

MCK GENERATOR 基板
MCK GEN-B 基板
 Master clock generator for Raspberry Pi
 製作マニュアル

<注意>

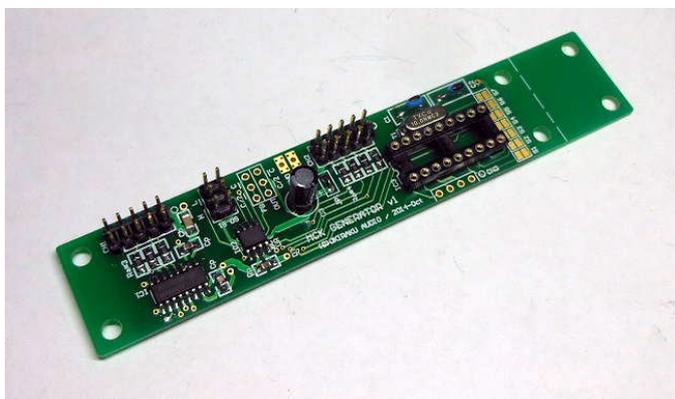
本キットをつかって生じた感電、火災等の一切のトラブルについては、当方は責任を負いませんのでご了承ください。また、基板、回路図、マニュアル等の著作権は放棄していませんので、その一部あるいは全体を無断で第三者に対して使用することはできません。

1. はじめに

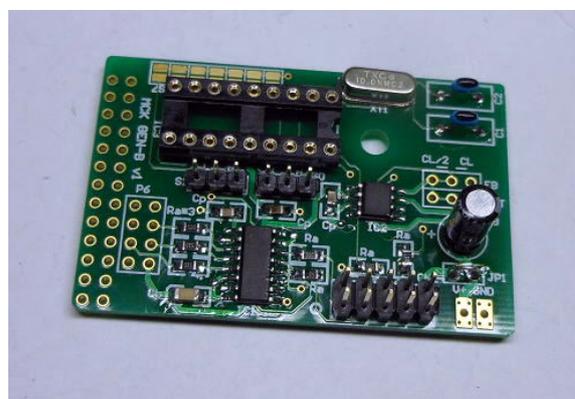
本基板はPCM信号におけるBCK(Bit Clock)よりMCK(Master Clock)を生成する基板です。通常のDAC-ICでは内部の信号処理のために外部よりMCKの供給が必要になりますが、音楽ソース源として使われるRaspberry Pi(以下RP)からはMCK信号はでていません。すなわちRPから出ている信号はDATA, LRCK, BCKの3信号のみになります。PCM5102など3信号線で動作するDACは数が少ないのですが、BCKを逡倍することでMCKを生成する基板があれば、一般的なDACが使用可能になります。本基板はRaspberry Piと多種のDACとの接続を目的にMCKを生成するために作成しています。

基板は2種類あり、1つはMCK GENERATOR基板であり本WEBでリリースしているDACにアドオンできる形状としたもので、もう1つはMCK GEN-B基板でありRP(TYPE-B)そのものに搭載可能な形状としたものです。どちらも機能および回路は同じですので、使用する状況に応じて使い分ければいいでしょう。

MCKを生成する場合のBCK設定倍率についてはジャンパーピンで2~16倍の範囲で設定可能です。またオプションのマイコンを搭載すればサンプル周波数(fs)毎に適切な倍率の自動設定が可能になります。また、本基板ではデジタルアイソレータを搭載しています。PRとのGND分離ができますので、電源に不要なノイズの重畳を抑えることができます。



(a)MCK GENERATOR



(b)MCK GEN-B

完成例(マイコンは搭載されていない状態です)

2. 機能&仕様

表 主な仕様

主使用素子	クロック逡倍器 : ICS570B、デジタルアイソレータ : Si8440
機能	BCK信号より逡倍しMCK信号を生成。
入力	Raspberry Pi(B)からのピンレイアウトに一致(8Pコネクタ)
出力	PCM信号(DATA, LRCK, BCK, MCK) 10Pコネクタより出力
必要電源	+3.3V (Raspberry Pi 給電、DAC側から給電の2系統) ※デジタルアイソレータにより入出力はGND分離。
基板	FR4、70μm銅箔厚、金フラッシュ MCK GENERATOR: 119mm×23mm MCK GEN-B: 35mm×54mm

3. 基板端子

3-1. 基板端子

表 基板端子機能

No	機能	備考
V+	3.3V 電源入力	DAC 側回路の電源入力端子 (CN2 を介して電源を供給 (*1) する場合は接続不要)
GND	電源 GND	

(*1) JP1 を接続することで CN2 の P9, 10 が 3.3V 電源ラインに接続されます。

3-2. コネクタ機能

(1) CN1 (MCK GENERATOR)、P6 (MCK GEN-B)

CN1 あるいは P6 は RP と接続する 8P コネクタになります。

表 CN1, P6 コネクタ配置

PIN	機能	説明	PIN	機能	説明
1	3.3V	3.3V 電源入力。RP より供給	2	NC	
3	LRCLK	ワードクロック入力	4	BCK	ビットクロック入力
5	DATA	データ入力	6	NC	
7	GND	GND: 信号リターン	8	GND	GND: 信号リターン

RP (TYPE B) は P6 コネクタを使用します。P6 コネクタの端子機能は下図のようになっています。部部品面からのコネクタ接続を想定しているため、下図のように 1 番ピンが「3.3V」になります。

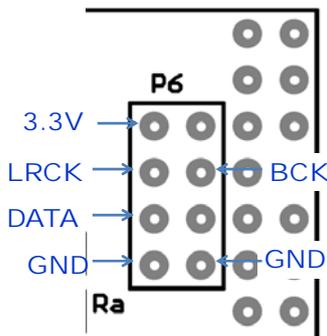


図 P6 コネクタ (RP TYPE B)

(2) CN2

外部 DAC に接続するための PCM 出力信号です。

表 CN2 端子機能 (PCM 出力)

PIN	機能	説明	PIN	機能	説明
1	DATA	データ出力	2	GND	GND: 信号リターン
3	LRCLK	ワードクロック出力	4	GND	GND: 信号リターン
5	BCLK	ビットクロック出力	6	GND	GND: 信号リターン
7	SCLK	システムクロック出力 (MCK)	8	GND	GND: 信号リターン
9	N.C	(*1)	10	N.C	(*1)

(*1) P9, 10 についてはコネクタの直近に基板内 Vcc (3.3V) と接続し易いようにジャンパーパッド有り。通常は DAC 側のコネクタの Pin9, 10 に 3.3V を供給し、コネクタを介して MCK GENERATOR あるいは MCK GEN-B に電源を供給するのが便利でしょう。

(3) CN3 (MCK GENERATOR のみ)

未使用です。なにも接続しないでください。

4. 部品表

次表に部品表例を示します。

表 部品表 (例)

品名	番号	規格	仕様	個数	
抵抗	Ra	チップ抵抗	51Ω	7	ダンピング抵抗
コンデンサ	C1, 2	セラミックコンデンサ	22pF	2	マイコン未搭載時は不要
	C3	電解コンデンサ	47uF/16V	1	
	Cb	チップコンデンサ	0.1uF	4	2012 サイズ
	Cp	チップコンデンサ	1uF	1	3216 サイズ
水晶	XT1	HC-49S	10MHz	1	マイコン未搭載時は不要
IC	IC1	アイソレータ	Si8440 など	1	
	IC2	クロック通倍器	ICS570B	1	
	IC3	マイコン	PIC16F819 など	1	

ハッチング部は基板キットに付属。マイコン(IC3)を使用しない場合は、XT1, C1, C2 の実装不要。

5. Raspberry Pi との接続における注意点

MCK GEN-B 基板と Raspberry Pi との固定はコネクタならびに 1本のネジで固定されます。この位置がずれると固定がしづらくなります。すなわちコネクタのピン穴 (ランド穴) の直径に余裕があるため、コネクタ単体を先に基板に半田付けすると傾いて取りつく可能性があります。そのため、コネクタの半田付けの前に Raspberry Pi と本基板を一旦組み立てた状態で、コネクタの半田付けをすることをお勧めします。このようにすれば、位置づれの発生を回避することができます。なお、本基板には P6 (8P コネクタ) 以外にも 26P コネクタも実装可能で Raspberry Pi と接続できるようになっていますが、このコネクタには配線はされていません。単純に Raspberry Pi との固定強化のために用いてもいいでしょう。

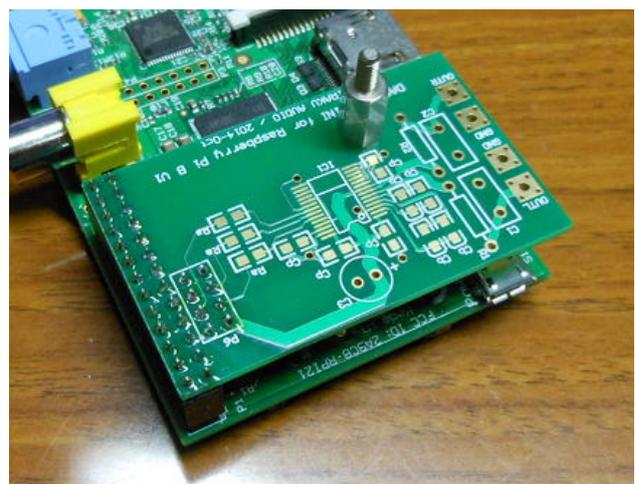
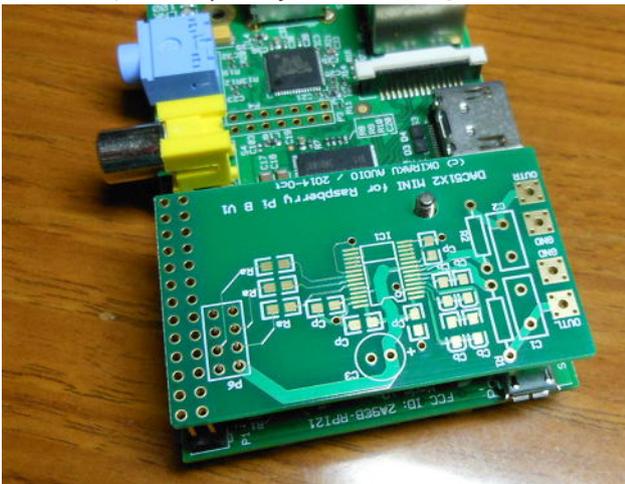


図 予めコネクタを取り付けて、組み立てた状態でコネクタを半田付けすると位置づれの発生を回避することができます (※図中の基板は別のものですが、接続のイメージを示しています)。

6. 通倍クロックの設定方法

6-1. ジャンパー設定する場合

基板上の S1, S0 ジャンパーを使用します (S1~S7 と連続している半田パッドとは名称は同じですが機能は異なります)。倍率の設定は下記の通りです。なお、実際に設定可能な周波数は ICS570B の入力および出力周波数の範囲から、次次表のハッチング部分になります。

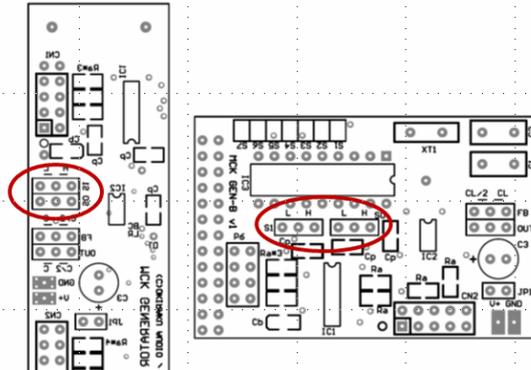


図 通倍クロックの設定ジャンパー (S1, S0)

表 クロック倍率の設定方法

S1	S0	倍率	MCK 周波数	BCK 周波数範囲 (MHz)
L	L	停止	-	-
L	M	3	192fs	3.75 to 28
L	H	4	256fs	2.75 to 19
M	L	8	512fs	2.5 to 9.5
M	M	6	384fs	2.5 to 12.5
M	H	10	640fs	2.5 to 7.5
H	L	1	64fs	11 to 75
H	M	16	1024fs	2.5 to 5
H	H	2	128fs	5.5 to 37.5

M はジャンパーピンを開放した状態。

下表よりわかるようにサンプル周波数 (fs) は 32kHz では動作できません。これは BCK 周波数が ICS570B の動作最低周波数を下回るためです。また fs=44.1kHz あるいは 48kHz 時においても、MCK 周波数を 128fs、192fs には設定できず 256fs 以上 (4 倍以上) とする必要があります。

図 設定可能な倍率範囲 (黄色のハッチング部分が設定可能範囲)

	fs(kHz)→	32	44.1	48	88.2	96	176.4	192	384
MCK		MCK(MHz)							
64fs	1 × BCK	2.048	2.8224	3.072	5.6448	6.144	11.2896	12.288	24.576
128fs	2 × BCK	4.096	5.6448	6.144	11.2896	12.288	22.5792	24.576	49.152
192fs	3 × BCK	6.144	8.4672	9.216	16.9344	18.432	33.8688	36.864	73.728
256fs	4 × BCK	8.192	11.2896	12.288	22.5792	24.576	45.1584	49.152	98.304
384fs	6 × BCK	12.288	16.9344	18.432	33.8688	36.864	67.7376	73.728	147.456
512fs	8 × BCK	16.384	22.5792	24.576	45.1584	49.152	90.3168	98.304	196.608
768fs	12 × BCK	24.576	33.8688	36.864	67.7376	73.728	135.4752	147.456	294.912
1024fs	16 × BCK	32.768	45.1584	49.152	90.3168	98.304	180.6336	196.608	393.216

6-2. マイコン(オプション) を使用する場合

(1) オプションのマイコンを搭載して周波数に応じた自動設定をする場合は、クロック倍率の設定ジャンパーの S1, S0 は必ず開放としてください。

(2) 周波数毎の倍率の設定方法

各周波数毎での倍率の設定は S1~S7 を用います。下表に S1~S7 の設定と倍率を示しています。一般的な設定としては S1...S7=HHHHHL とするのがいいでしょう。すなわち 44.1~192kHz までは 256fs とし、384kHz のみ 128fs とする設定です。

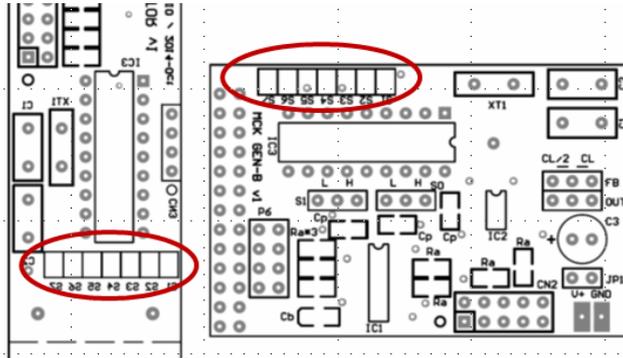


図 マイコン搭載時の周波数倍率設定ジャンパーS1~S7

表 S1, S2 の設定 (44.1、48kHz の入力時の倍率)

S2	S1	44.1/48kHz 時の倍率
H	H	4 × BCK (推奨)
H	L	6 × BCK
L	H	8 × BCK
L	L	16 × BCK

(*) H: 開放 L: 短絡

表 S3, S4 の設定 (88.2/96kHz の入力時の倍率)

S4	S3	88.2/96kHz 時の倍率
H	H	2 × BCK (推奨)
H	L	4 × BCK
L	H	6 × BCK
L	L	8 × BCK

(*) H: 開放 L: 短絡

表 S5, S6 の設定 (176.4/192kHz の入力時の倍率)

S6	S5	176.4/192kHz 時の倍率
H	H	2 × BCK (推奨)
H	L	3 × BCK
L	H	4 × BCK
L	L	6 × BCK

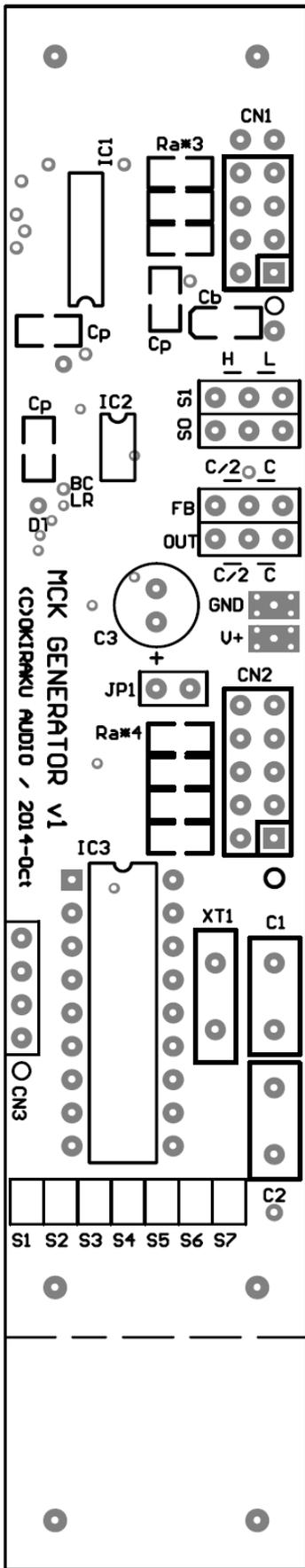
(*) H: 開放 L: 短絡

表 S7 の設定 (384kHz の入力時の倍率)

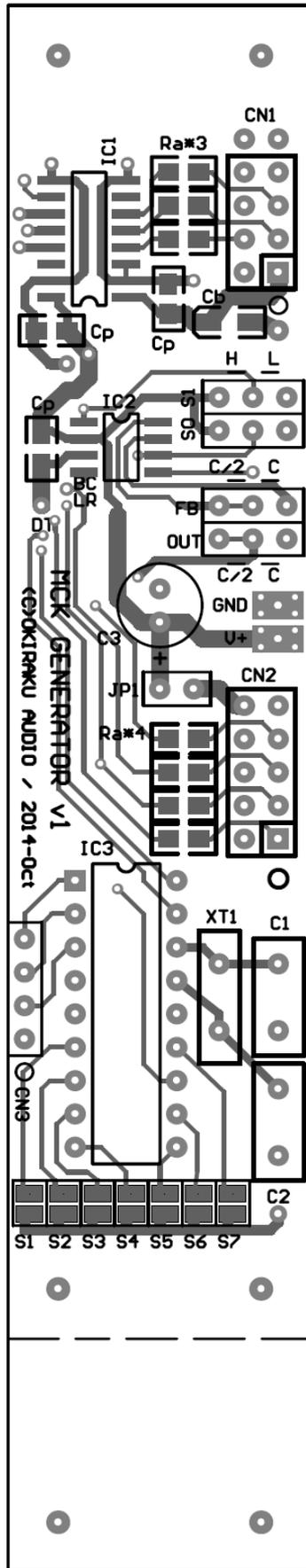
S7	384kHz 時の倍率
H	1 × BCK
L	2 × BCK (推奨)

(*) H: 開放 L: 短絡

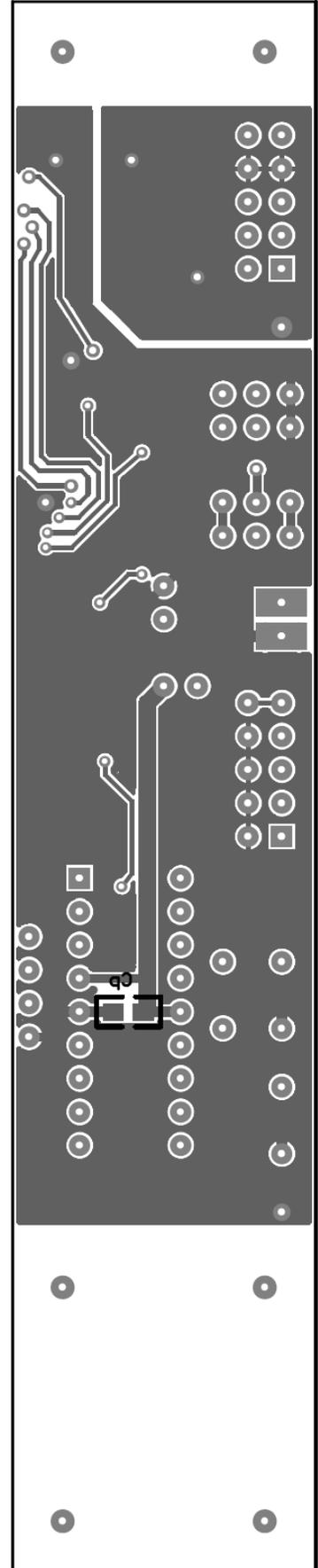
8. 基板パターン
8-1. MCK GENERATOR



(a) シルク



(b) 部品面パターン
☒ MCK GENERATOR



(c) 半田面パターン

8-2. MCK GEN-B

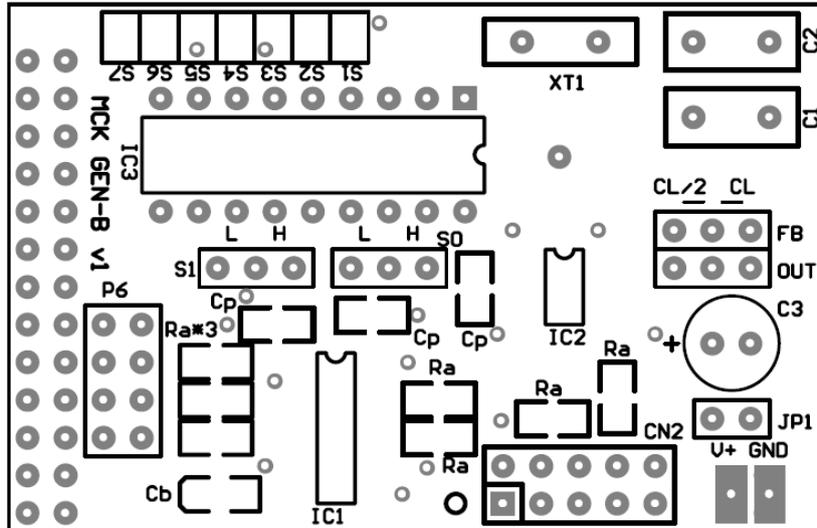


図 基板シルク

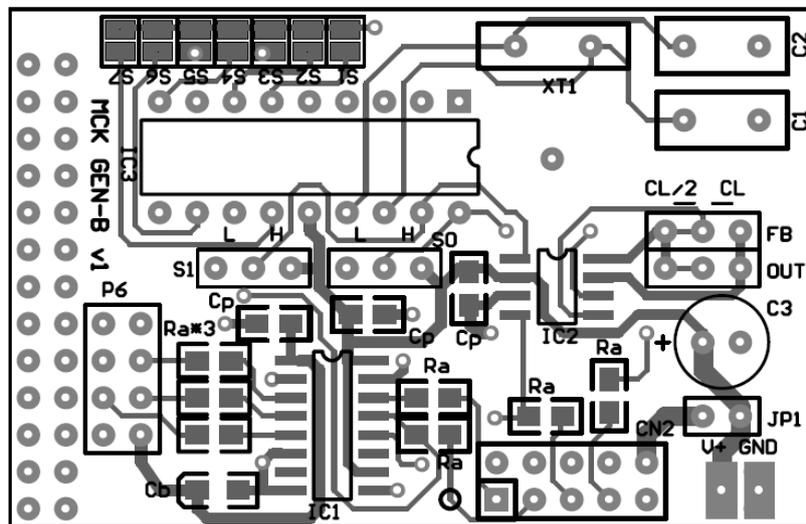


図 部品面パターン

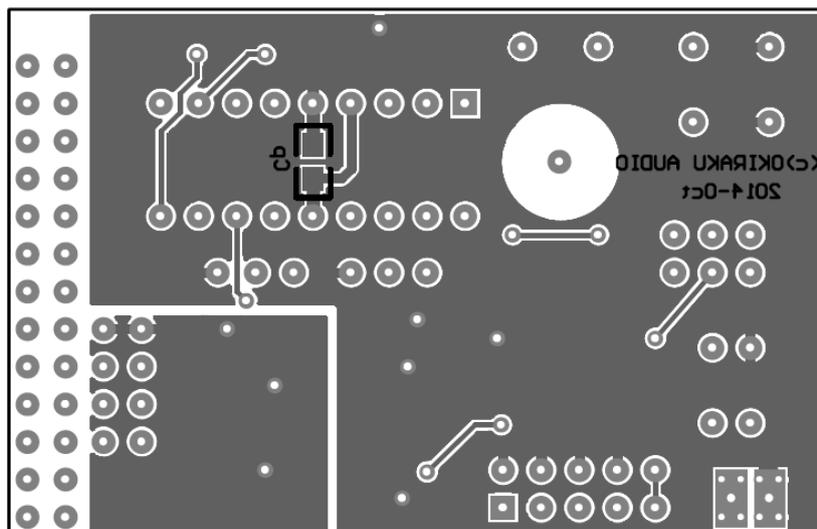
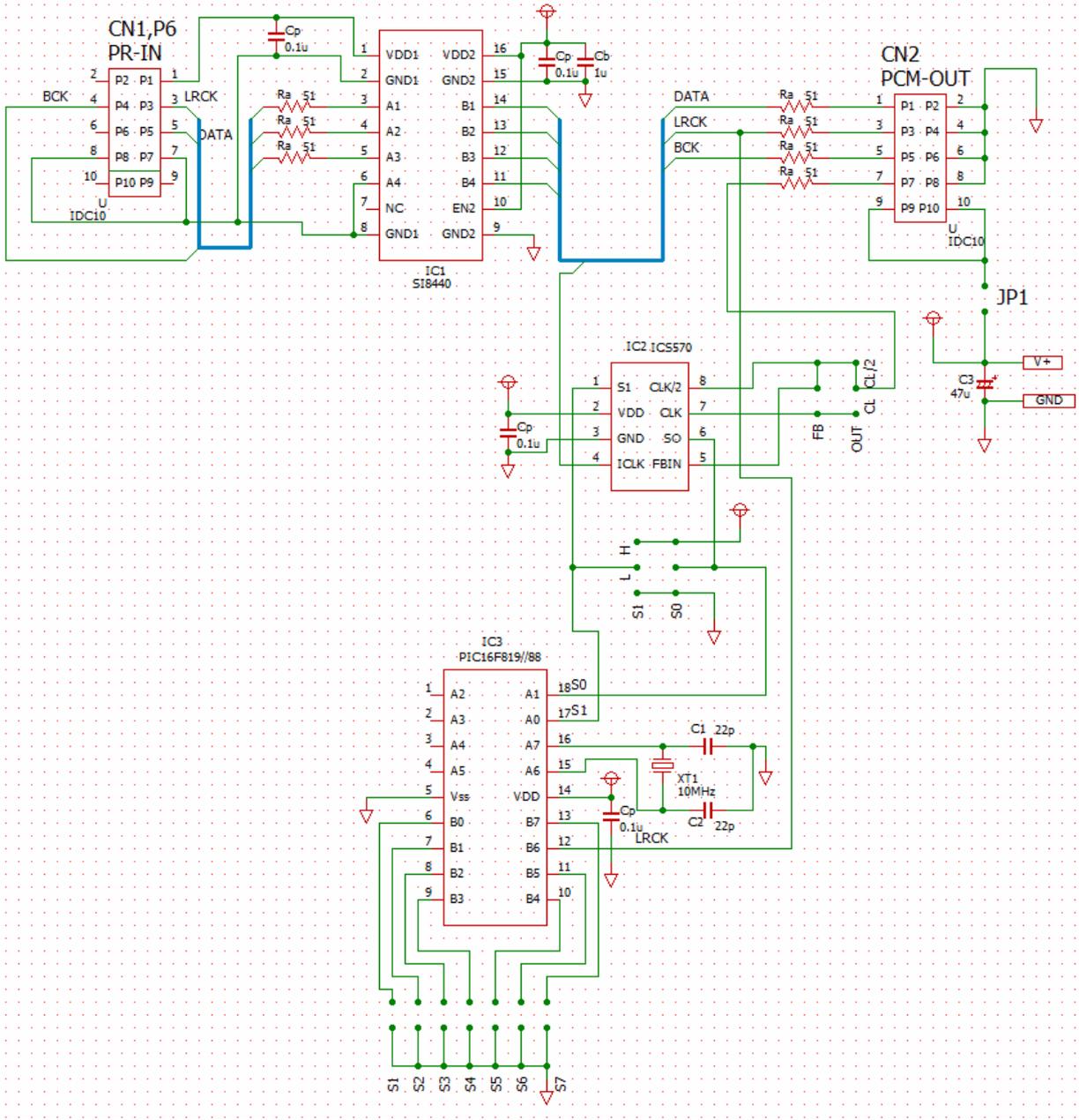


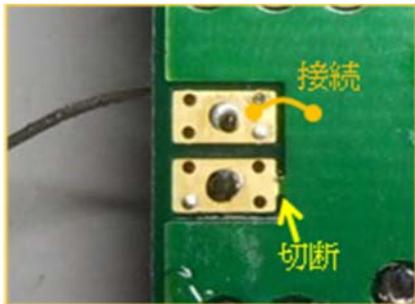
図 半田面パターン

9. 回路図



10. 【重要】MCK GENERATOR V1 基板のパターン修正

MCK GENERATOR V1 基板には一部修正箇所があります。修正箇所は下図の電源入力部分であり、黄色線部の接続を切断し、1つのパッドをベタ面に接続します。



(a) 修正箇所



(b) 修正方法

- ①黄色線部を切断
- ②レジスト削り半田でベタ面と接続

図 修正部分と修正方法

1 1. 注意点：補足

Raspberry Pi は LINUX で動作するマイコン基板であり、電源投入から立ち上がりまでには時間がかかります。例えば音楽再生ソフトの Volumio を用いる場合においても、LINUX が立ち上がり、アプリケーションソフトが立ち上がるまでは DAC に必要な信号である DAC、LRCK、BCK などは出力されません。そのため MCK GENERATOR 基板 (MCK GEN-B も同様) から出力される MCK 信号もアプリが立ち上がるまで出力されません。しかし多くの DAC (特にソフト制御される場合) においては内部処理に MCK 信号を使用している場合が多く、先に DAC が立ち上がった場合にはしばらく MCK が供給されないため、DAC の内部処理時にエラーが生じる場合があります。この事象を回避するには、先に Raspberry Pi のアプリを立ち上げる、あるいは Raspberry Pi のアプリが起動してから DAC 側の電源を投入するなどの処置が必要です。あるいは、DAC 側で内部処理のエラーの発生を検知して、対処するなどの対策が必要になります。

DAC179X-2.1 においても同様で、入力を PCM に設定して Raspberry と DAC を同時に電源投入すると、先に DAC が立ち上がりますが MCK 信号が供給されていない状態のため、内部処理にエラーが発生します。そこで DAC179X-2.1 の PIC プログラム「V3」においては、内部処理エラーが発生した場合は強制的に PIC マイコンをリセットする機能を追加しました。これにより Raspberry が立ち上がり MCK が供給されるまで、エラー処理 (リセット) を繰り返しますが (プチ、プチとノイズが発生する可能性ある)、MCK の供給が始まると通常の動作に入るようにしています。

1 2. 編集履歴

R1 2014.10.31