

## 100 絶滅危惧種 百円マイコンの 再評価の経緯

067 絶滅危惧種 百円マイコン 比較レビュー  
動画に、コメントを書き込まれた事に すぐに気付かず、5ヶ月経過した後に、気付きました。

で、コメントの内容に「*i<60000*ではなくて *i<30000* になっているようです」と、書き込まれていたので、びっくりして、どこに書き込まれているのか確認したところ AVRマイコンの ATMEGA328P の プログラムソースでした。

R8Cマイコン( R8C/M110A )と PICマイコン( PIC12F635 )は、1本しか Projectを作成していないので、どちらも確認しましたが、両方とも空ループの for 文は、“*i<60000*” でした。

AVRマイコンは、当初使用予定の百円マイコンの AT90S2313 は、何と AVRマイコン開発環境 AVR Studio 4 と Atmel Studio 7.0 で 確認しましたが 対応していませんでした。

AT90S2313は、かなり初期のマイコンで、AVRマイコンは、途中でマシン語の仕様が変わったらしく、それ以前のマイコンは 切り捨てられたのかもしれません。 AT90S2313は、有っても全く使えないでの、ゴミと化してますね。 秋月電子さんこんな古い使えないICは、売らないで下さいね。

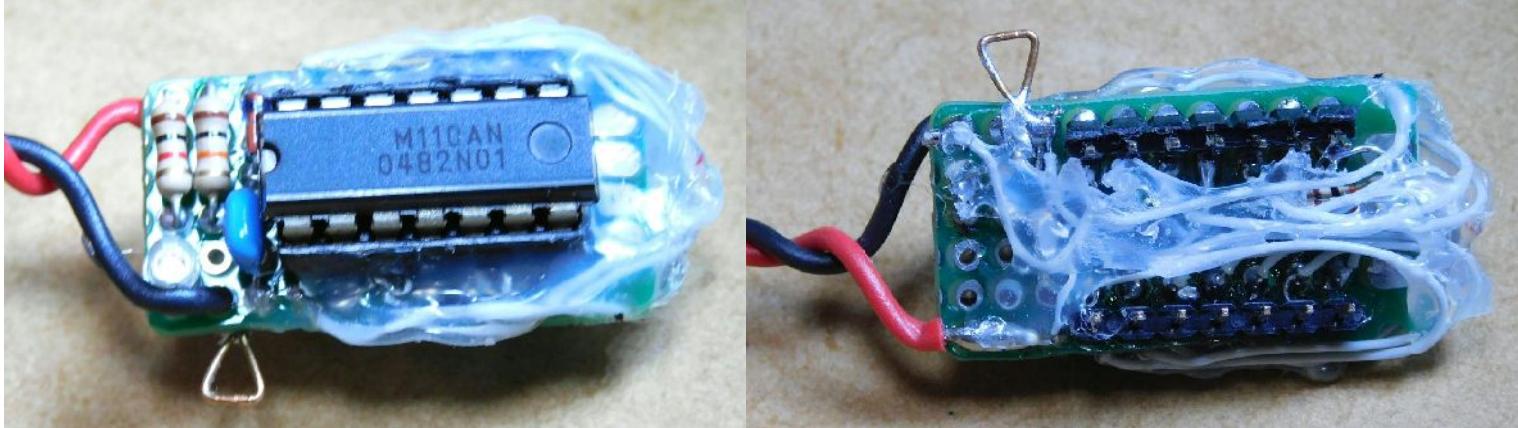
という事で 067の動画作成時に、困り果てていました。 で、書き込み器として使用していた秋月電子の Arduino互換基板に ATMEGA328Pが載っていたので、百円マイコンでは無いけど、CPUコアは、AVRマイコンなので ATMEGA328Pを使って速度比較をする事にしました。

で、どこで *i<30000* に入れ替わったのかは、分かりませんが、AT90S2313用のProjectのソースは *i<60000* に、なっていました。 ATMEGA328Pに 移植する際に、間違って *i<30000* に なった可能性が 高いです。

という事で、今回の動画は 拍子抜けの結末となってしまいました。これでは、せっかく来られた視聴者の方々に申し訳ないので、他に何か 見せるものを用意する事にします。

2つ前の動画で ダーリントン接続のパルス応答を計測する用途で R8C/M110Aマイコンを方形波パルス発振器として使用しました。

今回は、その詳細を 説明します。  
その時作成した小基板は ブレッドボード上で R8C/M1マイコンを使えるようにする目的も ありました。まずは、外観をお見せします。



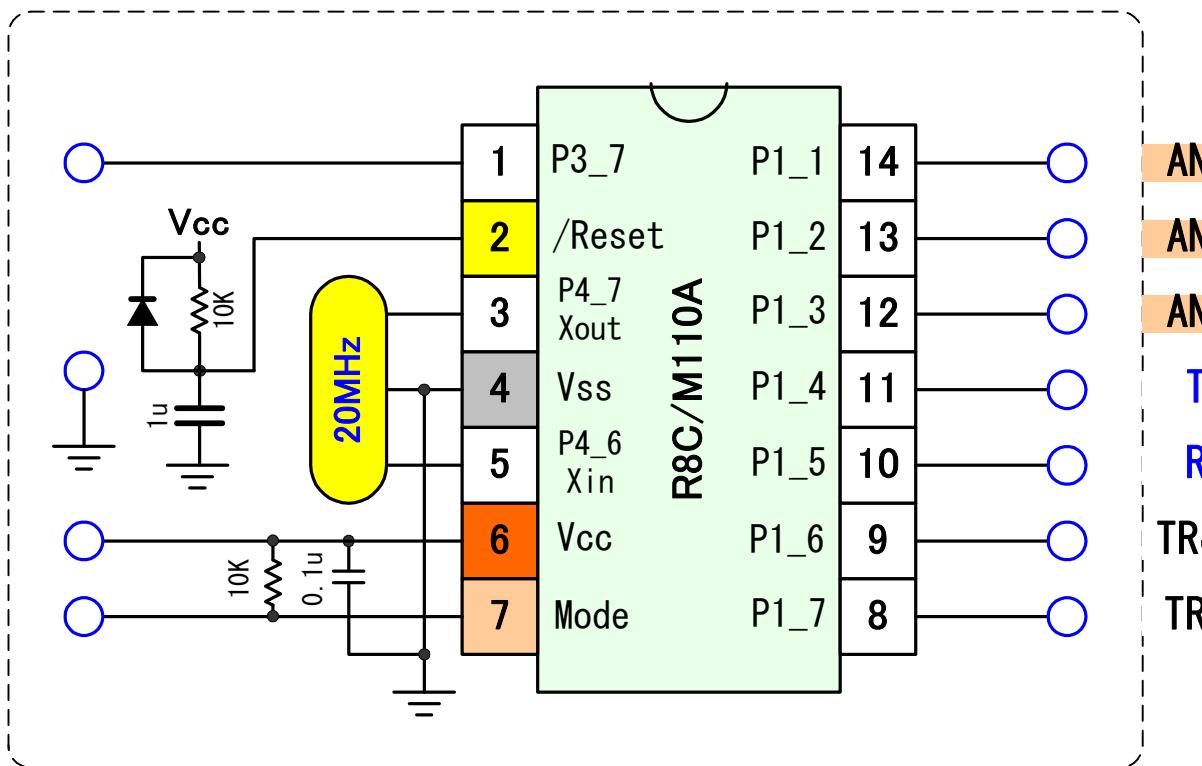
一応、ブレッドボードに マウント出来ますが、ハンダ付け時、ピンをブレッドボードに挿して 半田付けしてましたが、片方の列の足が 曲がってしました。一応ブレッドボードには挿せます。

赤、黒のリード線は 電源 5V とアースです。  
基板の横から 三角の細い銅線が出てますが、アース端子です。足ピンをハンダ付けするのに ICソケットは、裏面、連結ピンは、表面で、マイコンの各ピンの裏と表を リード線で接続したので配線が面倒でした。で、抜き差ししている時に配線が、切れる恐れもあると思い、ホットメルト

ボンドで、配線を包み込んだようにしています。

マイコン使用時に毎回必要となるパワーONリセット回路等は、子基板側に 実装してます。

## 今回の R8C/M110Aマイコン小基板 回路図



ブレッドボード上で インサーキット書き込みを行う場合

- ① 4pin: Vss(GND)、② 6pin: Vcc、③ 7pin: Modeを GND に接続、④ 11pin:TxD、⑤ 9pin: Pgm RxD に接続する。

ここに書いた機能は、一部分で  
ピンに割り付けられる機能は、まだ  
多数あります。

インサーキット書き込みですが  
TxD、RxDの電源レベル、及びブレ  
ッドボード側の電源と、TxD、RxDが  
USBから供給される別電源の場合  
電源の ON、OFFシーケンス等に  
注意する必要があります。

TRJ0 Pgm RxD

TRJ10

プログラム書き込み時 無難なのは、  
小基板に ICソケットを付けて  
マイコンに プログラムを書き込む  
時は、ターゲットの電源を 切って  
ICソケットから マイコンを取り外し  
て、ライターで書く方が 安全です。

## 高速な 定周期の 方形波パルス出力について

マイコンのプログラムによって、ON OFFパルスを出すのは、比較的ゆっくりした周期であれば、出せない事もないですが、**ソフトで周波数を出そうとすると、プログラムの分岐処理等の影響で 多少ジッタが 発生します。** パルスを出す周期のタイミングを タイマー周辺回路で作りだす事はあります。**タイマー割り込みで回すのは 最速で 10KHzぐらいが限度**と思います。

単純に、それ以上早いパルスを出したい場合は、マイコンのタイマー周辺回路に完全に任せた方が、安定した出力が出せます。

R8C/Mシリーズのマイコンは、タイマー周辺回路が、3種類あります。上位機種では、もつと多数のタイマー周辺回路を 実装してます。

今回は、PWMのような デューティ比の変調は出来ませんが、簡単にパルス出力、あるいはパルスカウントが出来る TRJ2 タイマー周辺回路を 使用します。前回も この TRJ2を使って 1MHz の パルスを出力しました。

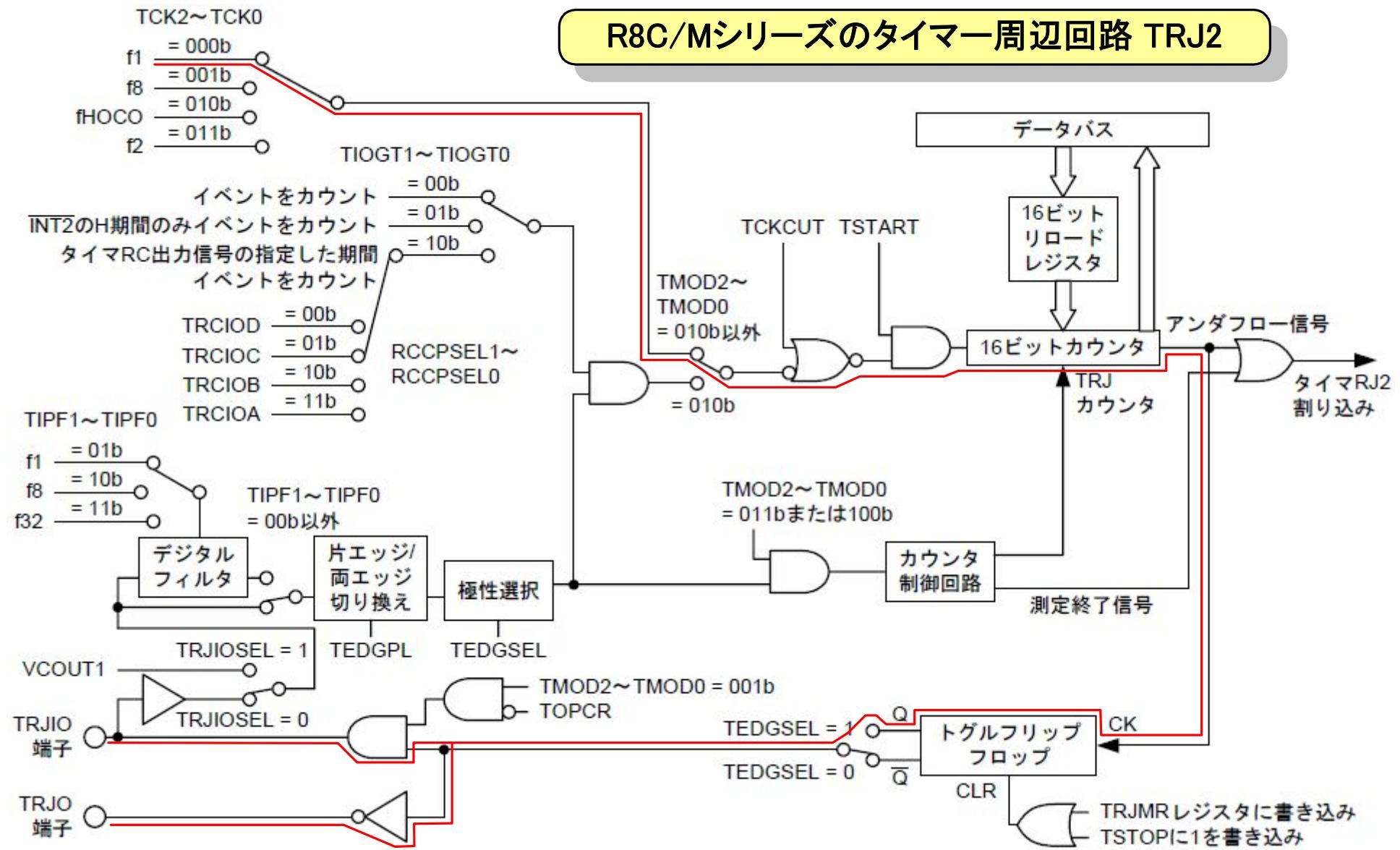
まだ、やった事はないですが、TRJ2の ブロック図を見る限りは 10MHzまで 出力できそうな気がします。最終的に 出口の I/Oポート部分で 10MHzが 出せるかが、気になるところです。

これは、出せるかどうか実験してみようと思います。

それと、TRJ2の信号出力は TRJOと TRJIOの2つが、ありますが、この2つは、極性が 反転します。これは、圧電ブザーを使うとき 面白い使い方が出来うるので 試してみます。

次に TRJ2タイマー周辺回路の ブロック図を示します。

## R8C/Mシリーズのタイマー周辺回路 TRJ2



## R8Cパルス発生器の ソフトの機能

前回の R8Cパルス発振器のソフトは、**10KHz**、**100KHz**、**1MHz**の 3通りの周波数を出せるようにしておきましたが、今回は 改造して **4bit**で **0 ~ 15** の値を指定して、右の表の周波数を出力出来るようにしました。

最低出力周波数は、**200Hz** で、最高出力周波数は、**10MHz**を出力させます。（**10MHz**を出す事は 出来ました。）

ちなみに 最低出力周波数 **200Hz**は、16bitの分周値で、**50000** となるので、それより低い周波数は、16bitのレンジを超えてしまうので出力出来ないという事です。（出来れば、**100Hz**を 出したかった。）

ややこしい話は、これぐらいにして実際に動かして信号を見てみましょう。

0	*** 無し ***				
No,	周波数	分周値	Port0	ビット設定	
1	200Hz	50000	( 1Ch : ***1 110* )		
2	500Hz	20000	( 1Ah : ***1 101* )		
3	1KHz	10000	( 18h : ***1 100* )		
4	2KHz	5000	( 16h : ***1 011* )		
5	5KHz	2000	( 14h : ***1 010* )		
6	10KHz	1000	( 12h : ***1 001* )		
7	20KHz	500	( 10h : ***1 000* )		
8	50KHz	200	( 0Eh : ***0 111* )		
9	100KHz	100	( 0Ch : ***0 110* )		
10	200KHz	50	( 0Ah : ***0 101* )		
11	500KHz	20	( 08h : ***0 100* )		
12	1MHz	10	( 06h : ***0 011* )		
13	2MHz	5	( 04h : ***0 010* )		
14	5MHz	2	( 02h : ***0 001* )		
15	10MHz	1	( 00h : ***0 000* )		