

MAMBA™ ファミリー
アプリケーション開発キット

サポート・デバイス
HI-6135, HI-6136
HI-6137, HI-6138

2016 年 6 月



株式会社ナセル

履歴

履歴	日付	変更内容
AN-6138, Rev. New	2015 / 7 / 20	初版リリース
Rev.A	2015 / 8 / 18	DIP スイッチのデフォルト変更
	2016 / 1 / 8	Cortex™ M3 マザーボード用の回路図と BOM を追加 MAMBA 回路図と BOM の更新

本文書は、Holt 社の『AN-6138』の和訳になります。

分かりにくい表現や誤訳がある場合は、Holt 社発行の英語版文書を参照してください。

Introduction [はじめに]

Holt MAMBA™ 評価ボードは、MIL-STD-1553 プロトコル・デバイスの MAMBA™ ファミリーで使用する場合の幅広い機能セットを示しています。

HI-6135 : RT

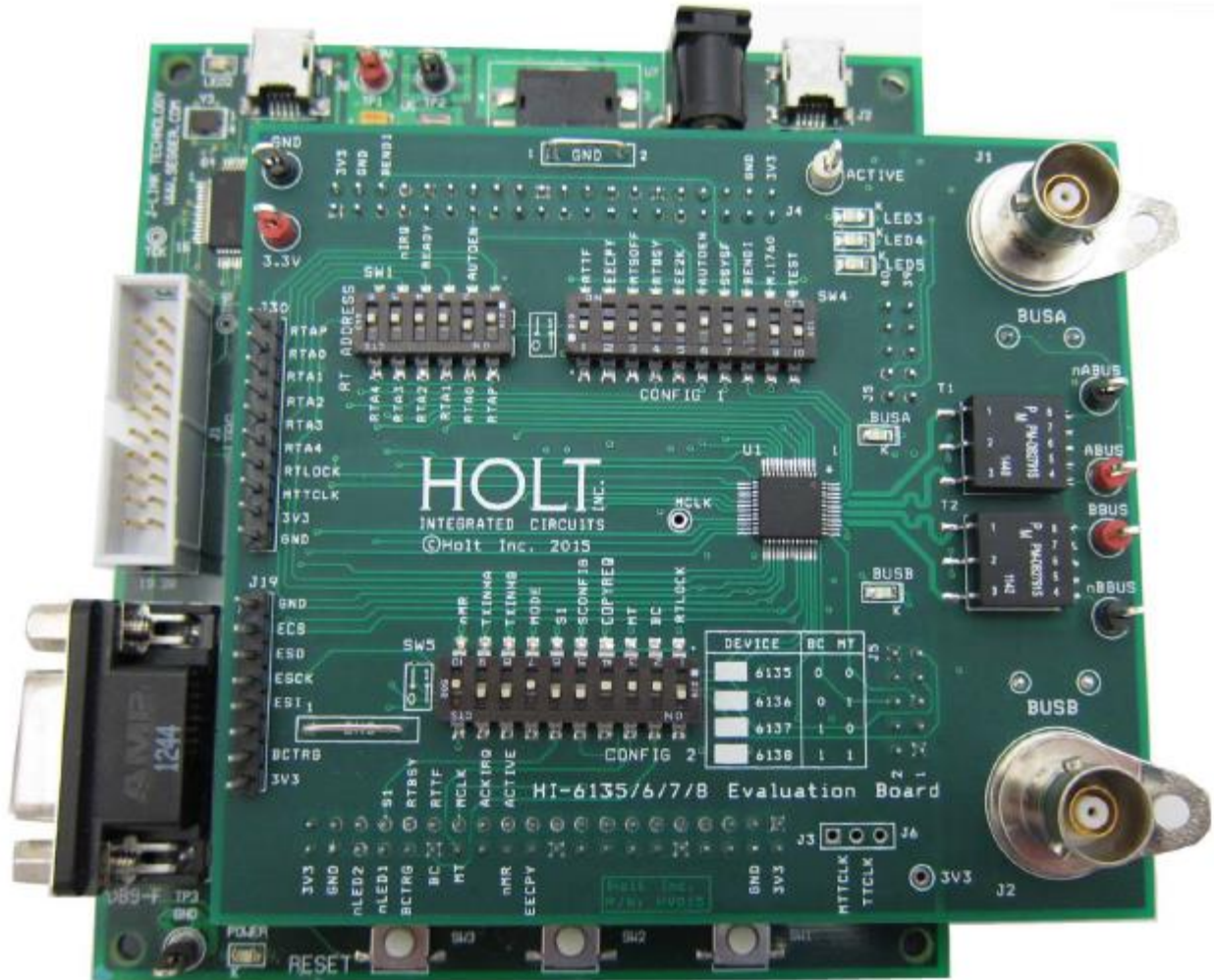
HI-6136 : RT および／または BM

HI-6137 : RT および／または BC

HI-6138 : RT、BC、および／または BM

MIL-STD-1553B バス通信デバイスの MAMBA™ ファミリーには極めて効率の良いプロトコル・ロジックと物理バス・インターフェイス回路が含まれています。2 ボードから構成されるキットと C プロジェクトのリファレンス・デザインは、すぐに実行できる BC、MT、RT の任意の組合せの同時動作を評価可能なプラットフォームを提供します。便宜上、このキットには、IAR システムズの ARM 用 Embedded Workbench®、そして ARM Cortex M3 マイクロコントローラ用の完全に統合されたデバッグ・インターフェイスが含まれています。この MAMBA™ ガイドのリファレンス・デバイスは HI-6138 です。この IC には利用可能なすべての機能が含まれているためです；他の MAMBA™ デバイスには HI-6138 機能のサブセットが含まれています。

このガイドでは、ボードの設定および、実行する方法について説明します。資料と必要なすべてのプロジェクト・ソフトウェアは、Holt の CD-ROM に含まれます。デモ・ソフトウェア・バージョンがすでにマイコンのフラッシュ・メモリにプログラムされています；提供されるソフトウェア開発ツールをインストールまたは実行する必要無く、ボードは箱から出してすぐに動作可能です。

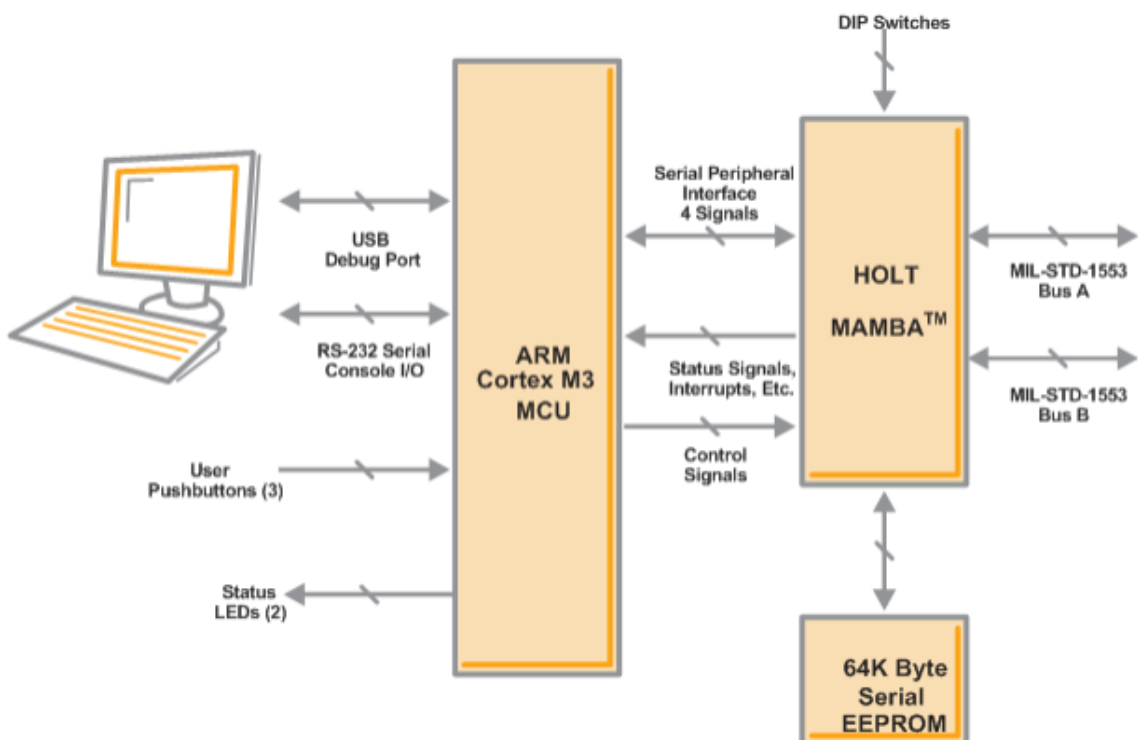


MAMBA™ 評価ボード、ARM Cortex M3 MCU ボードに搭載

Evaluation Kit Contents [開発キット構成]

- 本ユーザー・ガイド
- Holt MAMBA™ ソフトウェア・プロジェクトおよび、ドキュメント CD
- IAR システムズの ARM 用 Embedded Workbench® Ver.7.1 または、それ以上
- 5V DC 電源アダプタ
- デバッグ用 USB ケーブル
- RS-232 シリアル・ケーブル、DB9（オス）－DB9（メス）タイプ、PC 接続コンソール I/O 用
- 2 ボード構成で、
 - 上段 MAMBA™ デバイス、デュアル・トランスフォーマ・カップリング MIL-STD-1553 バス・インターフェイス。DIP スイッチにより、ボードの動作設定を行います。
 - 下段 MCU ボード：ARM Cortex M3 16/32Bit マイクロプロセッサ、デバッグ・インターフェイスおよび、3.3VDC レギュレート電源供給

Hardware Block Diagram [ハードウェア・ブロック図]



Default Switch Settings [デフォルト・スイッチ設定]

RT アドレス

スイッチ	ポジション	説明
SW1、6-2	00011 (OFF = 1)	RTA4:0 … RT アドレス設定。デフォルト設定 RT3
SW1、1	OFF = 1	RTAP … RT アドレス・パリティ・ビット。正しい奇数パリティを反映していなければ、RT 動作を開始することができません。

コンフィグ1

スイッチ	ポジション	説明
SW4、1	OFF	RTTF - ON … RT ステータス・ワードのターミナル・フラグ・ビットを設定
SW4、2	OFF	EECPY - ON … RAM とレジスタのコピーをEEPROMに作成
SW4、3	OFF	MTSTOFF - ON … 電源投入時のメモリテスト無効
SW4、4	OFF	RTBSY - ON … MCU が Busy Bit を RT ステータス・ワードにセット
SW4、5	OFF	EE2K - ON … EECPY と AUTOEN は、わずか 2K ワードのEEPROM を使用します。
SW4、6	OFF	AUTOEN - ON … 外部EEPROMからのレジスタ/RAM初期化を有効にします。
SW4、7	OFF	SSYSF - ON … RT ステータス・ワードにサブシステム・フェイル・ビットを設定します。
SW4、8	ON	BENDI - ON … ビッグ・エンディアンのメモリ&レジスタ・アクセスを設定します。
SW4、9	OFF	MODE1760 - ON … RT ステータス・ワードのビジー・ビットを、ハードウェア・マスタ・リセットの立ち上がりエッジですぐにセットします。
SW4、10	OFF	TEST - ON … セルフテストを可能にします。デモには追加のソフトウェアが必要です。

コンフィグ 2

スイッチ	ポジション	説明
SW5、1	ON	RTLOCK - OFF : RT アドレスをロックする、上書きされません。
SW5、2	OFF (6138, 6137)	BC - OFF : ソフトウェアによる BC モード有効
	ON (6136, 6135)	
SW5、3	OFF (6138, 6137)	MT - OFF : ソフトウェアによる SMT モード有効
	ON (6136, 6135)	
SW5、4	ON	COPYREQ - OFF : ソフトウェア起動後に EEPROM にコンフィグレーションを書き込みます
SW5、5	ON	SCONFIG : 未使用
SW5、6	ON	S1 : 未使用
SW5、7	OFF	MODE : デバイスはモード・ピンを High にする必要があります
SW5、8	ON	TXINHB - OFF : 1553 BUSB ドライバを無効
SW5、9	ON	TXINHA - OFF : 1553 BUSA ドライバを無効
SW5、10	OFF	nMR - ON : デバイスをリセット状態に保持

Default Jumper Settings [デフォルト・ジャンパ設定]

ジャンパ	位置	説明
JP2	OPEN	BUSA のマイナス・ラインをボードのグラウンドに接続するためのリンク
JP3	OPEN	BUSB のマイナス・ラインをボードのグラウンドに接続するためのリンク
JP8	OPEN	BUSA に 70Ω の不可抵抗を接続するためのリンク
JP9	OPEN	BUSB に 70Ω の不可抵抗を接続するためのリンク
JP10	CLOSED	50MHz クロックを下段 MCU ボードにリンク

Hardware Design Overview [ハードウェア設計概要]

上段のターゲット・ボードと下段の MCU ボードの部品表と回路図は、本ガイドの末尾を参照してください。

取り外し可能なターゲット・ボードは、ユーザー提供の代替マイクロプロセッサまたは FPGA ボードに接続するために、付属の MCU ボードから分離できます。ボード間のヘッダーは、0.1 インチ (2.54mm) ピッチの汎用コネクタで接続されています。すべてのホスト・インターフェイス信号は、ボード間のヘッダーを介して行われます。多数の MAMBA™ コンフィグレーション・ピン (リモート・ターミナル・アドレス設定端子など) は、上段のターゲット・ボードの DIP スイッチによって設定されます; これらの信号は、ボード間のヘッダー上の MCU では使用できません。

下段の ARM Cortex M3 ボードは、フラッシュ・プログラム可能な、Atmel AT91SAM3U-EK マイクロプロセッサを使用しています。GPIO ピンとして設定された NPCS0 を使用する Atmel プロセッサの 4 線シリアル・ペリフェラル・インターフェイス (SPI マスタ 24MHz) は、MAMBA™ (スレーブ) に接続されます。UART ベースのシリアル・ポートは、RS-232 コンソール I/O (オプション) を提供します。USB2.0 ポートは、将来の拡張用に使用可能です。2 つのプッシュ・ボタンは、ソフトウェアとの対話のために利用可能です。REST プッシュ・ボタンは、ターゲット・マスター・リセット信号を制御し、ARM マイクロプロセッサをリセットします。

ARM Cortex M3 ボードは、www.segger.com からライセンスされた「オンボード J-Link」デバッグ・インターフェイスが含まれています。高価な JTAG デバッグ・ケーブルを購入することなく箱から出してすぐに使用することができます。キットには、ご使用のコンピュータに、ボードのデバッグ・インターフェイスを接続するためのシンプルな USB ケーブルが含まれています (ユーザーがすでに ARM デバッグ・インターフェイス用のリボン・ケーブル・コネクタを所有している場合、ARM 標準の 2×10 のデバッグ・コネクタでデバッグ接続が可能です。この場合、下段ボードの底面のジャンパ JP2 をショートさせ、「オンボード J-Link」を無効にする必要があります。)

Initial Kit Set Up [キットの初期セット・アップ]

Holt の MAMBA™ アプリケーション開発キットは、MAMBA™ ファミリーの 4 つのデバイスすべてを示します。MAMBA™ は、3 つのターミナル・モード全てで動作します: RT、BC、MT。

1. ご使用の PC にはシリアル (COM) ポートと、TeraTerm のような「ターミナル・エミュレーション」プログラムが必要です。ほとんどのコンピュータには RS-232 COM ポートが無い場合、ADK 付属のシリアル/USB アダプタが必要になります。これをコンピュータの USB ポートに接続し、9 ピン・コネクタを ADK ボードに接続します。
2. Windows 2000 または、WindowsXP を使用している場合、ハイパーターミナルをターミナル・エミュレーションに使用できます。「スタート」→「すべてのプログラム」の順にクリックして、「Windows アクセサリ」→「通信プログラム」の順にクリックして、ハイパーターミナルを開きます。ハイパーターミナルをクリックして実行します。次の段落をスキップしてください。

Windows7 または、Vista を使用している場合…

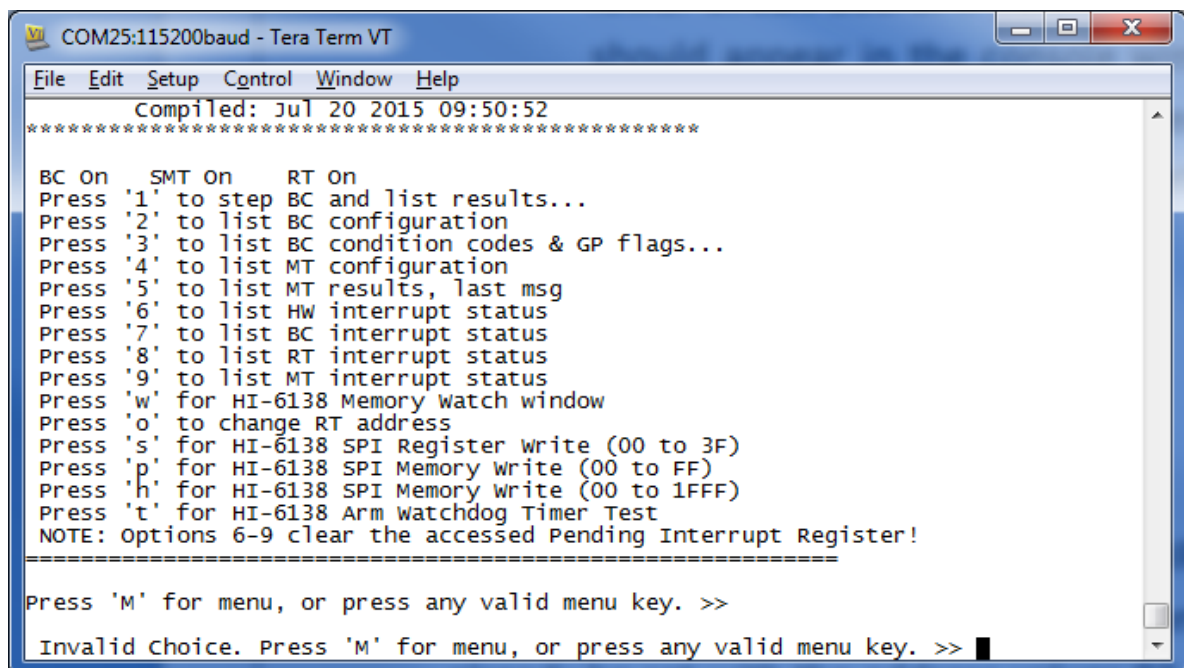
ハイパーターミナルは、これらのバージョンの Windows には含まれていません。無料のオープンソー

ス・ターミナル・エミュレーション・プログラム TeraTerm4.71 を、Holt 提供の CD 内にある、「teraterm-4.71.exe」インストール・プログラムを実行して、インストールしてください。再配布は、著作権表示が保持されることを条件に許可された旨のライセンス契約を受け入れます。通知は、「Help」→「About Tera Term」をクリックすることで Tera Term ウィンドウから表示することができます。インストールを継続します・・・

- デフォルトのインストール先をそのまま使用し、「Next」をクリックします。
- コンポーネント選択画面で、追加のプラグイン=TTXResizeMenu を除いたすべてのオプションを選択解除し、「Next」をクリックします。
- インストール言語を選択し、「Next」をクリックします。
- デフォルト・スタート・メニュー・フォルダを許可し、「Next」をクリックします。
- ショートカット作成を選択し、「Next」をクリックします。
- インストール画面で「Install」をクリックします。

Tera Term プログラムを実行します。「新しい接続」画面で (x) シリアルを選択し、COM ポートを選択します。シリアル・ポート設定ウィンドウを開くために、「設定」→「シリアル・ポート」をクリックします。設定を変更します、ボーレート：115200、データ：8 bit、パリティ：none、ストップ：1 bit、フロー制御：none。付属の DB9 シリアル・ケーブルを使用して、MCU ボードとコンピュータのシリアル (COM) ポートを接続します。

3. 付属の 5VDC 電源を差し込み、ケーブルを下段のボードの電源入力ジャックに接続します。Tera Term が正しく動作し、正しく設定されている場合、コンソール・ウィンドウに以下のコマンド・メニューが表示されます。このメニューは、ボードの電源が投入されたとき、または RESET プッシュ・ボタンが押された後に表示されます。評価ボードとの正しい Tera Term 通信を確認した後、セット・アップをクリックして保存することにより、ターミナルセットアップを保存できます。



```

COM25:115200baud - Tera Term VT
File Edit Setup Control Window Help
Compiled: Jul 20 2015 09:50:52
*****
BC On   SMT On   RT On
Press '1' to step BC and list results...
Press '2' to list BC configuration
Press '3' to list BC condition codes & GP flags...
Press '4' to list MT configuration
Press '5' to list MT results, last msg
Press '6' to list HW interrupt status
Press '7' to list BC interrupt status
Press '8' to list RT interrupt status
Press '9' to list MT interrupt status
Press 'w' for HI-6138 Memory watch window
Press 'o' to change RT address
Press 's' for HI-6138 SPI Register write (00 to 3F)
Press 'p' for HI-6138 SPI Memory write (00 to FF)
Press 'h' for HI-6138 SPI Memory write (00 to 1FFF)
Press 't' for HI-6138 Arm watchdog Timer Test
NOTE: options 6-9 clear the accessed Pending Interrupt Register!
=====
Press 'M' for menu, or press any valid menu key. >>
Invalid choice. Press 'M' for menu, or press any valid menu key. >>

```

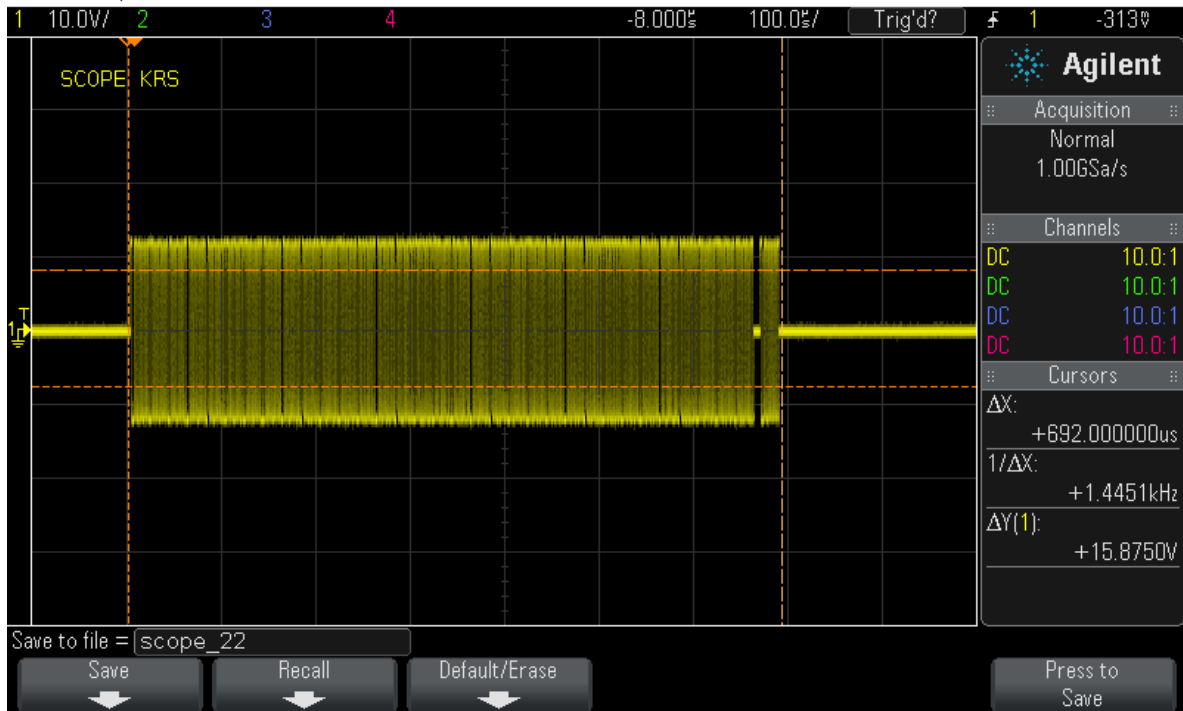
Testing Modes [テスト・モード]

RT ターミナル・アドレスは、電源投入前に DIP スイッチを使用して設定します。RT アドレス 3 および、4 は、あらかじめプログラムされたバスコントローラ・メッセージ・レパトリーによって利用されます。6 桁の DIP スイッチは、アドレス値 03 と奇数パリティで既に設定されているはずですが。

BC and RT Mode (HI-6138 and HI-6137) [BC および、RT モード (HI-6138 および、HI-6137)]

1. バスの動作を観察するには、オシロスコープを使用し赤い BUS A テスト・ポイントと赤い BUS B テスト・ポイントに接続します。ACTIVE と表示されたテスト・ポイントは、便利なスコープ・トリガです。
2. MIL-STD-1553 バスにケーブルで接続されていない場合は、半田ジャンパ JP2 と JP3 を接続することによって、バスにダミーの 70Ω 負荷がボード上に提供されます。
3. BC は、RT アドレス 3 と 4 への繰り返しの一連の MIL-STD-1553 コマンドを実行するようにプログラムされています。各バス・コマンドの前に BC 「Wait for Trigger」OP コードがあります。MCU は、コンピュータのキーボードで数字の「1」キーが押されるたびに、トリガ・パルスで BC に発行するようにあらかじめプログラムされています。613x_initialization.h コンソール I/O オプションを OFF にしてプログラムを再コンパイルすると、MCU ボードの SW1 ボタンを使用して BC のコマンド・ワードをトリガすることができますが、このモードでは Tera Term コンソール出力は無効になります。
4. コンピュータのキーボードで数字の「1」が押されるたびに、RT アドレス 3 または、4（または RT-RT メッセージの場合は RT アドレス 3 と 4 両方に）に新しい BC コマンドが発行されます。ブロードキャスト・コマンドも送信されます（これらはすべての RT アドレスを指定し、RT 応答はありません）
5. RT テストの場合、ボードは RT 有効で出荷され、アドレス 03 を設定するとデバイスはコマンドに回答します。RT の回答が観測され、TeraTerm コンソール画面では、キーを押すたびに新しいメッセージが表示されます。回答の例は、メッセージ#1 の下に表示されています。オシロスコープの画面は、BC が 32 ワードのデータを RT アドレス 03、サブアドレス 30 に送信することを示し、短い遅延の後に RT がステータス・ワードで回答しています。2 つ目の画像は、同じコマンドの後のターミナル画面を示しています。

MSO-X 3054A, MY50510474: Thu Jul 09 21:07:51 2015



```

COM24:115200baud - Tera Term VT
File Edit Setup Control Window Help
Message # 1
Message Type: Rx Subaddress Command, 32 data words
CW: 0x1BC0 = 03-0-30-00      Sw: 0x1800 = RT03 CS
BC Control word: 0x40A0
MEMask UseBusA maskBCR NonBcstSA
Block Status word: 0x8000
EOM Bus A
Condition Code Register: 0x8000
BC Running: No Condition Codes or Gen Purpose Flags Are Set.
Data Addr: 0x8000,
0x0101 0x0202 0x0303 0x0404 0x0505 0x0606 0x0707 0x0808
0x0909 0x1010 0x1111 0x1212 0x1313 0x1414 0x1515 0x1616
0x1717 0x1818 0x1919 0x2020 0x2121 0x2222 0x2323 0x2424
0x2525 0x2626 0x2727 0x2828 0x2929 0x3030 0x3131 0x3232
-----
Press 'M' for menu, or press any valid menu key. >>

```

RT Mode (HI-6135, 6136) [RT モード (HI-6135、HI-6136)]

1. これらのデバイスには、RT モードを示すためにメッセージを送信する BC モードは含まれていません。外部 BC コマンドを提供する必要があります。BC コマンドを生成するには別のボードが必要です。例えば、この目的で Holt 6130/31 ボードを使用できます。
2. 外部ボードを ADK ボードの BUSA または BUSB 端子に接続します。RT アドレス・スイッチ (SW1) が正しいアドレスに設定され、パリティが奇数であることを確認してください。応答が必要な RT にコマンドを送信し (Mode2 コマンド、「Transmit Status Word」)、BUS 上の RT 応答を確認します。

1760 Mode (all devices) [1760 モード (全デバイス)]

このモードでは、デバイスは 2msec 以内にセットされたビジー・ビットで応答し、リセットが解除されます。この機能をテストするには、ソフトウェアを実行せずにデバイスの電源を入れる必要があります。これは、マイコンをリセットしたままにすることで可能です。nMR スイッチが ADK (SW5/1) でトグルされると、デバイスは BC コマンドに応答し、「Busy」ビットがセットされます。

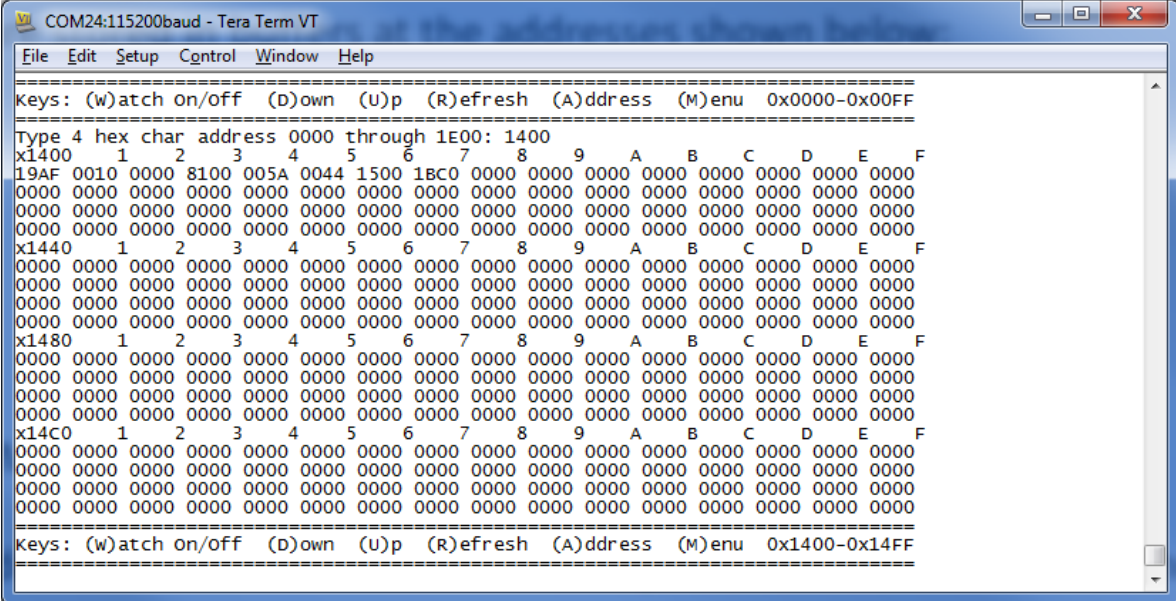
MT Mode (HI-6136, 6138) [MT モード (HI-6136、HI6138)]

このモードでは、デバイスは、バス上に送信されたすべてのコマンドとデータをモニタして保存します。コマンドは以下に示すアドレスのバッファに格納されます。

コマンド・ワード	0x1400 ~ 0x14FF
データ・ワード	0x1500 ~ 0x16FF

上記のアドレスの内容は、「w」コマンド・ユーティリティを使用して表示し、次に「a」を押してアドレスを入力し、0x1400 と入力します。格納されている 8 つのコマンド・ワードが一番上の行に表示されます。

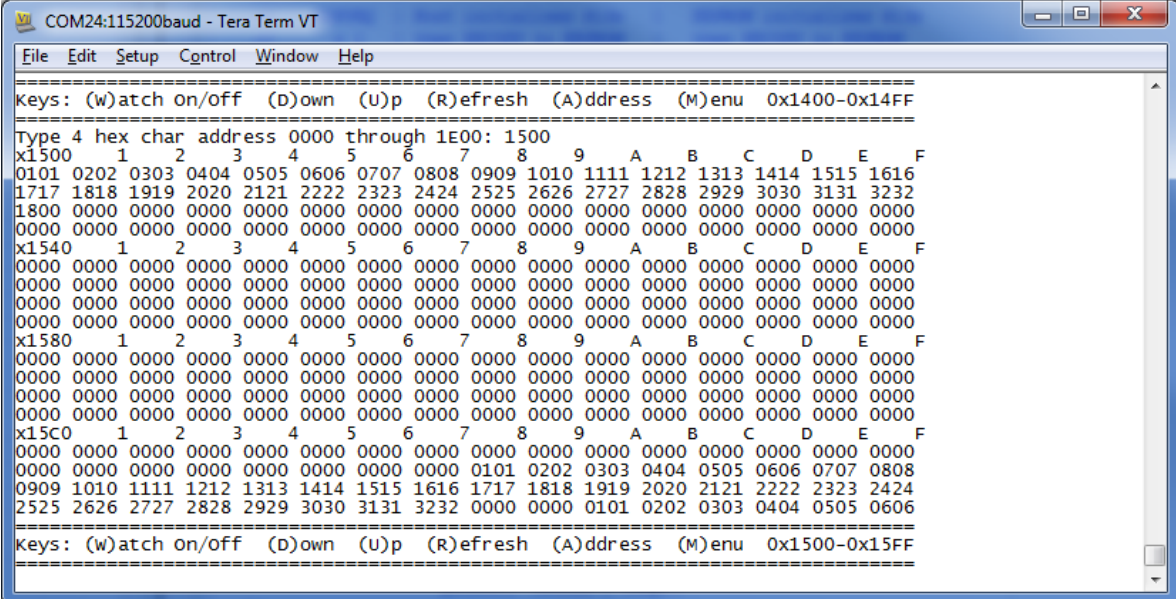
メッセージ#1 が送信された後に格納された 8 つのコマンド・ワードの例を、48Bit タイムタグの SMT について以下に示します：



The screenshot shows a terminal window titled "COM24:115200baud - Tera Term VT". The window contains a memory dump of 48-bit time tags. The data is organized into four sections, each starting with a hex address and a column of indices (1-9, A-F). The first section starts at 0000, the second at x1440, the third at x1480, and the fourth at x14c0. Each section contains 16 rows of data. The first row of each section shows non-zero values, while the subsequent rows are mostly zeros. The terminal also displays a keymap at the top and bottom of the dump area.

```
COM24:115200baud - Tera Term VT
File Edit Setup Control Window Help
=====
Keys: (w)atch on/off (D)own (U)p (R)efresh (A)ddress (M)enu 0x0000-0x00FF
=====
Type 4 hex char address 0000 through 1E00: 1400
x1400 1 2 3 4 5 6 7 8 9 A B C D E F
19AF 0010 0000 8100 005A 0044 1500 1BC0 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x1440 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x1480 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x14c0 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
=====
Keys: (w)atch on/off (D)own (U)p (R)efresh (A)ddress (M)enu 0x1400-0x14FF
=====
```

同じメッセージ#1 が送信された後、0x1500 番地に格納された 32 個のサブアドレス・データ・ワードは以下のとおりです：



```
COM24:115200baud - Tera Term VT
File Edit Setup Control Window Help
=====
Keys: (w)atch on/off (D)own (U)p (R)efresh (A)ddress (M)enu 0x1400-0x14FF
=====
Type 4 hex char address 0000 through 1E00: 1500
x1500 1 2 3 4 5 6 7 8 9 A B C D E F
0101 0202 0303 0404 0505 0606 0707 0808 0909 1010 1111 1212 1313 1414 1515 1616
1717 1818 1919 2020 2121 2222 2323 2424 2525 2626 2727 2828 2929 3030 3131 3232
1800 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x1540 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x1580 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x15C0 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0101 0202 0303 0404 0505 0606 0707 0808
0909 1010 1111 1212 1313 1414 1515 1616 1717 1818 1919 2020 2121 2222 2323 2424
2525 2626 2727 2828 2929 3030 3131 3232 0000 0000 0101 0202 0303 0404 0505 0606
=====
Keys: (w)atch on/off (D)own (U)p (R)efresh (A)ddress (M)enu 0x1500-0x15FF
=====
```

Using the C demo code [C デモ・コードの使用]

次の手順では、IAR C コンパイラをインストールおよび設定し、MAMBA™アプリケーション開発キットを使用してデモンストレーション・プロジェクトをロードおよび、変更する方法を説明します。

1. インストールのデフォルトを使用して、Windows コンピュータに ARM 用 IAR Embedded Workbench® (EWARM) をインストールします。EWARM は、プロジェクト・マネージャ、エディタ、コンパイラ、アセンブラ、リンカ、ライブラリ、およびデバッグツールを含む、完全に機能する統合開発環境です。これには最適化 C コンパイラが含まれており、幅広い ARM デバイスとハードウェア・デバッグ・システムをサポートしています。既成のデバイス・コンフィグレーション・ファイル、フラッシュ・ローダー、サンプル・プロジェクト・ファイルが含まれています。キットのインストール CD は IAR Embedded Workbench® の KickStart 版です。Embedded Workbench® のライセンス・ファイルを入手するには、IAR Web サイトで登録する必要があります。ライセンスは永続的です。唯一の制限事項はプログラム・サイズです。コンパイルされたプログラムが 32Kbyte を超える場合、コンソール I/O オプション（プロジェクト・ファイルの 613x_initialization.h）を無効にしてみてください。これにより、MIL-STD-1553 の機能を損なうことなく、コンパイルされたプログラム・サイズが大幅に削減されます。それ以外の場合は、ARM 用 Embedded Workbench® の無制限バージョン用の 30 日間評価ライセンスが IAR の Web サイト www.iar.com から入手できます。インストールの詳細については、IAR のリリース・ノート「Important Information」を参照してください。
2. デバッグを使用するには、IAR Embedded Workbench® を実行しているコンピュータと MAMBA アプリケーション開発キットとの間のインターフェイスが必要です。付属の USB ケーブルの小さい方の端を DEBUG とマークされた MAMBA™ 評価ボードの USB コネクタに接続します。USB ケーブルのもう一方の端をコンピュータの USB ポートに接続します。ARM 用の IAR C-SPY デバッグには、「J-link On Board」を組み込んだ、多数のターゲット・システム・インターフェイス用のドライバが含まれています。

評価ボードの USB ケーブルが最初に接続されたとき、Windows は J-Link デバイスのために「新しいハードウェアが検出されました」のメッセージを表示します。数秒後、Windows が適切なドライバをロードし、「使用できる準備ができました」のメッセージが表示されます。Windows が J-Link ドライバを見つけることができなかった場合、IAR Embedded Workbench® インストール CD の Driver ディレクトリからインストールします。

手順 5 でデバッグ・セッションを開始するときに困難が生じた場合、Project→Options をクリックします。ウィンドウが表示され、Category = Debugger で J-Link/J-Trace をハイライトにします。Connection タブを選択し、Communications = USB and Interface = SWD をクリックします。

3. 付属の IAR CD-ROM から IAR Embedded Workbench® をインストールします。ドキュメント「Holt Demo Project Installation for IAR Systems」を参照してください。インストールが完了し、少なくとも 1 つの Atmel サンプル・プロジェクトがインストールされたら、Holt Zip プロジェクトを PC のどこかにコピーし、2 つのプロジェクトを解凍します。

4. 付属の Holt CD には ANSI C で書かれた MIL-STD-1553 プロジェクトのサンプルが含まれています。プロジェクト・フォルダ全体（またはサブディレクトリ）は、コンピュータの HDD ドライブの特定の場所にコピーする必要があります。_Holt プロジェクト・フォルダ全体をこの場所に移動します：
C:\My Document\IAR embedded workbench\Atmel\atmel\at91sam3u - ek
5. IAR Embedded Workbench®を開きます。「File」→「Open Workspace」をクリックし、手順 4 で作成したプロジェクト・サブディレクトリに移動します。拡張子が.eww のプロジェクト・ファイルを選択し、「Open」をクリックします。（次回 Embedded Workbench®を開くと、このプロジェクトが Recent Workspaces リストに表示されます）
6. IAR インストールまたは IAR デバッガを使用して問題が発生した場合は、Holt CD-ROM に含まれているこれらの問題の解決に役に立つ 2 つの Holt テクニカル・ノートが提供されています。
7. MAMBA™ プロジェクトは符号なし整数のみを使用します。必要に応じて、変数の最上位ビットがトグルするときに発生するコンパイラ・メッセージをオフにします。メッセージは次のようになります：

Remark[Pe068]: integer conversion resulted in a change of sign

To disable this diagnostic message, click Project then click Options

Category = C/C++ Compiler

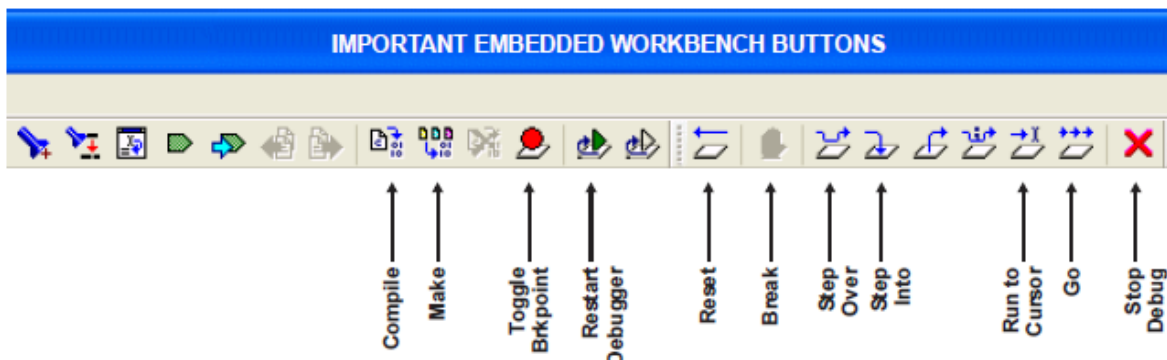
Tab = Diagnostics

Suppress these diagnostics: add "Pe068" to list

8. Holt IAR プロジェクトには、RT、BC、MT の 6 つの定義済み組み合わせが含まれています。これらはワークスペースのプルダウン・メニューから選択できます。選択すると、アプリケーションにその特定の組み合わせをコンパイルするように指示されます。使用可能な組み合わせは、そのモードが特定のデバイスで使用可能かどうかによって異なります。これらの組み合わせは、コンパイラのプリプロセッサ・ラベル BC_ena、RT_ena、SMT_ena を変更します。
9. デフォルトの組み合わせ BC_MT_RT は、HI-613X デバイスの BC、MT、RT のプライマリ・モードを有効にします。これらの組み合わせはすべてフラッシュ・ベースのプロジェクトです。RAM ベースのプロジェクトは、MCU の RAM 容量が限られているため、サポートされていません。設計上、ARM Cortex-M3 は Flash よりも RAM の方が遅く実行されるため、RAM ベースのプロジェクトはほとんど必要ありません。6 つの構成と対応するプリプロセッサのラベル値は次の表に示されています：

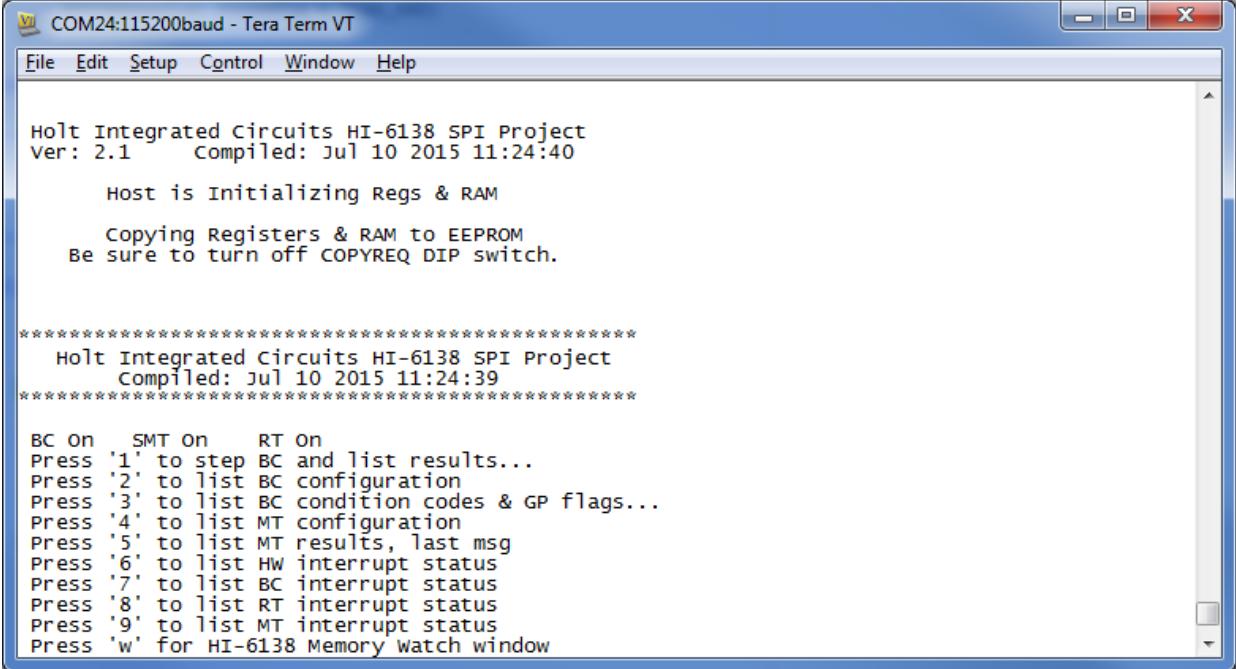
構成	BC_ena	RT_ena	SMT_ena	互換性
BC_MT_RT (デフォルト)	1	1	1	HI-6138
BC_ONLY	1	0	0	HI-6138、HI-6137
BC_RT	1	1	0	HI-6138、HI-6137
SMT_ONLY	0	0	1	HI-6138、HI-6136
RT_ONLY	0	1	0	All devices
RT_SMT	0	1	1	HI-6138、HI-6136

10. 他の構成も可能です。新しい構成を作成する簡単な方法は、「Project/Edit Configurations」を選択し、「New」を選択することです。このダイアログ・ボックスでは、新しい構成名を使用して既存の構成に基づいて新しい構成を行うことができます。新しい設定を選択し、必要に応じてプリプロセッサのラベルを編集し、新しい設定を保存します。新しい設定がプルダウン・メニューに表示されます。プロジェクト・ファイル **613x_initialization.h** は、タイムタグの分解能、コンソール I/O のオン・オフなどその他の重要なプロジェクト設定を構成します。
11. 「**Make**」ボタンをクリックしてプロジェクトをコンパイルします。次の図を参照してください。IAR Embedded Workbench®のビルド・メッセージ・ウィンドウにエラーまたは警告が表示されない場合は、続行できます。エラーが発生した場合は、修正してプログラムを再コンパイルしてください。
12. デバッガの「**Restart**」ボタンをクリックして、デバッグ・セッションを開始します。これにより、コンパイルされたプログラムが MCU にダウンロードされ、プログラム実行のためにボードが準備されます。「**Go**」をクリックして実行を開始します。実行を停止するには「**Break**」（通常、実行中は赤い手で表示される）をクリックします。
13. バスの動作を観察するには、オシロスコープを赤い BUSA テスト・ポイントと赤い BUSB テスト・ポイントに接続します。ACTIVE というラベルのテスト・ポイントは、便利なスコープのトリガ信号です。MIL-STD-1553 バスにケーブルで接続されていない場合、黒と赤のテスト・ポイント間に、70Ω 1W 抵抗を接続することにより、バス A とバス B にダミー負荷を提供します。



Auto - initialization from the EEPROM [EEPROM からの自動初期化]

1. 設定をシリアル EEPROM に保存するには :
プログラムの実行を開始する前に、AUTOEN と表示されている DIP スイッチを OFF にして、EEPROM からの自己初期化をする代わりに、MAMBA™ デバイスを初期化するように MCU に指示します。COPYREQ と表示されている DIP スイッチを「1」に設定すると、デバイスのレジスタのリセット後の初期化と RAM の完了後に、MCU の EEPROM コピー・シーケンスを開始するように指示します。以下のような画面メッセージが表示されます :



```
COM24:115200baud - Tera Term VT
File Edit Setup Control Window Help

Holt Integrated Circuits HI-6138 SPI Project
Ver: 2.1    Compiled: Jul 10 2015 11:24:40

Host is Initializing Regs & RAM

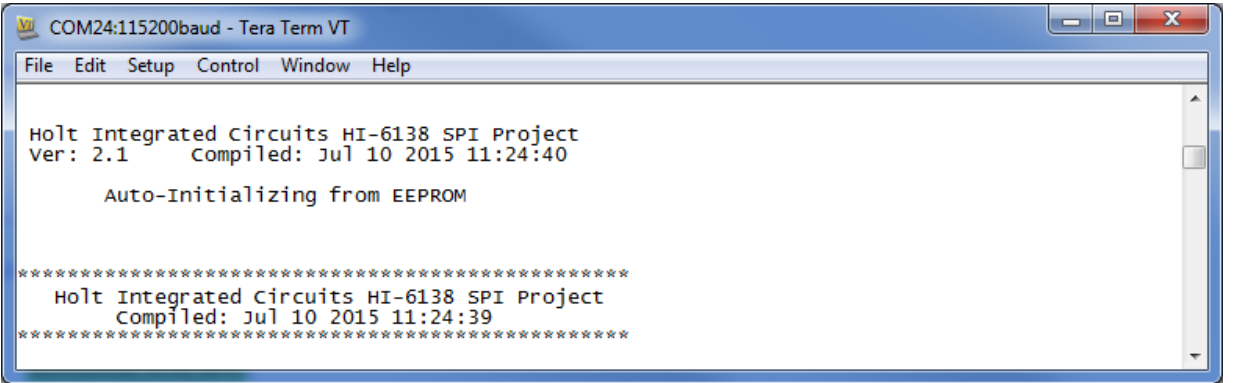
Copying Registers & RAM to EEPROM
Be sure to turn off COPYREQ DIP switch.

*****
Holt Integrated Circuits HI-6138 SPI Project
Compiled: Jul 10 2015 11:24:39
*****

BC On  SMT On  RT On
Press '1' to step BC and list results...
Press '2' to list BC configuration
Press '3' to list BC condition codes & GP flags...
Press '4' to list MT configuration
Press '5' to list MT results, last msg
Press '6' to list HW interrupt status
Press '7' to list BC interrupt status
Press '8' to list RT interrupt status
Press '9' to list MT interrupt status
Press 'w' for HI-6138 Memory watch window
```

赤色 LED は、EEPROM コピー処理中に点灯します。橙色の LED が点灯している間は、ロード・エラーが発生しています。LED が消灯したら、COPYREQ と表示されている DIP スイッチを「0」に設定する必要があります。

2. 電源を入れる前に、AUTOEN スイッチを「1」に設定し、電源を切ってから再び ON にするか、マスター・リセットを行うと、デバイスは EEPROM から自己初期化されます。次のようなメッセージが画面に表示されます。



```
COM24:115200baud - Tera Term VT
File Edit Setup Control Window Help

Holt Integrated Circuits HI-6138 SPI Project
Ver: 2.1    Compiled: Jul 10 2015 11:24:40

Auto-Initializing from EEPROM

*****
Holt Integrated Circuits HI-6138 SPI Project
Compiled: Jul 10 2015 11:24:39
*****
```

Project File List with Selected Descriptions [プロジェクト・ファイル選択項説明]

HEADER FILES WITHOUT CORRESPONDING C FILES

device_6138.h

HI-6135/6/7/8 (SPI) プロジェクトのみに使用
HI-6135/6/7/8 レジスタ・アドレスのマクロ定義

613x_initialization.h

重要な構成設定の定義

613x_regs.h

レジスタ・ビットとビット・フィールドのマクロ

C FILES WITH CORRESPONDING HEADER FILES

関数名のほとんどは自明のもので、一部の関数は元も 6131 の名前を保持していますが、MAMBA™ ファミリーでうまく動作するかどうか心配しないでください。

main.c

main();
プライマリ・プログラム・エントリ・ポータル **main()**は、EEPROM からの自己初期化を有効にしているかどうかにかかわらず、有効にされたターミナルの組み合わせについて、使用される初期化シーケンスを示しています。初期化が完了した後、関数呼び出しは、有効なターミナル・モードで使用されるすべての RAM 構造に対して強力なアドレッシング方法を提供します。

board_613x.c

board_613x.h ... ARM MCU I/O 定義を含む
ConfigureGpio(); ... ARM MCU 汎用 I/O 初期化
reset_613x();
autoinit_check();
initialize_613x_shared();
init_timer();
Delay_us(num_us);
Delay_ms(num_ms);
Delay_x100ms(num);
Flash_Red_LED();
Flash_Green_LED();
error_trap(count);
enable_check();
write_init_eeprom();

board_6138.c

board_6138.h … ARM MCU SPI I/O 定義と SPI コマンドのマクロ定義を含む

```
SPlopcodes(opcode) ;
Write_6131LowReg(reg_number, data, irq_mgmt) ;
Read_6131LowReg(reg_number, irq_mgmt) ;
Write_6131_1word(data, irq_mgmt) ;
Read_6131_1word(irq_mgmt) ;
Write_6131(write_data[], inc_pointer_first, irq_mgmt) ;
Read_6131(number_of_words, irq_mgmt) ;
Write_6131_Buffer(write_data[], inc_pointer_first, irq_mgmt) ;
Read_6131_Buffer(number_of_words, inc_pointer_first, irq_mgmt) ;
Read_Current_Control_Word(rt_num, irq_mgmt) ;
getMAPaddr() ;
enaMAP(map_num) ;
Read_Current_Control_Word(rt_num, irq_mgmt);
Read_RT1_Control_Word(txrx, samc, number, irq_mgmt);
Read_RT2_Control_Word(txrx, samc, number, irq_mgmt);
ReadWord_Adv4(irq_mgmt) ;
Read_Last_Interrupt(irq_mgmt) ;
Fill_6131RAM_Offset() ;
Fill_6131RAM(addr, num_words, fill_value) ;
Memory_watch(address);
Configure_ARM_MCU_SPI();
```

613x_BC.c

613x_BC.h … HI-6137/8 で使用される命令リストのマクロ

```
BC_bus_addressing_examples(); (HI-6130 専用)
initialize_bc_msg_blocks();
initialize_bc_instruction_list();
initialize_613x_BC();
bc_disable();
bc_enable();
bc_start();
bc_trigger();
bc_switch_tests();
```

デモの場合、この関数はプッシュ・ボタン SW1 をポーリング氏、次の BC メッセージをトリガします。

```
SW1_BC_Trigger();
SW2_BCtest ();
initialize_613x_BC();
```

613x_MT.c

613x_MT.h … バス・アドレス構造体を含む (HI-6130 専用)

```
initialize_613x_MT();
```

この関数は、シンプル SMT モニタ動作

613x_RT.c

613x_RT.h … HI-6135/6/7/8 で使用されるデスクリプタ・テーブル・アドレス・マクロ

```
initialize_613x_RT1();
```

```
RTAddr_okay(RTnum);
```

```
modify_RT_status_bits();
```

```
RTstatusUpdate();
```

この機能は DIP スイッチ設定に基づいて、RT ステータス・ビットを更新します

```
write_dummy_tx_data_RT1();
```

最後の機能は、デモ用の送信データ・バッファを初期化します

console.c

全てのターミナル・モードで使用されるコンソール機能 :

```
ConfigureUsart1();
```

```
text_header();
```

```
chk_key_input();
```

```
list_hw_ints_console();
```

BC モードで使用されるコンソール機能 :

```
bc_last_msg_console();
```

```
list_bc_config ();
```

```
list_bc_ccgpf_reg();
```

```
list_bc_ints_console();
```

RT1 および/または RT2 で使用されるコンソール機能 :

```
list_rt_ints_console();
```

SMT または IMT バスモニタで使用されるコンソール機能 :

```
list_mt_config();
```

```
mt_last_msg_console();
```

```
list_mt_ints_console();
```

冗長な文字列を「printf」してプログラム・サイズを縮小するプリミティブ・コンソール機能 :

```
print_null();
```

```
print_sp1sp();
```

```
print_b1sp();
```

```
print_b0sp();
```

```
print_dddn();
```

```
print_dd0n();
```

```
print_dd1n();
```

```
print_menuprompt();
```

```
print_line();
```

Memory_watch()関数によって呼び出されるコンソール関数

マニュアル SPI 書込みユーティリティ

```
ascii2int();
```

Application Development Kit Notes [アプリケーション開発キット注意事項]

MAMBA™は、SPI を搭載したマイクロコントローラとの互換性を考慮して設計されています。RAM とレジスタの位置は、8Bit SPI コマンドの助けを借りてリード/ライトされます。大部分のリード/ライト操作は、4つのメモリ・アドレス・ポインタ (MAP) のうちの1つを使用して、次にアクセスされる場所のアドレスを指定します。マルチワード転送を高速化するために、有効にされたメモリ・アドレス・ポインタは、各リード/ライトが実行されると自動的に次のアドレスにインクリメントされます。レジスタ・アドレス 0~15 十進は、メモリ・アドレス・ポインタを使用せずに直接読み取ることができます。レジスタ・アドレス 0~63 十進は、メモリ・アドレス・ポインタを使用せずに直接書き込むことができます。

デバッグ時に、このツールは MAMBA™ SPI ツールでは機能しないため、IAR Embedded Workbench®デバッガから見るができないので、レジスタまたは、RAM の値を監視するのに役立ちます。デモ・プログラムは、Memory_watch() という C 関数を使用して SPI を介して同様の機能を提供します。この関数呼び出しは、コンソール I/O が有効な場合にのみ機能します。提供されたメモリ・アドレス・パラメータで始まる 256 の連続するレジスタまたは RAM 値を表示します。メモリ・アドレス空間 0~0x1FFF 全体は、256 ワード単位でアクセス可能です。デモ・プログラムはキーボード入力をポーリングし、実行する必要があります。コンソール・メニューの「W」コマンドが入力されると、0x0000~0x00FF メモリ・アドレス空間が表示されます：

```

COM1:115200baud - Tera Term VT
File Edit Setup Control Window Help
0x0000 1 2 3 4 5 6 7 8 9 A B C D E F 0x00FF
x0000 19F2 A100 0000 0000 0000 0000 0000 0000 0180 0000 0000 0000 7818
FFF8 01F8 D400 0058 FBF8 01F8 D400 01CE 1C00 0400 0000 0000 AAAA BBBB 0000 ABCD
01CE 2000 0600 0000 0000 AAAA BBBB 0000 ABCD 0046 0000 0000 0000 0000 0000
0000 0000 D02D 1B70 1B70 0000 0000 8000 00C0 007F FE58 000E 0000 0000 0000
x0040 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 FF18 000E 0000 0000 0000 0000 FF45 0000 FF54 0000 9316 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 5EAE 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
x0080 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
5400 5400 5FFF 5DFF 6000 6000 7FFF 7DFF 0000 0000 0000 0000 0000 0000 0000
x00C0 1 2 3 4 5 6 7 8 9 A B C D E F
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
=====
Keys: <W>atch On/Off <D>own <U>p <R>efresh <A>ddress <M>enu 0x0000-0x00FF
=====

```

↑
アドレス範囲

画面下部のサブメニューには、使用可能なメモリ・ウォッチ・オプションがリストされています。「D」(DOWN コマンド) を押すと、表示されているアドレス範囲が 0x0100~0x01FF に変わります。上記の画面から「U」(UP コマンド) を押すと、デバイスのアドレス空間がラップアラウンドされ、表示されているアドレス範囲が、0x1F00~0x1FFF に変更されます。UP または DOWN コマンドを繰り返すと、アドレス範囲内を移動します。「R」を押すと現在選択されているアドレス範囲がリフレッシュされ、「A」(ADDRESS コマンド)

を押すと 4 つの 16 進文字を入力して、メモリ・ウォッチの開始アドレスを選択できます。「W」(WATCH)、「M」(MENU) を押すと、メモリ・ウォッチ表示がトグルし、5 ページのメニューに戻ります。

`Memory_watch()` が実行されると、表示されている各位置が再スキャンされることに注意してください。いくつかのレジスタ RAM 構造体ビットは、各読出しが発生した後に自動的にリセットされます。これには、保留割り込みレジスタのビットと RAM の RT デスクリプタ・テーブル・コントロール・ワードの DBAC データ・ブロック・アドレス・ビットが含まれます。これらのために、メモリ・ウォッチ・ウィンドウは、関数が実行されたときに有効な値を反映します。

Tera Term を使用したコンソール I/O オプションには、ペンディング割り込みレジスタのステータスを読み込んで表示するいくつかのメニュー・オプションがあります。ペンディング割り込みビットは、読出しが発生した後に自動的にリセットされることに注意してください。これらのレジスタの場合、メモリ・ウォッチ・ウィンドウは、実行が停止したときに有効な値が反映されます。

MAMBA™ デモ・プログラムは、MAMBA™ ファミリーのすべてのデバイスを対象としています。デフォルト設定では BC、RT、SMT すべてが有効になっています。これらのターミナルの機能のいずれかを有効または無効にするには、2 つの手順があります。ワークスペースの最上行のソフトウェア構成が、ハードウェア構成の DIP スイッチ (BCENE および、MTRUN) と一致する必要があります。

MAMBA™ SPI Interface [MAMBA™ SPI インターフェイス]

MAMBA™ はホスト MCU または FPGA への 4 線シリアル・ペリフェラル・インターフェイス (SPI) を備えています。このデバイスは、プラスチック QFP または、6mm×6mm QFN パッケージで提供されます。

MAMBA™ データ転送速度は、MCU SPI インターフェイスによって提供される SPI クロック周波数に依存します。SPI が最大 SCK 周波数 40MHz (このデモでは 24MHz を使用) でクロックされると、各 16Bit ワードは 400ns で転送され、SPI オペコードの実行に関するオーバー・ヘッドが加算されます。メモリ・アドレス・ポインタ (レジスタ) は、読出しまたは書込み操作が開始される前に、MCU または FPGA によって初期化されます。その後、8Bit の SPI オペコードを使用してリード/ライト動作が開始され、MCU または FPGA によって MAMBA™ にシリアル・シフトされます。その後ホストは、連続する RAM またはレジスタ・アドレスを読み書きするために、16 クロックの倍数で SCK クロックを継続します。クロックが継続されている限り、連続したアドレスが読み書きされます。マルチワード・アクセス中に割り込みがイネーブルされていると、潜在的な問題が発生します。プログラムの割り込みハンドラが SPI を補足して割り込みを処理すると、未完了のマルチワード転送が中断される可能性があります。適切なソフトウェア設計が無ければ、ハードウェアは割り込みが発生したことを知らず、メモリ・アドレス・ポインタは中断されたマルチワード転送の次の位置の RAM またはレジスタ・アドレスを含んでも含まなくてもよいので、割り込みからの簡単な復帰により、マルチワード転送が壊れます。

MAMBA™ SPI 転送中は、割り込みを無効にする必要があります。最も単純な実装では、オペコードを送信する前に割り込みを無効にし、マルチワード転送で最後のワードをリード/ライト後に割り込みを再び有効にします。これらが受け入れがたい割り込み遅延を引き起こす場合は、慎重なソフトウェア設計が必要です。適切な注意を払うことで、割り込みを一時的に再び有効にしてから、SPI ワード間ですぐに無効にすることができます。割り込みが再有効されると、割り込みインターバル中に発生した保留中の割り込みがすぐに認識されます。保留中の割り込みサービス・ルーチンが実行されます；割り込みからの復帰は次の「割り込み禁止」ステートメントにジャンプして実行します。

MAMBA™のソフトウェア例では、割り込みが一時的に無効にされてから SPI ワード間で再度有効になるたびにテストされた「SPI 割り込み発生」フラグを使用して、中断されたマルチワード・シーケンスを正常に完了します。ワード間の割り込み処理を検出すると、プログラムは次のワードのメモリ・アドレス・ポインタを再初期化し、次の元のオペコードを再発行して中断したマルチワード転送を再開します。3つ以上のレベルの割り込みをネストするのは難しいです。

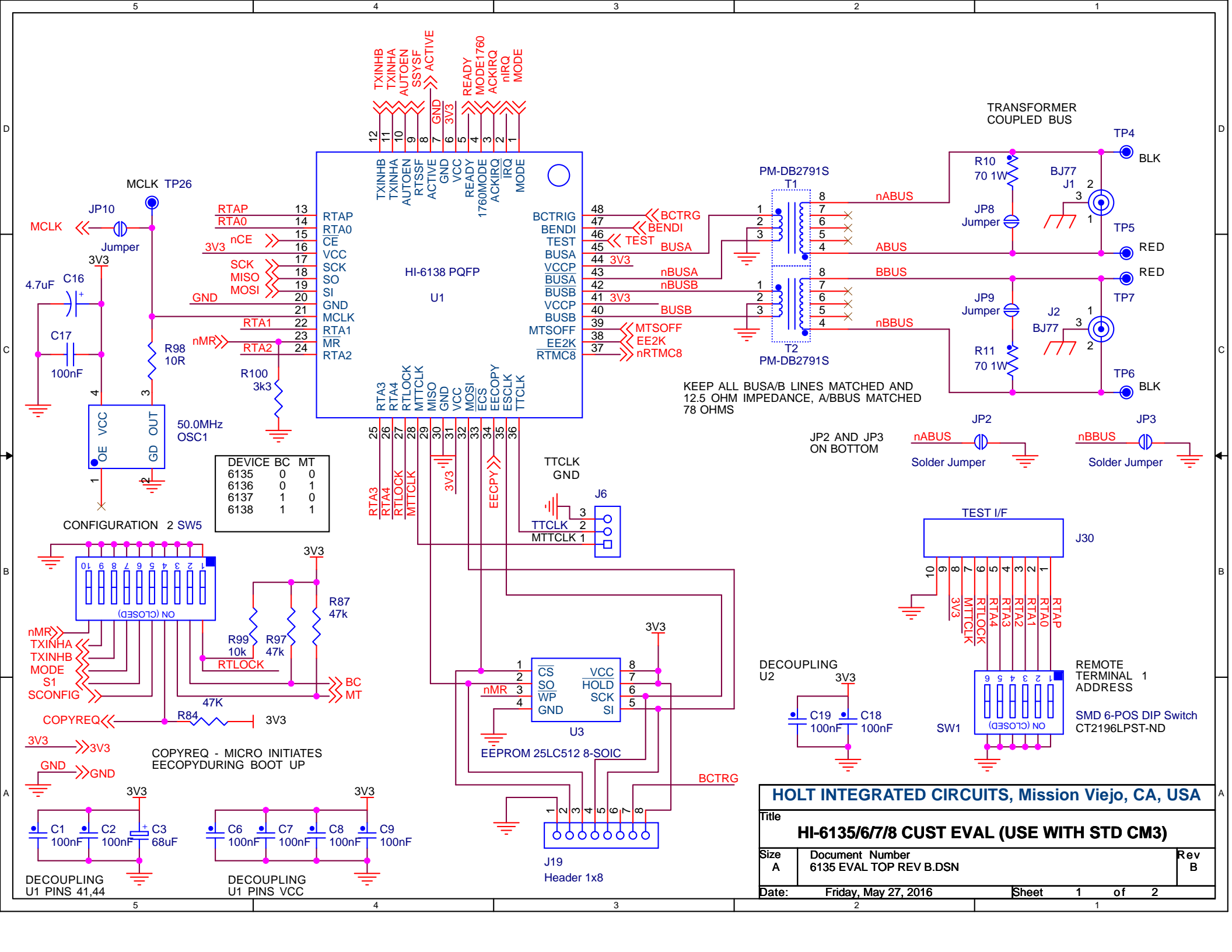
MAMBA™を使用する場合、SPI アクセスはテーブル・アドレスの C 構造体と互換性がありません。MAMBA™を使用しているときは、レジスタ/RAM 検査、ウォッチ・ウィンドウは使用できません。ユーティリティ機能の代わりに、HI-6135/6/7/8 デモ・プログラムの *Memory_watch()*関数のように、アドレス範囲を読み取るためには C で書かれていなければならない、コンソール I/O またはその他の表示手段を用いて表示データをフォーマットします。もちろん *Memory_watch()*が必要な場合は、アプリケーションプログラムが表示機能呼び出すために実行されている必要があります。上級ユーザー向けには、レジスタやメモリ・アドレスへの SPI 書込み用に他のユーティリティも用意されています。

まとめ

MAMBA™ SPI インターフェイスは、RAM またはレジスタにアクセスするためのわずか 4 つのホスト・インターフェイス信号を使用して、ハードウェア設計を簡素化しボード・スペースを節約します。

Bill of Materials
Mamba Evaluation board
Rev. B

Item	Qty	Description	Reference	DigiKey	Mfr P/N
1	1	PCB, Bare, Eval Board	N/A	-----	
2	10	Capacitor, Cer 0.1uF 20% 50V Z5U 0805	C1,C2,C6,C7,C8,C9,C17,C18,C19,C20	399-1176-1-ND	Kemet C0805C104M5UACTU
3	1	Capacitor, Cer 4.7uF 10% 6.3V X5R 0805	C16	399-3134-1-ND	Kemet C0805C475K9PACTU
4	2	Capacitor 68uF 10% 6.3V Tant 400 mOhm SMD EIA 6032-28	C3,C26	495-1507-1-ND	T495C686K006ZTE400
5	2	Connector 3-Lug Concentric Triax Bayonet Jack, Panel Front Mount TRB (BJ77)	J1,J2	MilesTek 10-06570	Trompeter Electronics BJ77
6	2	Header, Male 2x20 0.1" Pitch, 0.230" Pins, 0.120" Tails	J3,J4	S2012E-20-ND	Sullins PEC20DAAN
7	2	Header, Male 2x5, 0.1" Pitch, 0.230" Pins, 0.120" Tails	J5A,J5B	S2012E-05-ND	Sullins PEC05DAAN
8	1	Header, 1x10, 0.1" pitch	J6	DO NOT STUFF	
9	1	Header, 1x8, 0.1" pitch	J19	DO NOT STUFF	
10	1	Header, 1x3, 0.1" pitch	J6	DO NOT STUFF	
11	5	Solder Jumper	JP2,JP3,JP8,JP9,JP10	DO NOT STUFF	
12	1	LED Yellow 0805	LED5	160-1175-1-ND	Lite On LTST-C170YKT
13	3	LED Green 0805	LED1 - LED3	160-1179-1-ND	LiteOn LTST-C170GKT
14	1	LED Red 0805	LED4	160-1178-1-ND	LiteOn LTST-C170EKT
15	1	Osc, 50MHz 25ppm 3.3V SMD 5x7mm	OSC1	535-10087-1-ND	Abracon ASV-50.000MHZ-E-T
16	2	Res 69.8 Ohm 1W 1% 2512 SMD	R10,R11	RHM69.8BBCT-ND	Rohm MCR100JZHF69R8
17	5	Resistor, 150 Ohm 5% 1/8W 0805	R8,R9,R12,R80,R96	P150KACT-ND	Panasonic ERJ-6GEY0R151V
18	1	Resistor, 10 Ohm 5% 1/8W 0805	R98	P10ACT-ND	Panasonic ERJ-6GEY0R100V
19	1	Resistor, 10K 5% 1/8W 0805	R99	P10KACT-ND	Panasonic ERJ-6GEYJ103V
20	5	Resistor, 47K 5% 1/8W 0805	R82,R83,R84,R87,R97	P47KACT-ND	Panasonic ERJ-6GEYJ473V
21	1	Resistor, 3.3k 5% 1/8W 0805	R100	P3.3KACT-ND	Panasonic ERJ-6GEYJ332V
22	1	DIP Switch 6-Position SMD	SW1	CT2196MST-ND	CTS 219-6MST
23	2	DIP Switch 10-Position SMD	SW4,SW5	CT21910MST-ND	CTS 219-10MST
24	2	Transformer MIL-STD-1553 Single, 1:2.50,	T1,T2	Holt PM-DB2791S	Holt / Premier Magnetics
25	3	Test Point, Red Insulator, 0.062" hole	(+)BusA, (+)BusB, 3V3	5010K-ND	Keystone 5010
26	3	Test Point, Black Insulator, 0.062" hole	(-)BusA, (-)BusB, GND	5011K-ND	Keystone 5011
27	1	Test Point, White Insulator, 0.062" hole	TP8 (Active)	5012K-ND	Keystone 5012
28	1	IC HI-6135/6/7/8 Holt 48-PQFP	U1	HOLT IC	Holt IC
29	1	IC, Serial EEPROM 512Kbit 20MHz SPI 8SOIC, Microchip	U3	25LC512-I/SN-ND	Microchip 25LC512-I/SN
30	1	Hookup Solid wire - 20AWG - Black - 4" Long per Board	For J1 and J2	C2028B-XX-ND	General Cable C2028A.12.01



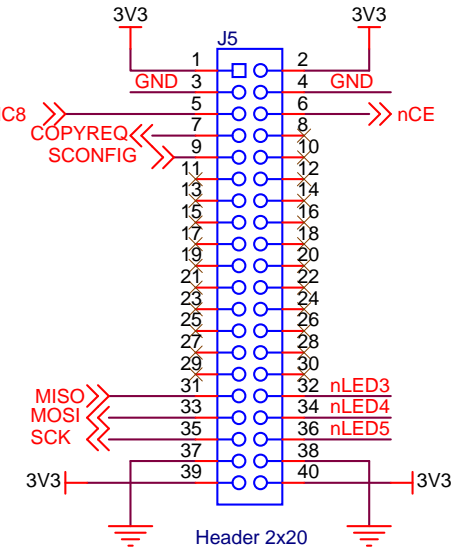
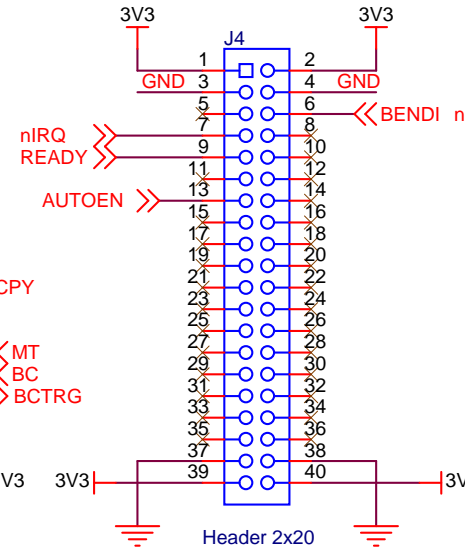
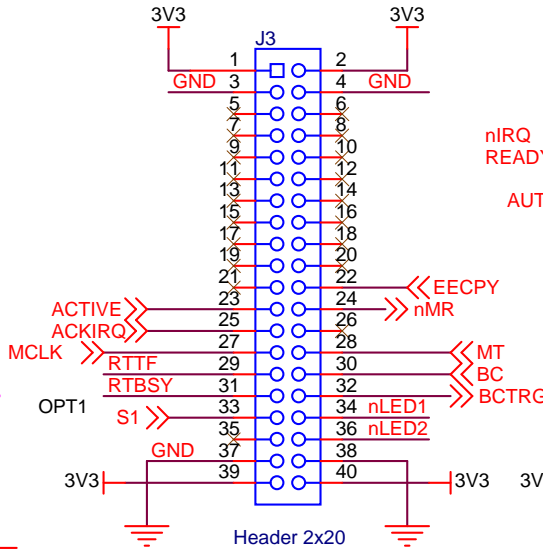
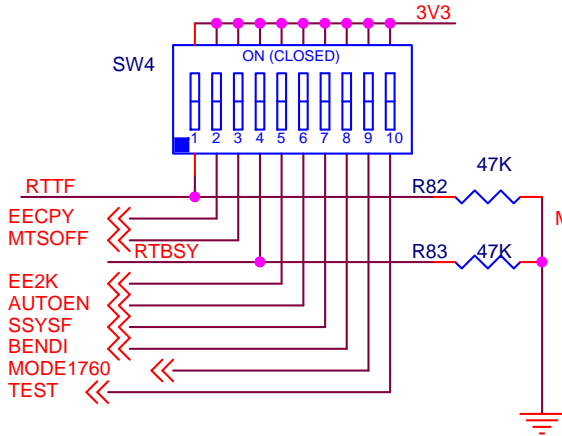
HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA

Title: **HI-6135/6/7/8 CUST EVAL (USE WITH STD CM3)**

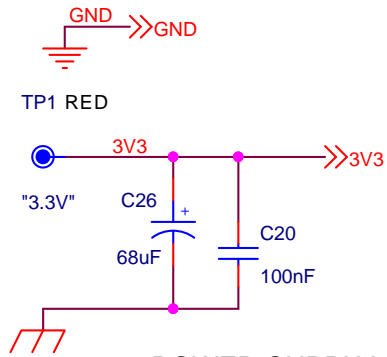
Size A	Document Number 6135 EVAL TOP REV B.DSN	Rev B
--------	--	-------

Date: Friday, May 27, 2016 Sheet 1 of 2

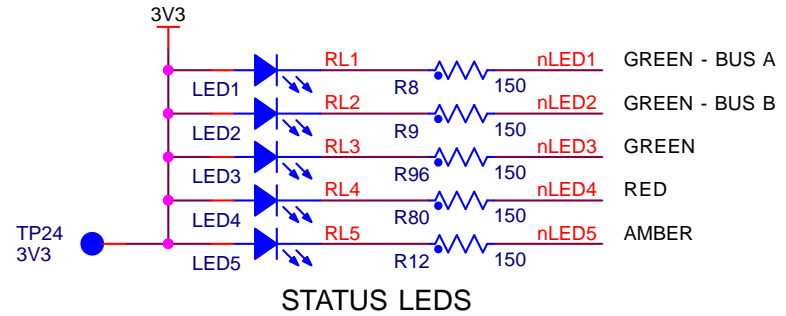
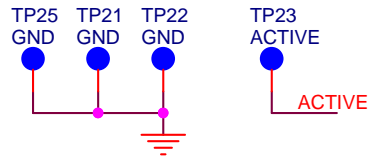
FOR ALL DIP SWITCHES
UP = LOGIC-1
CT21910LPST-ND
HW CONFIGURATION 1



RTTF - SETS RT TERMINAL FLAG
EECPY - MAKES COPY OF RAM AND REGS TO EPROM
MTSSOFF - DISABLE MEMORY TEST ON POWER UP
RTBSY - SETS BUSY BIT OF STATUS WORD
EE2K - SET HIGH EEPROM SAVES ONLY LOWER 2K WORDS
AUTOEN - LOADS FROM EEPROM ON POWER UP
SSSYF - SET SSSYF (SYSTEM FAIL) BIT IN RT STATUS WORD
BENDI SET HI: SPI USES BIG ENDIAN FORMAT
MODE1760 - POWERS UP WITH BUSY BIT SET
TEST - ENABLES TEST MODE, SEE DATA SHEET



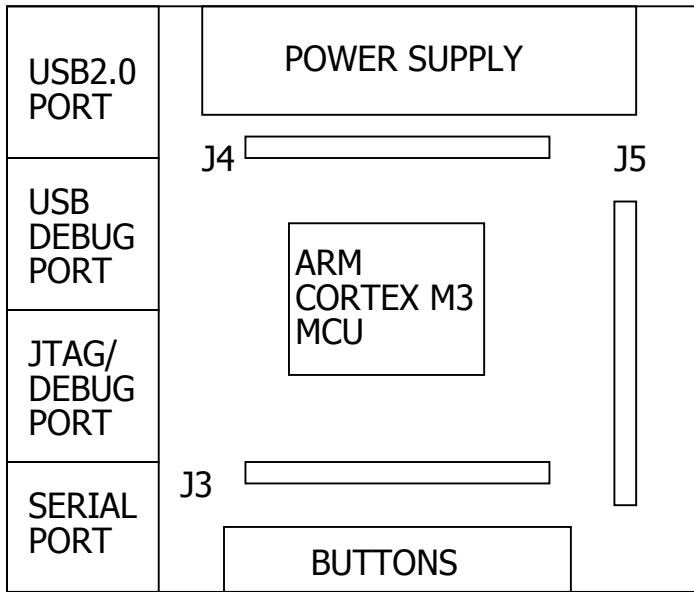
POWER SUPPLY -- 5VDC IN TO 3.3VDC OUT



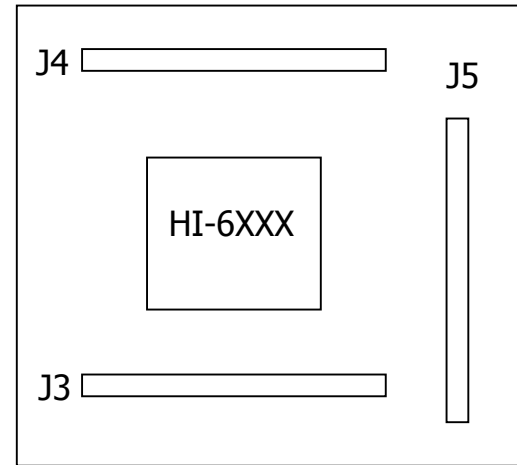
HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA		
Title HI-6135/6/7/8 CUST EVAL USE WITH STD CM3		
Size A	Document Number <Doc>	Rev B
Date:	Friday, May 27, 2016	Sheet 2 of 3

Bill of Materials
ARM Cortex M3 MCU Board
Rev. E

Item	Qty	Description	Reference	DigiKey	Mfr P/N
1					
2	1	PCB, Bare, Evaluation Board	N/A	-----	
3	1	Ferrite Bead, 220 Ohm @ 100MHz 300mA DC 0805	FB1	732-1602-1-ND	Wurth 742792034
4	2	Capacitor, Ceramic 10nF 10% 50V X7R 0603	C1,C42	490-1512-1-ND	Murata GRM188R71H103KA01D
5	2	Capacitor, Ceramic 10pF 10% NP0 C0G 0V 0603	C23,C34	490-1403-1-ND	Murata GRM1885C1H100JA01D
6	4	Capacitor, Ceramic 20pF 5% NP0 C0G 0V 0603	C14,C21,C25, C27	490-1410-1-ND	Murata GRM1885C1H200JA01D
7	29	Capacitor, Ceramic 100nF 10% 25V Y5V 0603	C2,C4,C6-C11, C13,C15-C19,C22,C24,C26,C28,C29,C33, C35-C40,C45-46,C54	490-1575-1-ND	Murata GRM188F51E104ZA01D
8	4	Capacitor, Tantalum 4.7uF 10% 10V Low ESR SMD 1206	C5,C20,C31, C32	478-2391-1-ND	AVX TPSA475K010R1400
9	4	Capacitor, Tantalum 10uF 10% 10V Low ESR SMD 1206	C3,C12,C30,C41	478-3317-1-ND	AVX TPSA106K010R1800
10	1	Capacitor 22uF 10% 6.3V Tantalum Low ESR SMD C	C43	399-10521-1-ND	Kemet T495C226K006ATE380
11	1	Capacitor 100uF 10% 6.3V Tantalum Low ESR SMD C	C44	495-1509-1-ND	Kemet T495C107K006ZTE150
12	1	Header, Male Shrouded 2x10, 0.1" Pitch	J1	HRP20H-ND	Assmann AWHW20G-0202-T
13	1	Connector, Receptacle USB Mini B Rt-Angle PCB Mount	J2	H2959CT-ND	Hirose UX60-MB-5ST
14	1	Connector DB9F, Right-Angle PCB Short Body, Board Lock	J6	AE10924-ND	Assman A-DF-09-A/KG-T4S
15	1	Jack, DC Power, 2.5mm ID x 2.1mm pin	J7	CP-102AH-ND	Cui PJ-102AH
16	3	Receptacle, Female 2x20, 0.1" Pitch, 8.5mm Height, 3.2mm Solder Tails	J3,J4,J5	S6104-ND	Sullins PPTC202LFBN-RC
17	1	Solder Jumper	JP1	SOLDER OPEN	
18	2	Inductor, 10uH,100mA 0805	L1,L2	490-4029-1-ND	Murata LQM21FN100M70L
19	1	LED Green 0805	LED1	160-1179-1-ND	LiteOn LTST-C170GKT
20	0	Resistor, Prov 1/8W 0805	R1,R15,R16, R44,R45	DO NOT STUFF	
21	7	Resistor, 0 ohm 1/8W 0805	R9,R12,R13, R14,R22,R23, R29	P0.0ACT-ND	Panasonic ERJ-6GEY0R00V
22	2	Resistor, 1.0 5% 1/8W 0805	R7,R8	P1.0ACT-ND	Panasonic ERJ-6GEYJ1R0V
23	2	Resistor, 39 5% 1/8W 0805	R4,R5	P39ACT-ND	Panasonic ERJ-6GEYJ390V
24	1	Resistor, 150 5% 1/8W 0805	R17	P150ACT-ND	Panasonic ERJ-6GEYJ151V
25	1	Resistor, 4.7K 5% 1/8W 0805	R3	P4.7KACT-ND	Panasonic ERJ-6GEYJ472V
26	1	Resistor, 6.8K 5% 1/8W 0805	R6	P6.8KACT-ND	Panasonic ERJ-6GEYJ682V
27	0	Resistor, 47K 5% 1/8W 0805	R18	DO NOT STUFF	Panasonic ERJ-6GEYJ473V
28	0	Resistor, 68K 5% 1/8W 0805	R19	DO NOT STUFF	Panasonic ERJ-6GEYJ683V
29	11	Resistor,100K 5% 1/8W 0805	R2,R10,R11, R20,R21,R24, R25,R26,R27, R28,R42	P100KACT-ND	Panasonic ERJ-6GEYJ104V
30	3	Switch Tactile SPST 6 x 6 mm SMT	SW1,SW2,SW3	P12932SCT-ND	Panasonic EVQ-Q2B03W
31	2	Test Point, Black Insulator, 0.062" hole	TP2,TP3	5011K-ND	Keystone 5011
32	1	Test Point, Red Insulator, 0.062" hole	TP1	5010K-ND	Keystone 5010
33	1	IC, MCU 32-Bit 256KB Flash, 144-LQFP	U1	ATSAM3U4EA-AU-ND	Atmel ATSAM3U4EA-AU
34	1	4-Ch TVS ESD Protection SOT23-6	U2	296-28203-1-ND	TI TPD4E001DBVR
35	1	IC, RS232 Driver/Receiver 3.0 to 5.5VDC 16-SOIC (3.9mm wide)	U3	296-19752-1-ND	Texas Inst MAX3232EIDR
36	1	IC Voltage Regulator 3.3V 1A LDO, SOT-223	U5	497-1228-1-ND	ST Micro LD1117AS33TR
37	1	PolyZen 5.6V PPTC protected Zener SMD	U6	ZEN056V130A24LSCT-ND	TE ZEN056V130A24LS
38	1	Filter, EMI 35dB 10A 1MHz-1GHz SMD	U7	490-5052-1-ND	Murata BNX022-01L
39	1	IC Voltage Ref 2.5V 1% Micropower SOT-23	VR1	576-1047-1-ND	Micrel LM4040DYM3-2.5
40	1	Crystal 12.00MHz, 50ppm 20pF, HC-49US leaded	Y1	631-1105-ND	Fox FOXSLF/120-20
41	1	Crystal, 32768 Hz 12.5pF cylinder leaded	Y2	535-9033-1-ND	Abracon AB26TRB-32.768KHZ-T
42	5	Rubber Foot, Bump on Black Hemisphere, .312 X.200 H	Place at 4 corners and center	SJ5746-0-ND	3M SJ61A1
47	1	Capacitor, Ceramic 100nF, -20% / +80% 25V Y5V 0603	C66	490-1575-1-ND	Murata GRM188F51E104ZA01D
48	1	Capacitor, Ceramic 33pF, 5% 50V C0G 0603	C59	490-1415-1-ND	Murata GRM1885C1H330JA01D
49	2	Capacitor, Ceramic 15pF, 5% 50V C0G 0603	C60,C61	490-1407-1-ND	Murata GRM1885C1H150JA01D
54	1	Ferrite Bead, 220 Ohm @ 100MHz 300mA DC 0805	FB2	732-1602-1-ND	Wurth 742792034
55	1	Solder Jumper	JP2	SOLDER OPEN	
56	1	Connector, Receptacle USB Mini B Rt-Angle PCB Mount	J8	H2959CT-ND	Hirose UX60-MB-5ST
57	1	LED Green 0805	LED2	160-1179-1-ND	LiteOn LTST-C170GKT
59	1	Resistor, 220 ohm 5% 1/10W 0603	R31	P220GCT-ND	Panasonic ERJ-3GEYJ221V
63	2	Resistor, 27 ohm 5% 1/10W 0603	R36,R38	P27GCT-ND	Panasonic ERJ-3GEYJ270V
66	1	4-Ch TVS ESD Protection SOT23-6	U4	296-28203-1-ND	TI TPD4E001DBVR



LOWER CIRCUIT BOARD



STACKING UPPER CIRCUIT BOARD

J3,J4 & J5 ARE DUAL-ROW STACKING RECEPTACLES (LOWER BOARD) AND HEADERS (UPPER BOARD).

HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA		
Title		
ARM CORTEX M3 MICROCONTROLLER BOARD		
Size	Document Number	Rev
A	CM3 BOARD REV E.DSN	E
Date:	Wednesday, June 01, 2016	Sheet 1 of 7

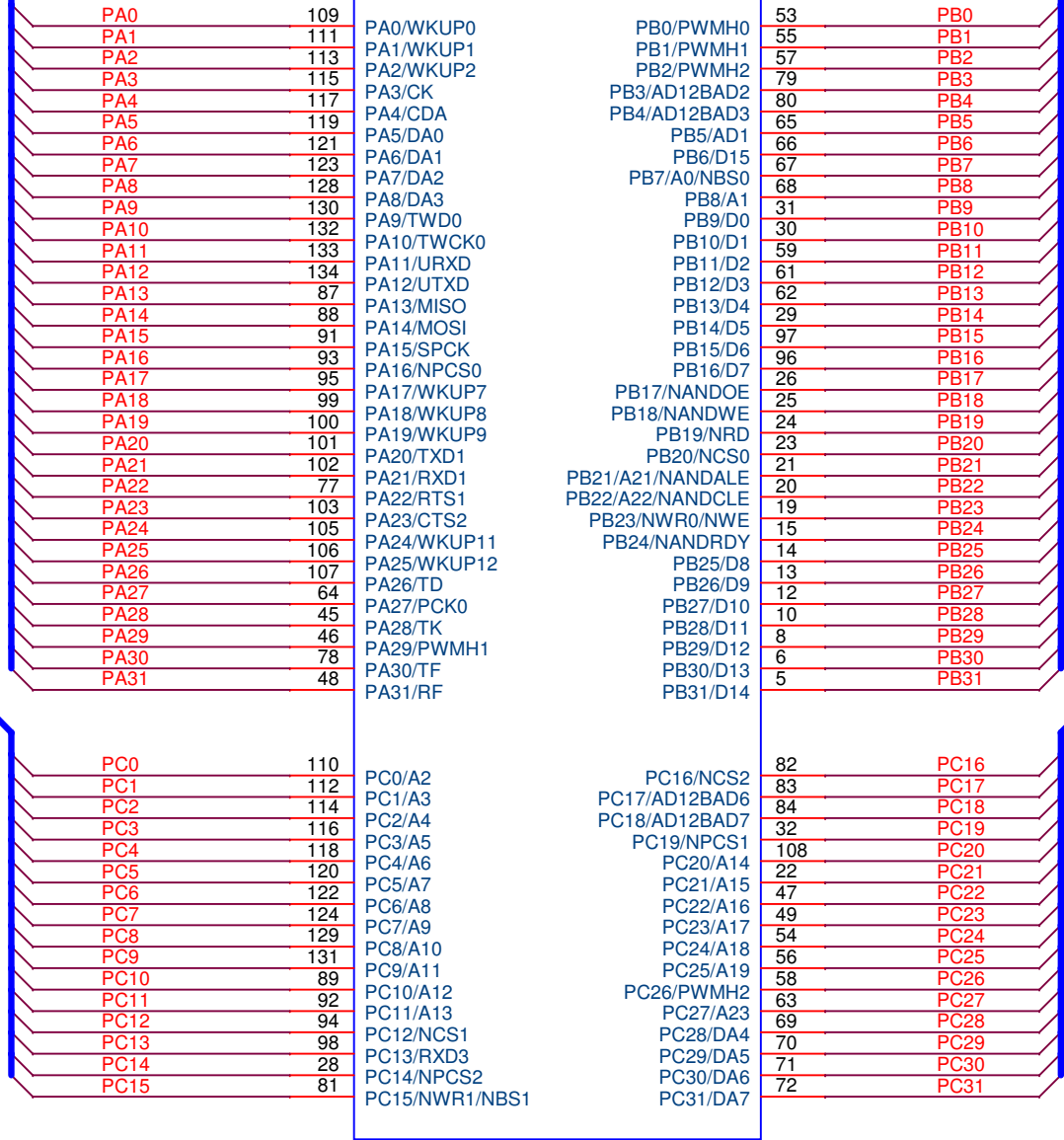
PA[31:0]

PB[31:0]

PC[31:0]

PC[31:0]

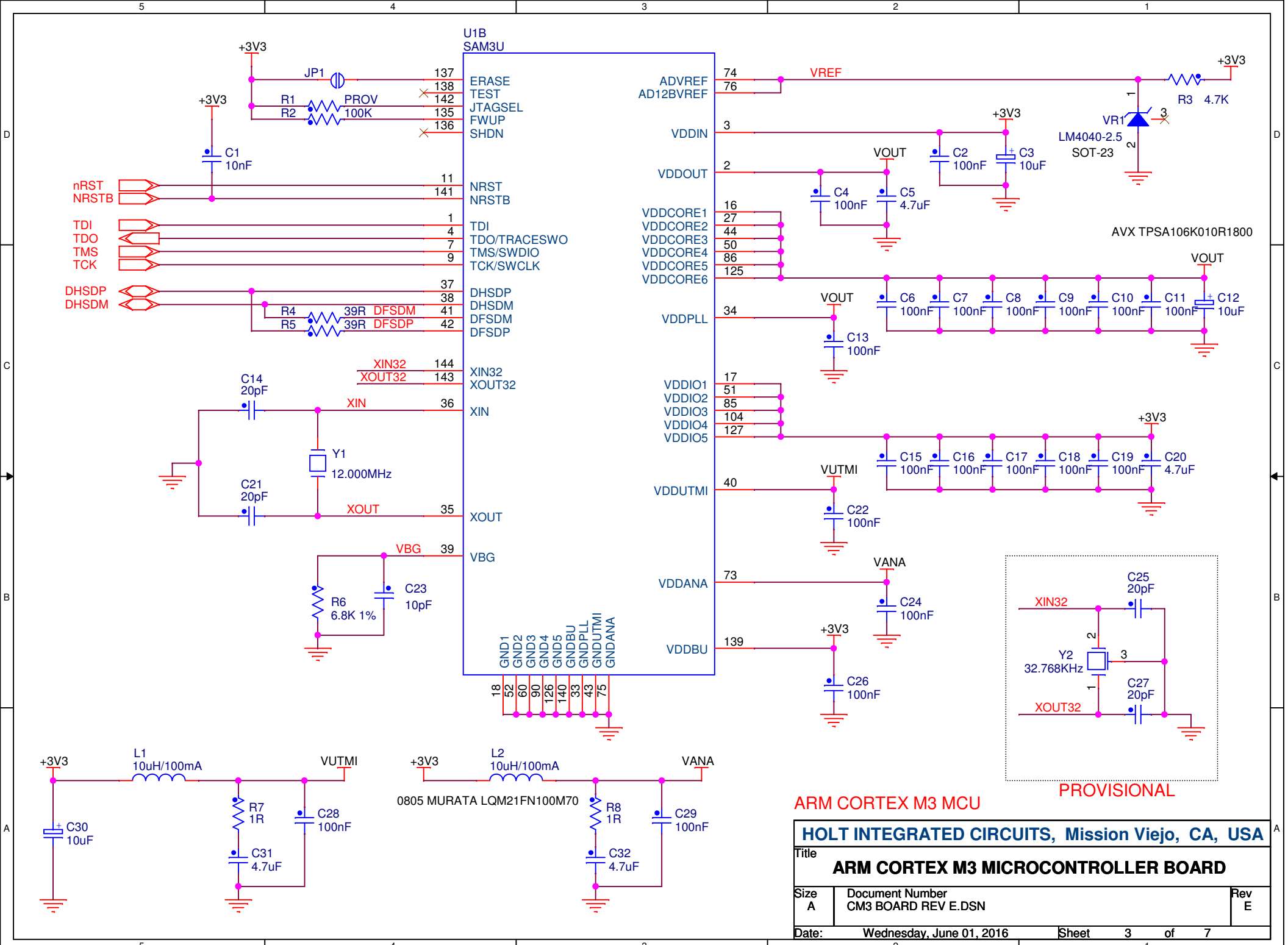
U1A
SAM3U



ARM CORTEX M3 PIO

HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA

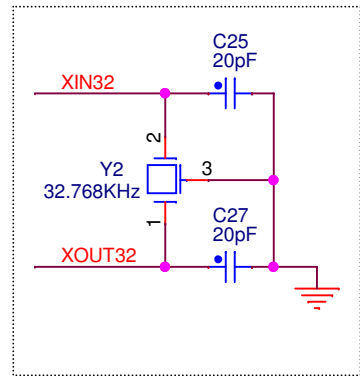
Title		
ARM CORTEX M3 MICROCONTROLLER BOARD		
Size	Document Number	Rev
A	CM3 BOARD REV E.DSN	E
Date:	Wednesday, June 01, 2016	Sheet 2 of 7



ARM CORTEX M3 MCU

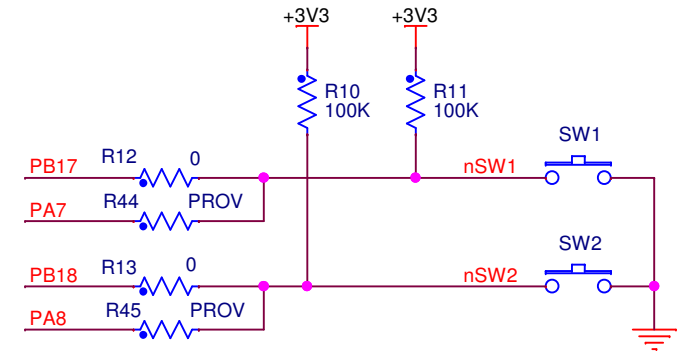
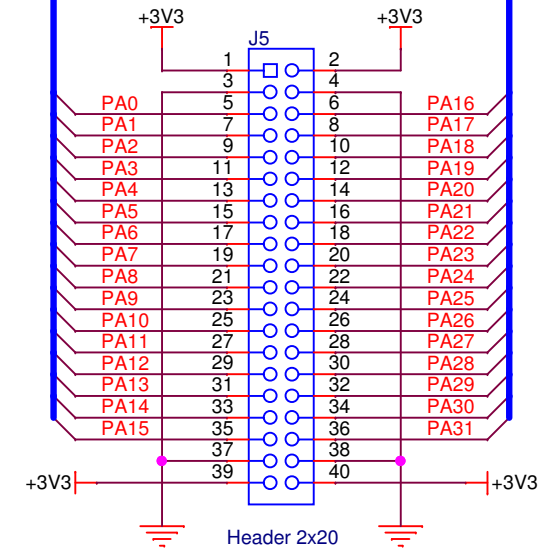
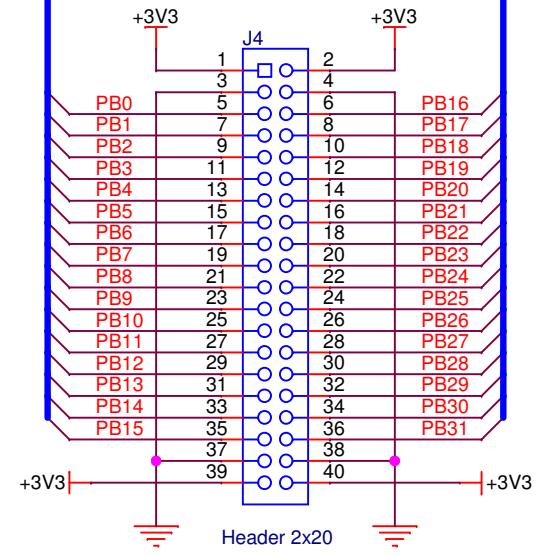
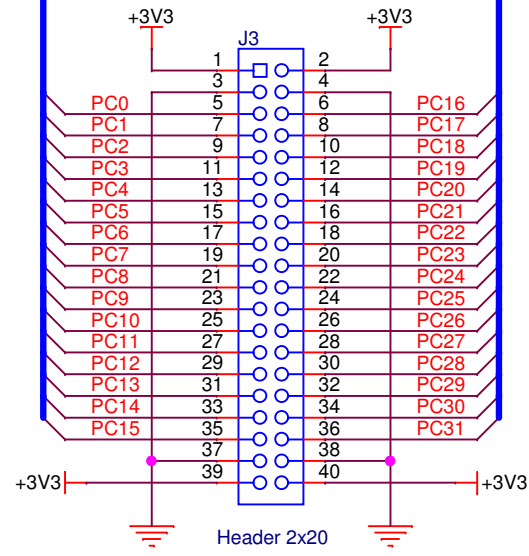
HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA

Title		
ARM CORTEX M3 MICROCONTROLLER BOARD		
Size	Document Number	Rev
A	CM3 BOARD REV E.DSN	E
Date:	Wednesday, June 01, 2016	Sheet 3 of 7



PROVISIONAL

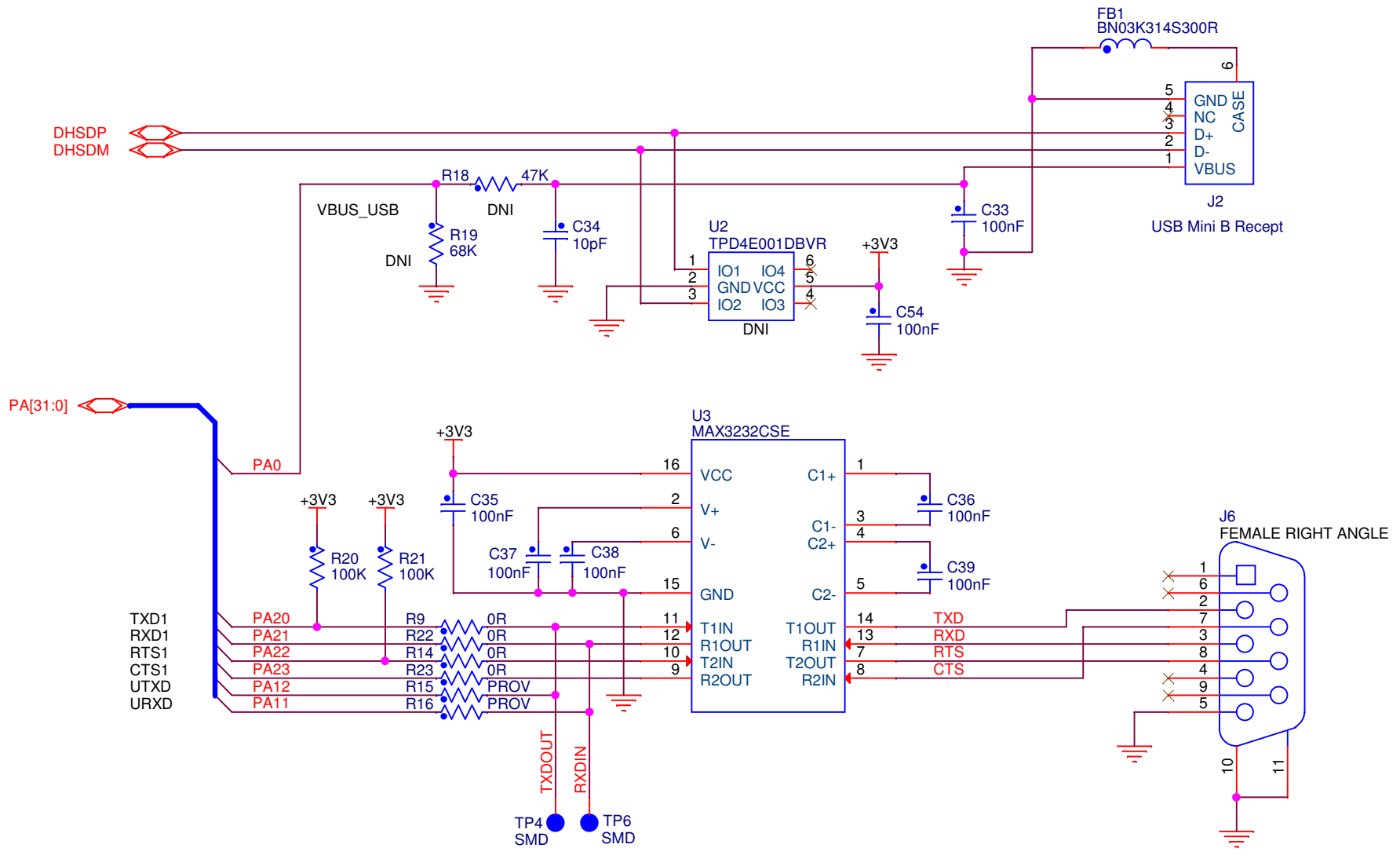
{1,4} PA[31:0]
 {1,3,5} PB[31:0]
 {1,3} PC[31:0]



BOARD I/O HEADERS, BUTTONS

HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA

Title		
ARM CORTEX M3 MICROCONTROLLER BOARD		
Size	Document Number	Rev
A	CM3 BOARD REV E.DSN	E
Date:	Wednesday, June 01, 2016	Sheet 4 of 7



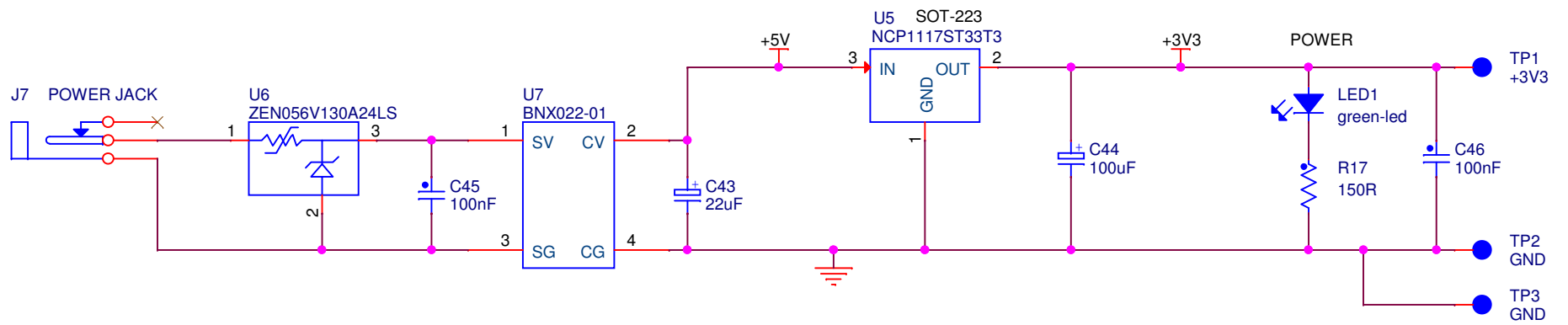
USB & RS-232 SERIAL

HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA

Title
ARM CORTEX M3 MICROCONTROLLER BOARD

Size A	Document Number CM3 BOARD REV E.DSN	Rev E
-----------	--	----------

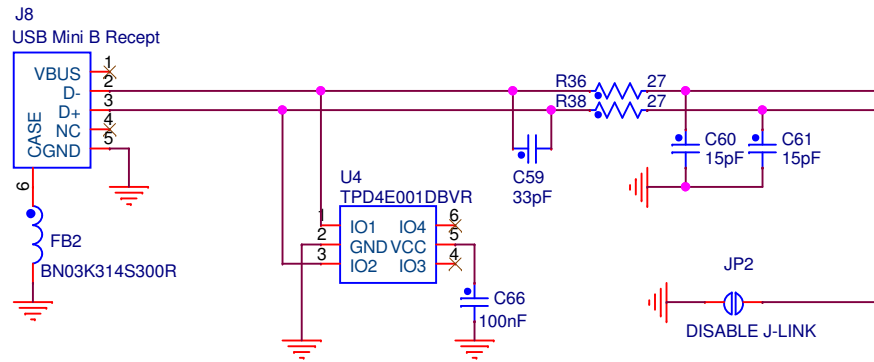
Date: Wednesday, June 01, 2016 Sheet 5 of 7



POWER SUPPLY

HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA		
Title		
ARM CORTEX M3 MICROCONTROLLER BOARD		
Size	Document Number	Rev
A	CM3 BOARD REV E.DSN	E
Date:	Wednesday, June 01, 2016	Sheet 6 of 7

USB DEBUG INTERFACE



**SEGGER J-LINK ON-BOARD
DEBUGGER INTERFACE**

(CONFIDENTIAL)

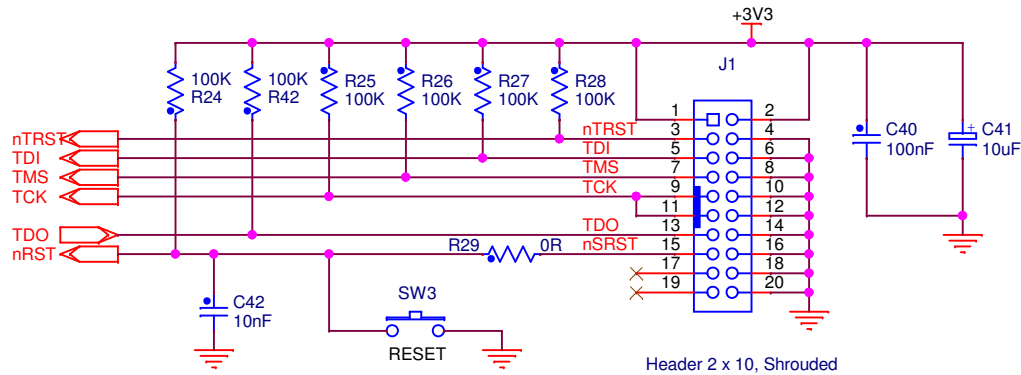
NOT PART OF A CUSTOMER DESIGN,
THIS BLOCK IS COMPRISED OF U8,
Y3, C47-C53, C55-C58, C62-C65, R30,
R32-R35, R37, R39-R41 AND R43.

- TDI
- TMS
- TCK
- TDO
- nRST

**DEBUGGER INTERFACE COPIED
FROM ATMEL ARM CORTEX M3**

USE THIS TO CONNECT J-LINK IF ABOVE
CIRCUITRY IS NOT POPULATED OR WHEN
IT IS DISABLED BY JUMPER JP2.

**PARALLEL
DEBUG INTERFACE**



HOLT INTEGRATED CIRCUITS, Mission Viejo, CA, USA

Title		
ARM CORTEX M3 MICROCONTROLLER BOARD		
Size	Document Number	Rev
Custom	CM3 BOARD REV E.DSN	E
Date:	Wednesday, June 01, 2016	Sheet 7 of 7