/カウンタを除く、一般的な同期動作のタイマ/カウンタについて、これらを考察します。尚、同期動作のタイマ/カウンタに対する比較動作に於けるタイミングは、基本的に第2世代プロセス製品の場合、対応するデータシートに記載されています。

タイマ/カウンタの制御は対応タイマ/カウンタ関係のレジスタ設定によって行いますが、これらのレジスタの配置位置によって使用できる命令が異なります。これらのレジスタが標準I/O空間にあるデバイスではOUT命令とST系命令が使用でき、拡張I/O空間にあるデバイスではST系命令だけが使用できます。標準I/O空間のレジスタをLDまたはST系命令でアクセスする場合、オペランドのI/Oレジスタ記述に注意してください。この場合、I/Oレジスタ名だけでなく、I/Oレジスタ名+\$20と書かなければなりません。標準I/O空間のレジスタはデバイス定義インクルード(.inc)ファイル内でそのI/Oレジスタ名に対して\$00~\$3FのI/O空間アドレスが.EQUで定義されています。従ってこれを\$20~\$5Fのデータ空間内アドレスにするために+\$20が必要です。

これらのレジスタ設定にOUT命令またはST系命令の何れを使用しても結果は同じです。その命令内でタイマ/カウンタがその命令に対応した何らかの動作を起こす場合はその命令の最終クロック周期で起きます。ST系命令の第1クロック周期で起きることは決してありません。

お断り: 本節の内容はAVR Studioのシミュレータでの動作を基に記述されています。従って実際のデバイスでの動作を保証するものではありませんので注意してください。

■ 同期動作タイマ/カウンタの計数開始/停止タイミング

同期動作のタイマ/カウンタの計数開始は対応タイマ/カウンタ制御レジスタのクロック選択(CSn2~0)ビットを'000'以外に設定することによって行ないます。計数開始/停止タイミングは基本的に第1世代プロセス製品と第2世代プロセス製品で動作が異なります。

○ 第1世代プロセス製品での計数開始

計数開始を指示するOUT命令の実行クロック周期、またはST系命令の第2実行クロック周期が計数クロック周期なら、その命令と同時に計数が開始されます。換言するとその命令と同時にタイマ/カウンタが+1されます。従って前置分周なしの場合、計数開始命令と同時に予め設定されていたタイマ/カウンタ値が必ず+1されます。前置分周を使用した場合、上記設定命令の該当クロック周期が計数クロック周期と一致しない限り、設定命令実行直後のタイマ/カウンタ値は予め設定された値のままで、その後訪れる計数クロック周期から計数を開始します。

○ 第2世代プロセス製品での計数開始

何れの命令による設定に於いても、その命令以降の計数クロック周期から実際の計数が開始されます。その命令の実行周期中は直前からの停止状態が保持されます。

○ 第1世代プロセス製品での計数停止

計数停止に関しても計数開始と同様に停止を指示するOUT命令の実行クロック周期、またはST系命令の第2実行クロック周期が計数クロック周期なら、その計数クロック周期は実行されず(その周期での+1を行わず)、直ちに停止します。ST系命令で行なうとき、第1実行クロック周期が計数クロック周期の場合、その周期で計数が行なわれ(+1され)、第2実行クロック周期で停止することに注意してください。

○ 第2世代プロセス製品での計数停止

何れの命令による設定に於いても、その命令完了直後に停止します。これらの命令中の計数クロック周期は実行(+1)されます。

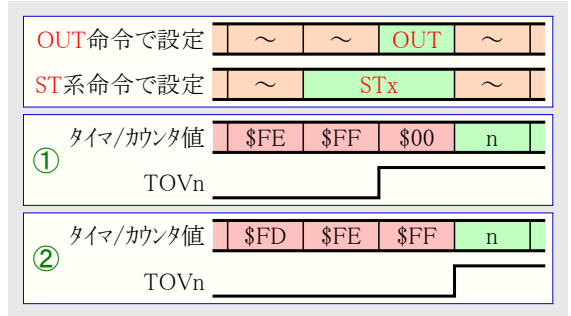
実行命令	OUT命令で設定	~	~	OUT	~	~	~
	ST系命令で設定	~	~	STx	~	~	~
前置分周なし	第1世代プロセス製品	n	n	n+1	n+2	n+3	n+4
	第2世代プロセス製品	n	n	n	n+1	n+2	n+3
前置分周あり (計数周期と一致)	計数クロック周期						
	第1世代プロセス製品	n	n	n+1	n+1	n+1	n+1
第2世代プロセス製品	n	n	n	n+1	n+1	n+1	
前置分周あり (計数周期と不一致)	計数クロック周期						
	第1世代プロセス製品	n	n	n	n	n+1	n+1
第2世代プロセス製品	n	n	n	n	n	n+1	

実行命令	OUT命令で設定	~	~	~	OUT	~	~
	ST系命令で設定	~	~	~	STx	~	~
前置分周なし	第1世代プロセス製品	n-3	n-2	n-1	n	n	n
	第2世代プロセス製品	n-4	n-3	n-2	n-1	n	n
前置分周あり (計数周期と一致)	計数クロック周期						
	第1世代プロセス製品	n-1	n-1	n-1	n	n	n
第2世代プロセス製品	n-1	n-1	n-1	n-1	n	n	
前置分周あり (計数周期と不一致)	計数クロック周期						
	第1世代プロセス製品	n-1	n-1	n-1	n	n	n
第2世代プロセス製品	n-1	n-1	n-1	n-1	n	n	

■ 同期動作タイマ/カウンタの計数中のタイマ/カウンタ値変更

計数中に於けるタイマ/カウンタ値変更(再設定)は、その設定命令実行完了後にタイマ/カウンタ値が新しい値に置き換えられます。従ってその設定命令中の計数クロック周期は実行され(+1され)、その結果が割り込み条件に該当する場合は対応する割り込み要求フラグも設定(1)されます。

右図は前置分周なしの8ビットタイマ/カウンタの例ですが、①の場合は設定命令実行後、直ぐに溢れ割り込み(TOV_n)が発生します。②の場合は一見すると溢れが起こっていないように見えますが、タイマ/カウンタ値が\$FFの周期の最後で、計数とそれに続いて新規値設定が行なわれます。従ってこの周期の最後でTOV_nが設定(1)されます。この場合の割り込みはタイマ/カウンタ値設定命令後ではなく、その次の命令実行後に行われることに注意してください。



■ 同期動作タイマ/カウンタの前置分周器リセット

基本的に第1世代プロセス製品は任意に前置分周器用カウンタをリセットできませんが、第2世代プロセス製品ではこれを行なえます。現在、このリセットに関するタイミングは不明ですが、計数開始/停止動作時の動きと基本的に同じならば、前置分周器用カウンタリセット命令の最終実行クロック周期の最後で前置分周器用カウンタがリセット(=0)されると思われます。その場合、その周期が前置分周数に1回訪れる計数周期なら、その計数周期は有効で、前置分周器用カウンタのリセットと計数が同時に行なわれると予測されます。

■ 第1世代プロセス製品と第2世代プロセス製品の違い

基本的に第1世代プロセス製品と第2世代プロセス製品間での際立って大きな違いはありませんが、前置分周あり時の比較一致タイマ/カウンタ解除(CTC)動作などでの比較一致直後の動作が異なります。タイマ/カウンタに関する実際の変化点は計数クロック周期の最後、換言すると計数クロック周期と次のクロック周期間と考えられます。

第2世代プロセス製品では一般的に考えられるよう動作します。即ちタイマ/カウンタ値は比較値に達すると、その値を前置分周したクロック数分の間維持し、次の計数クロック周期で0に再設定されます。従って0から比較値までの各値は全て前置分周したクロック数分ずつになります。

これに対して第1世代プロセス製品では比較一致時のクロック周期の最後でタイマ/カウンタが0に再設定されます。このため前置分周値の値に拘らず、比較値となるタイマ/カウンタ値は常に1システムクロック分となり、再設定後の0値は常に前置分周値-1システムクロックの長さになります。即ち比較値と再設定後の0値は、本来前置分周した2クロック分のところを1クロック分で処理されます。従って比較値は一般的に考えられる値+1でなければなりません。

「計数開始」と「計数停止」で記述されるように、第1世代プロセス製品と第2世代プロセス製品間で計数許可/禁止に関して1システムクロックの差があるようです。多くの応用ではこれが問題になることはないと思いますが、タイマ/カウンタの計数に関して微妙な扱いを行なっている応用プログラムを異なる世代プロセス製品に置き換える場合には十分に注意してください。

■ 前置分周クロックとの同期

応用によっては前置分周したクロックに同期して操作を行いたい場合があると思います。例えば溢れ割り込み(TOV_n)を使用した定周期繰り返し計時器などの場合です。これが最大値周期(8ビットタイマ/カウンタの256、16ビットタイマ/カウンタの65536)の場合はタイマ/カウンタ値の再設定が必要ないため問題ありませんが、他の周期の場合は割り込み処理ルーチン内でタイマ/カウンタ値の再設定が必要になり、一挙に面倒になります。これも前置分周なしの場合はタイマ/カウンタ値を読み取り、その後の再設定までの命令実行クロック数分補正した値でタイマ/カウンタを再設定することによって比較的簡単に実現可能です。

問題は低分周値の前置分周が使用されている場合で、この場合は命令実行を前置分周したクロックと同期化させ、問題のないように特定の位置でタイマ/カウンタを再設定しなければなりません。このために命令実行のどのタイミングで計数クロック周期になるかを判定しなければならなくなります。大きな前置分周値の場合は割り込み発生後の次の計数クロック周期までにタイマ/カウンタを再設定すればよいので、長い割り込み禁止時間がある場合や多重割り込みなどで複雑になっている場合を除いて、比較的簡単に実現できます。

低分周値の場合でも問題のない位置で再設定するために、NOP命令追加などで割り込み発生からタイマ/カウンタ再設定までの命令クロック数を調整する方法を使用できますが、それには割り込み発生からタイマ/カウンタ再設定までのクロック数が既知でなければならないため、多くの場合に多重割り込みや使用命令が制限されます。とは言え多くの場合、この方法は中程度の分周値の場合に有効な方法です。

低分周値で且つ多重割り込みや使用命令の制限なく行なうには、プログラムで計数クロック周期位置を特定し、それ以外の位置で且つその位置に対応した補正值でタイマ/カウンタを再設定しなければなりません。例えば8分周時の8ビットタイマ/カウンタに於ける1つの方法として、8つのレジスタへの連続するIN命令によるタイマ/カウンタ値取得、取得値から変化点を検出、その後の設定命令までの実行クロック数補正とその位置での設定値補正、の一連の手順があります。16ビットタイマ/カウンタについても変化点検出は下位バイトだけでよいので、同様な手法が使用できます。勿論最初にその時点のタイマ/カウンタ値を取得し、その直後で新たにタイマ/カウンタ値を取得して最初の値と比較し、不一致になるまで繰り返すことで、計数クロック周期でない位置を特定することはできます。低分周値時、この方法で多くの場合は対処可能でしょう。



© HERO 2014.

本書はAVRの補助情報の記録です。

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。