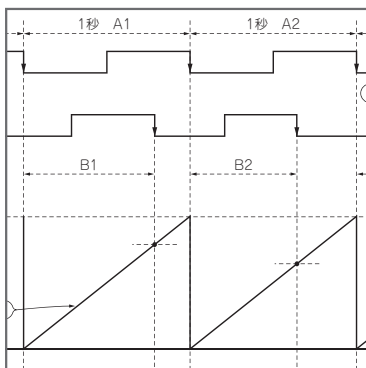




第7章

CPUレジスタ直接操作で性能を引き出す

Uno R4 真の実力テスト① 32ビットPWMタイマ

下間 憲行 Noriyuki Shimotsuma

Arduino Uno R4には32ビットのPWMタイマが2チャンネル入っています。この他にも16ビットのPWMタイマが6チャンネルあり、豊富なカウント・モードとパルス出力機能を持っています。

本稿では32ビット・タイマを使ったさまざまなカウント入力を試してみます。

ライブラリを使うのではなくCPUのレジスタを直接操作して性能を引き出します。従って、はじめにArduinoのI/Oピン番号とCPUのI/Oポート番号の関係を理解しておく必要があります。CPUのユーザーズ・マニュアルではI/Oポート番号として機

能が解説されています。まとめたものを表1に示します。本稿では汎用PWMタイマをGPT1といいます。

周波数カウンタを使いこなす

● 便利に使える、ゲート時間の短縮とクリア機能

32ビットのタイマGPT1を使った周波数カウンタを図1、リスト1、図2に示します。外部からやってくるパルスを32ビット・タイマで数えます。ゲート時間を1秒にすれば、数えたパルス数がHz単位の周波数になります。測定した周波数はシリアルで出力しています。シリアル・モニターで確認してください。

Uno R3のタイマ・カウンタ1は16ビットでした。高い周波数をカウントしようとすると桁が足りません。そこで、ゲート時間を短くしてカウント値を積算するという手法を用いました。Uno R4ではゲート時間を長くしても余裕です。

もう1つ便利なのがカウンタのクリア機能です。1秒ゲートを作るタイマ・インプット・キャプチャ入力

表1 Arduino Uno R4のI/Oピンと搭載マイコンRA4M1のI/Oポート対応

ライブラリを使わず、CPUのレジスタを直接操作して性能を引き出す。*印は本稿で使う信号

Arduino Uno R4のI/Oピン番号	RA4M1マイコンのポート番号
D0	P301
D1	P302
D2	P105
D3	P104
D4	P103
D5	P102
D6	P106
D7	P107
D8	P304
D9	P303
D10	P112
D11	P109
D12	P110
D13	P111
D14/A0	P014
D15/A1	P000
D16/A2	P001
D17/A3	P002
D18/A4	P101
D19/A5	P100

*：記事で使う信号

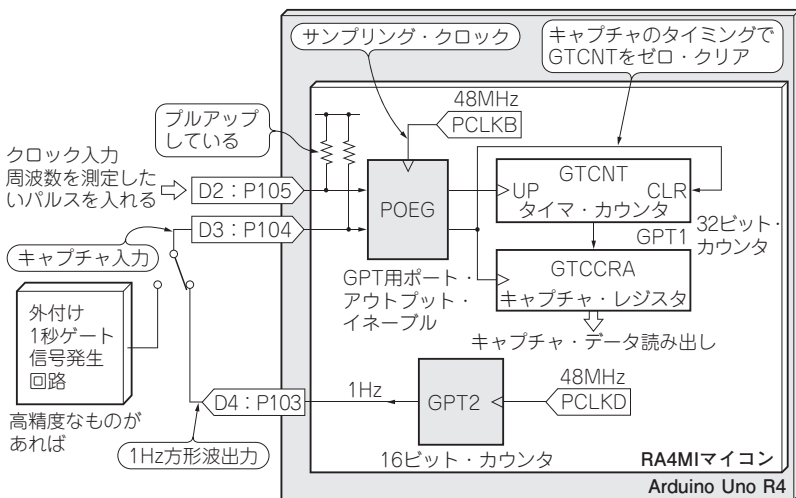


図1 GPT1を使った周波数カウンタ構成

外部から来るパルスを32ビット・タイマで数える。ゲート時間を1秒とすると、数えたパルス数がHz単位の周波数となる