

APB-2 組み立てマニュアル

2011/07/04 version 1.00

OjisanKoubou

(c)2011 OjisanKoubou, All Rights Reserved

■ 重要説明事項(必ずご一読ください)

- 本キットは個人が電子工作を楽しむことを目的としています。本キットで提供しているハードウェア、ソフトウェアの全部もしくは1部を製品に組み込んだり販売したりすることはおやめください。
- 本キットを使ったことにより直接的、間接的に被害、損害を被ったとしても一切補償しません。
- 組み立てに失敗した場合や使用中に故障した場合でも交換、保証、返金などには一切応じられません。
- この説明書およびホームページなどに掲載されている特性、性能はあくまで1製作例であり、すべてのキットで同一性能が保証されているわけではありません。良い場合もあれば悪い場合もあることをご承知おきください。
- 充分注意して設計をしていますが、重大な設計ミスやバグがないことの保証はありません。
- 回路、使用部品、基板、ソフトウェアなどは予告なく変更することがあります。
- 基板をむき出しの状態で使用すると予期せずショートしたりする危険性があります。完成したあとはケースに入れてお使いください。

■ 組み立てる前に

- 部品がすべて揃っているかご確認ください。不足部品がありましたら組み立て前にご連絡ください。
- 一部の部品では必要数より多く入っている場合があります。何個余るのかを袋や部品表にメモしておくとし組み立て後に必要な部品が全部マウントされたかどうかの確認が楽です。余った部品は他の工作にお役立てください。
- 全ページを印刷し、終わったところにはチェックマークを入れていきます。適宜気づいたことをメモしていくとよいでしょう。
- 一度、組み立て手順をすべて読んで、全体のイメージをつかんでおいてから始めると良いでしょう。

■ 組み立てに必要な工具など

- やに入り半田(太さ0.8mm以下の鉛入りのものを使いやすい)
- 無洗浄フラックス(Sunhayato HB20Fなど)
- マスキングテープ(部品の仮止めに使います)
- 半田ごて(温度調節付の60~80Wぐらいのものがお薦め)
- ピンセット(チップ部品をつかむため、竹串を削ったものでも可)
- ニッパー(先の細い良く切れるもの)
- ルーペ(5~10倍ぐらい)
- デジタルテスター

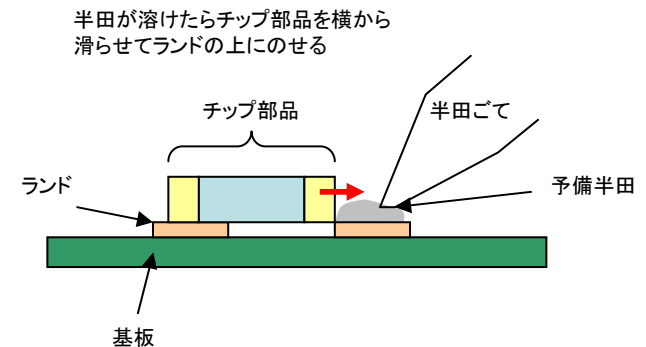
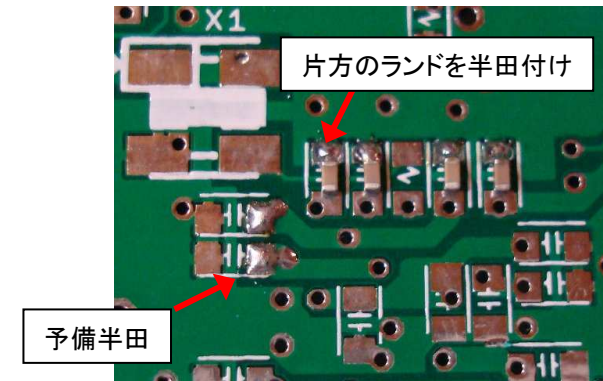
ピンセットは動作にあまり力がいらぬ先が曲がったもの(たとえばGootのTS-15など)がチップをつまむのに都合が良いようです。

■ 組み立てる際の注意点

- 半田付けは半田付けする部分の温度を十分に上げてから半田を供給するのがコツです。半田付けする場所に半田ごてをあてて2秒、半田をながして2秒ぐらいのイメージでやるとよいでしょう。半田がすぐにとんがってしまう場合は半田ごての温度を下げてください。
- 半田ごてでランドに力を加えないように気をつけてください。半田ごてでランドの上を動かす際も半田ごての先をランドに強く当てないで滑らすようにします。特にランドの短手方向への力は禁物です。ランドは最小幅0.25mmしかなく、一度剥がれてしまうと修復は非常に困難です。
- 各段階で半田付け箇所のルーペでの確認(イモ半田になっていないか、ブリッジしていないか、半田ボールがないか)を十分に行ってください。半田付けが良くないと一度は動作してもあとで動かなくなることがあります。特に電源関係の不具合はICなどの破壊に直結します。
- 静電気に弱い部品があります。部屋の湿度を上げる、手洗い、部品に触る前に静電気を逃がす、などの対策を行ってください。とはいっても真冬などのセーターを脱いだらバチバチいうようなとき以外はそれほど神経質になる必要はありません。

■ チップ部品などの半田付け方法

- 片側のランドに「予備半田」をしておきます。
- 半田ごてで「予備半田」を溶かし、部品を横から滑らせて半田付けします（右下図）。
- もう片方も半田付けします。このとき半田ごてで部品とランドを2秒程度暖めてから半田を流しさらに2秒で半田ごてを離します。半田がすーっと流れるようならOKです。GNDにつながっているランドなどは熱が逃げるのでもっと時間をかける必要があります。
- 先に半田付けしたところに再度半田を足します。
- コネクタはマスキングテープで仮止めしてまず1箇所半田付けします。その後、コネクタを押しながら最初に半田付けしたピンをこてで溶かすとパチンと収まり浮きがなくなります。浮きがないことを確認したら残りのピンの半田付けをします。
- GNDなどで半田が流れにくいときは十分に時間をかけて熱します。最近の部品は熱ではそう簡単には壊れません。



ランド land
 プリント基板 (PWB Printed Wiring Board) 上で部品がハンダ付けできるように銅が出ている部分。そのほかの部分にはソルダーマスク (Solder Mask) がかけられているのでハンダ付けできない。

■ QFP IC の半田付け方法

■ ICの下のスルーホールなどに半田が溜まって持ち上がっている場合は吸い取り線で平らにしておきます。

■ ランドにフラックスを塗ります。

■ ICの方向を間違えないように確認し、ICをランドの上ののせます。

■ ランドとICの足がすべての辺で正確にあっていることを確認します。ちょっとでもずれているとモワレになるのでわかります。

■ ICを動かさないように1箇所半田付けします(右図)。

■ もう一度ランドとICが合っているか確認し、先に半田付けしたのと対角位置の足を半田付けします。

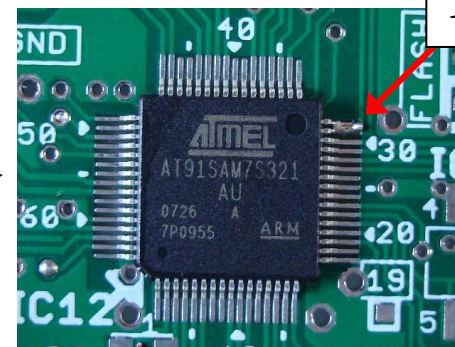
■ フラックスを再度ランドとICの足に塗ります。

■ 半田ごて先に少量の半田(1mm分ぐらい)をのせます。

■ 半田ごてをランドの端にあて、ICの足先までゆっくり移動させます(次ページ参照)。そのとき半田が一緒にすーっと流れるような感じです。だいたい2、3ピンを一緒に半田付けします。

■ 半田の量が多いとブリッジしてしまいますが気にせずすべての足を半田付けしてしまいます(右図)。

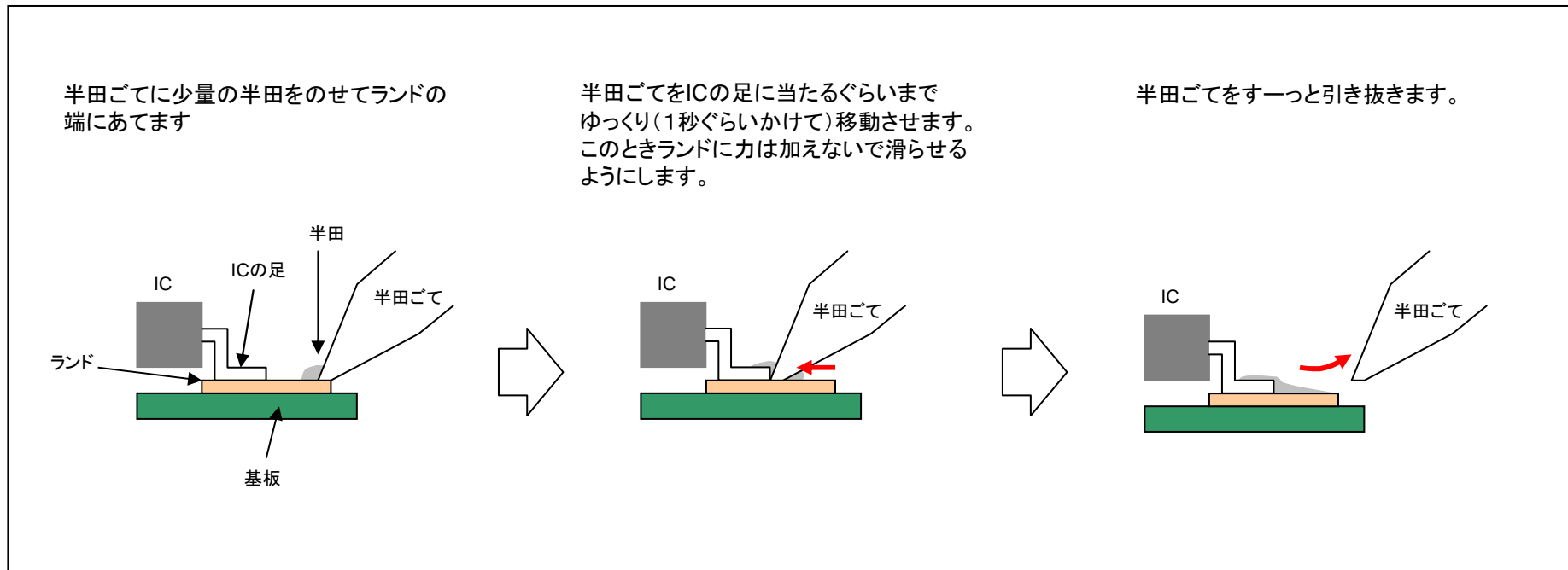
ここに示したのは私がやっている方法ですが、ほかにもいろいろなやり方があります。
たとえば”QFP 半田付け”でググるとたくさんヒットします。
うまくいかないときは試して見ると良いでしょう。



1箇所半田付けする



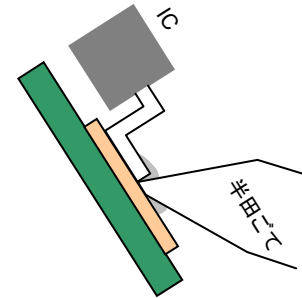
ところどころブリッジしているが気にしない



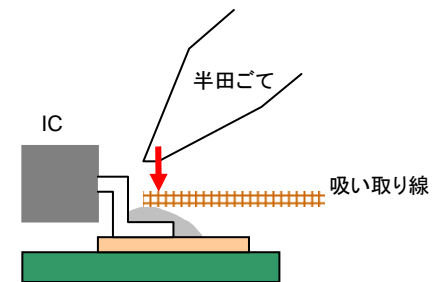
•半田付けが終わったらブリッジや未半田がないかルーペで確認します。基板の反対側から光が入るようにして見るとICの足の間がくっきり見えてブリッジを発見しやすくなります。

•もしブリッジしていた場合、次ページの方法で修正してください。

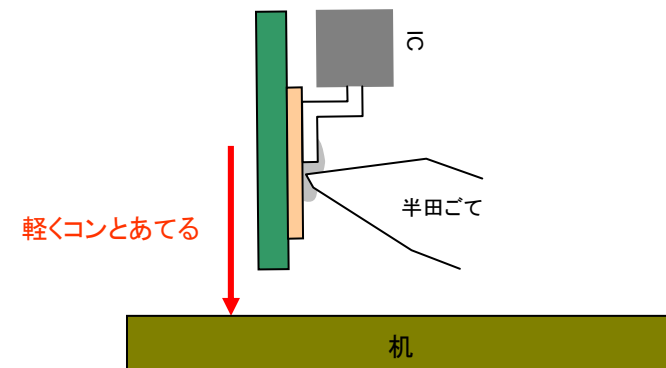
■ブリッジしているところにフラックスを塗り、右図のように基板を傾けて半田ごてをあてます。ブリッジしていた部分の半田が重力ですうーっと半田ごてのほうに流れてきます。



■それでもだめなときは、吸い取り線をICの足の上から半田ごてで押し付けます。吸い取り線を十分高温にするのがコツです。



■それでもだめなときは、フラックスを塗ってからブリッジした部分を半田ごてで溶かし、机の上などで軽くコンとショックを与えます。このとき落ちた半田が基板のほかの場所にくっついてしまうことがありますのですべて取り除いてください。



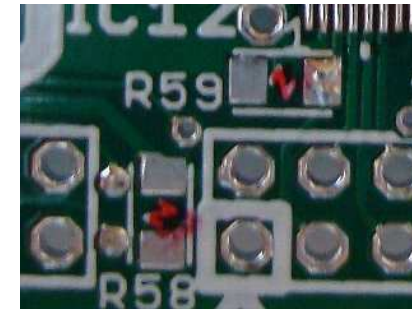
■ 組み立て

■ マウントした部品に順次マーキングしていきます。ピンク色がマウントする部品です。

■ マウント図で青の×印はマウントしない部品です。マウントしないチップ部品にはあらかじめ基板にマジック(赤が見やすい)でマーキング(右図)しておきますと間違えてマウントすることが防止できます。

■ 各ステップごとにマウントした部品のチェック、半田付け状態のチェック、動作確認をします。部品を間違えたり、半田付けが良くない状態で通電すると部品を損傷することがあります。確実な組み立てのため、面倒だと思わずにすべて実施されることをお勧めします。なにごとステップバイステップです。

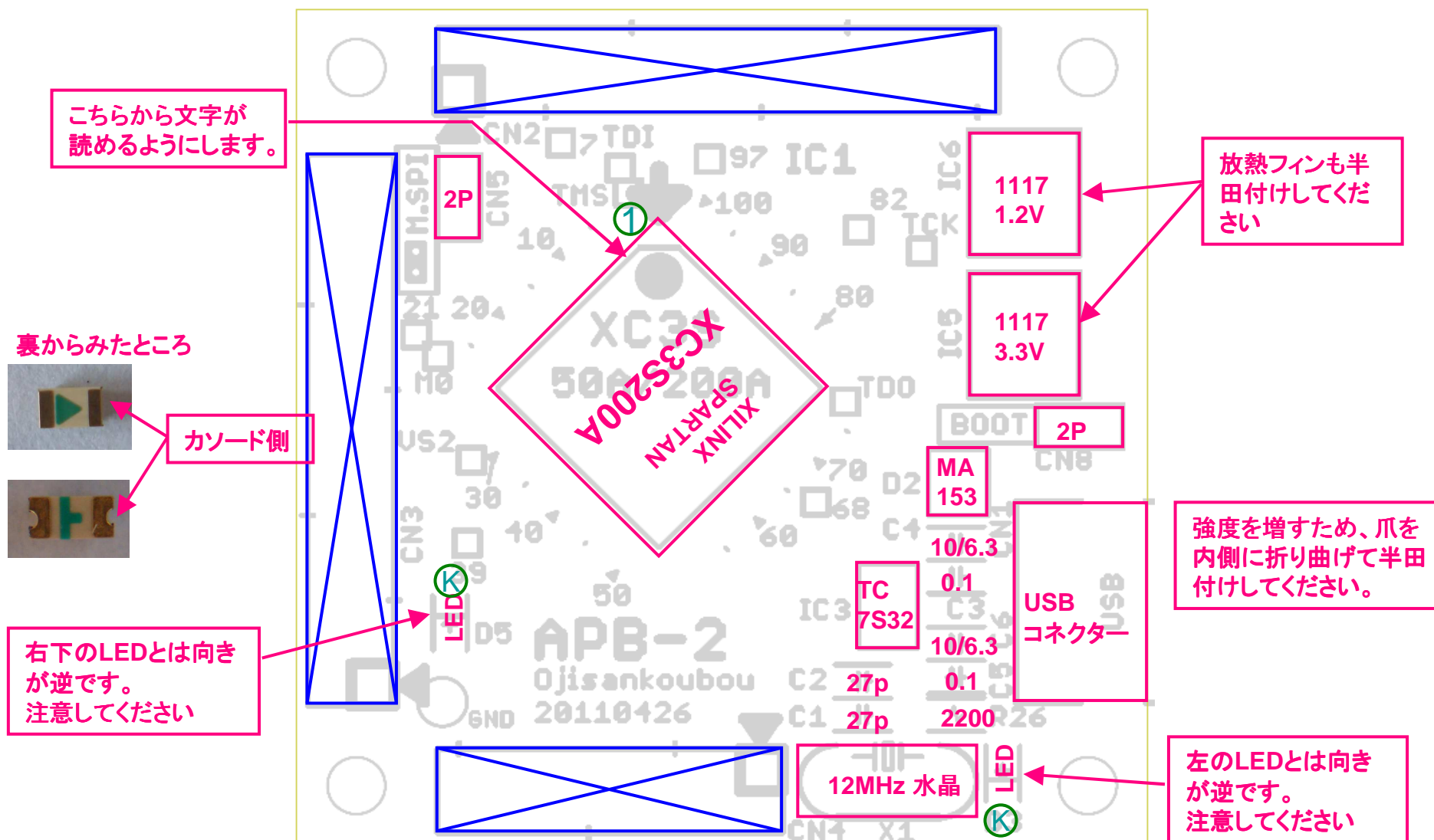
■ チップ部品は2012(2.0mm x 1.2mm)や1608(1.6mm x 0.8mm)サイズと非常に小さいので息を吹きかけただけで飛んでいってしまいます。1個ずつ取り出してマウントするようにします。別の種類の部品をマウントするときは机の上に他の部品が残っていないことを確認します。他の部品と混ざってしまうと探すのはほとんど不可能です。



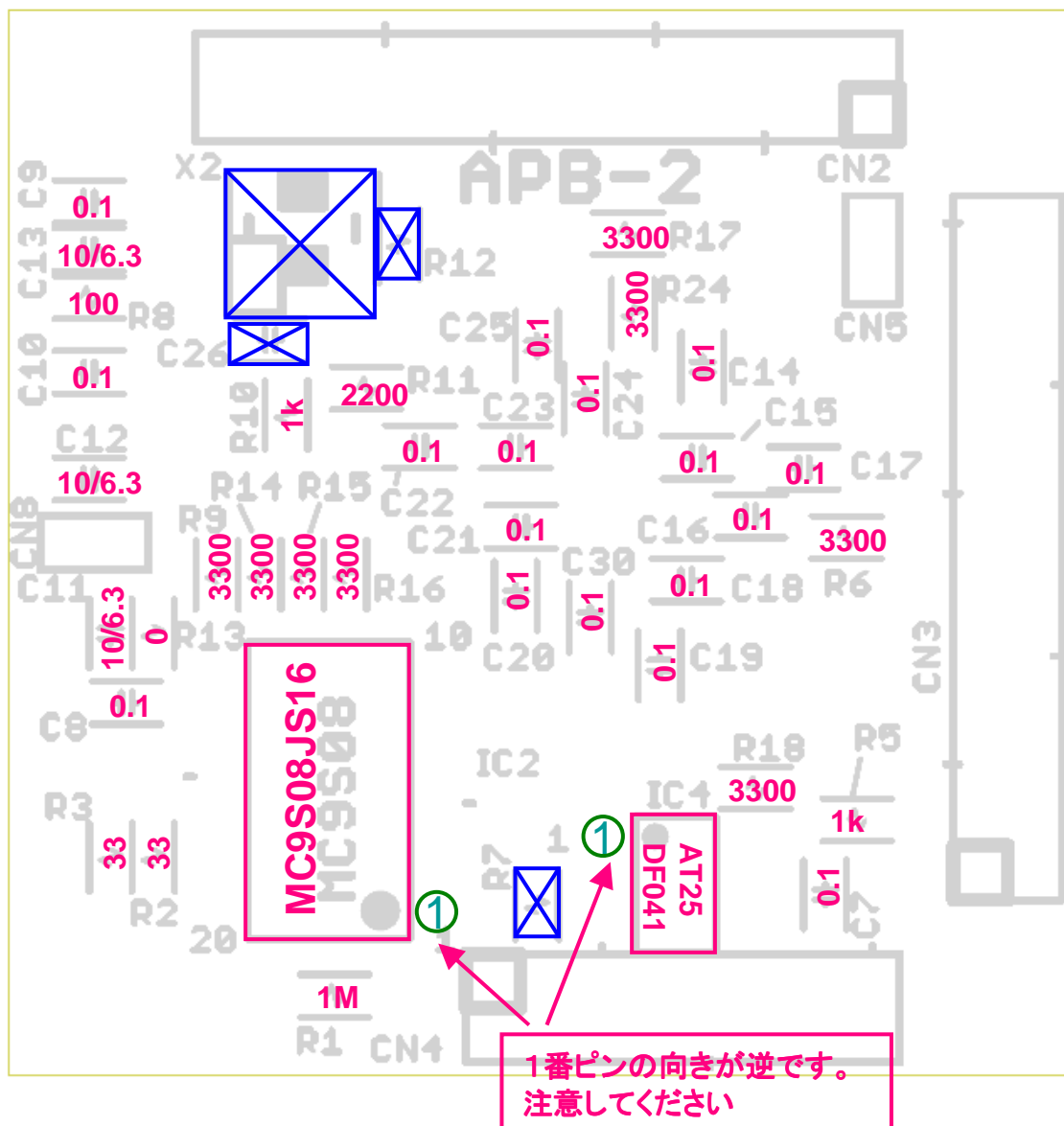
■ 部品表

Device	Description	Qty	Parts Reference
C	27p	2	C1, C2
C	0.1	19	C3, C5, C7, C8, C9, C10, C14, C15, C16, C17, C18, C19, C20, C21, C22, C23, C24, C25, C30
C	10/6.3 or 10/10	5	C4, C6, C11, C12, C13
C	xxx		C26
CN	2pin	2	CN5, CN8
CN	16pin		CN4
CN	26pin		CN2, CN3
CN	USB	1	CN1
D	LED	2	D3, D5
D	MA153	1	D2
IC	REG 1117 1.2V	1	IC6
IC	REG 1117 3.3V	1	IC5
IC	AT25DF021 or 041	1	IC4
IC	MC9S08JS16	1	IC2
IC	TC7S32	1	IC3
IC	XC3S200A_100	1	IC1
R	0	1	R13
R	33	2	R2, R3
R	100	1	R8
R	1k	2	R5, R10
R	2200	2	R11, R26
R	3300	8	R6, R9, R14, R15, R16, R17, R18, R24
R	1M	1	R1
R	xxx		R7, R12
X	12MHz	1	X1
X	xxx		X2
その他	ショートピン	1	
その他	基板	1	
その他	平田など	1	

マウント図(A面)



マウント図(B面)



1. チップ部品の半田付け

■ チップ部品を順番にマウントします。終わった部品にはチェックマークをいれていきます。

チップキャパシタ(コンデンサ)

- 27p
- 0.1
- 10/6.3 or 10/10

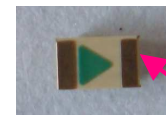
抵抗

- 0
- 33
- 100
- 1k
- 2200
- 3300
- 1M(水晶の取り付け穴をふさがないように仮付け程度にしておきます)

ダイオード

- MA153
- LED (A面にあります)

ダイオードには極性があります。
下記カソードマークを確認のうえ取り付けてください。



カソード側



これは裏面側です。
取り付けるときはひっくり返します

2. 電源IC

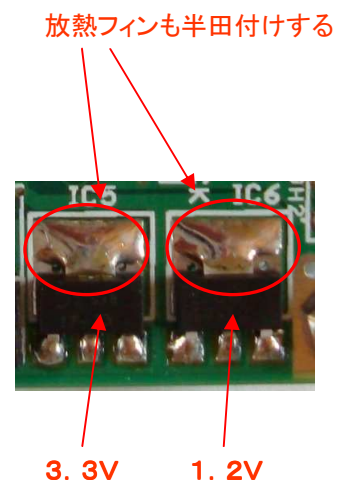
■ 電源 IC をマウントします。

IC

- REG 1117 3.3V (放熱フィンも半田付けします)
- REG 1117 1.2V (放熱フィンも半田付けします)

■ USB ケーブルを接続し、レギュレーターの電圧をチェックします。

- IC5 の放熱フィンの電圧が $3.3 \pm 0.1V$ であること
- IC6 の放熱フィンの電圧が $1.2 \pm 0.05V$ であること



正常動作しないときは、まず半田付けを確認してください。

イモ半田がないか、隣とブリッジしていないか、ルーペを使って一個一個確認していきます。ちょっとでもあやしいとおもったら再度半田づけします。

以降のチェックで動作しなかった場合も同様にまず半田付けを疑ってください。

3. MC9S08JS16

■ MC9S08JS16 と周辺部品をマウントします。

IC

- MC9S08JS16
- AT25DF041

IC をマウントするときはまず端の1本だけ半田付けし、浮きがないことを確認の上、対角のピン、もう一度浮きがないことを確認してから残りのピンを半田付けします。

水晶

- 12MHz
(足を伸ばしてマウントします。黒のスペーサーは間に挟みます)



スペーサーをはさむ

コネクタ

- CN8 2ピン オス(2ピン分切り取って使います)
(CN5 はFPGAの半田付けの邪魔になりますのでまだマウントしません)

CN8 をショートして電源を入れると MC9S08JS16 がブートモードになります。

4. MC9S08JS16 のチェック

- フリースケールのサイトから BOOTLOADER_GUI をダウンロードし、インストールします。

http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=S08JS&fosp=1&tab=Design_Tools_Tab

のページの下のほうにある、

Snippets, Boot Code, Headers, Monitors, etc. (1)

をクリックすると BOOTLOADER_GUI ができますのでダウンロードをクリックします。
すると Login 画面がでてきますのでメンバー登録するとダウンロードを開始します。

ダウンロードしたファイルを解凍し、中の Setup.exe を実行してインストールします。

- CN8 にショートピンを挿します。
MC9S08JS16 がブートモードになります。

- USB ケーブルで PC と接続します。
ドライバーのインストールが始まります。

- インストールしたMC9S08Bootloader を実行します(右図)。
画面のレイアウトが崩れますが何とか使えます。

- MC9S08Bootloader 画面の Product ID の部分に「 JS16 」と表示され、右下の USB アイコンが緑色になっていればOKです。



5. MC9S08JS16 にプログラム書き込み

□ 左側のファイル選択ボタンを押して MS9S08JS16 に書き込むファイル(CDC_monitor.abs.s19) を選択します。

- ① Mass Erase を押します。 →全消去
- ② Program を押します。 →プログラム書き込み
- ③ Reset を押します。 →書き込んだプログラムの実行

□ 上記それぞれのステップで右側の Status Window にエラーメッセージがでないことを確認します。

□ Reset を押したとき Product ID の「 JS16 」表示が消え、USBアイコンが赤に戻ることを確認します。

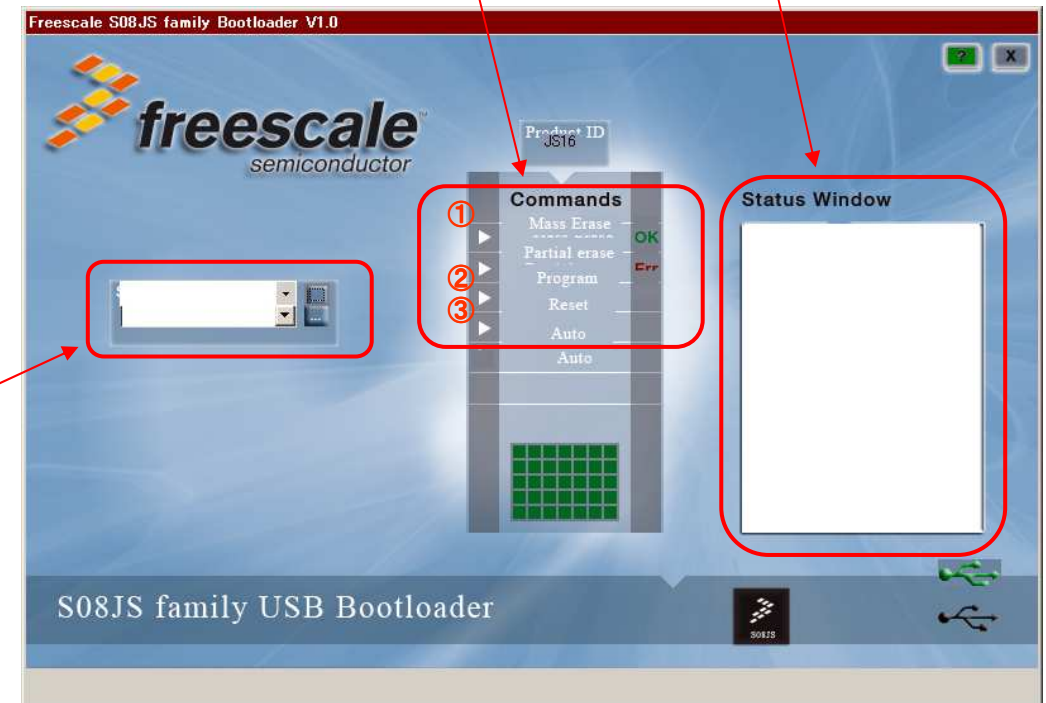
□ 基板上の LED(D3) が点灯していることを確認します。

□ USB ケーブルを抜きます。

□ CN8 のショートピンをはずします。
これで通常実行モードになり、電源オンすると書き込んだプログラムを実行します。

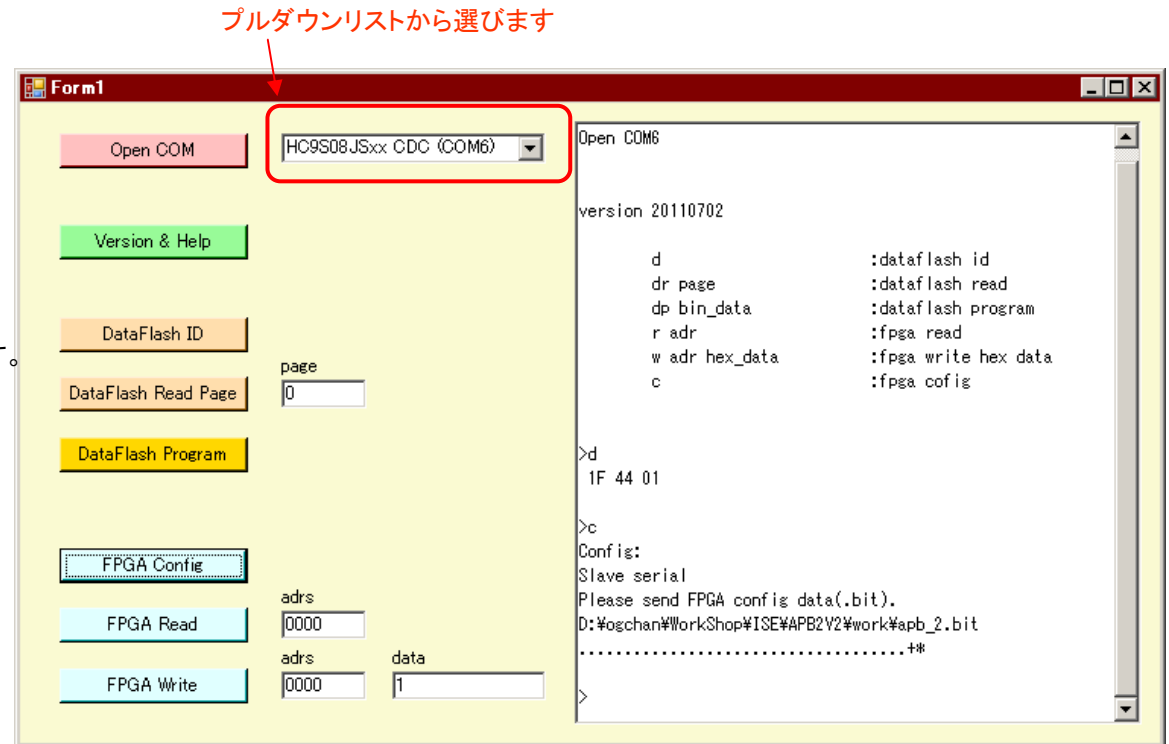
書き込む
ファイル選択

エラーメッセージがでないこと



6. MC9S08JS16 の通信チェック

- USB ケーブルで PC と接続します。
- APB-2 テストプログラムを起動します。
- シリアルポートのリストから HC9S08JSxxCDC を選びます。
- Open COM ボタンを押します。
正常にオープンできると Open COMx というメッセージがでます。
- Version & Help ボタンを押すとバージョン表示がでます。
またボタンを押すたびに基板上的 LED(D3) が点灯・消灯することを確認します。
- DataFlash ID ボタンを押すとデータフラッシュの ID(1F 44 01) が表示されることを確認します。
- DataFlash Read Page ボタンを押すとデータフラッシュの 1 ページ分のデータが表示されます。
まだ何も書き込まれていないのですべて FF になっています。



シリアル通信でコマンドを送り、返ってきたデータを表示させているだけなので TeraTerm などのターミナルソフトでも同じことができます。

使えるコマンド一覧は p.21 をごらんください。

7. FPGA(XC3S200A)

■ FPGA と周辺部品をマウントします。

IC

XC3S200A

1番ピンの方向に注意してください。

コネクタ

CN5 (2ピン分切り取って使います)

IC をマウントするときにはまず端の1本だけ半田付けし、浮きがないことを確認の上、対角のピン、もう一度浮きがないことを確認してから残りのピンを半田付けします。

ブリッジしても気にせずすべてのピンを半田付けしてしまいます。

その後、7ページ目に書いてある方法でブリッジを直します。

電源を入れる前にループで十分チェックを行なってください。

CN5 は、

オープン： Slave Serialモード

ショート： Master SPI モード

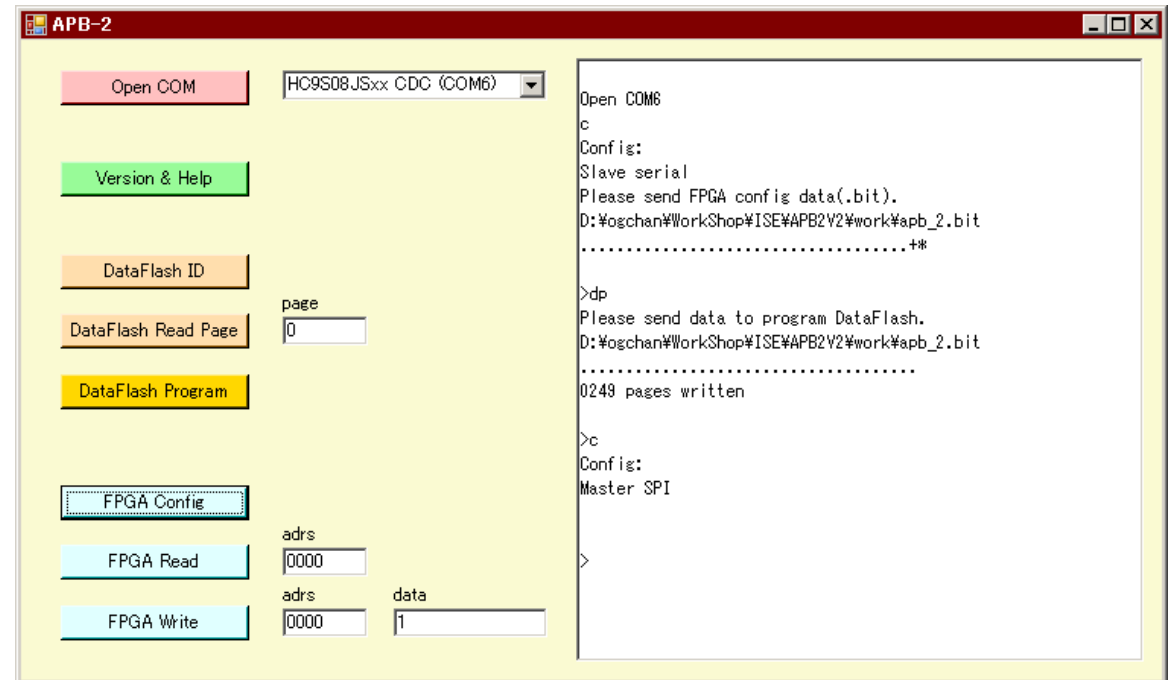
となっています。

Slave Serial モードでは、MC9S08JS16 からFPGAにコンフィグデータを転送してコンフィグします。

Master SPI モードでは FPGA がデータフラッシュ (At25DF041) に書き込まれたコンフィグデータを読み込んでコンフィグします。
あらかじめコンフィグデータを書き込んでおけば、電源オンすると同時にコンフィグしますのでスタンドアローン動作にできます。

8. FPGA(XC3S200A) のチェック

- USB で PC と接続、APB-2 テストプログラムを起動し、COM ポートをオープンします。
- FPGA Configボタンを押すと Slave Serial モードでコンフィグを開始します。
- ファイル選択ダイアログでコンフィグデータ(apb_2.bit) を選択します。
- コンフィグが終了すると LED(D5) が点滅することを確認してください。
- DataFlash Programボタンを押し、コンフィグデータ(apb_2.bit) を選択してデータフラッシュに書き込みます。
- DataFlash Read Pageボタンを押すとデータフラッシュの 0 ページを読み出して表示します。すべて 00 とか FF になっていないことを確認します。
- CN5 にショートピンを挿します (Master SPIモードにします)。
- FPGA Configボタンを押すとコンフィグを開始します。データフラッシュからのコンフィグは一瞬で終了します。
- 同じく LED(D5) が点滅することを確認してください。



ファイルを選択し転送するのは30秒以内に行なってください。

MC9S08JS16 はデータが送られてこないと30秒でタイムアウトします。タイムアウトするとプロンプトをだしてコマンド待ちに戻ります。

9. MC9S08JS16 プログラム開発

- MC9S08JS16 のプログラム開発は Freescale から提供されている CodeWarrior を使いC言語で行ないます。

http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=S08JS&fsrch=1&sr=2

このページの下の方にスクロールし、Featured Software and Tools の下にある See more をクリックします。

CWX-MCU-SE: Special Edition: CodeWarrior for Microcontrollers (Eclipse, Windows hosted)

をダウンロードします。

Evaluation ではなく Special Edition のほうです。注意してください。
Linux 用もあります。

USB には、CMX_USB_LITE_MC9S08JS16: CMX USB LITE FOR MC9S08JS16. を使っていますが、サンプルコードの中に私が改良したバージョンが入っていますので Freescale からあえてダウンロードする必要はありません。

プログラムを組むときは、FPGA(XC3S200A) や データフラッシュ(AT25DF041) とのインターフェースをオープンドレインで行なうことに注意してください。これは電源電圧が違うためで、普通のポート出力だと高電圧をかけてしまい最悪破壊につながります。
定義されているマクロを使うか、すでにある入出力ルーチンを使うと安全です。

すでにインプリされているコマンド(次ページ参照)を使えばたいのことはできますのでスタンドアローン動作をさせたいときを除けば MC9S08JS16 のプログラムを組む必要性は低いです。

10. MC9S08JS16 のコマンド

■ サンプルプログラムでインプリされているコマンド一覧。

d	データフラッシュのID読み出し
dr page	データフラッシュの指定したページから1ページ分（256バイト）読み出し
dp bin_data	データフラッシュにFPGAコンフィグデータ書き込み
r adr	FPGA 内部のデータ（16バイト）読み出し
w adr hex_data	FPGA 内部にデータ書き込み
c	FPGAコンフィグ <ul style="list-style-type: none"> - Slave Serial モードのときはコンフィグデータ（.bit ファイル）をターミナルソフトから転送します。 - Master SPI モードのときはデータフラッシュからコンフィグを実行します。 - CN5 がオープンの場合はSlave Serialモード、ショートすると Master SPI モードになります。
0x01 adrs(2) size(2) bindata	FPGA バイナリーデータ書き込み コマンド1バイトのあとにアドレス2バイト、サイズ2バイト、書き込みデータを送ります。
0x02 adrs(2) size(2)	FPGA バイナリーデータ読み出し コマンド1バイトのあとにアドレス2バイト、サイズ2バイトを送るとデータ読み出しします。

11. XC3S200A プログラム開発

- XC3S200A のプログラム開発は Xilinx から提供されている ISE Webpack を使います。

<http://japan.xilinx.com/support/download/index.htm>

このページの、

Full Installer for Windows(TAR/GZ - 3.88 GB)

をダウンロードし、インストールしてください。最新バージョンは 13.1 です。
Linux 用もあります。

プログラムを組むときは、VCCAUX が 3.3V になっていることに注意してください。
UCF ファイルに
CONFIG VCCAUX=3.3;
と記述します。

サンプルプログラムに入っている UCF ファイルを参考にしてください。

12. PC側 プログラム開発

- PC側はシリアルポートを使った通信ソフトになりますので言語は問いません。私は Visual C# Express を使っています。

Visual C# Express は、

<http://www.microsoft.com/japan/msdn/vstudio/express/>

からダウンロードできるようです。

サンプルプログラム(この説明書で使っているテストプログラム)を改造すれば簡単にアプリが作れると思います。

サンプルプログラムを使うには「プロジェクト」メニューから「参照の追加」を選び System.Management を追加してください。