

S1C17W23

光電式容積脈波記録法(PPG)

アプリケーションノート

評価ボード・キット、開発ツールご使用上の注意事項

1. 本評価ボード・キット、開発ツールは、お客様での技術的評価、動作の確認および開発のみに用いられることを想定し設計されています。それらの技術評価・開発等の目的以外には使用しないで下さい。本品は、完成品に対する設計品質に適合していません。
2. 本評価ボード・キット、開発ツールは、電子エンジニア向けであり、消費者向け製品ではありません。お客様において、適切な使用と安全に配慮願います。弊社は、本品を用いることで発生する損害や火災に対し、いかなる責も負いかねます。通常の使用においても、異常がある場合は使用を中止して下さい。
3. 本評価ボード・キット、開発ツールに用いられる部品は、予告無く変更されることがあります。

本資料のご使用につきましては、次の点にご留意願います。

本資料の内容については、予告無く変更することがあります。

1. 本資料の一部、または全部を弊社に無断で転載、または、複製など他の目的に使用することは堅くお断りいたします。
2. 本資料に掲載される応用回路、プログラム、使用方法等はあくまでも参考情報であり、これらに起因する第三者の知的財産権およびその他の権利侵害あるいは損害の発生に対し、弊社はいかなる保証を行うものではありません。また、本資料によって第三者または弊社の知的財産権およびその他の権利の実施権の許諾を行うものではありません。
3. 特性値の数値の大小は、数直線上の大小関係で表しています。
4. 製品および弊社が提供する技術を輸出等するにあたっては「外国為替および外国貿易法」を遵守し、当該法令の定める手続きが必要です。大量破壊兵器の開発等およびその他の軍事用途に使用する目的をもって製品および弊社が提供する技術を費消、再販売または輸出等しないでください。
5. 本資料に掲載されている製品は、生命維持装置その他、きわめて高い信頼性が要求される用途を前提としていません。よって、弊社は本（当該）製品をこれらの用途に用いた場合のいかなる責任についても負いかねます。
6. 本資料に掲載されている会社名、商品名は、各社の商標または登録商標です。

要旨

本資料は、S1C17W23 にて、LED(発光ダイオード)と光学センサ(フォトトランジスタまたはフォトダイオード)を用いた光電式容積脈波記録法(PPG: Photo Plethysmography、フォトプレチスモグラフィ)により脈波を計測して、心拍数の値を得るための参考資料です。

動作環境

- S5U1C17W23T (以下 SVT17W23: Software Evaluation Tool for S1C17W23)
ICDmini との接続には専用ケーブル (4pin－4pin) 2 種が必要です
- ICDmini (S5U1C17001H)
パソコンとの接続には USB ケーブルが必要です
- パソコン
 - ・ GNU17 (S5U1C17001C) 開発ツールインストール済み ※
 - ・ ICDmini USB ドライバインストール済み
- 最新版の FLS17W23 (ファイル名: fls17w23.elf)
内蔵 FLASH へのプログラムの書き込みに必須
- S1C17W23 PPG 用プログラミングパッケージ (本パッケージ)
 - ・ 光学センサの出力値の時間ごと変化の視覚化 プログラミングパッケージ
(slc17w23_ppg_mon_gnu17vx)、VBA マクロを含む Excel ファイル (PpgMon64.xlsm)、
VBS ファイル (DoPpg64.vbs)、Active-X コントロールファイル (NonComSck.ocx)
 - ・ 脈波から心拍数を計算してその値を LCD 表示 プログラミングパッケージ
(slc17w23_ppg_demo_gnu17vx)

※ 本パッケージは、GNU17 V2.3.0 で動作確認しています。

目 次

要旨	i
動作環境	i
1. 仕様	2
2. 使用機能説明	3
3. 動作原理	5
3.1 検出原理	5
3.2 検出方法	6
3.3 検出回路	7
3.4 FIR フィルタ	11
3.5 矩形波相関フィルタ	12
4. ソフトウェア説明	13
4.1 s1c17w23_ppg_mon_gnu17vx について	13
4.1.1 ファイル構成 (src 内)	13
4.1.2 ファイル構成 (inc 内)	13
4.1.3 モジュール説明	14
4.1.4 操作手順	16
4.1.5 PpgMon64.xlsm の使い方	18
4.1.6 サンプルプログラム動作概要	20
4.2 s1c17w23_ppg_demo_gnu17vx について	22
4.2.1 ファイル構成 (src 内)	23
4.2.2 ファイル構成 (inc 内)	23
4.2.3 モジュール説明	24
4.2.4 操作手順	27
4.2.5 サンプルプログラム動作概要	28
Appendix A. 脈波検出実験の一例	31
(1) 計測波形の安定に必要な時間	31
(2) FIR フィルタ、矩形波相関フィルタの効果の確認	32
参考文献	36
改訂履歴表	37

1. 仕様

1. 仕様

本アプリケーションノートは、S1C17W23 に内蔵されている AD コンバータ (ADC12A) で取り扱うのに適した電圧レベルの波形を得るために、光学センサ (フォトトランジスタまたはフォトダイオード) の出力信号を AFE で増幅、フィルタリング処理しています。

サンプルプログラムのうち、s1c17w23_ppg_mon_gnu17vx は、VBA マクロ、VB スクリプト (VBS) を含む Excel と連携して、一定の周期で光学センサの出力値をサンプリングし、UART を用いて PC に送信して Excel の表に書き込みます。Excel の表に書かれた配列の値はグラフ化されるため、時間ごと光学センサ出力値の変化を視覚化することができます。

またサンプルプログラムのうち、s1c17w23_ppg_demognu17vx は、16 ビットタイマ (T16) で一定時間ごと繰り返し光学センサの出力値を取り込み、脈波から心拍数を計算してその値を LCD に表示します。あわせて、脈波と拍動の様子もリアルタイムで表示します。

図 1-1 は、Appendix A で用いた評価システムの写真です。

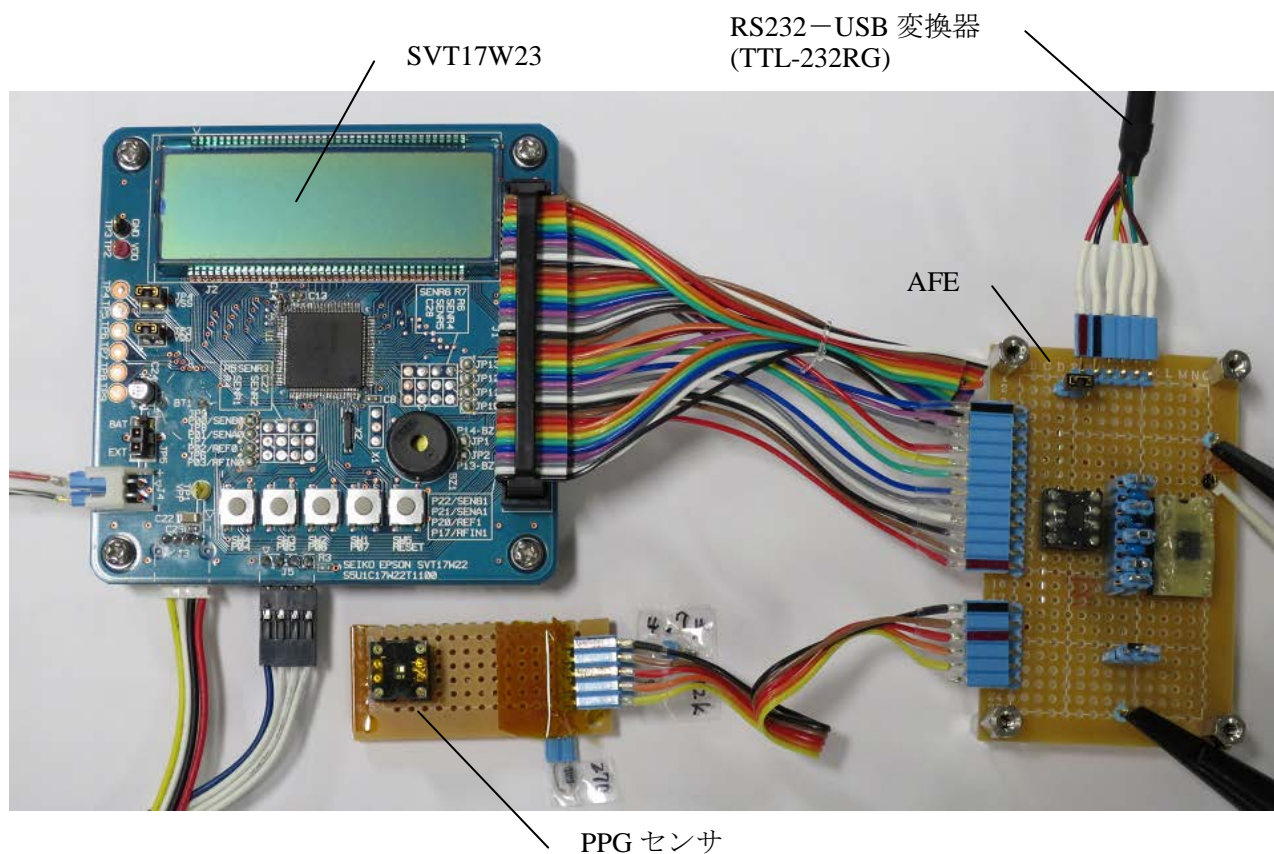


図 1-1 Appendix A で用いた評価システム

2. 使用機能説明

PPOINT	P10、P15、P16、P17、P20 計 5 ポートを FET のゲート、アナログスイッチに接続し、LED の駆動、アナログ回路の切換に用います。
ADC12A Ch.0	AFE で増幅、フィルタリング処理された光学センサの出力値をデジタル値に変換するために使用します。
T16 Ch.0	ADC12A を定期的に動作させるためのインターバルタイマ。
T16 Ch.1	間歇駆動時にサンプリングするまでの安定化待ち時間用タイマ。
T16 Ch.3	ADC12A の動作クロックを発生させるためのタイマ。

以下は、プログラム `slc17w23_ppg_mon_gnu17vx` の場合です。

UART Ch.0	UPMUX で PPOINT の P36 を USIN0 に、P37 を USOUT0 に割り当てます。PC 上で実行する VBA マクロ、VB スクリプトを含む Excel との通信に使用します。
-----------	---

以下は、プログラム `slc17w23_ppg_demo_gnu17vx` の場合です。

LCD24	脈波から心拍数を計算してその値を LCD に表示するために使用します。
-------	-------------------------------------

システムクロック システムクロックは OSC3（内蔵発振 4MHz）を使用します。
`slc17w23_ppg_demo_gnu17vx` は OSC1（32.768kHz）も使用します。

割り込み	ADC12A Ch.0 のベクタ番号とベクタアドレスは以下の通りです。 ADC12A Ch.0 ベクタ番号： 26 (0x1a) ベクタアドレス： 0x8068 本サンプルでは、次の 2 種類の割り込みを使用します。 アナログ入力信号 0 A/D 変換完了割り込み アナログ入力信号 0 A/D 変換結果オーバーライト エラー割り込み
------	---

T16 Ch.0 のベクタ番号とベクタアドレスは以下の通りです。
 T16 Ch.0 ベクタ番号： 9 (0x09)
 ベクタアドレス： 0x8024
 本サンプルでは、次の割り込みを使用します。
 アンダーフロー割り込み

以下は、`slc17w23_ppg_mon_gnu17vx` の場合。
 UART Ch.0 のベクタ番号とベクタアドレスは以下の通りです。
 UART Ch.0 ベクタ番号： 10 (0x0a)
 ベクタアドレス： 0x8028
 本サンプルでは、次の割り込みを使用します。
 受信バッファ 1 バイトフル割り込み

図 2-1 に ADC12A の構成を示します。

2. 使用機能説明

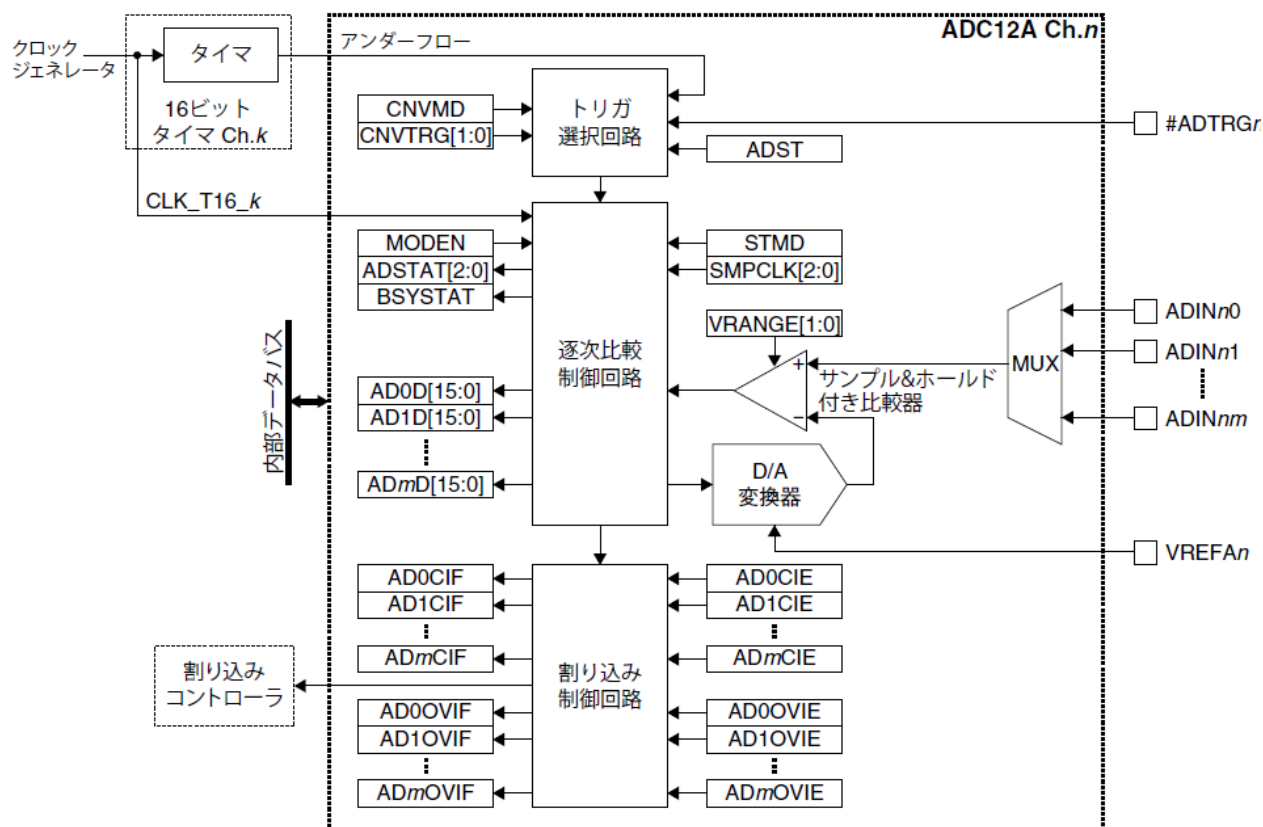


図 2-1 ADC12A の構成

3. 動作原理

3.1 検出原理

脈波とは血液が心臓の収縮により大動脈に押し出された時に発生した血管内の圧力変化が末梢方向に伝わって行く時の波動のことです。この波動による血管内の圧力変化を捕らえたものが圧脈波であり、血管の容積変化を捕らえたものが容積脈波です。

容積脈波の検出には光学的技法が用いられ、光電式容積脈波記録法(PPG: Photo Plethysmography、フォトプレチスモグラフィ)と称されます。光電式脈波の検出には透過式と反射式とがあります。

透過式は発光部と受光部の間に測定部位を挟む方式で、測定部位は指先(指尖部)、耳朶に限られます。反射式は測定部位に貼り付ける方式であり、測定は任意の部位を選ぶことができます。血中ヘモグロ빈は、ある波長帯の光に強い吸収スペクトルを持っています。この波長帯の光を照射した時の生体の透過光や反射光は、血管の容積変動に伴い変化するヘモグロ빈量に応じて変化するのです。この透過光や反射光の強度を電気信号に変えて脈波を検出することができます。¹⁾

この技術を発展させたものとしては、プローブを指先や耳朶などに付け、侵襲せずに脈拍数と経皮的動脈血酸素飽和度(SpO₂)をモニターするパルスオキシメーター(Pulse Oximeter)が良く知られており、手術中の患者の容体などを把握するのに用いられています。このパルスオキシメーターは、図 3-1 の人血液の光吸収率(酸化ヘモグロ빈、ヘモグロ빈)の波長依存性²⁾のように動脈血酸素飽和度が波長の異なる光源を用いることで算出可能なことを原理に用いています。具体的には異なる波長の複数の LED を光源に用いて皮膚を照らし、フォトランジスタやフォトダイオードなどの光学センサで反射光や透過光の強度信号を得る方式が一般的です。

本アプリケーションノートでは脈拍を計測するのが目的ですので、図 3-1 でわかるように動脈血酸素飽和度の影響を受けにくく、かつ吸光率が高い緑色の光源を実験条件に含めることにします。

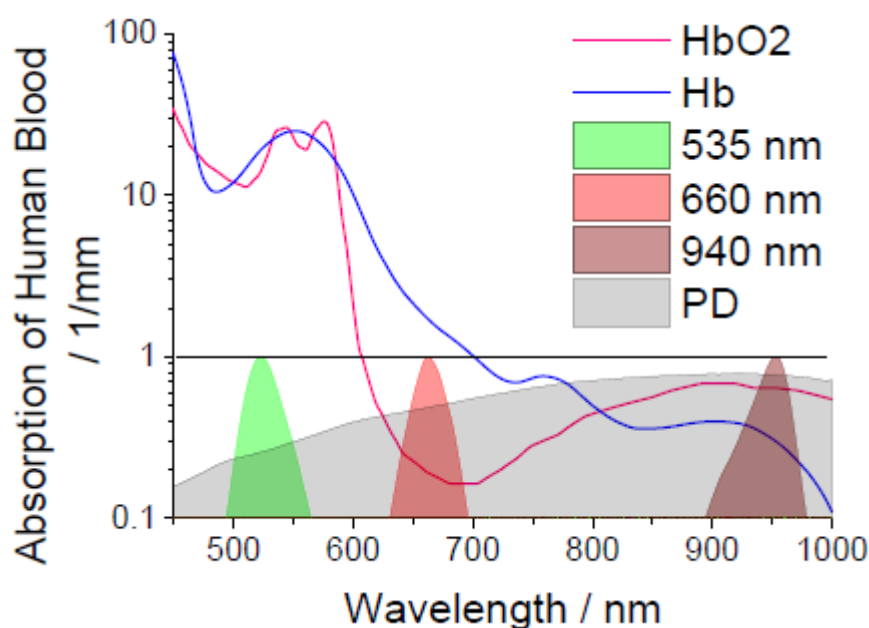


図 3-1 人血液(酸化ヘモグロ빈、ヘモグロ빈)の光吸収率の波長依存性²⁾

3. 動作原理

3.2 検出方法

本アプリケーションノートでは、光電式脈波の検出には反射式を用いました。その回路の概要を、図 3-2 の反射式的光電式脈波計測に用いる回路のブロック図に示しました。

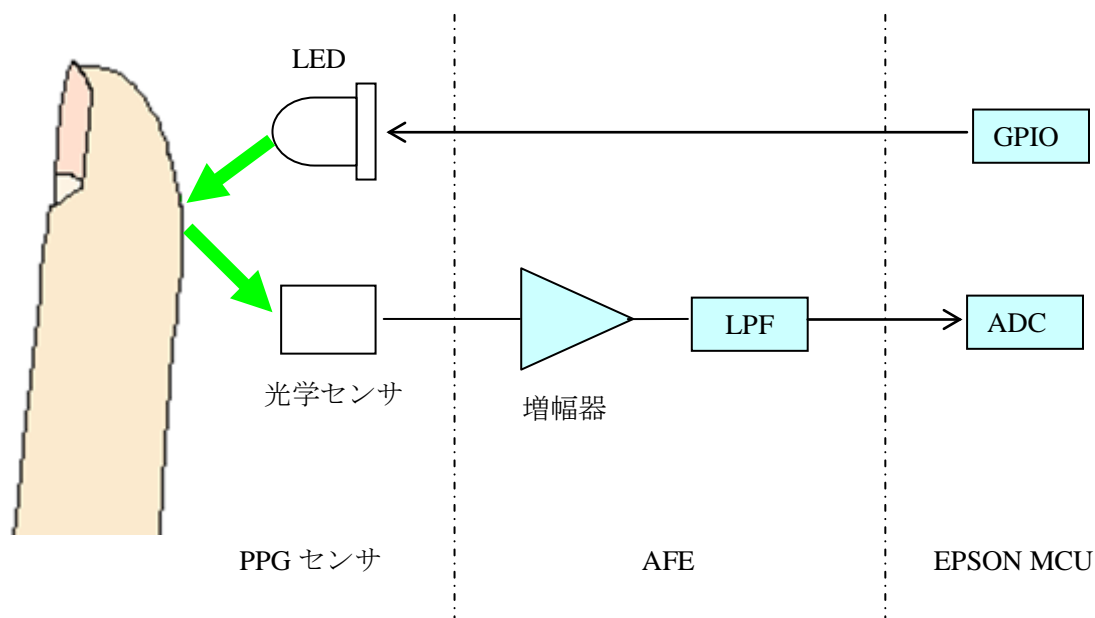


図 3-2 反射式的光電式脈波計測に用いる回路のブロック図

GPIO 出力により LED はオン／オフ制御され、光が指先に照射されます。指の毛細血管中のヘモグロビンで吸収された後に骨などで反射された光を光学センサが受光、脈波に応じた光起電流の信号が AFE の増幅器、LPF で処理され、MCU の ADC でデジタル値に変換されるというのが基本的な動作になります。

この方式は、図 3-3(a)のように脈の立ち上がり時に血流が多くなることで血液体積が増え、光を吸収する赤血球が増え、図 3-3(b)の静状態に比べ反射光や透過光の強度が減少し、脈波として検出できるという動作原理を応用しています。

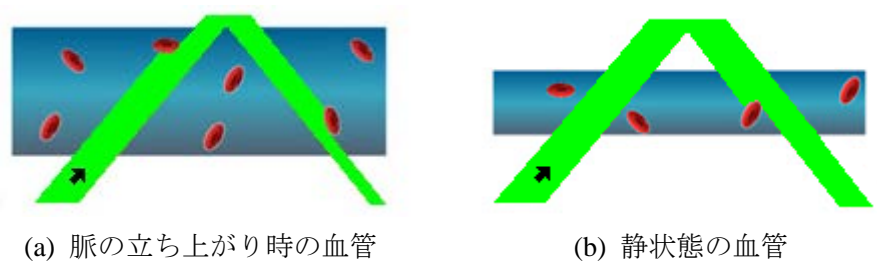


図 3-3 光強度が脈動に応じて変化する理由³⁾

3.3 検出回路

脈波の検出回路としては、大きく2つの方式が考えられます。一つは脈波計測中にLEDが点灯しっぱなしの連続駆動方式、もう一つはADCで光学センサの出力値を変換している間だけLEDを点灯する間歇駆動方式です。図3-4に連続駆動方式の脈波検出AFE回路の一例を⁴⁾、図3-5に間歇駆動方式の脈波検出AFE回路の一例を示します。

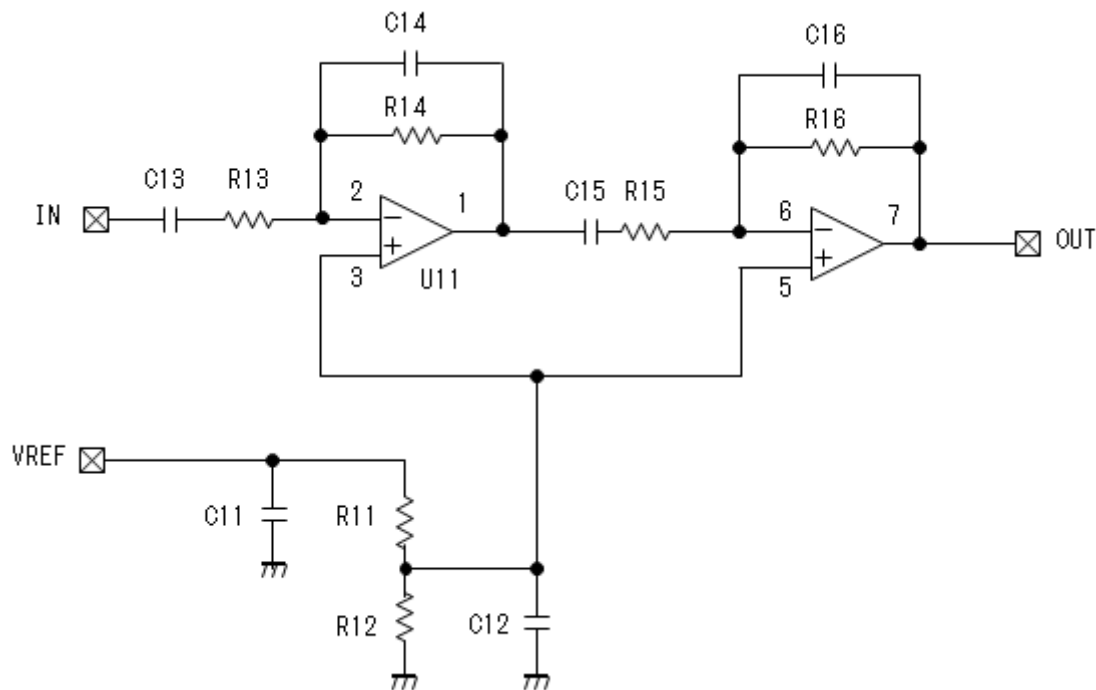
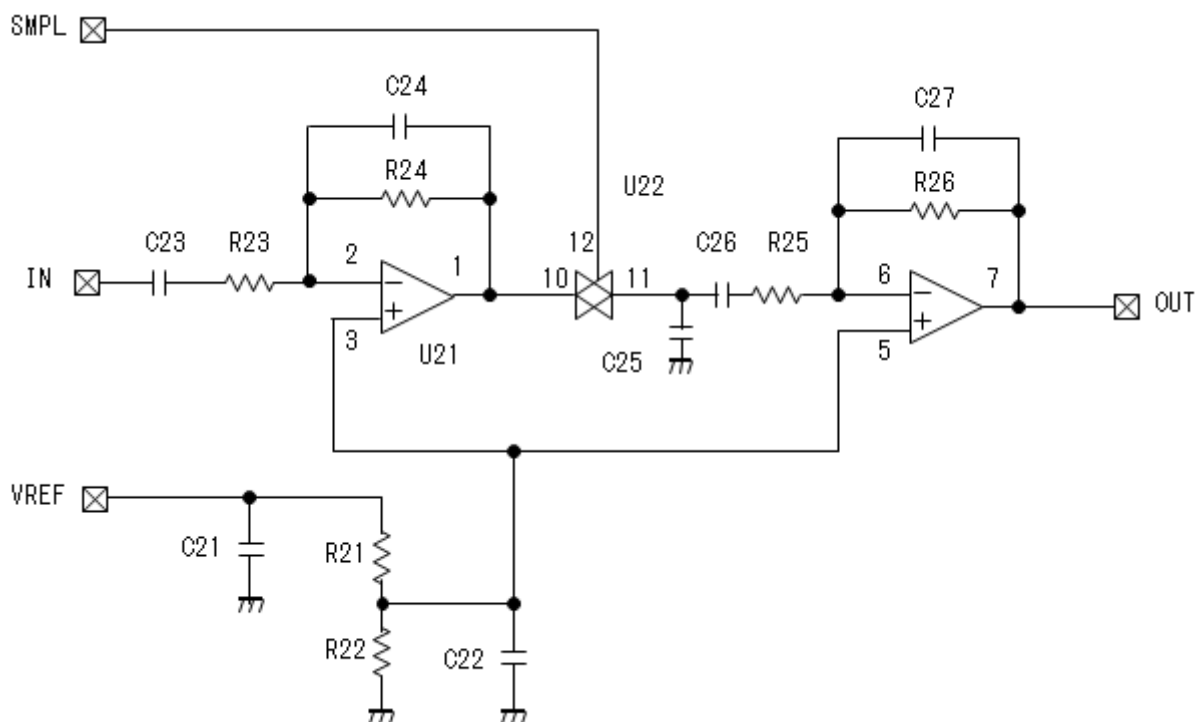
図 3-4 連続駆動方式の脈波検出 AFE 回路の一例⁴⁾

図 3-5 間歇駆動方式の脈波検出 AFE 回路の一例

3. 動作原理

本アプリケーションノートの Appendix A では、図 3-6 の回路を用いることで、両駆動方式を電氣的に切り替えながら挙動を調査しました。

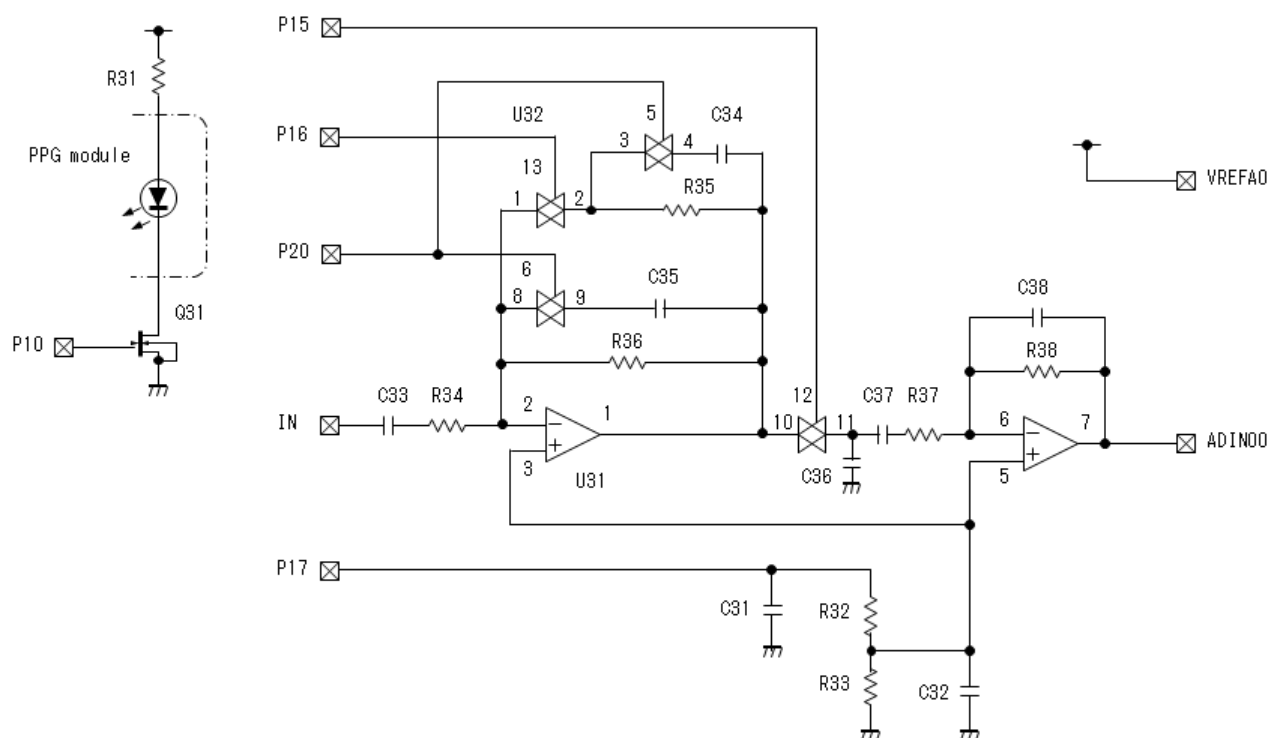


図 3-6 本アプリケーションノートで使用了回路

上記回路の P10、P15、P16、P17、P20 は、SVT17W23 の GPIO に接続されており、表 3-1 に示した機能を持っています。

表 3-1 P10、P15、P16、P17、P20 の機能

GPIO	“L”レベル時	“H”レベル時	備考
P10	LED 消灯	LED 点灯	
P15	データホールド	データサンプリング	連続駆動方式では常に“H”
P16	アンプゲイン高	アンプゲイン低	
P17	V _{REF} 供給なし	V _{REF} 供給中	
P20	プリアンプの LPF 無効	プリアンプの LPF 有効	

なお、PPG モジュールの LED に直列接続された抵抗器 R31 以外の、上記回路で用いた部品の BOM を、表 3-2 に示しました。

表 3-2 図 3-6 の本アプリケーションノートで使用した回路の BOM

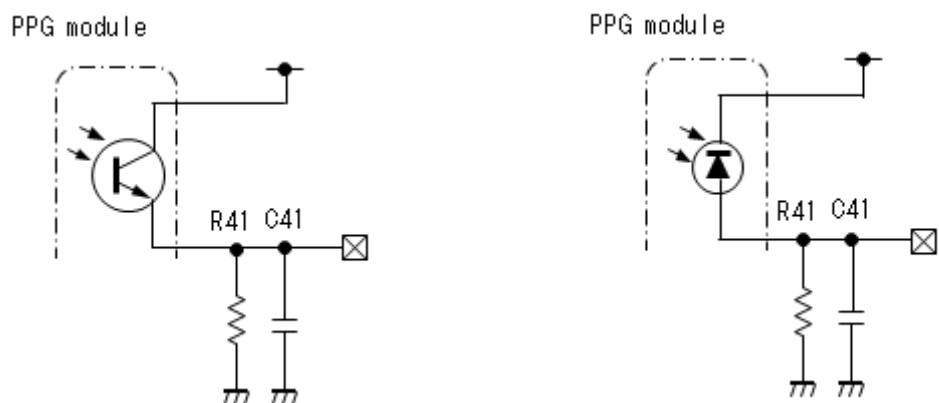
品名	記号	メーカー	メーカー品番	スペック
オペアンプ	U31	STMicroelectronics	LMV358LIDT	
アナログスイッチ	U32	Texas Instruments	SN74HC4066PWR	
FET	Q31	NXP Semiconductors	2N7002P,215	
抵抗	R32	ROHM	MCR01 Series Size=1005(0402),Tol=F	47k Ω
	R33			47k Ω
	R34			15k Ω
	R35			560k Ω
	R36			2.7M Ω
	R37			15k Ω
	R38			180k Ω
コンデンサ	C31	muRata	GRM185B10J105KE21	1 μ F,6.3V,B
	C32		GRM185B10J105KE21	1 μ F,6.3V,B
	C33		GRM21BF10J106ZE01	10 μ F,6.3V,F
	C34		GRM155B31H103KA88	0.01 μ F,50V,B
	C35		GRM155B11H222KA01	2200pF,50V,B
	C36		GRM155B31C104KA87	0.1 μ F,16V,B
	C37		GRM21BF10J106ZE01	10 μ F,6.3V,F
	C38		GRM188B11E333KA01	0.033 μ F,25V,B

また、光学センサは、フォトトランジスタの場合とフォトダイオードの場合では接続方法が異なり、図 3-7(a)のフォトトランジスタ回路の一例、図 3-7(b)のフォトダイオード回路の一例のようにします。

本アプリケーションノートでは、PPG センサとして、表 3-3 に示す 4 種類のモジュールを用いて評価しました。各々のモジュールで用いた部品の回路定数を、表 3-4 に示します。この表に示した R31 の値には、@5mA、@10mA というワードが添えられていますが、これは LED に流す電流を意味します。本アプリケーションノートの Appendix A. ではこのうち 5mA の条件の回路定数を用いて評価したのですが、10mA でも絶対最大規格条件以下なので、簡単な方法で感度を上げたいと思った場合、こちらの回路定数に変えてみるのも一つの方法です。

なお、ノイズが気にならない場合、C41 を省略することで、センサの時間応答性が向上するというメリットがあります。

3. 動作原理



(a) フォトトランジスタ回路の一例

(b) フォトダイオード回路の一例

図 3-7 光学センサの回路の一例

表 3-3 本アプリケーションノートで使用した PPG センサ

メーカー	メーカー品番	LED 発光色	光センサ
New Japan Radio	NJL5303R-TE1	緑	フォトトランジスタ
Fairchild Semiconductor	QRE1113GR	赤外	フォトトランジスタ
OSRAM Opto Semiconductors	SFH 7050	緑、赤、赤外	フォトダイオード
ROHM Semiconductor	RPR-220C1N	赤外	フォトトランジスタ

表 3-4 本アプリケーションノートで使用した PPG センサの回路定数

メーカー品番	LED 発光色	R31@5mA	(R31@10mA)	R41	C41
NJL5303R-TE1	緑	270 Ω	120 Ω	82k Ω	4.7 μ F
QRE1113GR	赤外	470 Ω	220 Ω	15k Ω	10 μ F
SFH 7050	緑	68 Ω	15 Ω	220k Ω	0.22 μ F
	赤	330 Ω	150 Ω	47k Ω	
	赤外	470 Ω	220 Ω	56k Ω	
RPR-220C1N	赤外	470 Ω	220 Ω	82k Ω	10 μ F

なお、回路図に記載はありませんが、光学センサの出力値の時間ごと変化の視覚化のプログラミングパッケージ(s1c17w23_ppg_mon_gnu17vx、PpgMon64.xls 等)を実行する際には UART の接続が必要ですが、詳細については Excel ファイル PpgMon64.xls のシート「Note」を参照してください。

3.4 FIR フィルタ

FIR フィルタはデジタルフィルタの一種で、有限パルス応答(Finite Impulse Response)フィルタのことです。この FIR フィルタの設計手法は様々な文献で紹介されていることから、詳細についてはそれらを参照してください。

図 3-8 に、FIR フィルタの構成の一例を示します。図中の文字は、回路の配線・節(ノード)です。⁵⁾

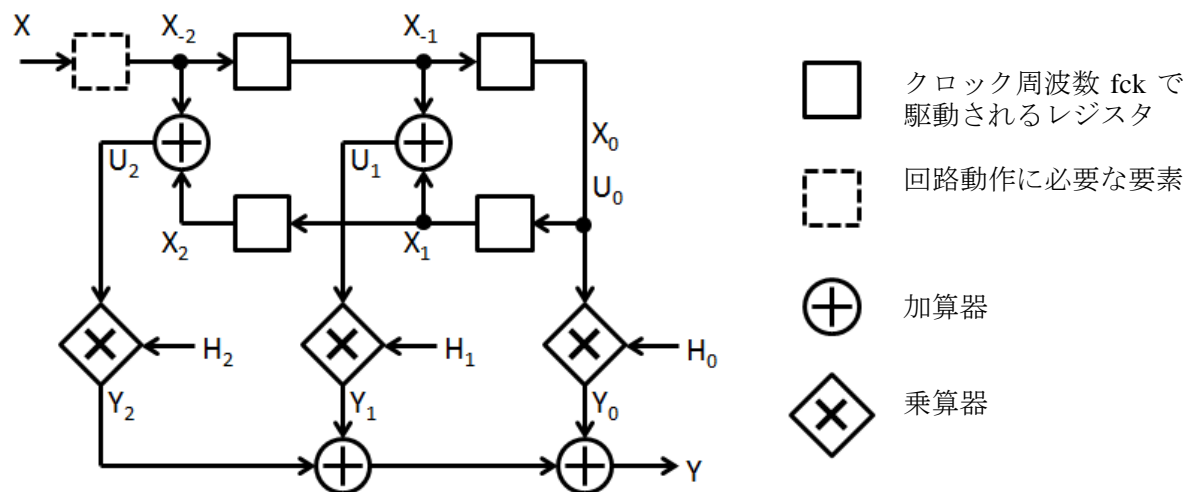


図 3-8 FIR フィルタの構成

本アプリケーションノートで用いている **FIR** フィルタは、上図のように構成されていますが、その設計情報の概略について説明します。

人間の脈拍数は多くの場合 40~250 回/分と、比較的低い周波数です。従い脈波測定時にはこれより高い周波数のノイズ成分を取り除く LPF(Low Pass Filter)が必要になります。今回の FIR フィルタは、サンプリング周期を 20 ミリ秒に固定し、カットオフ周波数を 4Hz として設計しました。

サンプルプログラムに搭載した **FIR** の窓関数の設計に用いた **Excel** ファイル(FIRdesign.xls)を添付しますので、具体的な計算についてはそちらを参照してください。^{6) 7)}

3. 動作原理

3.5 矩形波相関フィルタ

矩形波相関フィルタはずっと昔に考案された技術で、組み込み用 MCU でもリアルタイムに実行可能なアルゴリズムです。^{8) 9)} 心臓収縮時に検出光量が減少して AD 変換された値が小さくなるので、測定された脈波データのボトム位置の間隔を測定することで心拍数を計算することができますが、脈波データのノイズを抑えるのに扱いやすいフィルタなので、本アプリケーションノートのサンプルプログラムにも機能を搭載しました。

具体的には測定された脈波データに対して、図 3-9 のような矩形パルス窓を使って、相互相関値を求めます。脈波データから相互相関値を求める式は、以下の通りです。¹⁰⁾

$$\begin{aligned} y(t) = & (-1) * \{ x(t) + x(t-1) + \dots + x(t-n+1) \} \\ & + (+1) * \{ x(t-n) + x(t-n-1) + \dots + x(t-3n+1) \} \\ & + (-1) * \{ x(t-3n) + x(t-3n-1) + \dots + x(t-4n+1) \} \end{aligned}$$

ここで、 $y(t)$: 相互相関値、 $x(t)$: 脈波データ、 n : 1/4 窓幅のフレーム数 です。

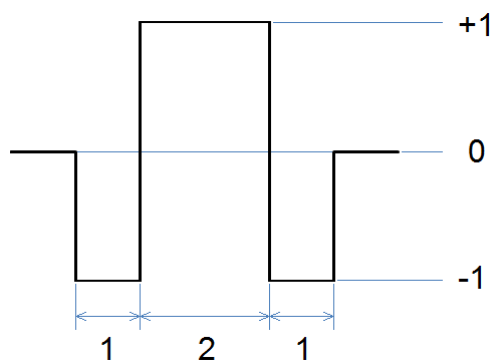


図 3-9 矩形パルス窓

4. ソフトウェア説明

4.1 s1c17w23_ppg_mon_gnu17vx について

VBA マクロ、VB スクリプト (VBS) を含む Excel と連携して、一定の周期で PPG センサの出力値をサンプリングし、UART を用いて PC に送信して Excel の表に書き込むソフトウェア、s1c17w23_ppg_mon_gnu17vx について説明します。

4.1.1 ファイル構成 (src 内)

ファイル名	機能
ad12.c	ADC12A ドライバファイル
boot.c	スタートアップモジュールファイル
init.c	初期化関数ファイル
main.c	メイン関数ファイル
osc.c	OSC ドライバファイル
t16_ch0.c	T16 Ch.0 ドライバファイル
t16_ch1.c	T16 Ch.1 ドライバファイル
t16_ch3.c	T16 Ch.3 ドライバファイル
uart.c	UART ドライバファイル

4.1.2 ファイル構成 (inc 内)

ファイル名	機能
reg	S1C17W23 周辺機器レジスタ定義ファイル格納フォルダ
c17w23_reg.h	S1C17W23 周辺機器ヘッダ定義ファイル
init.h	初期化関数ヘッダ定義ファイル
osc.h	OSC ドライバヘッダ定義ファイル
ppg_mon.h	脈波計測値視覚化ヘッダ定義ファイル
t16_ch0.h	T16 Ch.0 ドライバヘッダ定義ファイル
t16_ch1.h	T16 Ch.1 ドライバヘッダ定義ファイル
t16_ch3.h	T16 Ch.3 ドライバヘッダ定義ファイル
uart.h	UART ドライバヘッダ定義ファイル

4.1.3 モジュール説明

ファイル中のモジュールのうち、一般的でない設定を中心に、関数名とその機能、変数名とその内容について説明します。

ファイル名： main.c

関数名	機能
execFir	Excel ファイル PpgMon64.xlsm で FIR フィルタ処理が選択されている場合、PpgMon64.xlsm のシート FIR に記載された条件に従い FIR フィルタリング処理計算し、pm.firRes に格納します。
calcCc	PpgMon64.xlsm で CC フィルタ処理が選択されている場合、PpgMon64.xlsm の Num. Frame の設定値に従い矩形波相関フィルタ処理計算し、得られた相互相関値を pm.ccRes に格納します。
intUartCh0	UART Ch.0 割り込み関数。PpgMon64.xlsm の VBA、VBS からのコントロール情報の受信に伴い、脈波の測定を開始するため、T16 Ch.0 の動作をスタートします。
intT16Ch0	T16 Ch.0 割り込み関数。PpgMon64.xlsm の Scan Interval の設定時間毎に呼び出される。すでに受け取っているコントロール情報に従い、脈波の ADC 値を測定し、PC 上の VBS に送信します。PpgMon64.xlsm の Data Set#の個数のデータの送信が終了したところで、T16 Ch.0 の動作をストップします。
intAdc12Ch0	ADC12A Ch.0 割り込み関数。AD 変換が終了した時に測定結果を pm.measuredVal に格納し、pm.stage=1(測定終了)に設定します。

構造体名： ppg_mon(ファイル ppg_mon.h 中で定義)

変数名	内容
rawData[]	脈波計測結果の生データ。ADC12A でサンプリングした値なので、0～4095 の範囲の値。
firRes[]	FIR フィルタ処理後の値。フィルタリングされることで変換元の値よりも振幅が小さな値になります。
ccRes[]	矩形パルス窓で計算した相互相関値。
firCoef[]	PC 側で値を 65536 倍してから送られた FIR フィルタの係数。
rcvData[]	PC から送られた制御情報。
sndData[]	PC に送るための測定値。生データ、FIR フィルタ処理後データ、相互相関値の順に各 2Byte 送られます。相互相関値はマイナスの場合もあるので、+32768 した結果を送ります。送るデータがない場合は 29999 を送ります。
interm	PC から送られた間歇駆動か連続駆動かの制御情報。 0: 連続駆動、1: 間歇駆動

highGain	PC から送られたゲインの制御情報。 0: アンプゲイン低、1: アンプゲイン高
lpf	PC から送られた LPF の制御情報。0: LPF オフ、1: LPF オン
filter1, filter2	PC から送られたフィルタの制御情報 2 つ。 0: なし、1: FIR フィルタ、2: 矩形波相関フィルタ
skipNum	PC から送られた測定開始後にデータ測定する迄に読み飛ばすデータの数。
dataSetNum	PC から送られた測定するデータの数。
scanInterval	PC から送られたサンプリング周期。
samplingDelay	PC から送られた間歇駆動時にサンプリングするまでの安定化待ち時間の追加時間。
firNum	PC から送られた FIR フィルタの係数の個数。タップ値=N の場合、(N-1)/2。
numFrame	PC から送られた矩形パルス窓の 1/4 窓幅のフレーム数。
dataPtr	測定データのポインタ。
measuredVal	測定値。
measCount	測定数のカウンタ。
skipCount	測定開始後にデータ測定する迄に読み飛ばす際のカウンタ。
availDataNum	矩形波相関フィルタの計算を行うのに十分なデータがあるか判断するためのカウンタ。
setNum	4 * numFrame
step	状態フラグ。0: PC からの制御データ送信待ち、1: データ読み飛ばし中、 2: 測定中、3: 測定終了→0 に移行
stage	ADC 測定状態フラグ。0: 測定中、1: 測定終了
plxBase	GPIO P10-P17 に出力するデータの元となるデータ。これに対して必要なビットのみをオンオフ操作して出力する。

4.1.4 操作手順

サンプルソフトウェアは GNU17 Version 2(以下 GNU17v2 と呼びます)用と GNU17 Version 3(以下 GNU17v3 と呼びます)用の 2 つのプロジェクトを含んでいます。

GNU17v2 か GNU17v3 のためのサンプルソフトウェアを使用するにあたり、以下に示した手順に従い、対象機種を構成してください。

- (1) サンプルソフトウェアの `s1c17w23_ppg_mon_gnu17vx` フォルダの `inc` サブフォルダーに `c17w23_reg.h` ファイルと `reg` フォルダーをコピーしてください。
- (2) GNU17 IDE を起動後、インポートウィザードを起動するためにメニューの[ファイル]→[インポート...]を選択し、以下のどちらかを選択します。
[一般]→[既存プロジェクトをワークスペースへ] (GNU17v2) または、
[General]→[Existing Projects into Workspace] (GNU17v3)
- (3) サンプルプログラムを含むプロジェクトを選択します。
`s1c17w23_ppg_mon_gnu17v2` フォルダ (GNU17v2) または、
`s1c17w23_ppg_mon_gnu17v3` フォルダ (GNU17v3)
- (4) 以下の手順でインポートウィザードから抜けてください。
[プロジェクトをワークスペースにコピー]を選択し、[完了]をクリック (GNU17v2) または、
[Copy projects into workspace] を選択し、[Finish]をクリック (GNU17v3)
- (5) ターゲットとなる CPU を変更します。
(GNU17v2)
 1. [C/C++プロジェクト]一覧のインポートしたプロジェクトを選択し、メニューの[プロジェクト]→[プロパティ]を選択する。
 2. 現れた[プロパティ]ダイアログボックスのプロパティリストから[GNU17 一般設定]を選択します。
 3. [ターゲット CPU 機種]ドロップダウンリストから目標の CPU を選びます。
 4. [適用]ボタンを押します。
(GNU17v3)
 1. [Project Explorer]一覧のインポートしたプロジェクトを選択し、メニューの[Project]→[Properties]を選択する。
 2. 現れた[Properties] ダイアログボックスのプロパティリストから[GNU17 Setting] を選択します。
 3. [Target CPU]ドロップダウンリストから目標の CPU を選びます。
 4. ダイアログボックスを閉じるために[OK]ボタンを押したら、ステップ(7)に移ります。
- (6) デバッガーのスタートアップオプションを設定します。(GNU17v2 のみ)
 1. プロパティリストから[GNU17 GDB コマンド]を選択します。
 2. [初期起動コマンドの生成]ダイアログボックスを表示させるため、[雛形からコマンドを生成する]ボタンをクリックします。
 3. [デバッガ:]ドロップダウンリストから[ICD Mini]を選び、[Flash ROM 書き込み実行]を選択します。
 4. ダイアログボックスを閉じるために[上書き]ボタンをクリックします。また、[プロパティ]ダイアログボックスを閉じます。
- (7) プロジェクトのビルド
IDE を使用して、`s1c17w23_ppg_mon_gnu17vx` プロジェクトをビルドしてください。

(8) Excel VBA を使用するための準備

1. デスクトップに PpgMon64 フォルダをコピーしてください。このフォルダには、Excel ファイル「PpgMon64.xlsm」、VB スクリプトファイル「DoPpg64.vbs」、Active-X コントロールファイル「NonComSck.ocx」が入っています。同梱されていない、マイクロソフトが提供する Active-X コントロールファイル「MSCOMM32.OCX」は、種々のサイトからダウンロードすることが可能ですが、それ入手し、先の PpgMon64 フォルダ内にコピーしてください。
2. [すべてのプログラム]→[アクセサリ]→[コマンド プロンプト]上で右ボタンを押し、[管理者として実行...]を選択すると、「管理者: コマンド プロンプト」が起動します。
3. 「cd C:¥Users¥user¥Desktop¥PpgMon64」、「regsvr32.exe MSCOMM32.OCX」、「regsvr32.exe NonComSck.ocx」を実行して Active-X コントロールファイルを有効にします。
4. 上記操作でうまくいかない場合、Active-X コントロールファイルを「C:¥Windows¥System32」にコピーし、「cd C:¥Windows¥System32」、「regsvr32.exe MSCOMM32.OCX」、「regsvr32.exe NonComSck.ocx」を実行してください。

(9) 接続、電源投入

1. 光学センサと AFE、SVT17W23、ICDmini、USB ケーブル、PC を接続してください。
2. SVT17W23 の UART Ch.0 と、Excel VBA を実行する PC のシリアル端子（USB 接続のシリアル通信 I/F を使うのが一般的です。市販品の例を Excel ファイル「PpgMon64.xlsm」のシート「Note」に示しました。）を接続してください。PC に搭載された CPU の処理能力が十分なら、この PC は先の PC と兼用することもできます。
3. SVT17W23、ICDmini をリセットしてください。

(10) Excel の VBA の実行

Excel ファイル「PpgMon64.xlsm」をダブルクリックして実行してください。

(11) 実行

1. IDE を使用して、「s1c17w23_ppg_mon_gnu17vx」プロジェクトを実行してください。
2. 「s1c17w23_ppg_mon_gnu17vx」の実行後、Excel VBA からのコマンドで、PPG センサの出力値の収集を開始します。

4.1.5 PpgMon64.xlsm の使い方

PpgMon64.xlsm の外観を図 4-1 に示します。

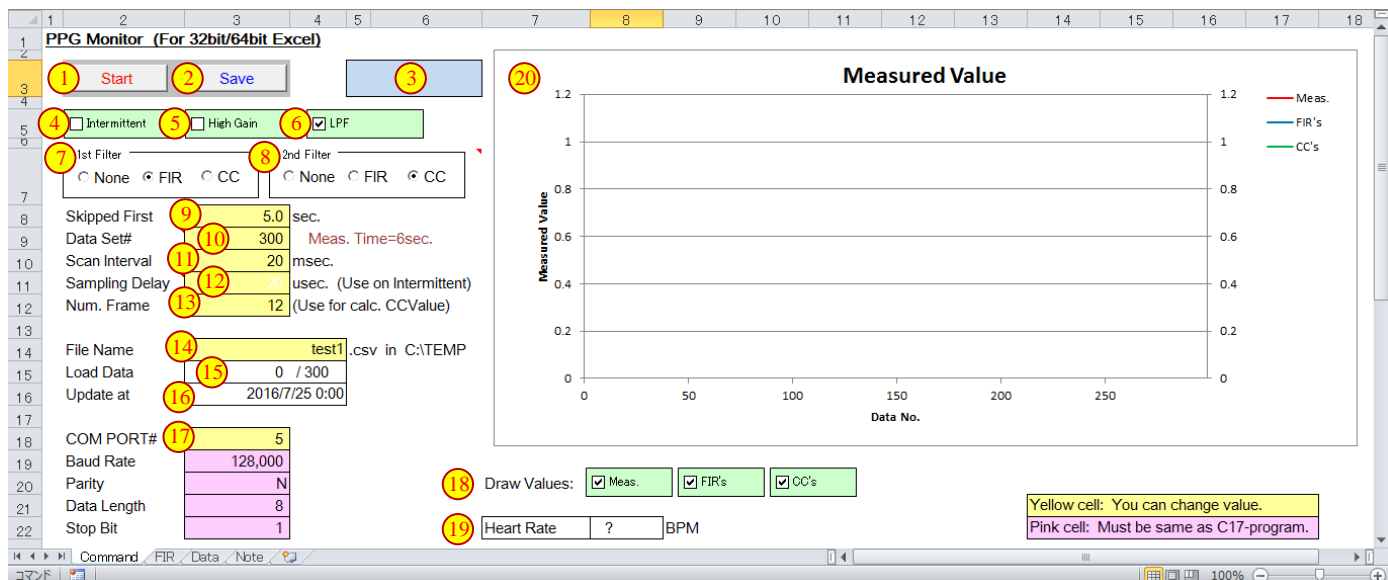


図 4-1 PpgMon64.xlsm の外観

- | | |
|---------------|---|
| ① 測定開始ボタン | 脈波の計測値の取得を開始します。 |
| ② データセーブボタン | 計測データを、CSV 形式でセーブします。 |
| ③ 状態インジケータ | 測定中、セーブ中にその旨 表示します。 |
| ④ 駆動方式選択ボタン | <input type="checkbox"/> : 連続駆動、 <input checked="" type="checkbox"/> : 間歇駆動 |
| ⑤ アンプゲイン選択ボタン | <input type="checkbox"/> : アンプゲイン低、 <input checked="" type="checkbox"/> : アンプゲイン高 |
| ⑥ LPF 選択ボタン | <input type="checkbox"/> : LPF オフ、 <input checked="" type="checkbox"/> : LPF オン |
| ⑦ 第 1 フィルタ | None: なし、FIR: FIR フィルタ、CC: 矩形波相関フィルタ 注 1 |
| ⑧ 第 2 フィルタ | None: なし、FIR: FIR フィルタ、CC: 矩形波相関フィルタ |
| ⑨ 読み飛ばし時間 | 測定の最初に測定値を読み飛ばす時間。 |
| ⑩ データセット数 | 取り込むデータセット数を指定します。 |
| ⑪ スキャンインターバル | データの取り込み間隔を指定します。 |
| ⑫ サンプリング遅延時間 | 間歇駆動時、LED 発光後のサンプリングの遅延時間(追加分)。 |
| ⑬ フレーム数 | 矩形パルス窓の 1/4 窓幅のフレーム数。 |
| ⑭ セーブファイル名 | 計測データを保管するファイルの名称を指定します。 |
| ⑮ データ取り込み数 | 計測したデータセットの数を逐次更新して表示します。 |
| ⑯ データ採取日時 | データ採取の最終日時を表示します。 |
| ⑰ シリアル通信条件設定 | 上から、COM ポート番号、ボーレート、パリティ、データ長、ストップビット数。 |
| ⑱ グラフ表示項目選択 | 左から、測定生データ、FIR フィルタ処理後、矩形波相関フィルタ処理後の相互相関値。 |

- ⑱ 脈拍数 測定された脈波から計算した脈拍数。
- ⑳ 測定データグラフ 各データをプロットします。相互相関値のみ縦軸は右側。

- 注 1) 第 1 フィルタ、第 2 フィルタの組み合わせで、実際にどのフィルタが適用されるかの詳細は、**PpgMon64.xlsm** のシート「**Command**」の第 7 行 6 列のコメントを参照してください。
- 注 2) 本サンプルプログラムで用いている UART によるシリアル通信は 2 線式通信で、**SVT17W23** はデータを一方的に送る仕様で作られているため、PC 側の受信バッファが追従できなくなると通信データが欠落します。通信が追従できない場合、⑩、⑪の設定値を変更することで回避するようにしてください。
- 注 3) シリアル通信に関する設定⑰の **COM** ポート番号以外の項目は、通常は変更する必要のない情報です。もしもこれらの情報を変更した場合、**s1c17w23_ppg_mon_gnu17vx** の当該箇所の変更が必要になりますのでご注意ください。
- 注 4) グラフ⑳の描画は全データ取得後に行われます。すなわち、データ取得時には逐次更新されず、データクリアされている状態をキープします。

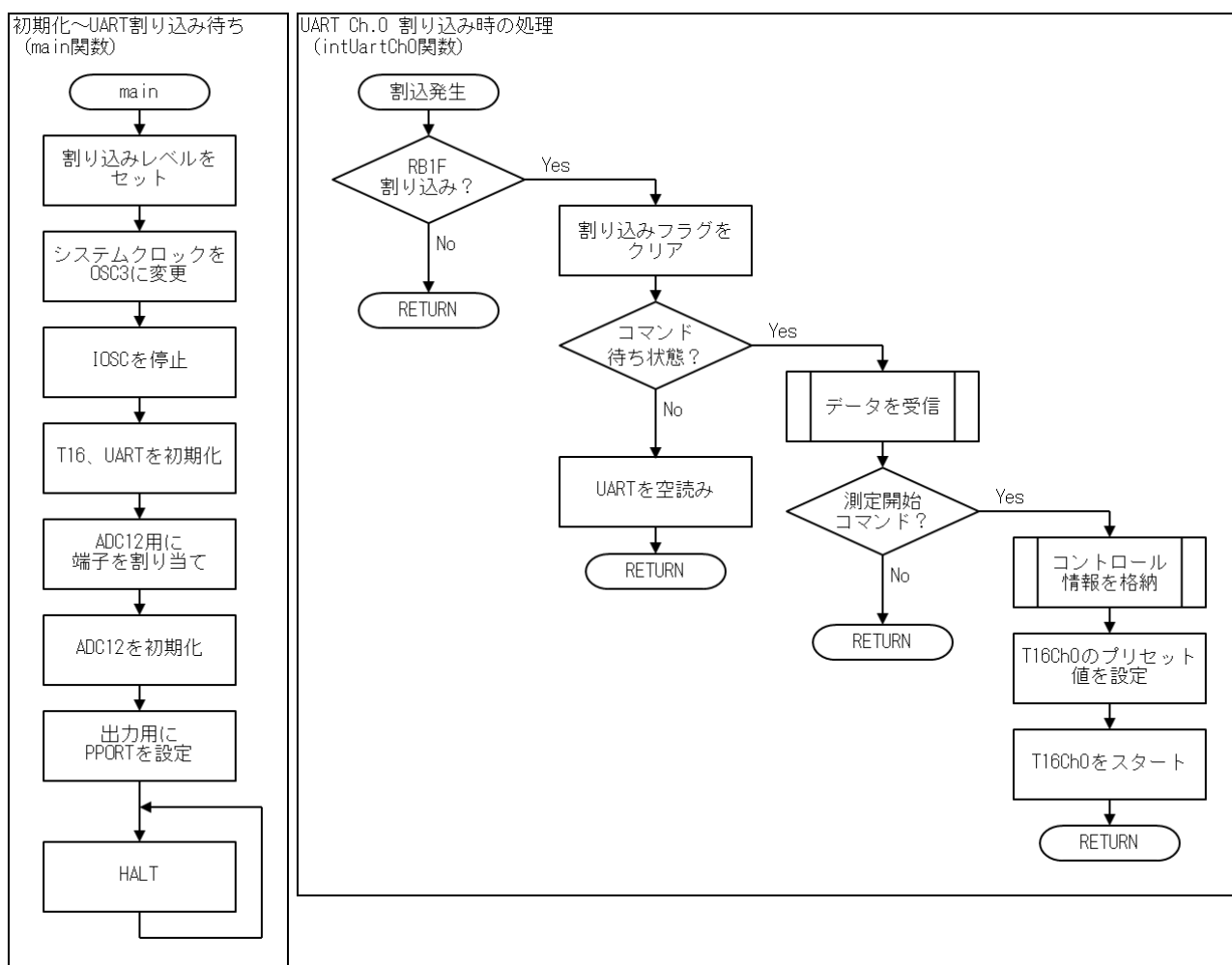
4.1.6 サンプルプログラム動作概要

サンプルプログラムは、以下の処理を実行します。

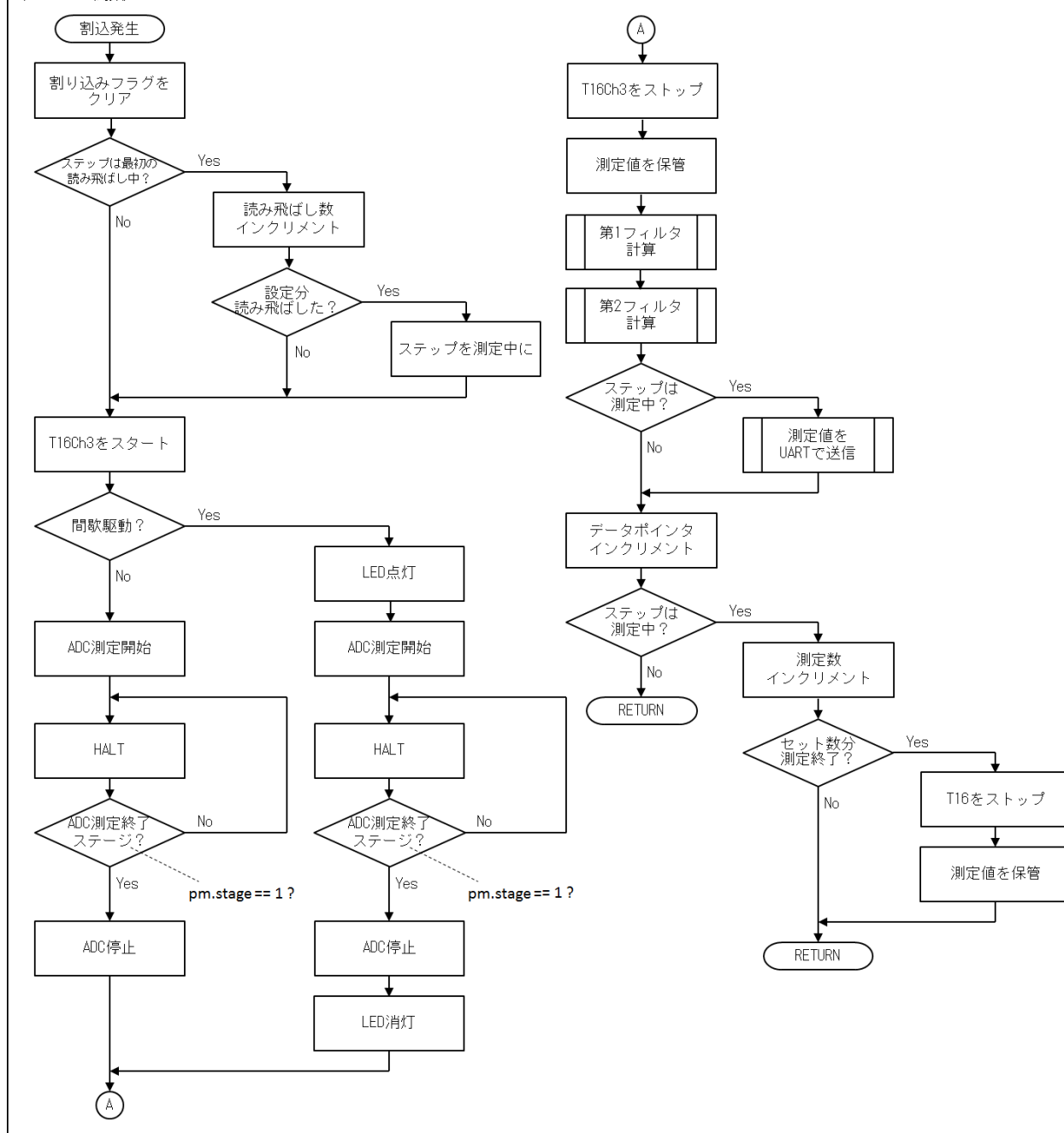
- (1) 使用するために以下の周辺機器/機能を初期化します。
 - 割り込みレベルの設定
 - システムクロックを IOSC から OSC3（内蔵発振 4MHz）に切り替え
 - T16、UART を初期化
 - ADC12 用に端子を割り付けて ADC12 を初期化
 - 出力用に PPORT を設定
- (2) UART の割り込みを待つために HALT します。
- (3) UART の割り込みが発生すると、割り込みハンドラ intUartCh0 では以下の処理を実行します。
 - PC から送られてきたコントロール情報を受信
 - 脈波の測定を開始するため、T16 Ch.0 の動作をスタート
- (4) T16 Ch.0 の割り込みが発生するたび、割り込みハンドラ intT16Ch0 で、受け取っているコントロール情報を基に以下の処理を実行します。
 - 脈波の AD 変換値を読み取り
 - フィルタ計算を実行
 - PC に測定データを送信

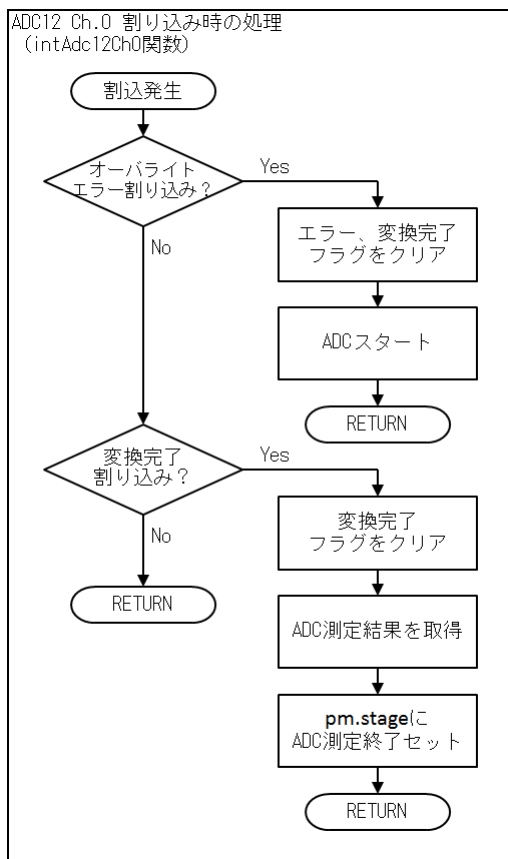
指定個数のデータの送信が終了したところで、T16 Ch.0 の動作をストップします。

フローチャートを以下に記します。



T16 Ch.0 割り込み時の処理
(intT16Ch0関数)





4.2 s1c17w23_ppg_demo_gnu17vx について

PPG のデモ用に、拍動、脈拍数、脈波を液晶表示するソフトウェア、s1c17w23_ppg_demo_gnu17vx について説明します。

4.2.1 ファイル構成（src 内）

ファイル名	機能
ad12.c	ADC12A ドライバファイル
boot.c	スタートアップモジュールファイル
display.c	LCD 表示関数ファイル
init.c	初期化関数ファイル
main.c	メイン関数ファイル
osc.c	OSC ドライバファイル
t16_ch0.c	T16 Ch.0 ドライバファイル
t16_ch1.c	T16 Ch.1 ドライバファイル
t16_ch3.c	T16 Ch.3 ドライバファイル

4.2.2 ファイル構成（inc 内）

ファイル名	機能
reg	S1C17W23 周辺機器レジスタ定義ファイル格納フォルダ
ad12.h	ADC12A ドライバヘッダ定義ファイル
c17w23_reg.h	S1C17W23 周辺機器ヘッダ定義ファイル
display.h	LCD 表示関数ヘッダ定義ファイル
init.h	初期化関数ヘッダ定義ファイル
lcd_font.h	LCD 表示フォントヘッダ定義ファイル
osc.h	OSC ドライバヘッダ定義ファイル
ppg_data.h	脈波計測データヘッダ定義ファイル
t16_ch0.h	T16 Ch.0 ドライバヘッダ定義ファイル
t16_ch1.h	T16 Ch.1 ドライバヘッダ定義ファイル
t16_ch3.h	T16 Ch.3 ドライバヘッダ定義ファイル

4.2.3 モジュール説明

ファイル中のモジュールのうち、一般的でない設定を中心に、関数名とその機能、変数名とその内容について説明します。

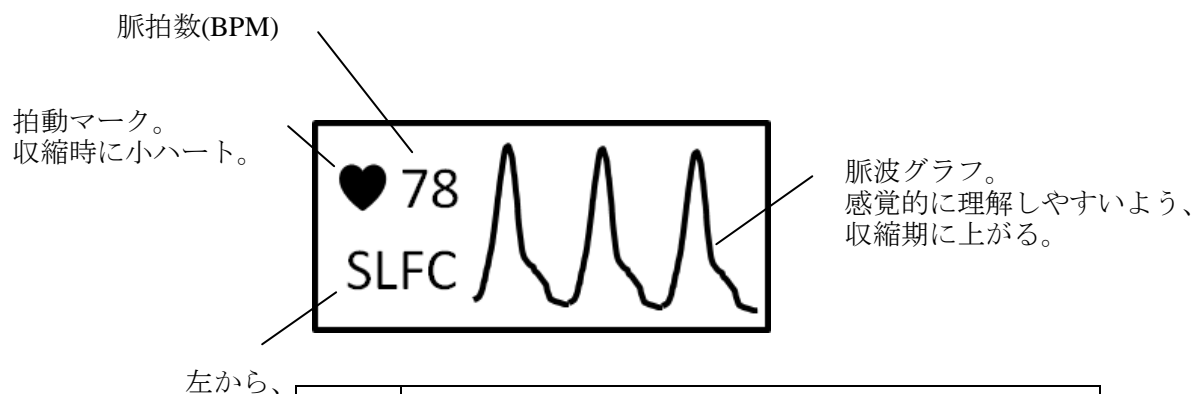
ファイル名： main.c

関数名	機能
calcHeartRate	相互相関値の最小ピーク(拍動時に暗くなる)の間隔から脈拍数を計算します。1/4 窓幅のフレーム数も一定の範囲で追従して変化します。
getMinMax	拍動を見分ける際に使うため、生データの最大値、最小値を求めます。
checkHeartBeatTiming	最新の測定生データが収縮期の判断値以下の時に 1 を、そうでないときに 0 を返します。
execFir	生データを FIR フィルタリング処理計算し、pd.firRes に格納します。
calcCc	pd.firRes を矩形波相関フィルタ処理計算し、得られた相互相関値をpd.ccRes に格納します。
val2Str4	正の整数値 val を digit 桁(最大 4 桁)の数値文字列に変換し、dst に入れて返します。zeroSup が 1 の時はゼロサプレスし、0 の時はゼロサプレスしません。
dispHeartMark	mark が 0 の時、収縮期を示す小さなハートマークを、そうでないときに大きなハートマークを LCD の決められた位置に表示します。
dispHeartRate	心拍数が計算されている時、その値を LCD の決められた位置に表示し、計算されていない時は、「?」マークを値の代わりに表示します。
dispStat	現在選択されている測定条件を LCD に表示します。詳細については、図 4-2 を参照してください。
setP1Base	選択されている条件をもとに、P1 ポートに出力するデータのベースとなる値を pd.p1xBase に設定します。
clearGraph	LCD のグラフエリアの表示をクリアします。
intPport	SW1～4 が押されたときに呼ばれる割り込みハンドラ。
intT16Ch0	20 ミリ秒毎に呼び出され、コントロール情報に従い、脈波の ADC 値を測定して pd.rawData に格納します。その後、FIR フィルタリング処理計算、矩形波相関フィルタ処理計算します。グラフの更新もこの関数で行います。一定間隔毎に脈拍数を更新したり、ハートマークを更新したりする処理も行います。
intAdc12Ch0	ADC12A Ch.0 割り込み関数。AD 変換が終了した時に測定結果を pm.measuredVal に格納し、pm.stage=1(測定終了)に設定します。

構造体名： ppg_data(ファイル ppg_data.h 中で定義)

変数名	内容
rawData[]	脈波計測結果の生データ。ADC12A でサンプリングした値なので、0～4095 の範囲の値。
firRes[]	FIR フィルタ処理後の値。フィルタリングされることで変換元の値よりも振れ幅が小さな値になります。
ccRes[]	矩形パルス窓で計算した相互相関値。
dataPtr	測定データのポインタ。
availDataNum	矩形波相関フィルタの計算を行うのに十分なデータがあるか判断するためのカウンタ。
measuredVal	測定値。
calcHRIter	心拍数を計算するタイミングのためのイタレーションカウント用。
numFrame	矩形パルス窓の 1/4 窓幅のフレーム数。
heartRate	心拍数。単位は BPM(Beats Per Minute)。
rawMinVal	生データの最小値。
rawMaxVal	生データの最大値。
rawRange	生データの範囲＝ (最大値－最小値)。
rawSystoleLevel	収縮期判断値。この値より生データが小さいと収縮期と判断。
currHeartMark	現在のハートマーク。LCD の更新頻度を最小化するために使用。
systoleCycleCounter	心臓収縮の間隔カウンター。
hRFromSystole	心臓収縮のサイクルから計算した心拍数。これを用いて numFrame を自動調整する。
running	計測状況。0: 停止、1: 計測中
interm	間歇駆動か連続駆動かの制御情報。0: 連続駆動、1: 間歇駆動
highGain	ゲインの制御情報。0: アンプゲイン低、1: アンプゲイン高
lpf	LPF の制御情報。0: LPF オフ、1: LPF オン
samplingDelay	間歇駆動時にサンプリングするまでの安定化待ち時間の追加時間。
stage	ADC12A 測定状態フラグ。0: 測定中、1: 測定終了
p1xBase	GPIO P10-P17 に出力するデータの元となるデータ。これに対して必要なビットのみをオンオフ操作して出力する。

overRunTimes	T16Ch0 のオーバーランカウンタ。SW4 押下による測定開始回数までは許容値。それを超えたら、intT16Ch0 関数の処理が間に合っていないので、その中の処理を main 関数に移動させ、イベント処理する必要がある。
--------------	---



SW	表示とその意味			
SW4	S	計測停止	R	計測中
SW3	L	アンプゲイン低	H	アンプゲイン高
SW2	F	LPF オン	-	LPF オフ
SW1	C	連続駆動	I	間歇駆動

図 4-2 LCD 表示情報の説明

4.2.4 操作手順

サンプルソフトウェアは GNU17 Version 2(以下 GNU17v2 と呼びます)用と GNU17 Version 3(以下 GNU17v3 と呼びます)用の 2 つのプロジェクトを含んでいます。

GNU17v2 か GNU17v3 のためのサンプルソフトウェアを使用するにあたり、以下に示した手順に従い、対象機種を構成してください。

- (1) サンプルソフトウェアの `slc17w23_ppg_mon_gnu17vx` フォルダの `inc` サブフォルダーに `c17w23_reg.h` ファイルと `reg` フォルダーをコピーしてください。
- (2) GNU17 IDE を起動後、インポートウィザードを起動するためにメニューの[ファイル]→[インポート...]を選択し、以下のどちらかを選択します。
[一般]→[既存プロジェクトをワークスペースへ] (GNU17v2) または、
[General]→[Existing Projects into Workspace] (GNU17v3)
- (3) サンプルプログラムを含むプロジェクトを選択します。
`slc17w23_ppg_mon_gnu17v2` フォルダ (GNU17v2) または、
`slc17w23_ppg_mon_gnu17v3` フォルダ (GNU17v3)
- (4) 以下の手順でインポートウィザードから抜けてください。
[プロジェクトをワークスペースにコピー]を選択し、[完了]をクリック (GNU17v2) または、
[Copy projects into workspace] を選択し、[Finish]をクリック (GNU17v3)
- (5) ターゲットとなる CPU を変更します。
(GNU17v2)
 1. [C/C++プロジェクト]一覧のインポートしたプロジェクトを選択し、メニューの[プロジェクト]→[プロパティ]を選択する。
 2. 現れた[プロパティ]ダイアログボックスのプロパティリストから[GNU17 一般設定]を選択します。
 3. [ターゲット CPU 機種]ドロップダウンリストから目標の CPU を選びます。
 4. [適用]ボタンを押します。
(GNU17v3)
 1. [Project Explorer]一覧のインポートしたプロジェクトを選択し、メニューの[Project]→[Properties]を選択する。
 2. 現れた[Properties] ダイアログボックスのプロパティリストから[GNU17 Setting] を選択します。
 3. [Target CPU]ドロップダウンリストから目標の CPU を選びます。
 4. ダイアログボックスを閉じるために[OK]ボタンを押したら、ステップ(7)に移ります。
- (6) デバッガーのスタートアップオプションを設定します。(GNU17v2 のみ)
 1. プロパティリストから[GNU17 GDB コマンド]を選択します。
 2. [初期起動コマンドの生成]ダイアログボックスを表示させるため、[雛形からコマンドを生成する]ボタンをクリックします。
 3. [デバッガ:]ドロップダウンリストから[ICD Mini]を選び、[Flash ROM 書き込み実行]を選択します。
 4. ダイアログボックスを閉じるために[上書き]ボタンをクリックします。また、[プロパティ]ダイアログボックスを閉じます。
- (7) プロジェクトのビルド
IDE を使用して、`slc17w23_ppg_mon_gnu17vx` プロジェクトをビルドしてください。

(8) 接続、電源投入

1. 光学センサと AFE、SVT17W23、ICDmini、USB ケーブル、PC を接続してください。
2. SVT17W23 の UART Ch.0 と、Excel VBA を実行する PC のシリアル端子（USB 接続のシリアル通信 I/F を使うのが一般的です。市販品の例を Excel ファイル「PpgMon64.xlsm」のシート「Note」に示しました。）を接続してください。PC に搭載された CPU の処理能力が十分なら、この PC は先の PC と兼用することもできます。
3. SVT17W23、ICDmini をリセットしてください。

(9) 実行

1. IDE を使用して、「s1c17w23_ppg_mon_gnu17vx」プロジェクトを実行してください。
2. 「s1c17w23_ppg_mon_gnu17vx」の実行後、Excel VBA からのコマンドで、PPG センサの出力値の収集を開始します。

4.2.5 サンプルプログラム動作概要

サンプルプログラムは、以下の処理を実行します。

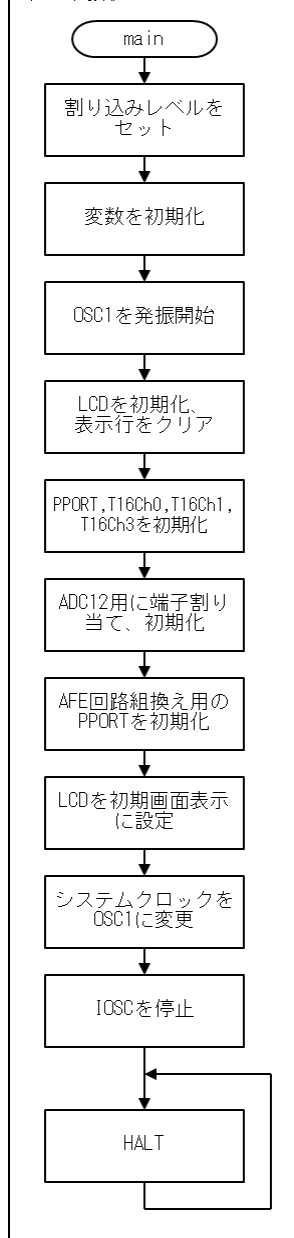
- (1) 使用するために以下の周辺機器/機能を初期化します。
 - 割り込みレベルの設定
 - 変数の初期化
 - OSC1 (32.768kHz) の発振を開始
 - LCD を初期化し、表示をクリア
 - PPORT、T16 Ch0、T16 Ch1、T16 Ch3 を初期化
 - ADC12 用に端子を割り当て、初期化
 - AFE 回路組換え用の PPORT を初期化
- (2) LCD を初期画面表示に更新します。
- (3) システムクロックを IOSC から OSC1 (32.768kHz) に切り替えます。
- (4) PPORT の割り込みを待つために HALT します。
- (5) SW1～SW4 の押下による PPORT の割り込みが発生すると、以下の処理を実行します。
 - 測定停止状態のときは、SW4 を押すと測定を開始するために T16 Ch.0 を始動します。SW1～SW3 を押すと AFE の回路を組み替えることができ、それに応じて LCD の表示が更新されます。
 - 測定中に SW4 を押すと計測を終了するために T16 Ch.0 を停止します。SW1～SW3 の操作は無効で、20 ミリ秒毎に T16Ch0 の割り込みが発生して intT16Ch0 関数が呼ばれます。
- (6) intT16Ch0 関数では、間歇駆動の場合は LED を点灯後、指定の遅延時間経過後に ADC12 をトリガして AD 変換値を得、LED を消灯します。連続駆動の場合は ADC12 をトリガして AD 変換値を得るだけです。その後、FIR フィルタ計算、矩形波相関フィルタ計算し、定期的に脈拍計算し、心臓収縮判断されたら心臓マークの表示を更新し、グラフを描画します。

以下の関数については、s1c17w23_ppg_mon_gnu17vx/src/main.c と全く同じ処理フローで、構造体の変数名のみ、pm から pd に変更になります。従い、フローチャートは掲載していません。

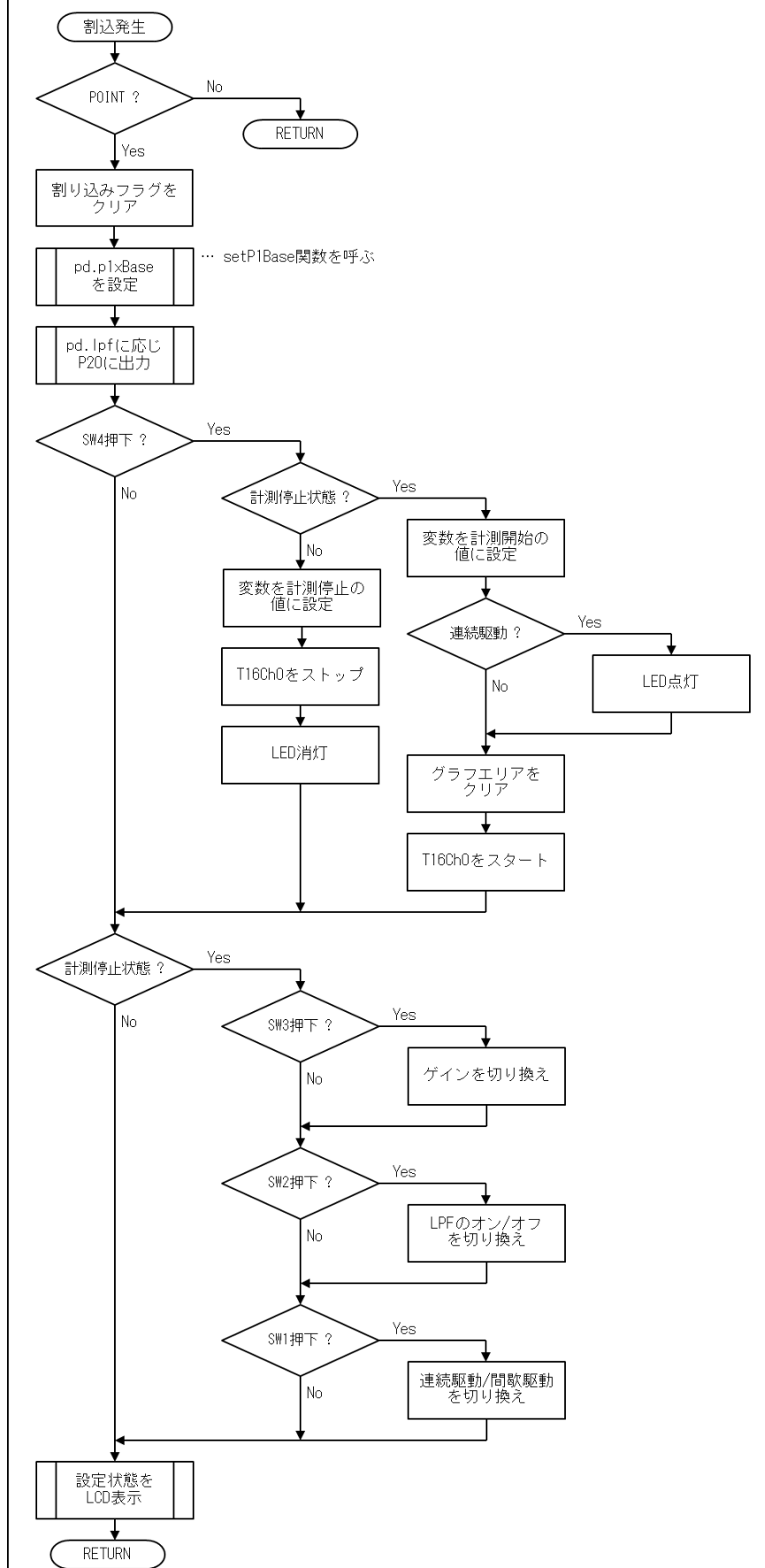
○ ADC12Ch.0 割り込み時の処理 (intAdc12Ch0 関数)

その他の処理のフローチャートを以下に記します。

初期化～T16Ch0割り込み待ち
(main関数)

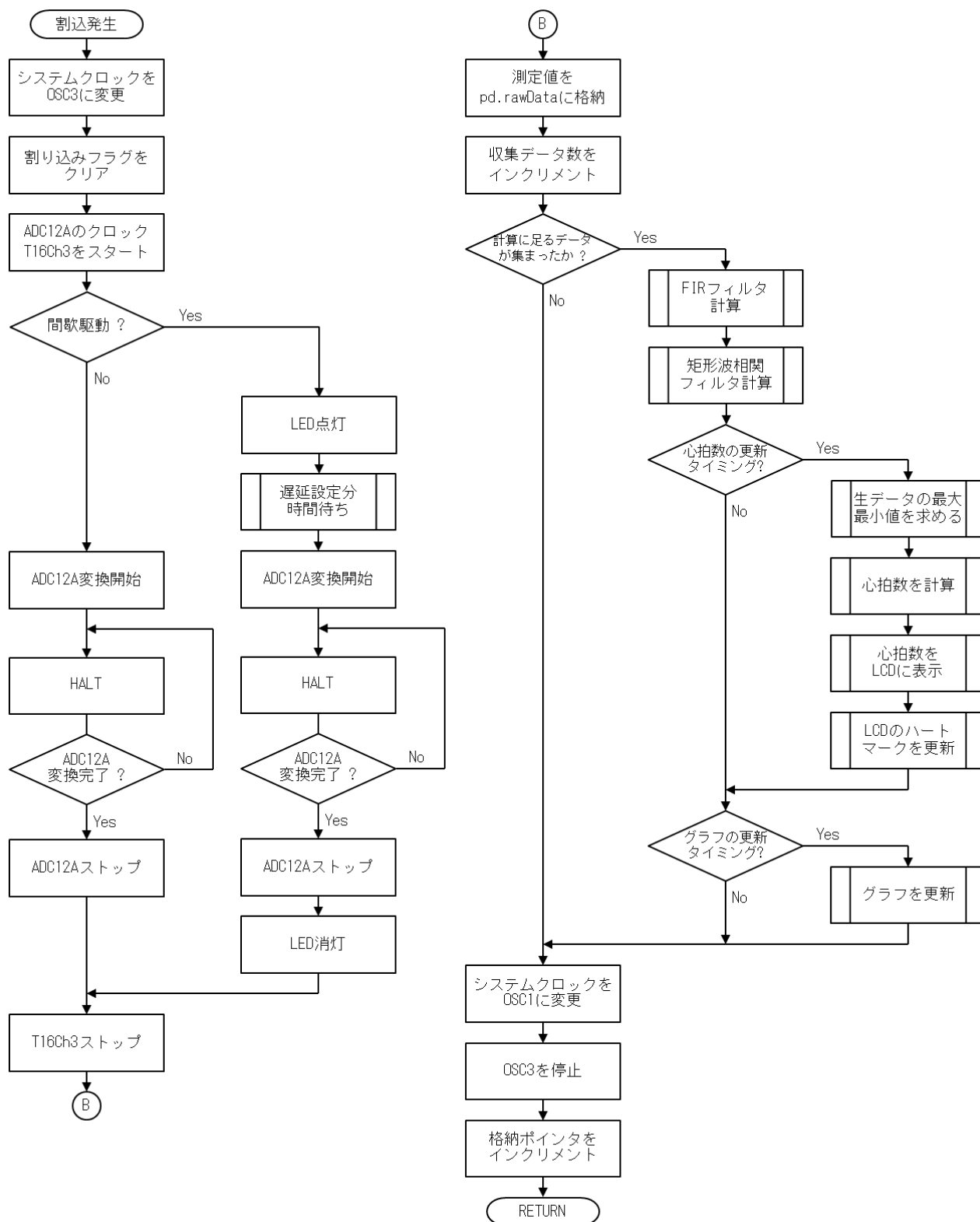


PPORT P0x 割り込み時の処理
(intPport関数)



ソフトウェア説明

T16 Ch.0 割り込み時の処理
(intT16Ch0関数)



Appendix A. 脈波検出実験の一例

この章では、実際に脈波を計測し、条件の違いによりどのように計測されるか調査した実験結果の一例を示します。

PPG センサには、表 3-3 の一番上に記載した、緑色 LED とフォトトランジスタを一体化した NJL5303R-TE1 を用いて評価しましたが、他の PPG センサでも類似した結果が得られました。PPG センサの回路定数は、表 3-4 の $R31@5mA=270\Omega$ 、 $R41=82k\Omega$ 、 $C41=4.7\mu F$ の部品を選定しました。

(1) 計測波形の安定に必要な時間

光学センサ(フォトトランジスタまたはフォトダイオード)の出力を絶対値で見ると変化があまりにも小さいため、脈波を観察することは困難です。従い、交流成分のみを取り出して増幅することで実用的なレベル変化を見られるような回路構成にしたのですが、その結果、波形が安定する迄に時間がかかるという副作用が生じます。そこで測定条件を変えた時に波形が安定する迄にどれほどの時間が必要か調査しました。調査に用いた条件、結果は、表 A-1 の通りです。(全条件の組み合わせでないのは、例えば指先を連続駆動、高アンプゲインで計測すると、脈波波形が飽和するなど不都合な条件があるため。) 実際の計測波形は、図 A-1 の通りです。

なお測定部位で、指先は人差し指の腹側を、手首は ^{とうこつ}橈骨動脈上としました。また Scan Interval=20 ミリ秒に固定しました。

表 A-1 調査に用いた条件と結果

No.	測定部位	駆動方法	アンプゲイン	LPF	波形安定時間
1	指先	連続駆動	低	なし	3 秒
2	指先	連続駆動	低	あり	5 秒
3	指先	間歇駆動	低	なし	20 秒
4	指先	間歇駆動	低	あり	20 秒
5	指先	間歇駆動	高	なし	36 秒
6	指先	間歇駆動	高	あり	36 秒
7	手首	連続駆動	低	なし	6 秒
8	手首	連続駆動	低	あり	8 秒
9	手首	連続駆動	高	なし	11 秒
10	手首	連続駆動	高	あり	8 秒

上記結果より、間歇駆動の場合は波形安定にかかる時間が長く、回路との相性は良くないと思われます。また LPF を使っても、波形安定時間はさほど変わらないので、FIR フィルタ、矩形波相関フィルタの効果を調べる次の実験は、LPF ありの条件について評価することにしました。(表 A-1 の No.欄、波形安定時間欄が薄黄色の条件。すなわち、No.2、No.8、No.10。)

Appendix A. 脈波検出実験の一例

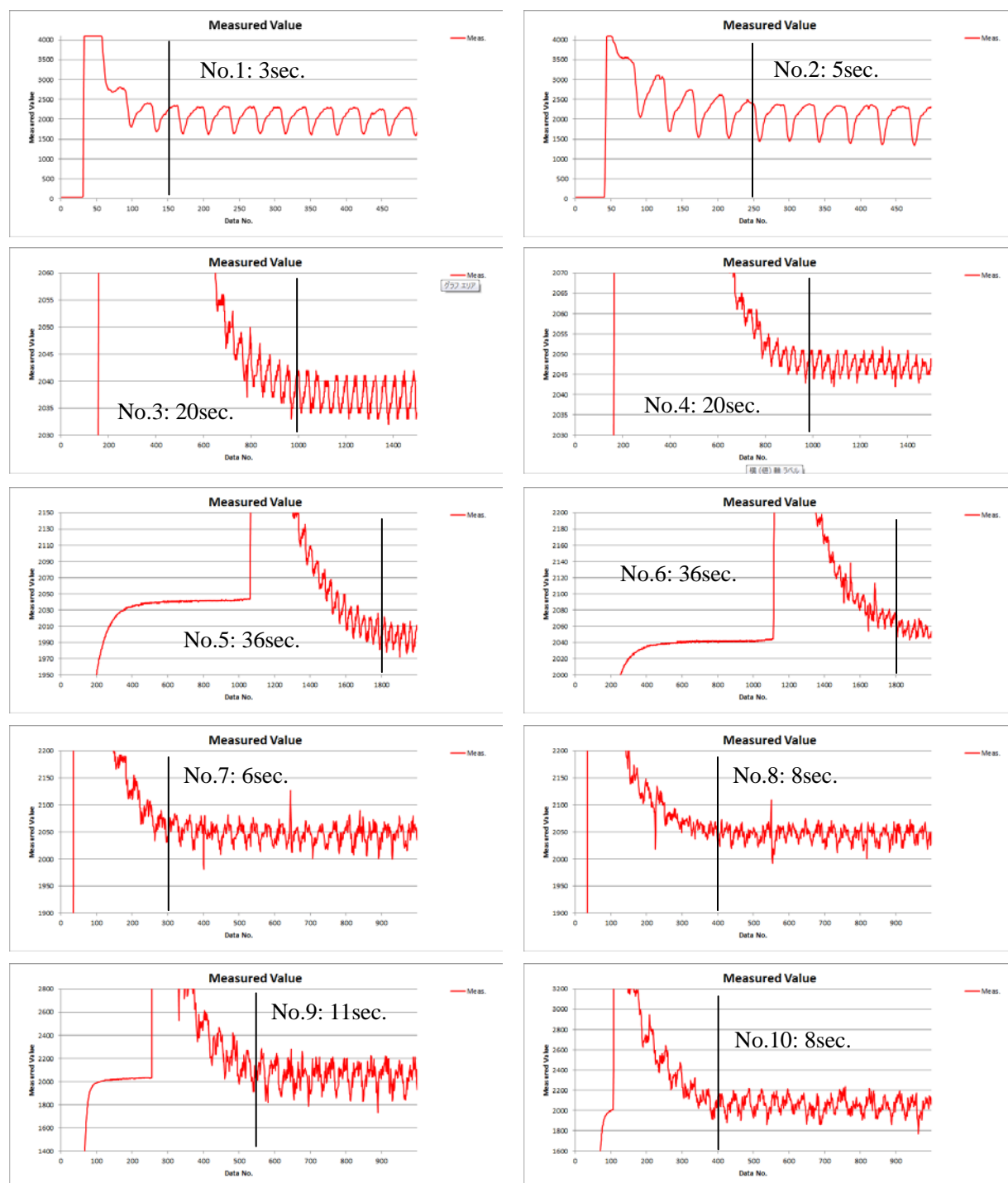


図 A-1 実際の計測波形

(2) FIR フィルタ、矩形波相関フィルタの効果の確認

No.2、No.8、No.10 の 3 つの条件について、Skipped First=10 秒に固定し、FIR フィルタと矩形波相関フィルタの効果、処理の順番を変えた時の挙動と共に確認しました。

図 A-2 は、生データを矩形波相関フィルタだけで処理した時の結果で、図 A-3 は、FIR フィルタ、矩形波相関フィルタの順で処理した時の結果です。

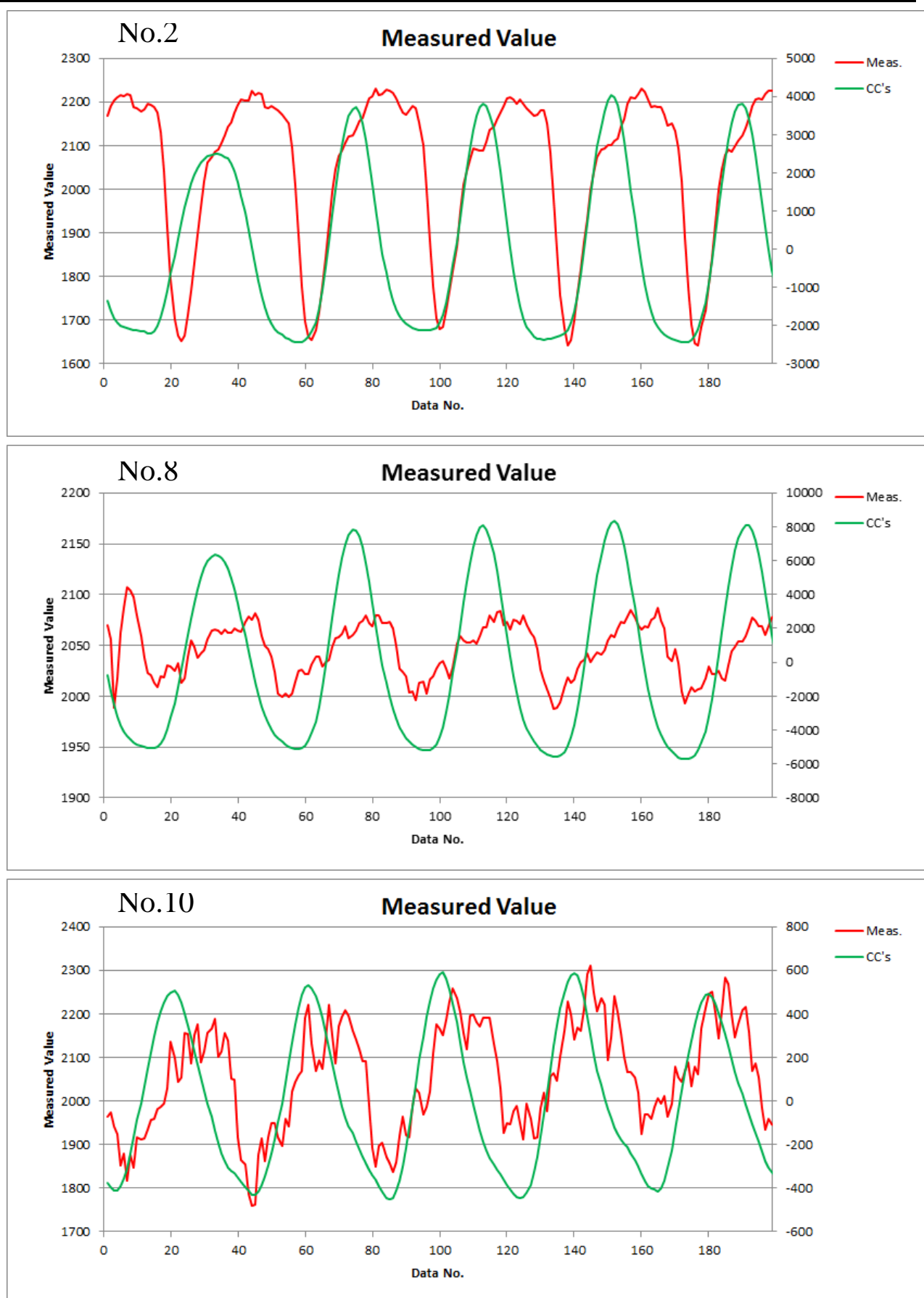


図 A-2 生データを矩形波相関フィルタだけで処理した時の結果

Appendix A. 脈波検出実験の一例

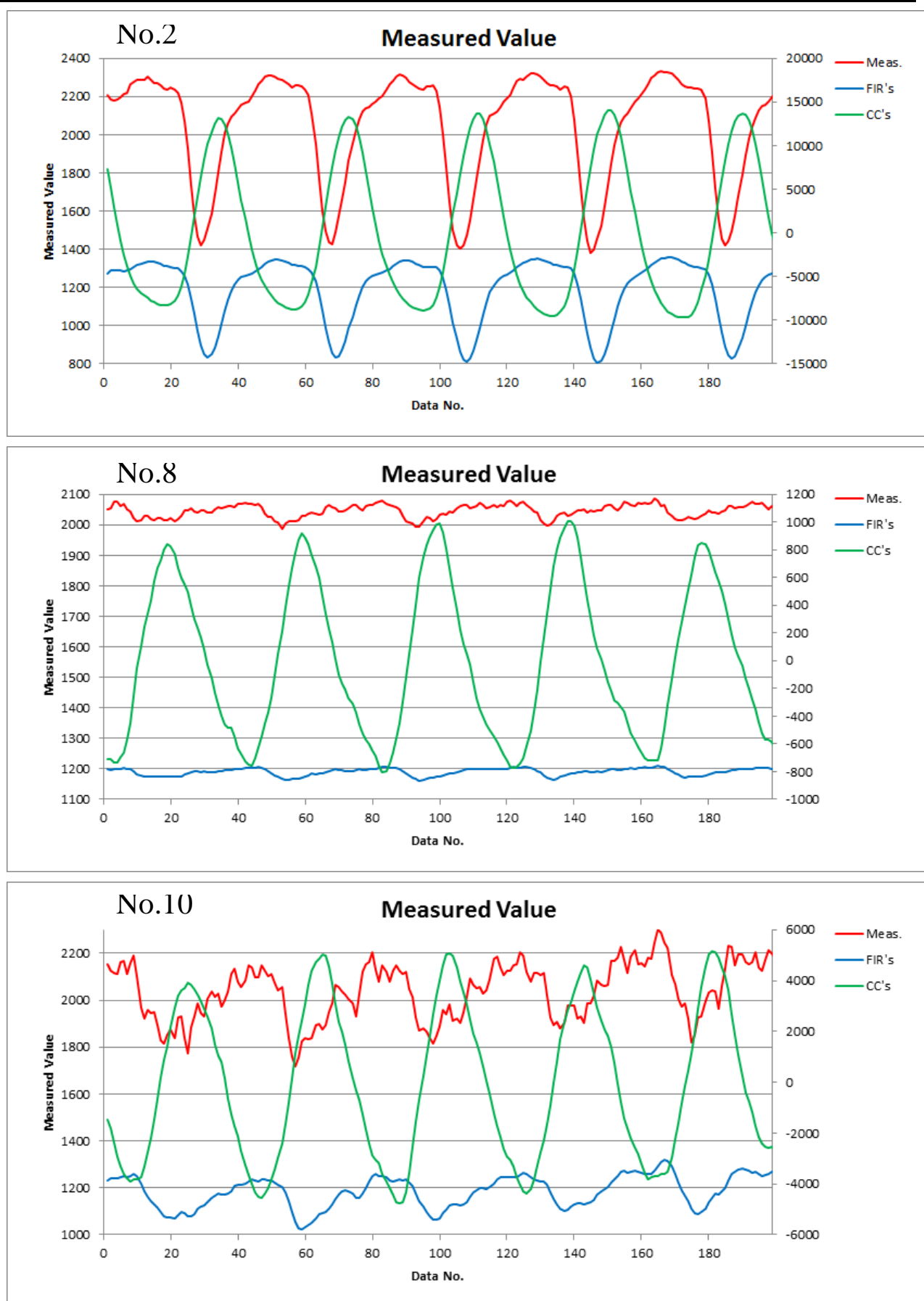


図 A-3 FIR フィルタ、矩形波相関フィルタの順で処理した時の結果

図 A-2 の生データを矩形波相関フィルタだけで処理した時の結果と、図 A-3 の FIR フィルタ、矩形波相関フィルタの順で処理した時の結果を比較しても、明瞭に優劣付けられる状況ではありません。すなわち、FIR フィルタよりも矩形波相関フィルタの方が脈波の整形に効果があり、単位時間当たりの相互相関値の最小ピークの出現数をカウントしたりすることで、脈拍数を算出することができると結論付けることができそうです。もちろんこれはハードウェアで LPF 処理していることも影響しています。

なお、サンプルプログラム `s1c17w23_ppg_demo_gnu17vx` は、ノイズの影響を低減することを目的に、FIR フィルタ、矩形波相関フィルタの順にデータ処理して心拍数を算出しています。

ただ、手首で脈波計測する場合、信号変化がとても小さくノイズの影響を受けやすいため、特に運動時の脈拍数を取得する場合には、モーションセンサなどの別のセンサと連携して算出する必要があります。また、本アプリケーションノートでは、結果の再現性の確認を容易にするために、特殊な光学系や入手困難な部品を用いていません。従い実用性を高めるためには、レンズを使った集光性能の高い光学系や、大面積の感度の高い受光素子、高輝度 LED などの採用を検討する必要があることは言うまでもありません。

参考文献

- 1) 特許庁資料. 医療用診断器具. 4.2.2 脈波計(光電式) (1997).
https://web.archive.org/web/20130331080635/https://www.jpo.go.jp/shiryou/s_sonota/map/ippan04/4/4-2.htm
- 2) SFH7050 – Photoplethysmography Sensor Application Note. OSRAM. Fig.2. p.2 (2014).
<http://www.osram-os.com/media/resource/HIRES/615710/989832/sfh-7050---photoplethysmography-sensor.pdf>
- 3) ケイタイ用語の基礎知識 第 377 回：脈拍センサーとは (2008).
<http://k-tai.watch.impress.co.jp/cda/article/keyword/40664.html>
- 4) 電脳伝説 Vintagechips 脈波計が完成 (2012).
<https://vintagechips.wordpress.com/2012/08/13/%E8%84%88%E6%8B%8D%E8%A8%88%E3%81%8C%E5%AE%8C%E6%88%90/>
- 5) デジタルフィルタ設計の家（実装担当者向デジタル信号処理入門）
http://www7b.biglobe.ne.jp/~dfd_house/
- 6) 石川高専 山田洋士 研究室ホームページ Digital Filter Design Services (2006).
<http://dsp.jpn.org/dfd/design/>
- 7) 滴了庵日録 Excel で信号処理 (2009).
<http://d.hatena.ne.jp/licheng/20091127/p1>
- 8) 特公平 07-067440 (1990).
- 9) 【俺センシング】『PC のカメラで非接触バイタル・センシングができる』 (2013).
<http://www.neo-tech-lab.co.uk/WebCam/>
- 10) Google OS 実験室 ～Moonlight 明日香～ 脈拍センシングにチャレンジ(3) (2015).
<http://google-os.blog.jp/archives/50818067.html>

[illegible]

セイコーエプソン株式会社

マイクロデバイス事業部 デバイス営業部

東京 〒191-8501 東京都日野市日野 421-8
TEL (042) 587-5313 (直通) FAX (042) 587-5116

大阪 〒541-0059 大阪市中央区博労町 3-5-1 御堂筋グランタワー15F
TEL (06) 6120-6000 (代表) FAX (06) 6120-6100

ドキュメントコード:413346600

2016 年 10 月 作成