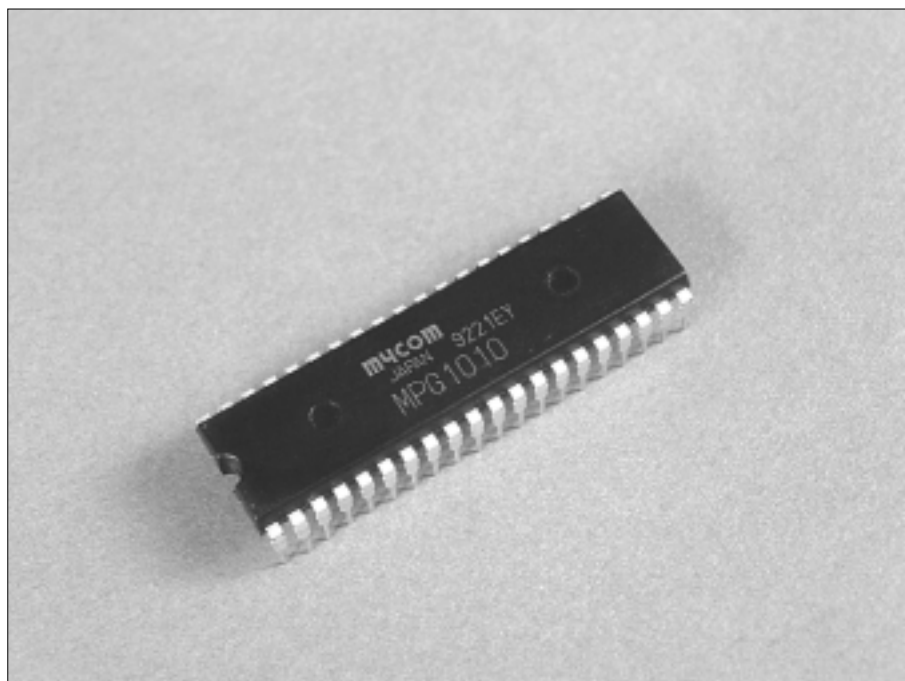


# パルスジェネレータ MPG1010

最高速、最大8MHzの出力パルスレートを達成。  
特定基準クロックを必要としない出力周波数設定。  
最適な減速タイミングをハードウェアで高速演算。  
全ての回転域で滑らかに加減速。

## 〈特 徴〉

- ステッピングモータ・パルス列入力サーボモータ用パルスジェネレータLSI
- 最大8MHzの出力パルスレート（基準クロック16MHz時）
- 出力パルス分周設定による、特定基準クロックを必要としない出力周波数設定
- 65,535刻みの出力周波数および加減速傾斜設定
- 減速開始点自動演算機能
- 滑らかな出力周波数、加減速

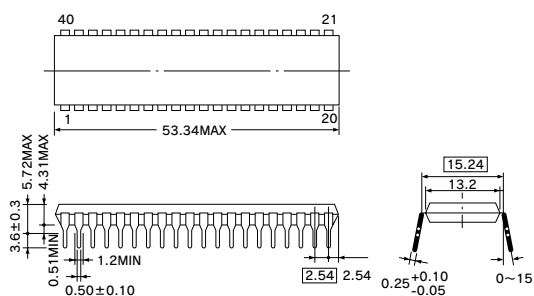


# MPG1010 仕様

項目	内容
電源電圧	DC+5V±5%
基準クロック	131K~16MHz
総パルス数設定範囲	1~16,777,216パルス
減速開始設定範囲	0~16,777,215パルス
周波数設定ステップ数	65,536 [step]
加減速傾斜設定ステップ数	65,536 [step]
出力パルス分周ステップ数	256 [step]
出力周波数範囲例 [Hz]	0.004~256 (CLK=131KHz, 分周比=256) 1~65,535 (CLK=2.1MHz, 分周比=61) 122~8,000,000 (CLK=16MHz, 分周比=1)
動作及び機能	台形及び三角駆動 減速開始点自動演算機能 フォワード/リバースオーバーラン停止 スローダウン停止 即時非常停止 サーボモータ用パルス出力完了制御 サーボモータ用偏差カウンタオーバーフローによる非常停止 出力パルス数モニタ 加減速速度別設定
汎用入出力端子	2端子
パルス出力	1パルス/2パルス切替
パッケージ	40ピンDIP リードピッチ 2.54mm (100mil)

## 外形寸法(mm)

40 Pin Plastic DIP (600mil)



# MPG1010 特徴的な性能

## 1) 最大8MHz出力パルスレート

高速化の要請に先駆けて、必要十二分な出力パルスレート（パルスの周波数）を達成しました。

## 2) 特定基準クロックを必要としない出力周波数設定。

従来は特定の周波数の基準クロックを使用するものが多く、そのため特定周波数の水晶発振器などが必要でしたが、MPG1010では出力パルス分周比設定レジスタ（1~256）により、基準クロックに限定されずに任意の出力周波数範囲が設定可能です。これによりマイコンやパソコンから供給されるクロックを直接利用して、制御システムを作れるようにしました。

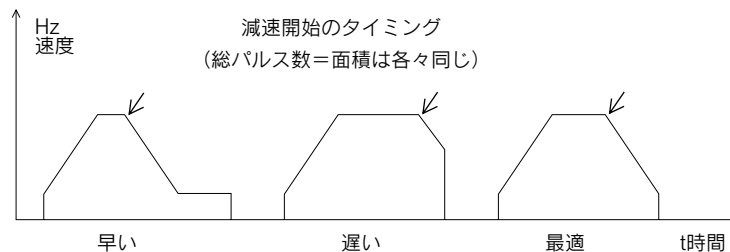
最大16MHzの基準クロック入力が、可能です。

同一の基準クロックから256種の出力周波数範囲が設定可能で、いずれも速度・傾斜設定は65,535刻みです。

## 3) 減速開始点自動演算機能

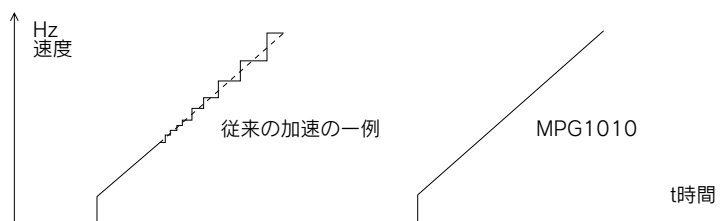
加速した後、減速をして停止する場合、発生するパルスの総数は希望した通り正確でなければなりません。しかし減速開始のタイミングが早ければ最後に低速でもたつき、タイミングが遅ければ充分低速に達しない前に停止するため、慣性で設定位置よりズレてしまいます。

減速開始点の計算は従来ソフトウェアで行いましたが、最適なタイミングで減速を自動的に開始することができます。



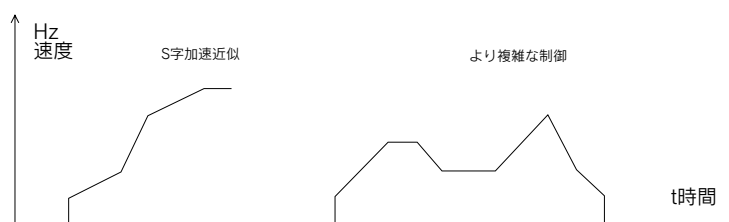
## 4) 滑らかな出力周波数・加減速

全て出力パルスレート帯域で極めて滑らかに加減速（出力パルスレートの増減）を行います。従来は直線的加減速と言っても高回転での傾斜が段階状であったりしました。またジッター（一定速度での出力パルスレートのブレ）も0.0015%~0.000006%（理論値）と最高レベルです。



## MPG1010 特徴的な性能

加速と減速の傾斜（出力パルスレートの増減の割合）の設定が2つのレジスタ（データを与える所）で別々に設定できます。また動作中に速度（出力パルスレート）や傾斜を自由に変える複雑な制御が可能です。これらにより、理想的とされているS字加減速などの折線近似が可能です。例としてS字加速とは、最初は比較的ゆっくり加速して徐々に加速を増し、最高速に達する前で再度ゆっくり加速するものです。



また速度や加速度レジスタに与えるデータは、希望データをそのまま2進数に変換するだけで計算不要のモードがあります。10,000 [Hz] まで1,000 [Hz/msec：加速度の単位の一つ] で加速したい場合、単純に1万と1千を与えればよいのです。

### 5) その他

本LSIはCMOSロジック構成になっており、既成のプログラムを焼き込んだいわゆるワンチップマイコンではありませんので、チップ自体が暴走してしまうという事はありません。