

Atmel START

Atmel START 使用者の手引き

概要

Atmel STARTはマイクロコントローラ開発を始めるあなたを手助けします。これは使用可能で最適化された規則であなたの 組み込み応用を誂えるために、MCUを選び、ソフトウェア構成部品、ドライハ、、ミドルウェアと例プロジェクトを構成設定することを 許します。一旦これを終えると、生成したコート、プロジェクトをダウンロート、してAtmel Studioや他の第三者開発ツールで開くこ とができます。

Atmel STARTで以下を行うことができます。

- ・ソフトウェアとハートウェアの両必要条件に基づくMCU選択の手助け
- ・あなたの基板用の例を見つけて開発
- ・トライバ、ミドルウェア、例プロジェクトを構成設定
- ・有効なPINMUX割り付けの構成設定での手助け
- ・システム クロック設定の構成設定

本書は一般の方々の便宜のため有志により作成されたもので、Microchip社とは無関係であることを御承知ください。しおりの[はじめに]での内容にご注意ください。

目次

概要	Ę		61
1.	序説		3
	1.1.	Atmel START、ソフトウェア内容、IDE ······	3
	1.2.	即時開始と作業の流れの概要・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	5
2.	必要多	€件と互換性 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	6
	2.1.	支援するウェフ゛フ゛ラウサ゛・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	6
	2.2.	支援するIDEとコンパイラ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	6
3.	開始に	こ際して: Atmel START ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	7
	3.1.	文書資料 •••••••	7
	3.2.	開始に際して ・・・・・	7
	3.3.	ソフトウェア構成部品の追加と構成設定の方法 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	13
	3.4.	構成設定の保存と復元・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	14
4.	構成詞	と定画面 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	15
	4.1.	ሳባック構成設定部 ·····	15
	4.2.	事象システム構成設定部	17
	4.3.	Atmel START QTouch [®] 容量性感知ライブラリ ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	25
	4.4.	PINMUX構成設定部・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	25
	4.5.	CCL構成設定部 ····································	26
5.	Trust	Zone支援 ····································	30
	5.1.	安全ブロジェクトの作成 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	30
	5.2.	非安全フ ロシ エクトの作成 ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	32
	5.3.	Atmel Studioでの安全フロシェクトのインホート	32
	5.4.	Atmel Studioでの非安全フロン エクトのインホート メ、ッ®センシ のンビ ®女の中へコ゚ン゙ りのひた !	33
	5.5. E.C	Kell®からのµVision®での女生フuンエクトのインホート Kell®からの」ノにに限っのまた合心にたりのクレキ゚レ	36
	5.6. 5.7	Kell®からのµVision®での非女主ノuシェクトのインホート	38 49
	5.7. БО	IAR Embedded Workbenchでの安主/ログェアのイノホート	+2 49
	5.0. 5.0	Makefileを使う安全プロジェクトのイルポート	±2 19
	5.5. 5.10	Makefileを使う非安全プロジェクトのインポート・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	±2 49
6	め.10.	Marcine を使う学 ダエンロンゴーの 12 mm - 1 Iールでの Atmel START出力の 使田 · · · · · · · · · · · · · · · · · ·	12
0.	61	Atmel Studio	43
	6.2	IAR Embedded Woekbench [®]	48
	6.3.	Keil [®] uVision [®]	53
	6.4.	GNU C Makefile	56
	6.5.	SOMNIUM DRT	56
	6.6.	MPLAB [™] X IDE	59
7.	Atmel	STARTでの内容 ····································	32
	7.1.	ARM内容 ····································	62
	7.2.	AVR [®] 内容 ····································	62
8.	キーボー	-ドショ―トカット ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	33
9.	Atmel	STARTへのリンク ····································	34
10.	改訂層	夏歴 ••••••••••••••••••••••••••••••••••••	35
Mic	rochip	ウェブ᠊サイト ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	36
おそ	≩様へ	の変更通知サービス ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	36
お客	⋜様支	援	36
Mic	rochip		66
法的	句诵知		36
商材			67
	" いこよ-	って認証された品質管理システム・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	37
₩ 9	していていていていていていていていていていていていています。	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~	38
120	1.4.3,9-3		50

1. 序説

1.1. Atmel START、ソフトウェア内容、IDE

本項はAVR[®]とSAMのツール体系内の様々な部分の概要とそれらが互いにどう関連するかを与えます。 開始に際しての話題



映像: AVRとSAMのツール体系概要

1.1.1. Atmel START

Atmel STARTは様々なソフトウェア枠組みに対するウェブに基づくソフトウェア構成設定ツールで、これはMCU開発開始を手助けします。新規 プロジェクトまたは例プロジェクトのどちらかからの開始でも、Atmel STARTは便利で最適化した規則で、(ASF4とAVR Codeから)あなたの 組み込み応用を仕立てるためのトライバやミドルウェア(中間ソフトウェア)のようなソフトウェア部品を選んで構成設定することを許します。一旦最 適化されたソフトウェア構成設定が行われると、生成されたコート、プロジェクトをダウンロードしてAtmel Studio 7、IAR embedded Workbench[®]、 Keil[®] µVsionを含む、あなたの選ぶIDEでそれを開くか、または単にmakefileを生成することができます。

Atmel STARTは以下を許します。

- ・ソフトウェアとハードウェアの両要件に基づくMCU選択を助けます。
- ・あなたの基板用の例を探して開発します。
- ・トッライバ、ミドルウェア、例プロジェクトを構成設定します。
- ・有効なPINMUX設計の構成設定を助けます。
- ・システムクロック設定を構成設定します。



1.1.2. ソフトウェア内容 (ト・ライハ・とミト・ルウェア)

高度なソフトウェア枠組み(ASF)

ASF(Advanced Software Framework)はお客様の設計時間を減らすために専門家によっては開発された検証済みのドベイベとコード単 位部の豊富な組を提供します。これはドライベと高価値のミドルウェアを通してハードウェアに対する抽象化を提供することによってマイクロ コン トローラの使用を簡単化します。ASFは評価、試作、製造段階に使われるように設計された無料の開放ソース コード ライブラリです。

SAM製品部門を支援するASF4はASFの第4主要世代です。Cはメモリ量、コード性能を改善してAtmel STARTウェブ使用者インターフェースと 調和するように、枠組み全体の完全な再設計と再実装になります。ASF4はAtmel STARTと共に使われなければならず、これはASF 2 と3のASFウィザードを置き換えます。

Michochip.com: ASF製品頁

AVR[®] Code

AVR製品部門を支援するAVR Codeは8ビットと16ビットのPIC MCUを支援する創設サービスと等価なAVR 8ビットMCU用の簡単なファーム ウェア枠組みです。AVR Codeはコート、量とコート、速度だけでなく、コートの簡潔性と信頼性に対しても最適化されています。AVR Codeは Atmel STARTによって構成設定されます。

1.1.3. 統合開発環境 (IDE)

IDE(Integrated Development Environment)はAtmel STARTで構成設定されてエクスポートされたト・ライハ・とミト・ルウェアのようなソフトウェア部品 に基づく応用の開発(または例応用の更なる開発)に使われます。Atmel STARTはAtmel Studio 7、IAR embedded Workbench[®]、 Keil[®] µVsion[®]を含むIDEの範囲を支援します。

Atmel Studio 7は全てのAVRとSAMのマイクロコントローラ応用を開発してデバックするための統合開発基盤(IDP:Integrated Development Platform)です。Atmel Studio 7 IDPはC/C++またはアセンブリ言語のコートで書かれた応用を書いて構築してデバッグするための継ぎ目がなく使い易い環境を与えます。それはAVRとSAMのデバイスを支援するデバッガ、書き込み器、開発キットに対しても継ぎ目なく繋げます。Atmel STARTとAtmel Studio 7間の開発体験は最適化されています。Atmel Studio 7でのAtmel STARTに基づくプロジェクトの反復的な開発は再構成設定と機能併合を通して支援されます。

このAtmel Studio 7に対する練習開始はIDEの主な機能の全てを通してあなたを導きます。これは実践を伴う映像系列として設計されています。各項はその項を網羅する映像で始まります。

1.2. 即時開始と作業の流れの概要

ASF4は周辺機能トライバと例応用から成るソフトウェアライブラリです。Atmel STARTウェブ「頁(http://start.atmel.com)は使用者の応用で必要とされるソフトウェア単位部を選び、使用者の求めに従ってそれらの単位部を構成設定するのに使われます。プロジェクトェクスポート(Exp ort Project)画面使用時、対応する生成したCコート、プロジェクトを使用者のコンピュータにダウンロート、してAtmel Studio 7、IAR™ Embedded Workbench、またはKeil µVisonのような使用者が選ぶIDEにインポートすることができます。IDEはプロジェクトの変更、コンハペイル、書き込み、デバックに使われます。

インストール

ASF4はAtmel STARTウェブ頁(http://start.atmel.com)を使って構成設定され、構成設定された応用は、その後にzipファイルとして使用 者のコンビュータにダウンロートされます。ASF4はどの構成部分も手元にインストールする必要がありません。使用者は通常、それらのコンビュー タにインストールされたIDEや独立型ツールチェーンを持つことを望み、故に使用者はダウンロートじしたコート、をコンハペイル、書き込み、デバッグをする ことができます。

作業の流れ

作業の流れはかなり素直で、新規プロジェクト作成(Create New Project)画面または既存例検索(Browse Existing Examples)画面のどちらかで始めます。

既存例検索(Browse Existing Examples)画面で開始:

Atmel START例は開発に対する開始点として使われるように設計されました。従って、効率的な選別機構はそれらの必要条件に最も近いプロジェクトを見つける開発者を手助けするのに利用可能で、直ぐに使える高品質で製造準備が整ったコートをそれらに与えます。けれども、これらの例プロジェクトはピン配置の変更(PINMUX画面)または、付加的なドライバやミドルウェアの追加(プロジェクト計器盤(DAS HBOARD))によってソフトウェア構成設定を拡張することができるため、変更することも容易です。例えば、追加計時器に対する支援追加や、BLEインターフェースの追加さえもです。Atmel STARTを使うプロジェクト構成設定任意選択を理解するにはAtmel START使用者の手引きの「構成設定画面」章をご覧ください。Atmel Studio 7(ファイル(File)⇒新規(New)⇒Atmel STARTプロジェクト(Atmel Start project))から直接Atmel STARTプロジェクトを作成または再構成設定することも可能です。

新規プロジェクト作成(Create New Project)画面で開始:

この画面はソフトウェアとハードウェアの両方に基づいてあなたのプロジェクトに対してMCUを選ぶのを手助けするように設計されました。

- 1. プロジェクト開始前に必要条件によってMCUを選別
- 2. あなたのプロジェクトに構成部品を追加、例えば、周辺機能トライバとミドルウェア
- 3. 各構成部品を構成設定
- 4. プロジェクトをエクスポートして更なる開発のため、支援するIDEの1つにそれを追加

IDEの役割とコート、走行

ー旦使用者がソフトウェア構成設定に満足すると、そのプロジェクトは開発者の選ぶIDE用のzipファイルとしてェクスポートされます。これを行う 方法についての指示だけでなく、支援するIDEの一覧の提供についてもAtmel START使用者の手引きの「**外部ツールでのAtmel STA RT出力の使用**」をご覧ください。ASF4/Atmel STARTはどの構成部分もあなたの手元のコンピュータにインストールする必要がありません。 IDEは最終製品に対して例プロジェクトの機能を拡張するのに必要とされるコートを開発するだけでなく、ダウンロート゛したコート゛をコンパイル、 書き込み、デバッグをするのにも使われます。ダウンロート゛した応用はコンパイルの準備が整っています。コート゛をコンパイルする上での指示に ついてはあなたのIDE資料を参照してください。ダウンロート゛したコート゛の動きは応用依存です。

・"新規プロジェクト"の構成設定は全てのドライバを初期化するmain()関数を生成しますが、他の操作はしません。

・ "例プロジェクト"の構成設定はもっと複雑な操作を実行するmain() 関数を生成します。

関連リンク 4. 構成設定画面

6. 外部ツールでのAtmel START出力の使用

2. 必要条件と互換性

2.1. 支援するウェブ ブラウザ

Atmel STARTはHTML5適合ブラウザで最も良く動きます。 これは以下の最新版を含みます。

- Chrome
- Firefox
- ・ Internet Explorer (11版またはそれ以上)
- Edge
- Opera
- Safari

2.2. 支援するIDEとコンパイラ

Atmel STARTからの出力は次のようなソフトウェア ツールの組で使うことができます。

- ・ インストールされたAtmel START拡張を持つAtmel Studio 6.2
- ・Atmel Studio 7.0またはそれ以降
- IAR Embedded Workbench

+ Keil $\mu Vision$

また、Atmel START出力は生成されたMakefileを利用してコマント行GNUコンハパラで使うことができます。

Atmel STARTはコードを生成するのにCMSIS一括を使います。いくつかの有用なリンクは次のとおりです。

- ・CMSIS一括の一般情報 http://www.keil.com/pack/doc/CMSIS/Pack/html/index.html
- Atmel Studio用デバイス系統一括(DFP:Device Family Packs)はhttp://packs.download.atmel.comからダウンロードすることができませ すが、Atmel Studioの一括管理部(Pack Manager)から最も良く管理されます。
- ・Keilデバイス支援、DFPに関してはAtmel下をご覧ください。http://www.keil.com/dd2
- 関連リンク 6. 外部ツールでのAtmel START出力の使用

3. 開始に際して: Atmel START

ブラウサを開いてhttp://start.atmel.comに行ってください。

3.1. 文書資料

文書の主な供給元はプロジェクトの開始に際して(Getting Started projects)、Atmel START使用者手引き(Atmel START User Manual) だけでなく、ASF4と基礎サービス枠組み内容用の参考手引書(reference manuals)もです。

プロジェクトの開始に際して

あなたの組み込み応用を構築するため、それらを共に動くようにするのにシステムの各種断片をどう使うか、例えば、Atmel Studio 7の ようなIDEとでAtmel STARTを効率的に使う方法。

プロジェクトの開始に際してはプロジェクト使用者手引きに繋げられた映像系列や実地訓練手引きのような訓練材料を持ちます。例プロ ジェクトそれ自身は度々関連する実地や映像の訓練材料の目的や最終点を表します。訓練材料はAtmel STARTとあなたのIDEが共 にどう動くかの作業の流れ概要を与えます。

- ・ 例検索(Browse Examples)を開き、区分(Category)引き落としメニューをクリックし、開始に際して(Getting Started)区分を選んでください。
- ・訓練材料へのリンクが例プロジェクト使用者手引きで見つけられ、これはプロジェクトを開くことなく、アクセスすることができます。
- ・ Atmel START使用者手引き(Atmel START User Manual)をご覧ください。

START使用者手引き

- ・Atmel STARTで何?
- ・即時開始と作業の流れ概要
 - プロジェクトの開始に際して(Getting Started projects)を使い、適切な例プロジェクトを見つけて/再構成設定してください。
- ソフトウェアとハードウェアの両要件に基づいてMCUを選ぶことによって新規プロジェクトを作成(Create New Project)してください。
- ・様々なAtmel START構成設定画面の使い方
 - 計器盤 (Dashboard)
 - PINMUX
 - 事象システム構成設定部
 - QTouch[®]構成設定部
- ・以下のような様々なIDEへプロジェクトをエクスポートする方法
 - Atmel Studio 7
 - IAR Embedded Workbench
 - Keil µVision
 - Makefile
- ・Atmel STARTが構成設定して生成することができる内容概要とソフトウェア
 - ASF4
 - 基礎サービス

ASF4 API参考手引書

- ・ASF4ソフトウェア基本構造
- ・トライバ実装と設計慣例
- ・API参照基準

AVR[®] コードAPI参考手引書

- ・基礎サービスソフトウェア基本構造
- ・トライバ実装と設計慣例
- ・API参照基準

3.2. 開始に際して

Atmel START例は組み込みプログラム作成者の優れた開始点です。例プロジェクトは直ぐに動きますが、Atmel STARTを使うことによって変更することも容易です。

例プロジェクトは各互換基板とデバイス用に誂えられ、小さなメモリ量で高品質で製造準備が整ったコートに帰着します。各例についてより 多くの情報を得るには利用可能な使用者手引きを読んでください。これはプロジェクトを開くことさえなしに利用可能です。

プロジェクトの開始に際してはプロジェクト使用者手引きに繋げられた映像系列や実地訓練手引きのような訓練材料を持ちます。例プロ ジェクトそれ自身は度々関連する実地や映像の訓練材料の目的や最終点を表します。このシステム水準の訓練材料は、システムの各種断 片を共にどう動かすか、即ち、あなたの組み込み応用を構築するため、Atmel Studio 7のようなIDEとでAtmel STARTをどう効率的に 使うかを示すように設計されています。

・ 例検索(Browse Examples)を開き、区分(Category)引き落としメニューをクリックし、開始に際して(Getting Started)区分を選んでください。

・訓練材料へのリンクが例プロジェクト使用者手引きで見つけられ、これはプロジェクトを開くことなく、アクセスすることができます。

具体的に、ASF4の入門のため、感知器プロジェクトの開始に際してと訓練材料が特に適切です。Atmel START頁(http://start.atmel.c om/)の"映像指導"のリンクでこのプロジェクトを実演する映像の系統を見つけるでしょう。この映像系統でI/O1 Xplained Pro拡張基板上 のアナログ光とデジタルI²C温度の感知器を使ってプロジェクトを構築しています。映像系統は実践訓練文書によって支援されます。両訓 練資料は下の方法一覧を含むASF4プロジェクトで作業する以下の重要な概念を網羅します。 方法:

- ・必要とされるドライバとミドルウェアを追加して構成設定し、またピンとクロックを構成設定するためのAtmel STARTの使用
- そのレジスタの説明の文脈に於いてデータシートを開くための入出力表示部(I/O View)(<F1>)のような多くの最新機能が使われるAtmel Studio 7にAtmel STARTプロジェクトをエクスポート
- ・効率的にプロジェクト機能を構築するためにExamplesフォルダとAtmel_Start.cからの応用レベル関数の使い方の最良の実践を学習
- ・データ可視器(Data Visualizer)を使って各開発段階を容易に確認
- ・循環的に機能性を構築するためにAtmel Studio 7からAtmel STARTプロジェハを再構成設定

3.2.1. 開始に際して: 例プロジェクトを開く

- 1. ブラウザを開いてhttp://start.atmel.comに行ってください。
- 2. 例検索(BROWSE EXAMPLES)を選んでください。利用可能な全ての例プロジェクトを示すウィントウが開きます。
- 3. 感知器開始(Getting Started Sensors)を見つけるために区分(Category)から開始(Getting Started)を選んで(または検索選別部 で"感知器(Sensors)"を入力して)ください。プロジェクトの概要を得るために使用者手引き(User guide)をクリックしてください。

助言: 例プロジェクト使用者手引きはプロジェクトが何を行うかの説明、必要とされるインターフェース設定だけでなく、プロジェクトを走らせるための簡単な手順一覧も与えます。開始プロジェクトはそれらの使用者手引きで訓練材料や映像へのリンクも含みます。

4. 例が有用であれば、例上でクリックし(それを青の強調表示にし)、その後にそれを開くために選んだ例を開く(OPEN SELECTED EXAMPLE)をクリックしてください。

mel START exa ART. ample project	amples are ts are tailor	a great starting point ed for each compatib	for embedded program	mmers. Example projects will work "out sulting in high quality production ready	of the box" but are also easily to	o modify using Atmel orint. To get more
ormation abo arch:	Filter	ample, please read th	e available user guides.	Getting started (3)	Board: All	
Name		Categories	Description		Board(s) supported	Liser quide
AN17644 Getti STARTED with	ing AVR	· · ·	This is Getting ATmega328PB optimization a reading and Lt	STARTED with AVR using 8.In this application power ind use of sleep mode is shown, ADC ED dimming with TC1 is also done.	Custom board (ATmega328PB)) User guid
etting START vents	ed AVR	÷ 4	This example o communicatio overflow even will be connec	demonstrate peripheral-to-peripheral n using Event System. An RTC t will trigger the ADC conversion, both ted via an event channel	ATtiny817 Xplained Mini	User guid
etting START	ed	1/0 -	This example of	demonstrates sensor integration with	SAM L21 Xplained Pro	User guid
ATMEL: GETTING This proje and digita • Lea • Us	APPLICAT G STARTE ect represe al I2C temp arn how to se some of	TON_EXAMPLES:0. D: SENSORS Ints the point we will g rerature sensors on th use the data visualize the latest Atmel Studi	0.1::APPLICATION:C get to by video 12 in a G ie Atmel IO1 Xplained P er to test each developr o 7 features through th	SETTING_STARTED_SENSORS: Setting STARTed series. In this video ser ro extension board. ment step le course of this development	ies, we'll be building a project usi	ing the analog light
ATMEL: GETTING This proje and digitz • Lea • Us	APPLICAT G STARTE ect represe al I2C temp arm how to e some of control of con	TION_EXAMPLES:0. D: SENSORS Ints the point we will g lerature sensors on th use the data visualize the latest Atmel Studi	et to by video 12 in a G te Atmel IO1 Xplained P er to test each developr o 7 features through th	SETTING_STARTED_SENSORS: Setting STARTed series. In this video ser Pro extension board. ment step ne course of this development	ies, we'll be building a project usi	ing the analog light
ATMEL: GETTING This proje and digit: Us Us	APPLICAT G STARTE ect represe al I2C temp arn how to se some of StART	TION_EXAMPLES:0. D: SENSORS Into the point we will g the rature sensors on th use the data visualize the latest Atmel Studie	o.1::APPLICATION:C	SETTING_STARTED_SENSORS: Setting STARTed series. In this video ser Yro extension board. ment step He course of this development	ies, we'll be building a project usi	ing the analog light
ATMEL: GETTING This proje and digita • Lea • Us	APPLICAT G STARTE ect represe al I2C temp arm how to te some of arm how to te some of arm how to te some of ar	TION_EXAMPLES:0. D: SENSORS Ints the point we will g erature sensors on th use the data visualize the latest Atmel Studie the latest Atmel Studie ser son trainings, which f	o.1::APPLICATION:C et to by video 12 in a G e Atmel IO1 Xplained P er to test each developr o 7 features through th	SETTING_STARTED_SENSORS: Setting STARTed series. In this video ser Yo extension board. ment step le course of this development	ies, we'll be building a project usi	ing the analog light
ATMEL: GETTING This proje and digits Lee Us Us Atmet Atmet STA	APPLICAT G STARTE ect represe al I2C temp arn how to se some of weistArt weistArt ARTed Serve e two hand	TION_EXAMPLES:0. D: SENSORS Ints the point we will g ierature sensors on the use the data visualize the latest Atmel Studie the latest Atmel Studie the latest Atmel Studie source sensors of the sensors source sensors of the sensors source sensors of the sensors of the sensors source sensors of the sensors	o.1::APPLICATION:C et to by video 12 in a G te Atmel IO1 Xplained P er to test each developr o 7 features through th	SETTING_STARTED_SENSORS: Setting STARTed series. In this video ser Pro extension board. ment step le course of this development	ies, we'll be building a project usi	ing the analog light



助言: 例一覧は検索用語、区分、基板で選別することができます。 検索(Search)はプロジェクト名または説明のどちらかの語で例を選別します。 区分(Category)はBLEのような鍵となる技術で例を選別します。 基板(Board)は開発キット名で例を選別します。

3.2.2. 開始に際して: 新規プロジェクト作成

新規プロジェクト作成(Create New Project)画面はソフトウェアとハードウェアの両必要条件に基づいてあなたのプロジェクトに対してMCUを選ぶのを手助けするように設計されました。この例は新しいプロジェクトをどう作成するかを記述します。

1. ブラウザを開いてhttp://start.atmel.comに行ってください。

2. 新規プジェクト作成(CREATE NEW PROJECT)を選んでください。



下図で示されるように新規プロジェクト作成(CREATE NEW PROJECT)画面が現れます。選別(FILTERS)部分では、プロジェクトにミト ルウェア(MIDDLEWARE)とトライハ'(DRIVERS)を追加する時に、それらの必要条件に合うMCUの一覧が絞り込まれます。ソフトウェア検 索(SEARCH FOR SOFTWARE)は全てのソフトウェア構成部品を検索し、ミトルウェアとトライハ'の両方に関連する結果を表示します。トラ イハ'はMCU周辺機能ト'ライハ'と外部構成部品を支援するために書かれた構成部品ト'ライハ'の両方を参照します。

FILTERS		RESULTS						
HARDWARE	CO	WiFi (1) 🗙						
		Filter on device		Show all Show only boards Show only devices				
EARCH FOR SOFTWARE		Name	Architecture	Package	Pins	Flash	SPAM	
wifi		ATSAME70119	CORTEX-M7	LOEP64	64	512 KB	256 KB	
MIDDLEWARE	\odot	ATSAME70I20	CORTEX-M7	LOEP64	64	1 MB	384 KB	
WiFi		ATSAME70I21	CORTEX-M7	LOEP64	64	2 MB	384 KB	
		ATSAME70N19	CORTEX-M7	LOFP100	100	512 KB	256 KB	
DRIVERS	\otimes	ATSAME70N19	CORTEX-M7	TFBGA100	100	512 KB	256 KB	
		ATSAME70N20	CORTEX-M7	LOFP100	100	1 MB	384 KB	
		ATSAME70N20	CORTEX-M7	TFBGA100	100	1 MB	384 KB	
		ATSAME70N21	CORTEX-M7	LQFP100	100	2 MB	384 KB	
		ATSAME70N21	CORTEX-M7	TFBGA100	100	2 MB	384 KB	
		ATSAME70Q19	CORTEX-M7	LQFP144	144	512 KB	256 KB	
		ATSAME70Q19	CORTEX-M7	UFBGA144	144	512 KB	256 KB	
		ATSAME70Q19	CORTEX-M7	LFBGA144	144	512 KB	256 KB	
		ATSAME70Q20	CORTEX-M7	LQFP144	144	1 MB	384 KB	
		ATSAME70Q20	CORTEX-M7	UFBGA144	144	1 MB	384 KB	
		ATSAME70Q20	CORTEX-M7	LFBGA144	144	1 MB	384 KB	
		ATSAME70Q21	CORTEX-M7	LQFP144	144	2 MB	384 KB	
		SAM E70 Xpla	ined					
		ATSAME70Q21	CORTEX-M7	UFBGA144	144	2 MB	384 KB	
		350 of 542 boards an	d devices	LEDGAAAA	***	2.440	204.62	_
		MODEINFORMAT						

3. 例えば、ソフトウェア検索(SEARCH FOR SOFTWARE)バーで"WiFi"を入力して検索によって見つかるWi-Fiミドルウェア上をクリックす ることにより、より多くの情報(MORE INFORMATION)下で選んだソフトウェア構成部品の説明を見ます。ここで、これがSPI/USAR Tインターフェースを通してATWINC1500網制御器SoCを支援するライブラリであることを知ることができます。 4. Wi-Fiミドルウェア数を1に増やすと、今やデバイスを選別し、それらのMCUとATWINC1500 Wi-Fi SoCで動くことができる基板だけを示します。どのキットがWi-Fi拡張を支援するかを知るには、結果(RESULTS)ウィントウで基板のみ表示(Show only boards)をチェックしてください。下の画像はこの段階の結果を示します。新規プロジェクト作成(CREATE NEW PROJECT)をクリックしてください。

HARDWARE Image: Starcch FOR SOFTWARE Wifi Image: Starcch FOR SOFTWARE Wifi Image: Starcch FOR SOFTWARE Image: Starcch FOR SOFTWARE <td< th=""><th>ds Show only devices Pins Flash SRAN</th></td<>	ds Show only devices Pins Flash SRAN			
SEARCH FOR SOFTWARE	ds) Show only devices			
Name Architecture Package Pins MIDDLEWARE Image: SAM E70 Xplained Image: SAM G53 Xplained Pro Image: SAM G53 Xplained Pro WiFi Image: SAM C53 Xplained Pro Image: SAM C53 Xplained Pro Image: SAM C21 Xplained Ultra DRIVERS Image: SAM C21 Xplained Pro Image: SAM C21 Xplained Pro Image: SAM C21 Xplained Pro SAM C21 Xplained Pro Image: SAM C21 Xplained Pro Image: SAM C21 Xplained Pro Image: SAM C21 Xplained Pro SAM C21 Xplained Pro Image: SAM C21 Xplained Pro Image: SAM C21 Xplained Pro Image: SAM C21 Xplained Pro SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro Image: SAM D21 Xplained Pro	Pins Flash SRAM			
Wifi SAM E70 Xplained MIDDLEWARE SAM C53 Xplained Pro WiFi I DRIVERS SAM C21 Xplained Pro SAM C21 Interface with ATSAMBLDCHV-STK SAM C21 Xplained Pro SAM D10 Xplained Pro SAM D10 Xplained Pro SAM D21 Xplained Pro SAM D10 Xplained Pro SAM D21 Xplained Pro SAM D10 Xplained Pro SAM D21 Xplained Pro SAM D21 Xplained Pro				
 MIDDLEWARE WIFI I DRIVERS SAM G53 Xplained Pro SAM G55 Xplained Pro SAM C55 Xplained Pro SAM C21 Interface with ATSAMBLDCHV-STK SAM C21 Xplained Pro SAM D10 Xplained Mini SAM D10 Xplained Pro SAM D10 Xplained Pro SAM D10 Xplained Pro SAM D10 Xplained Pro SAM D20 Xplained Pro SAM D21 Xplained Pro 				
WIFI I I SAM G55 Xplained Pro SAM V71 Xplained Ultra AM V71 Xplained Ultra SAM C21 Xplained Pro SAM C21 Xplained Pro SAM C21 Xplained Pro SAM D10 Xplained Mini SAM D10 Xplained Mini SAM D10 Xplained Pro SAM D20 Xplained Pro SAM D20 Xplained Pro SAM D21 Xplaine				
WiFi SAM V71 Xplained Ultra S AM V71 Xplained Ultra S AM C21 Interface with ATSAMBLDCHV-STK S AM C21 Xplained Pro S AM D10 Xplained Mini S AM D10 Xplained Mini S AM D10 Xplained Pro S AM D20 Xplained Pro S AM D21 Xplained Pro				
DRIVERS				
SAM C21 Xplained Pro AMD C21 Xplained Pro AMD C21 Xplained Mini AMD C21 Xplained Pro AMD C21				
SAM D10 Xplained Mini SAM D10 Xplained Pro SAM D20 Xplained Pro SAMD21E16L interface with ATSAMBLDCHV-STK SAM W25 Xplained Pro SAMD21118A interface with ATSAMBLDCHV-STK SAM D21 Xplained Pro SAM D21				
SAM D11 Xplained Pro SAM D20 Xplained Pro SAMD21E16L interface with ATSAMBLDCHV-STK SAM W25 Xplained Pro SAMD21118A interface with ATSAMBLDCHV-STK SAM D21 Xplained Pro SAM D21				
SAM D20 Xplained Pro AMD21E16L interface with ATSAMBLDCHV-STK AMD21J1BA interface with ATSAMBLDCHV-STK AMD21J1BA interface with ATSAMBLDCHV-STK AMD21Xplained Pro AMD21Xplain				
SAMD21E16L Interface with ATSAMBLDCHV-STK SAM W25 Xplained Pro SAMD21J18A interface with ATSAMBLDCHV-STK SAM D21 Xplained Pro				
SAM W25 Xplained Pro SAMD21J18A interface with ATSAMBLDCHV-STK SAM D21 Xplained Pro SAM D21 Xplained Pro SAM D21 Xplained Pro SAM D21 Xplained Pro				
SAMD21/18A interface with ATSAMBLDCHV-STK SAM D21 Xplained Pro SAM D41 Xplained Pro SAM J41 Xplained Pro				
SAM D21 Xplained Pro SAM DA1 Xplained Pro SAM U21 Xplained Pro	SAMD21J18A interface with ATSAMBLDCHV-STK			
SAM DA1 Xplained Pro SAM U11 Xplained Pro				
CAMI 21 Velaiged Bro.				
SAM 221 Aplaned PTO				
SAM L22 Xplained Pro				
SAM R21 Xplained Pro				
SAM R30 Xplained Pro				

Atmel STARTでプロジェクトが作成されたら、下で示されるように計器盤(DASHBOARD)タブを開いてください。ATWINC1500単位 部用Wi-Fiミドルウェア ライブラリが既に関連するドライハ、依存性と共にプロジェクトへ追加されていること注目してください。Wi-Fiミドルウェア のより多くの情報については使用者手引き(User Guide)をクリックしてください。ここでATWINC1500に対する主なプログラミング参考 基準のATWINC1500ソフトウェア設計の手引きへのリンクを見つけるでしょう。



注: SAM L21 Xplained Pro基板を選ぶと、デバイスとしてそれ に実装されるATSAML21J18B(TQFP64)を自動的に選ぶ ことを確認することができます。

SELECTED DEVICE: ATSAML21J18B

GENERAL

Name	ATSAML21J18B
CPU	CORTEX-MOPLUS
Flash	264 KB
SRAM	40 KB
Package	TQFP64
Package Change packag	TQFP64

5. 次に、ソフトウェア構成部品追加(Add software component)をクリックして"ADC"を検索してください。その後、一覧からADCドライハ(AD C driver)を選んで構成部品追加(Add component(s))をクリックしてください。

	Atmel	START ATSAML	21J18B				🗲 Return To Fro	ont Page
		<pre>{ } view code</pre>	Ct	- 8	SAVE CONFIGUR/	ATION	B	EXPC
	MY SOFTW	ARE COMPONEN	ITS					
DASHBOARD	Application Middleware Driver	2		• •	dd software compo Add a new software (shortcut 'a') MY PROJECT	nent e component to t	he current project)
	MY SOFTWAR	E COMPONENTS						0
разнвоакр	Application Middleware Driver	ADD SOFTWARE COMPON	+ IENTS	Add software c	omponent		Show dependence Show system driver Show hardwar X	s (0) s (0) e (0)
š		Name		Description			Add	
MNIA	ĢCL	- O Drivers O ADC		Analog-to-dig	ital converter (ADC).		①	0
	SELECTED BO					Add compone Add select	nt(s) Cancel ted component(s) to the	e project

6. ADCトライハ・やWi-Fiミト・ルウェア ライブラリが今やプロジェクト計器盤に追加されます。トライハ・(Driver)引き落としメニューはトライハ・形式を選ぶことができます。ADC非同期(ADC_Async)任意選択をクリックしてください。



7. 使用者手引き(User guide)でのクリックは下で示されるように今やADC_Async API参照基準の文脈に於いてASF4 API参照基準手 引書のHTML版を開きます。

Contents Index Search	Atmet
ASF4 API Reference Manual	< → @
Contonto	ADC Asynchronous Driver
Contents ASF4 Introduction and Context Getting Started Software Architecture AC Driver Implementation and Design Convention AC Drivers ADC Drivers ADC Asynchronous Driver ADC Synchronous Driver Analog Glue Function CAN Drivers Calendar Drivers Camera Drivers Camera Drivers	 The driver <u>adc async read channel</u> function will attempt to read the required number of conversion results from a ringbuffer. It will return immediately (i.e. not block) if the requested number of data is not available. If data was not available, or fewer data than requested was available, this will be indicated in the function's return value. The asynchronous driver uses interrupts to communicate that ADC data is available, and the IRQ handler reads data from hardware and into ringbuffers in memory. These ringbuffers decouple the generation of data in the ADC with the timing of the application request for this data. In a way, the producer and consumer events are asynchronous to each other. The user can register a callback function to piggyback on the IRQ handler routine for communicating with the main application. Summary of the API's functional features The API provides functions to: Initialize and de-initialize driver and associated hardware Select single shot or free running conversion modes Configure major ADC properties such as resolution and reference source Hookup callback handlers on conversion done, error and monitor events Start ADC conversion
Cryptography Drivers	Read back conversion results
 DAC Drivers Delay Driver Digital Glue Logic Ethernet MAC Drivers Event System Driver External Bus Driver External IRQ Driver 	Summary of configuration options Below is a list of the main ADC parameters that can be configured in START. Many of these are used by the adc async init function when initializing the driver and underlying hardware. Most of the initial values can be overridden and changed runtime by calling appropriate API functions, such as adc async set resolution. • Selecting which ADC input channels to enable for positive and negative input • Which clock source and prescaler the ADC uses • Various aspects of Event control
 Flash Driver Frequency Meter Drivers Hash Algorithm Drivers Helper Drivers I2C Drivers 	 Single shot or free running conversion modes Sampling properties such as resolution, window mode and reference source Run in Standby or Debug mode Driver Implementation Description
I2S Controller Drivers MCI Drivers PAC Driver	Channel Map The ADC asynchronous driver use a channel map buffer to map the channel number of each enabled channel and the index of the descriptor for the channel.

Add software component

3.3. ソフトウェア構成部品の追加と構成設定の方法

3.3.1. ソフトウェア構成部品追加

1. Atmel STARTプロジェクトで計器盤(DASHBOARD)を選んでソフトウェア構成部品追加(Add software component)をクリックしてください。

2. "選別(Filter)"領域でUSARTを入力することによってUSARTトライハを追加し、その後に+符号をクリック、または行上をダブルクリックすることによってそれをあなたのプロジェクトに追加してください。

Ð

				s
0				
Δ.	N			
		2		
	~	з.		
		9	-	

助言: それをドライバ(Driver)一覧で探すこともできます。

次のように選択構成部品(SELECTED COMPONENTS)表示部で表示されたそれを得るでしょう。

Count Add/remove Remo	ve all
1 ⊝⊕ 🔳	D
Add component(s) Ca	ncel
	Count Add/remove Remo 1 ⊙⊕ I Add component(s) Ca

3. 今や構成部品追加(Add component(s))卸をクリックすることによってUSARTドライベの追加を完了することができます。それでUSART ドライベが応用に追加されます。

3.3.2. ソフトウェア構成部品の構成設定方法

ソフトウェア構成部品はプロジェクト作成後に計器盤(DASHBOARD)から追加や削除をすることができます。

ソフトウェア構成部品を構成設定するには計器盤(DASHBOARD)タブで私のソフトウェア構成部品(MY SOFTWARE COMPONENTS)へ行っ てください。構成部品上に(マウスを)浮かせることで他のソフトウェア構成部品に対する依存性の表示を与えます。構成部品のクリックはその ソフトウェア構成部品のェディタを開きます。ここでは以下を行うことができます。

- ・構成部品の削除と改名
- ・利用可能な場合に使用者手引きを開く
- ・利用可能な場合にドライバ、動作形態、実体の選択
- ・(利用可能な場合の)信号の選択と他の構成部品とで有り得る衝突の解決
- ・選んだソフトウェア構成部品に対するパラメータ指定の構成設定

基本的な構成設定は、どの周辺機能の実装でも、例えば、全てのADCハートウェア周辺機能やUSARTハートウェア周辺機能に対して共通する利用可能な共通機能やパラメータです。

基本動作:領域名、引き落とし任意選択、マウス浮かし文章を使い、組み込み開発者は適切な構成設定を直感的に選ぶことができるべきです。

高度な構成設定はハートウェア特有構成設定パラメータを固有機能や区別された機能で表します。この機能を使うことはコートを違う基本 構造に移すのにより多くの作業が必要されるかもしれないことを意味します。Microchip M0+のような共通MCU基盤では、例え高度な 構成設定でもD21とL21間で移植することができます。

MY SOFTWARE COMPO	NENTS		0
разнвоако разнавание разна разнавание разна разнавание разна ра разна разна ра разна разна разна разна разна разна разна ра разна разна ра ра разна ра ра разна ра разна ра ра разна ра ра ра ра ра ра ра ра ра ра ра ра ра		ISART_0	
OSC32KCTRL	MCLK 🔅	DMAC OSCO	TRL 🔅 GCLK 🔅
	U	ISART_0	8
	Universal Asynchronous receiver/transmitter (UART) communication in synchronous/b	llocking mode
S GENERAL	COMPONENT SETTINGS	SIG	SNALS
ਹੋਂ User guide	Driver: HAL:Driver:U	ISART_Sync V RX:	PA05 ~
Θ	Mode: UART	~тх:	PA04 ~
Rename component	Instance: SERCOM0	×	
m Remove component	CLOCKS		
uu	Slow: Generic cloc	k generator 3 (400 kHz) 🛛 🗸	
	Core: Generic cloc	k generator 0 (4 MHz) 🛛 🗸	
HAL DRIVER USART SYNC (114	RT) CONFIGURATION ON SERCOMO		
BASIC CONFIGURATION		ADVANCED CONFIGURATION	Enable: 🗸
Receive buffer enable:		Run in stand-by:	
Transmitt buffer enable:		Immediate Buffer Overflow	
Frame parity:	No parity	Start of Frame Detection Enable:	
Character Size:	8 bits	Collision Detection Enable:	
Stop Bit:	One stop bit	Operating Mode:	USART with internal clock
Baud rate:	9600	Sample Rate:	■ 16x arithmetic
		Sample Adjustment:	7-8-9 (3-4-5 8-bit over-sampling)
		Fractional Part:	0
		Data Order:	LSB is transmitted first
		Encoding Format:	No encoding
		Debug Stop Mode.	keep running V

3.4. 構成設定の保存と復元

計器盤(DASHBOARD)から構成設定保存(Save Configuration)をクリックすることによってあなたの構成設定を保存することができます。 ここでは構成設定ファイルをダウンロードすることができ、これは入口頁でプロジェクト復元任意選択を使うことによってアップロードすることがで きます。構成設定ファイルは生成されてダウンロードされる全てが一括で含まれます。同じ復元任意選択を使うことによって一括全体をアッ プロードすることもできます。

- 1. 新しいプロジェクトを作成してそれを構成設定してください。
- 2. 構成設定保存(Save Configuration) へ行ってください。ファイルに適切な名前を与えて構成設定ダウンロート(Download Configuration) をクリックしてください。

後の段階で、再構成設定を望む時は次のとおりです。

- 1. Atmel START入口頁へ行ってください。
- 2. 入口頁の既存プロジェクト読み込み(Load Existing Project)部分でファイルからプロジェクト読み込み(Load Project From File)釦をクリック してください。
- 3. 検索(Browse)をクリックして保存したあなたの構成設定を選んでください。選んだファイルを開く(Open Selected File)をクリックしてください。

プロジェクトが今や読み込まれ、(以前に)止めた所からまた始めることができます。

4. 構成設定画面

Atmel STARTはあなたの設計の様々な側面を準備するために柔軟な多数の構成設定画面を提供します。

- ・ PINMUX構成設定部: PINMUX構成設定部は割り当てられたプロジェクト ピンの概要だけでなく、GPIOピン構成設定も提供します。
- ・ クロック構成設定部: クロック体系は発振器と各種形式の供給元から成ります。 クロック構成設定を使うことにより、各供給元と発振器を構成設定し、前置分周器を設定して計算された出力周波数を見ることができます。
- ・事象システム構成設定部:事象システム構成設定部は事象発生部、チャネル、事象使用部を接続するための簡単なドラッグ&ドロップ機能を 提供します。
- ・QTouch構成設定部: QTouch構成部機能はQTouch釦の構成設定のために今やAtmel STARTに持ち込まれています。 摺動子と 輪がまもなく利用可能です。

4.1. クロック構成設定部

クロック体系は発振器と各種形式の供給元から成ります。クロック構成設定を使うことにより、各供給元と発振器を構成設定し、前置分周 器を設定して計算された出力周波数を見ることができます。

表示部は選んだ構成部品に対する正しいクロック元も選ばせます。

- ・発振器(OSCILLATORS)部分は選んだデバイスに対して利用可能な発振器を表示します。殆どのクロック元は固定周波数を持ち、一方でその他は直接か、または依存する入力信号を選ぶことによってのどちらかが指定されるかもしれません。発振器パラメータを構成 設定するには(歯車アイコンで)設定ダイアログを開いてください。
- ・供給元(SOURCE)部分は入力信号を選び、利用可能ならば逓倍器や分周器を変更することによってクロック周波数を構成設定させます。
- ・構成部品(COMPONENTS)部分は構成設定に追加される単位部実体を一覧にします。この表は入力信号を選んで結果の周波数 を見ることを許します。
- ・禁止されたクロックからの接続は点線を使うことによって表示されます。



全てのデバイスが供給元を許可/禁止するチェック枠を持っている訳ではありません。これらは(歯車アイコンで)設定(Settings)ダイアログを呼び出すことによって設定されなければなりません。チェック枠のない発振器は常に許可されます。

クロック構成設定部の使い方

- ・既定構成設定が提供されます。殆どの構成部品は一般クロック生成器0を使います。
- ・枠間のドラッグ&ドロップ、または望む入力を選ぶために設定ダイアログを開くことによって接続を設定します。
- ・各枠の歯車アイコンをクリックすることによって設定ダイアログを呼び出します。
- ・枠をクリックすることによって個別クロック経路を表示します。
- ・禁止アイコンをクリックすることによって禁止されたクロックとクロック経路を許可します。アイコンのツールチップ(マウス浮かし助言)はより多くの情報 を提供します。
- ・全てのクロック設定をリセットします。

許可と禁止

プロジェクト作成時、発振器の多くは初めに禁止されています。これらは左下角で空のチェック枠を持ちます。禁止された発振器を使う供 給元と構成部品は警告標示を持ちます。供給元を許可または禁止するには、利用可能ならばチェック枠を使うか、または設定メニューを 開いて適切なパラメータをチェックしてください。

発振器供給元入力

いくつかの発振器は入力として供給元生成器を持つかもしれません。これらは周波数が定義された制限外の場合に警告を表示します。供給元と提供する周波数を示すツールチップを表示するには文章上に(マウスを)浮かせてください。警告は、巡回依存性を作る、発振器が同じ発振器の入力である供給元への入力の場合にも表示されます。

複数クロック

いくつかの構成部品は複数のクロックを持ちます。これがその場合なら、ト・ラッグ&ト・ロップに使われるクロックはより暗いフォントを持ちます。切り替えるにはト・ラッグ&ト・ロップで使われるべきクロックの名前をクリックしてください。

クロック元は右のように構成部品設定(COMPONENT S

(FDPLL96M) Warning: Circular Dependency Input Frequency above: 2 MHz Frequency: 70311 MHz

USART(0)

Using: SERCOM0

Fractional Digital Phase Locked Loop

COMPONENT	SETTINGS	\otimes
Clock	Source	
Slow	Generic clock generator 4 (655.36 Hz)	
Core	Generic clock generator 6 (32.77 kHz)	~
-	Generic clock generator 0 (8 MHz)	
	Generic clock generator 1 (2 MHz)	
	Generic clock generator 2 (1.02 kHz)	
	Generic clock generator 3 (32.77 kHz)	
	Generic clock generator 4 (655.36 Hz)	
	Generic clock generator 5 (32.77 kHz)	e
	Generic clock generator 6 (32.77 kHz)	_
	Generic clock generator 7 (32.77 kHz)	

Core 32.77 kHz

Slow 960 kHz

ETTINGS)で設定することができます。

構成部品クロック元選択

クロック元は次のように構成部品エディタで選ぶこともできます。

COMPONE	NT SETTINGS	SIGNALS		
Driver:	HAL:Driver:USART Async	~.]	RX:	PA04 💙
Mode:	UART	~	TX:	PA06 ~
Instance:	SERCOM0	~		
CLOCKS				
Slow:	Generic clock generator 0	· · · · ·		
7.7.1.				

クロック設定

発振器と供給元の設定(CLOCK SETTINGS)ダイアログは選ばれた項目によって定義される設定を表示します。ツールチップは任意選択 での手助けを提供します。任意選択の編集を許可するには上部角の"Enable(許可)"チェック枠をチェックしてください。

CLOCK SETTINGS		\otimes				
EXTERNAL MULTIPURPOSE CRYSTAL OSCILLATOR CONFIGURATION						
EXTERNAL MULTIPURPOSE CRYSTAL OS	SCILLATOR (XOSC) CONTROL	_				
Enable:						
ON Demand:						
Run In Standby:						
Enable XTAL:						
Automatic Amplitude Control Enable:						
Gain of the Oscillator:	2Mhz V					
Start up time for the External Oscillator:	31 us ~]				
Clock Frequency:	400000 🗘					
USAGE						
This clock is not used by any component	is.					
		Close				

4.2. 事象システム構成設定部

事象システムは直接の周辺機能間通信と合図を許します。事象システムは発生部、チャネル、使用部から成ります。周辺機能の状態での変化が事象として参照されます。

(供給元と言われる)1つの周辺機能で起こる事象は違う周辺機能(行き先)で別の事象を起動するのに使うことができます。用語の発生部は供給元周辺機能の事象を示し、用語の使用部は行き先周辺機能の事象を示します。発生部と使用部は事象チャネル経由で接続されます。チャネル部分は特定の事象発生部供給元に対して利用可能なチャネルを構成設定することができます。

同期と非同期の2つの事象チャネルの形式があります。非同期事象制御は出力を直接修飾するための事象入力を許します。事象チャネルのより多くの詳細についてはハートウェア手引書を参照してください。この資料は同期チャネルを通してADC変換(事象使用部)を起動する計時器事象(事象発生部)の例で事象システムを説明します。



4.2.1. 概念と原則

代表的な例はADC変換を開始するための計時器(溢れまたは比較一致)事象を持つことです。この例では計時器(TCA)が供給元で ADCが作動部分です。両方は事象チャネル経由で接続されます。

- 1. ATtiny817を使って新しいプロジェクトを作成してください。
- 2. 事象システムト・ライハ・を追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。画面の左側に事象(E VENTS)と名付けられたタブが現れます。

BOARD	Application Middleware	(+ Add software	component			Show dependencies 🕊 Show system drivers 🖸
HSAD 🛃		Click "Add	software components" to add dri	vers and middleware to you	r project.	×	Show hardware Q
ХПИ		Filter					
INI		Name	Description			Add	
		+ 🏶 Middleware				^	
πř.	SELECTED BOAR	- O Drivers	10. 10. 10. 10. 10. 10. 10. 10. 10. 10.				
		O AC	Analog Comparator	Analog Comparator (AC).		(±)	
6	You are using a custom	are using a custom		Analog-to-digital converter (ADC).			
ř,				code.	(±)		
÷	SELECTED DEVIC	O DAC	Digital-to-analog converter (DAC).				
Ð		C Digital Glue Logic	Event system allows from the CPU	peripherals to interact wi	for external use. thout intervention	Ð	
-		O Flash	Flash controller to a	ccess the flash memory.		(+)	
	GENERAL Name	O 12C	Inter-Incegrated Circ for on-board low-sp control ers and peri	uit (I2C), two wire serial in eed bi-directional commu	terface usually use nication between	ed 🕒	1
2	CPU	O RTC	Real-time clock (RTC) is used to keep track of t	he current time.	(+)	1
4	Flash	O SPI	Serial Peripheral Int interface.	erface (SPI), synchronous s	serial communicati	on 🕀	1
-	Package	Ö Timer	Task timer functiona	lity using a timer counter	perioheral.	(+) *	1
È.	Order Code	SELECTED COMPONENTS					1
	Change package	Name	+	Count	Add/remove	Remove all	
		O Event System		1	⊝⊕	đ	1
							-
L .				ſ			

3. 事象(EVENTS)タブをクリックしてください。既定によってシステムトライハ・からポート(Port)とポート多重器(PORTMUX)のような発生部(GEN ERATORS)と使用部(USERS)が現れます。

4.2.2. 計時器溢れ事象でADC変換開始

この例では前に事象システムドライバだけが選ばれたと仮定されます。

1. 計時器トライバを追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。発生部と使用部としてTC A0が追加されます。溢れ事象は発生部として一覧にされます。



2. ADCト ライハ を追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。



ADCが使用部として追加されます。

3. TCA事象発生部(溢れ)を非同期事象チャネル0へ引き摺ることによって開始し、その後にこの事象チャネルをADCに引き摺ってみてください。これはADCに対して未だ許可されたSTARTEI(開始事象入力許可)を持たないために動きません。ADC構成設定歯車をクリックし、その後に"STARTEI"をチェックして閉じてください。今や特定事象使用部として"事象での変換起動(Trigger Conversion on Event)"がADC0に存在するのを見るでしょう。今や事象チャネルをこの事象に引き摺ることができます(注:ADC0は特定事象使用部が許されていない時に灰色表示にされるべきです)。

	Add software component	Reset 🄶 Undo 🦽 Redo
GENERATORS	CHANNELS	USERS
TCA9	Asynchronous Event Channel 0	ADC0
Timer/Counter AD overflow	ASVNCCH0: Asynchronous Multiplexer Channel	Trigger Conversion On Event
		PORTMUX
EVENT ACTIONS		
PC TRIGGER CONVERSION ON E	VENT	
STARTEI: Start Event Input En	able: 🗸	TCAD
ASYNCUSER1: Asynchronous Selection Ch 1 - ADC0:	User Off	m event input
PO		
		Close

4. 次に、TCA事象をADCと接続します。: タイマ/カウンタA0溢れ(Timer/Conter A0 overflow)をクリックして同期事象チャネル0(Synchronous Event Channel 0)チャネルに引き摺ってください。トラック後トロップの間、支援されるチャネルだけが強調表示されます。接続はTCAと事 象チャネル間に作られます。



5. 同期事象チャネル0(Synchronous Event Channel 0)をクリックしてADC0に引き摺ってください。 接続は事象チャネルと事象での変換起動 (Trigger Conversion On Event)間に作られます。



これは以下のコードに帰着します。

EVSYS.ASYNCUSER1 = EVSYS_ASYNCUSER1_SYNCCH0_gc /* 同期事象チャネル0 */;

EVSYS.SYNCCH0 = EVSYS_SYNCCH0_TCA0_OVF_LUNF_gc /* タイマ/カウンタA0溢れ */;

事象入力はadc.cで許可されます。

ADCO.EVCTRL = 1 << ADC_STARTEI_bp /* 開始事象入力許可:許可 */;

4.2.3. 事象システム構成設定部

事象システムは直接の周辺機能間通信と合図を許します。事象システムは発生部、チャネル、使用部から成ります。周辺機能の状態での変化が事象として参照されます。

(供給元と言われる)1つの周辺機能で起こる事象は違う周辺機能(行き先)で別の事象を起動するのに使うことができます。用語の発生部は供給元周辺機能の事象を示し、用語の使用部は行き先周辺機能の事象を示します。発生部と使用部は事象チャネル経由で接続されます。チャネル部分は特定の事象発生部供給元に対して利用可能なチャネルを構成設定することができます。

同期と非同期の2つの事象チャネルの形式があります。非同期事象制御は出力を直接修飾するための事象入力を許します。事象チャネルのより多くの詳細についてはハードウェア手引書を参照してください。この資料は同期チャネルを通してADC変換(事象使用部)を起動する計時器事象(事象発生部)の例で事象システムを説明します。

4.2.3.1. 事象システム構成設定部の使い方

代表的な例はADC変換を開始するための計時器(溢れまたは比較一致)事象を持つことです。この例では計時器(TCA)が供給元で ADCが作動部分です。両方は事象チャネル経由で接続されます。

1. ATtiny817を使って新しいプロジェクトを作成してください。

2. 事象システムトライハを追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。画面の左側に事象(E VENTS)と名付けられたタブが現れます。

Pilter SELECTED BOAR You are using a custom You are using a custom CRC ACC ADC Analog Comparator (AC). C AC ADC ADC ADC C AC Digital Courter (AC). C Digital Courter (AC). C Digital Courter (AC). C Digital Courter (DAC). C Plash Plash C Distribution Courter (DAC). C Distribution Courter (DAC). C Distribution Courter (DAC). C Distribution Courter (DAC). C Distribution Courte		Add	
SELECTED BOAR Name Description You are using a custom AC Analog Comparator (AC). You are using a custom AC Analog to-digital converter (ADC). SELECTED DEVIC CRC Cyclic redundancy check (CRC) error-detecting code SELECTED DEVIC CRC Cyclic redundancy check (CRC) error-detecting code GENERAL Digital Glue Logic Custom zable control logic module for internal or eigented from the CPU. GENERAL Flash Flash controller to access the flash memory. 12C Inter-integrated Circuit (12C), two wire serial interfa		Add	
SELECTED BOAR * Middleware - Orivers AC Analog Comparator (AC). ADC Analog to-digital converter (ADC). ADC ADC ADC Cyclic rodundancy check (CRC) error-detecting code OAC Digital Gue Logic Custom zable control logic module for internal or e SELECTED DEVIC Event System Event System		-	
SELECTED BOAR - O Drivers You are using a custom AC Analog Comparator (AC). You are using a custom O CRC Cyclic redundancy check (CRC) error-detecting code DAC Digital-o-analog converter (DAC). SELECTED DEVIC O Event System Event system custom throws peripherals to interact without from the CPU. GENERAL O Flash Flash Flash controller to access the flash memory.			
You are using a custom AC ADC Analog Comparator (AC). You are using a custom CRC Cyclic redundancy check (CRC) error-detecting code DAC Digital - o-analog converter (DAC). Digital Glue Logic Custom zable control logic module for internal or e Exerct System Event System Event System From the CPU. GENERAL Tash Flash Flash Flash Converter (Cac), wo wire serial interface Converter (Cac), wo wire serial interface 			
You are using a custom ADC Analog to digital converter (ADC), CRC Cyclic redundancy check (CRC) error-detecting code O DAC Digital o-analog converter (DAC), O Digital Glue Logic Custom zable control logic module for internal or e Vent system Event System Cyclic redundancy check (CRC) error-detecting code O DAC O Figstal Glue Logic Custom zable control logic module for internal or e Vent system Cyclic redundancy check (CRC) GENERAL O Flash Flash Flash Flash Custom conclusion control logic custom control logic module for internal or e Vent system Cyclic redundancy check (CRC) Vent system Vent system Cyclic redundancy check (CRC) Vent system Vent sys		(\pm)	-
You are using a custom CRC Cyclic redundancy check (CRC) error-detecting code O DAC Digital-o-analog converter (DAC). O Digital Glue Logic Custom zable control logic module for internal or error SELECTED DEVIC Event System GENERAL O Flash Plac inter-infegrated Circuit (I2C), two wire serial interface		(+)	
O DAC Digital- Digital- SELECTED DEVIC O Digital Glue Logic Custom zable control logic module for internal or e Selected Device SELECTED DEVIC © Fiesh Event system Event system GENERAL © Flash Flash controller to access the flash memory. 0 12C Inter-invegrated Circuit (12C), two wire serial interface	Cyclic redundancy check (CRC) error-detecting code.		
SELECTED DEVIC Digital Glue Logic Custom zable control logic module for internal or e SELECTED DEVIC Event System Event system allows peripherals to interact without from the CPU. GENERAL Flash Flash controller to access the flash memory. inter-integrated Circuit (I2C), two wire serial interface inter-integrated Circuit (I2C), two wire serial interface		(±)	
SELECTED DEVIC © Event System Event s stem allows peripherals to interact without from the CPU. GENERAL © Flash Flash controller to access the flash memory. inter-integrated Circuit (I2C), two wire serial interfaces inter-integrated Circuit (I2C), two wire serial interfaces	external use.	Ð	
GENERAL O IZC Inter-Ingegrated Circuit (IZC), two wire serial interfa	tintervention	Ð	
GENERAL Disconstruction of the serial inter-incegrated Circuit (12C), two wire serial interfa		Ð	
Name for on-hoard low-speed bi-directional communicate	ice usually used tion between	Đ	1
CPU O RTC Real-time dock (RTC) is used to keep track of the cu	urrent time.	(+)	1
Flash O SPI Serial Perioheral Interface (SPI), synchronous serial	d communication	0 (#)	1
SRAM interface.			1
Package Task timer functionality using a timer counter period	oheral.	(†) *	1
Order Code SELECTED COMPONENTS			1.
Change package Name Count A	hdd/romoup	Remove all	1
	000	Nerriove on	1
Vevent System	ΘÐ		1
	1		

3. 事象(EVENTS)タブをクリックしてください。既定によってシステムト・ライハ・からポート(Port)とポート多重器(PORTMUX)のような発生部(GEN ERATORS)と使用部(USERS)が現れます。

4.2.3.2. 計時器溢れ事象でADC変換開始

この例では前に事象システムトライバだけが選ばれたと仮定されます。

1. 計時器ドライバを追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。発生部と使用部としてTC A0が追加されます。溢れ事象は発生部として一覧にされます。

EV	ENT SYSTEM CONFIGURATOR		3
SHBOARD		Add software component	Reset 🄄 Undo 🏞 Redo
DAS	GENERATORS	CHANNELS	USERS
-A	TCA0	Asynchronous Event Channel 0 🔅	PORTMUX
Xn	Timer/Counter A0 overflow Timer/Counter A0 underflow high byte (split mode) Timer/Counter A0 compare 0	Asynchronous Event Channel 1	Event Output 0 Event Output 1 Event Output 2
PINM	Timer/Counter A0 compare 1 Timer/Counter A0 compare 2	Asynchronous Event Channel 2	
	PORTA	Asynchronous Event Channel 3	TCA0 Enable count on event input
	Port A pin 0	Synchronous Event Channel 0 👸	
C) CLOCKS	Port A pin 1 Port A pin 2 Port A pin 3 Port A pin 4 Port A pin 5	Synchronous Event Channel 1	
12	Port A pin 6 Port A pin 7		
EVENT	PORTB		
*	Port B pin 0		
	Port B pin 1		
	Port B pin 2 Port B pin 3		
	Port B pin 4		
	Port B pin 5		
	Port B pin 6		
	Port B pin 7		

2. ADCドライバを追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。

EV	ENT SYSTEM CONFIGURATOR		(?
BOARD		Add software component	Reset 🕽 Undo C Redo
DASH	GENERATORS	CHANNELS	USERS
國 (TCA0	Asynchronous Event Channel 0	ADCO
š	Timer/Counter A0 overflow Timer/Counter A0 underflow high byte (split mode) Timer/Counter A0 compare 0	Asynchronous Event Channel 1	\$
E PINM	Timer/Counter A0 compare 1 Timer/Counter A0 compare 2	Asynchronous Event Channel 2	PORTMUX Event Output 0
	PORTA	Asynchronous Event Channel 3	Event Output 1 Event Output 2
SKS	Port A pin 0 Port A pin 1	Synchronous Event Channel 0 🔅	TCA0
00 CLO	Port A pin 2 Port A pin 3 Port A pin 4 Port A pin 5 Port A pin 6	Synchronous Event Channel 1	\$
EVENTS	Port A pin 7		
*	Port B pin 0 Port B pin 1 Port B pin 2		
	Port B pin 3 Port B pin 4 Port B pin 5 Port B pin 6 Port B pin 7		

ADCが使用部として追加されます。

3. TCA事象発生部(溢れ)を非同期事象チャネル0へ引き摺ることによって開始し、その後にこの事象チャネルをADCに引き摺ってみてく ださい。これはADCに対して未だ許可されたSATRTEI(開始事象入力許可)を持たないために動きません。ADC構成設定歯車を クリックし、その後に"SATRTEI"をチェックして閉じてください。今や特定事象使用部として"事象での変換起動(Trigger Conversion on Event)"がADC0に存在するのを見るでしょう。今や事象チャネルをこの事象に引き摺ることができます(注:ADC0は特定事象使用 部が許されていない時に灰色表示にされるべきです)。

		Add software component	Reset 🍝 Undo 🦽 Redo
	GENERATORS	CHANNELS	USERS
	TCA0	Asynchronous Event Channel 0	ADC0
Times	Timer/Counter A0 overflow Counter A0 underflow high byte (split mode)	ASVNCCH0: Asynchronous Multiplexer Channel	Trigger Conversion On Event
	PO P	/ENT Ible: User Off	PORTMUX TCA0 n event input

4. 次に、TCA事象をADCと接続します。: タイマ/カウンタA0溢れ(Timer/Conter A0 overflow)をクリックして同期事象チャネル0(Synchronous Event Channel 0)チャネルに引き摺ってください。トラック後ト・ロップの間、支援されるチャネルだけが強調表示されます。接続はTCAと事 象チャネル間に作られます。



5. 同期事象チャネル0(Synchronous Event Channel 0)をクリックしてADC0に引き摺ってください。接続は事象チャネルと事象での変換起動 (Trigger Conversion On Event)間に作られます。



これは以下のコードに帰着します。

EVSYS. ASYNCUSER1 = EVSYS_ASYNCUSER1_SYNCCH0_gc /* 同期事象チャネル0 */ ;
EVSYS.SYNCCH0 = EVSYS_SYNCCH0_TCA0_OVF_LUNF_gc /* タイマ/カウンタA0溢れ */;
事象入力はadc.cで許可されます。
ADCO.EVCTRL = 1 << ADC_STARTEI_bp /* 開始事象入力許可:許可 */;

4.3. Atmel START QTouch[®]容量性感知ライブラリ

周辺機能接触制御器(PTC:Peripheral Touch Controller)は1~256チャネルの自律的感知を支援する容量性接触感知単位部です。PT Cは同じ応用で自己と相互の両容量性感知設計を支援し、これはシステム設計者に大きな柔軟性を提供します。自律動作のため、例 え多数キーの設計でも、PTCは非常に小さな資源と電力しか使いません。

鍵となる特徴

- ・ 釦、摺動子、輪を支援。外囲器に依存して256接触チャネルまで。
- ・同時に外部部品が必要とされなくても、相互と自己の容量性接触を支援
- 自己校正、調整不要
- ・低CPU負荷用自律的動作
- ・非常に低い電力消費
- ・厳しい緩衝での動作に対する素晴らしい伝導耐性(CI:Conducted Immunity)
- ・組み込み耐水性支援
- ・表面データのデータ流し
- ・塊と低電力を支援

QTouch[®]ライブラリは周辺機能接触制御器を持つAVR[®]とSAMのデバイスを使って接触応用を開発するための著作権使用料無料のソフト ウェアライブラリです。QTouch構成設定部はAtmel STARTに基づく接触例プロジェクト構築部で、同時にデータ可視器(Data Visualizer)が 感知器デバックゲータを可視化してQTouchライブラリパラメータ設定の調整を許します。この指導の以降の頁はQTouchライブラリ接触例プ ロジェクトで作業する時にQTouch構成設定部とデータ可視器ツールを使うことが必要とされる手順を網羅します。

4.3.1. QTouch[®]構成設定部

QTouch構成設定部は標準評価基盤用のQTouchプロジェクトをダウンロードするためだけでなく、使用者独自基板用のQTouchプロジェクトを生成するための任意選択も提供します。

- ・使用者基板接触プロジェクトを生成する手順
- ・2D接触表面と手ぶりのプロジェクトを生成する手順
- ・Microchip基板プロジェクトをダウンロートする手順

4.4. PINMUX構成設定部

PINMUX構成設定部は構成設定される全てのピンの概要を表します。

	Pin	label	Board	label		S	ignal	Show labels ~	① Zoom in	Zoom out Auto fit	
W T	Pad	User	Header	Pin	Mode	Label	Mode				
	RT						1			00 06	5
1	PA00				Digital	P/0				ADIN XDIN ZB	12
2	PA01				Digital	P/1			d.	N N N	1 (d)
3	PA02				Digital	P/2		POPT (P/0)	PAGE 1		R X
4	PA03				Digital	P/3		PORT (P/0)	PAUL		24 PA25
	software o	omponent	5					PORT (P/1)	PAUL		23 97424
5	PA04							PORT (P/2)	PAUZ		22 PA23
6	PA05						_	PORT (P/3)	PA03 4	~	21 PA22
7	PA06		1	1					PA04 5		20 PA19
8	PA07								PA05 6		19 PA18
9	VDDI								PA06 7	-	18 PA17
10	GND								PA07 8		17 PA16
11	PA08								0		
12	PA09								0	9410 9410 9408	0A14
13	PA10								100	001	
14	PA11										
10	0414										
Pin 3	(PA02) is u	sed as P/2	2 with PORT.		User label:			Pull configu	uration:	Off Y	
				3	Pin mode:	Digi	tal input	V			

ピン割り当て(Pin Assignment)表では以下を行うことができます。

- ・ピン番号、パッド、ソフトウェア構成部品、動作形態、信号の一覧
- ・並び順を変更するために列先頭部をクリック
- ・先頭部をクリックして列を選ぶことによる列の追加

・行をクリックすることによってピンを選択、クリック時に<Shift>または<Ctrl>のキーを押さえるこによってより多くを選択

- デバイス配置(Device Layout)では以下を行うことができます。
- ・ピン割り当て表で対応するパットを強調表示するためにピン上に(マウスを)浮かす
- ・ピン割り当て表での編集用に対応するパットを開くのためにピンをクリック
- ・デバイスに対するウェブに基づく情報をアクセスするためにデバイスの中央をクリック

ピン詳細エディタ(Pin Details Editor)はGPIOピンに対して特性を設定するのに使われます。ピン割り当て表で1つ以上のピンを選択する と、画面の下部でピン詳細エディタが開きます。ここで使用者標識を割り当て、ピン動作形態を選び、選んだピンの動作形態を構成設定 することができます。



助言:一覧でクリックする間に<Shift+Ctrl>キーボート、キーを押すことによって一度に複数ピンを選択することができます。これは一度 で複数ピンを構成設定することを許します。

選択したピンが違う形式の場合、それらは同じピン動作形態を支援しないかもしれません。選択したピンに対して共通の 設定だけを構成設定することができます。

4.5. CCL構成設定部

構成設定可能な注文論理回路(CCL:Configurable Custiom Logic)はデバイスピン、事象、または他の内部周辺機能に接続することが できる設定可能な論理回路周辺機能です。これはPCB上の簡単な接続論理回路機能用の論理回路ゲートを省き、論理回路動作は CPUの介入なしに設定可能な参照表(LUT:Lookup Table)と順次論理回路を使って実行することができます。

各LUTは個別に構成設定可能な3つの入力と真理値表、加えて任意選択の濾波器、同期部、端検出器から成ります。入力は入出 カピン、内部帰還、周辺機能、または事象システムからの事象のどれかにすることができます。各LUTは3つの入力から使用者設定可能 な論理式として出力を生成することができます。2つの隣接するLUTは順次論理回路部に接続され、単位部を形成します。任意選択 の順次論理回路はJKフリップフロップ、RSラッチ、Dラッチ、または門付きDフリップフロップのような複雑な波形生成を許すことができます。出力 は入出力ピン、周辺機能、事象システム、または対応する単位部への帰還として配線することができます。

CCL構成設定部はCCL単位部の図画的な末端間構成設定を支援します。

4.5.1. CCL構成設定部の使い方

以下の応用はCCLへの入力としての内部周辺機能と入出力ピンの使用を実演します。CCLは入力に基づく出力を生成するのにLUT と順次単位部を使います。下の例応用は低照度状態での動きを検出してピンに出力信号を提供します。信号は信号をOFFに切り替 えるためにスィッチが押されるまでそのピンで利用可能です。

- 1. ATtiny817を使って新しいプロジェクトを作成してください。
- 2. ソフトウェア構成部品追加(Add software component)をクリックしてデジタル接続論理回路トライハを追加してください。一旦デジタル接続 論理回路トライハ(Digital Glue Logic driver)が追加されると、画面の左側にCCLと名付けられたタフブが現れます。

1	MY SOFTWARE C	OMPONENTS		_			(?)
	Application Middleware Driver		+ Add software compor	ent			
		ADD SOFTWARE COMPONENTS	software components" to add drivers and n	nddleware to you	r project.	×	
		Filter					
		Name	Description			Add	
١.		- O Drivers					
	SELECTED BOAR	O AC	Analog Comparator (AC).			(
		O ADC	Analog-to-digital converter (A	DC).		Ð	
	You are using a custom	O CRC	Cyclic redundancy check (CRC) error-detecting	code.	\odot	
		O DAC	Digital-to-analog converter (D	AC).		٠	
		O Digital Glue Logic	Customizable control logic (Co use.	CL) module for in	iternal or external	Ð	
	SELECTED DEVIC	O Event System	Event system allows peripher from the CPU.	als to interact wi	thout intervention	۲	
	and the second se	O Flash	Flash controller to access the	flash memory.		•	
	Name	Ö I2C	Inter-Integrated Circuit (I2C), t for on-board low-speed bi-dir controllers and peripherals.	two wire serial in rectional commu	terface usually use nication between	d 📀	Ţ
	CPU	O PTC	Peripheral Touch Controller (PTC) for capacitiv	e touch	Ð	Ť
	Flash		measurement on sensors tha	t function as but	tons, sliders, and	~	1
	SRAM		wieets,			*	Ť.
	Package						1
	Order Code	SELECTED COMPONENTS			1.1.0	D	î.
	Chapee package	Name		Count	Add/remove	Remove all	1
	And the start and the start of the	O Digital Glue Logic		1	$\Theta \oplus$	m	1
							1
					\dd component(s)	Cancel	t
				_	1M Stars		
		DATA SHEET	ORDER SAMPLES		PURCHASE D	EVICE	

- 3. 再びソフトウェア構成部品追加(Add software component)をクリックしてACと事象システムのドライバを追加してください。
- 4. それの入力に基づいて望む(明るいまたは暗い)出力を提供するようにアナログ比較器単位部を構成設定してください。
- 5. それの入力に基づいて望む(検出した動き)出力を提供するように事象システム(EVSYS)単位部を構成設定してください。
- 6. CCLタブへ行って最初のLUT(LUT0)を許可してください。AC0 OUT入力供給元(ACO OUT input source)をLUT0の最初の入力 (A)にドラッグ&ドロップしてください。その後にLUT0事象0(LUT0 Event 0)をLUT0の2つ目の入力(B)にドラッグ&ドロップしてください。3 つ目の入力は切断されたままにしてください。

下図に於いて真理値表の行と列が選んだ入力に基づいて更新され、無効な行は禁止されます。

	CCL CONFIGURATOR		(?)
HBOARD	Enable CCL 🗸 🔅 Settings	Add software component Reset	D Undo C Redo
DAS	INPUT	LOOKUP TABLE	OUTPUT
B	EVSYS 🔷	A UTO 0x00	VO Pins 🔕
_	LUTO Event 0		PA4
Š	Courses ····	Custom	PB4
INNI			PAZ PC1
	AC0	Settings 🏠	Sinks 😭
	AC0 OUT input source	0.00	EVSYS 🔕
			Asynchronous Multiplexer Channel 0
CKS	IO Pins 🛇	B	Asynchronous Multiplexer Channel 1 Asynchronous Multiplexer Channel 2
CLO	PA0 PA1		Asynchronous Multiplexer Channel 3
\odot	PA2		
	PC3 PC4	· · · ·	
	PC5		
ರ			
Å			
		LUTO CONFIGURATION	
	Gate Type: Custom V	C B A VALUE Filter Options: Disable Filter	Y
2	Truth Logic:	0 0 0 Enable Edge Detector:	
VENT	Expression:	0 0 1 0 Sequential Logic - (lut0 & lut1):	
A E	O Value (hex):	0 1 1 0 Sequential Logic Options: Sequential logic disab	led ~
	-	1 0 0 0	
		1 0 1 0	
		1 1 0 0	

7. 以下の方法の1つによって真理値表を構成設定することができます。

7.1. 予め定義された論理回路ゲート型(AND, NAND, OR, NOR, XOR, XNOR)の1つを選択

7.2. 式を入力

ANDは'*'を使って表現されます。演算子が指定されない場合は(*)が仮定されます。

ORは'+'を使って表現されます。

XORは'^'を使って表現されます。

NOTは'!'を使って表現されます。

NANDは式 -!(a*b)を使うことによって表現されます。演算子が指定されない場合、NAND!(*)が仮定されます。

NORは式 - !(a+b)を使うことによって表現されます。

XNORは式 - !(a^b)を使うことによって表現されます。

複雑な組み合わせの論理回路は上で一覧にされた演算子を伴う式を使って作成することができます。式の部分を群にする ことによって演算子の優先順位を制御するのに括弧を使うことができます。例えば、a+((b[^]c*(!c)

7.3. 真理値論理回路に値を入力

真理値表の値(VALUE)列は個別ビットを設定することが許されます。

7.2.と7.3.の任意選択を使うためにケート型(Gate Type)で"独自(Custom)"が選ばれるべきです。真理値表は選んだ任意選択 に基づいて自動的に更新されます。

ここで考察されている例に基づき、"LUT0出力"は"低照度状態での動き"を表します。これを構成設定するために、下図で示されるように引き落とし枠からANDゲートを選んでください。真理値表がANDゲートを反映するように更新されることに留意してください。



8. PINMUX表示部を使ってPC3ピンをデジタル入力として構成設定してください。

9. 2つ目のLUT(LUT1)を選んで許可してPC3を"LUT1"の最初の入力(A)にドラッブ&ドロップしてください。PC3の信号は逐次回路へ 与えるように"LUT1出力"を利用可能にされるべきです。ゲート型(Gate Type)引き落とし一覧で"独自(Custom)"を選び、真理値論 理回路(Truth Logic)下の値(Value)任意選択を選んでください。真理値表格子内の値を交互切り替えするか、または次図で示さ れるように値(Value)領域に直接的に値を入力してください。

	CCL CONFIGURATOR		0
IBOARD	Enable CCL 🖌 🏟 Settings	Add software component Re	set 🕽 Undo C Redo
DASH	INPUT	LOOKUP TABLE	OUTPUT
	EVSYS 🙆	A Z LUTO 0x02	I/O Pins 🔕
MNNUX	LUT0 Event 0	Custom	PA4 PB4 PA7 PC1 Sinks 🔅
C) CLOCKS	UO Pins O PA0 PA1 PA2 PC3 PC4	B Custom C 2 Settings	Asynchronous Multiplexer Channel 1 Asynchronous Multiplexer Channel 1 Asynchronous Multiplexer Channel 2 Asynchronous Multiplexer Channel 3 Cinha
U C C	CS Sources		
		LUT1 CONFIGURATION	
	Gate Type: Custom	C B A VALUE Filter Options: Disable Filter	~
13	Truth Logic:	0 0 0 Enable Edge Detector:	
VEN	O Expression: Try a*b*c	Sequential Logic - (lut0 & lut1):	
а т	Value (hex):	0 1 1 0 Sequential Logic Options: Sequential logic	disabled 🗸
		1 0 0 0	
-		1 0 1 0	
		1 1 0 0	
		1 1 1 0	

10. JKフリップフロップを選んでください。"LUT0出力"がJ入力に配線され、"LUT1出力"がK入力に配線されます。 構成設定部はこれらの接続を示すために自動的に更新されます。

JKフリップフロップの出力をPB4ピンへ出力先変更してください。

低照度状況下で動きが検出される時は必ずPB4で論理1(High)が利用可能で、この信号はスィッチ(PC3)が押されるまでPB4で利用可能に留められます。



5. TrustZone支援

TrustZoneプロジェクト

ARMv8-M技術用TrustZoneはMicrochip SAM L11マイクロコントローラで許され、MCUデバイスでの信頼される資源と信頼されない資源間 のハートウェア強制された隔離を提供します。

TrustZone管理部構成部品はCortex-M23コアに基づくSAM L11 MCUの広域的なTrustZone設定を構成設定する方法を提供します。

ARMv8-Mに基づくプロジェクトには次のように2つのTrustZone形式があります。

- ・安全プロジェクト(Secure project): 全域的なTrustZone設定へのアクセスを得るためにプロジェクトにTrustZone管理部構成部品が追加されなければなりません。
- ・非安全プロジェクト(Non-Secure project): 安全でないプロジェクトはTrustZone全域設定に対して安全なプロジェクトに依存するため、プロ ジェクトにTrustZone管理部構成部品を追加しないでください。

5.1. 安全プロジェクトの作成

1. SAM L11デバイスで新しいプロジェクトを作成してください。

2. TrustZone管理部ミドルウェアを追加するためにソフトウェア構成部品追加(Add software component)をクリックしてください。



3. 一旦ミドルウェアが追加されると、計器盤(DASHBOARD)へ行ってプロジェクト構成部品を選んでください。プロジェクト構成部品下にAtm el Studio用ツールチェーン設定(TOOLCHAIN SETTINGS)が表示されます。IARとKeil用のツールチェーン設定はこのUIで提供されず、使用者は各々のIDEでそれらの設定を手動で選ぶことが必要です。

	MY SOFTWARE COMPONENTS	(?
DASHBOARD	Application • Add software component Middleware • Add software component Driver • System driver Click "Add software components" to add drivers and middleware to your project.	Show system drivers ඟ 🌰
FINMUX	MY PROJECT	
crocks	GCLK (). MCLK (). OSC32KCTRL (). OSCCTRL ().	DMAC 🔅
Θ	GENERAL	
	TOOLCHAIN SETTINGS	
	COMPILER CONFIGURATION LINKER CONFIGURATION Miscellaneous Other Flags: Image: Configuration of the state of the stat	(bsecure_gateway_veneer.lib

- 4. コンパイラその他のフラグ(Miscellaneous Other Flags)は'-std=gnu99 -mcmse'設定を入れられます。'-mcmse'フラグはCMSE安全実 行可能ファイル作成をコンパイラに許可するのに使われます。
- 5. リンカその他のフラグ(Miscellaneous Linker Flags)は、-Wl,--out-implib=libsecure_gateway_veneer.lib -Wl,--cmse-implib、設定を入 れられます。、-Wl,--cmse-implib、はインボートライブラリで安全交換部被覆層(Secure gateway veneer)とシンボルの生成用です。安全 交換部ライブラリの生成を望まない場合、この設定を削除して文書枠を空に保ってください。
- 6. TrustZone管理部ミドルウェア ソフトウェア構成部品を選んで各種メモリの安全と非安全の大きさ、または周辺機能の安全性属性のような 設定を構成設定してください。
- 7. これらの設定についてより多くを知るには各設定近くの疑問符上に(マウスを)浮かせてください。

TRUSTZONE MANAGER CONEL	GURATION			C
USER ROW (UROW)		BOOT CONFIGURATION ROW (BOCOR)	0
Enable UROW fuse programming:	0 Enable	Enable BOCOR fuse programming:	Disable	6
USER ROW (UROW) - TRUSTZONE R	ELATED	BOCOR_WORD_0.IDAU_BOOTPROT: 6	0x0	hexiv
USER_WORD_2.IDAU_AS:	Oxff	PORT SECURITY ATTRIBUTION		
USER_WORD_2.IDAU_ANSC:	USER_WORD_2.IDAU_AS	PA00:	Secure I/O Pin	~
USER_WORD_2.IDAU_DS:	Flash Application Secure Size =	Nex v PA01:	Secure I/O Pin	~
USER_WORD_2.IDAU_RS:	© 0x7f	PA02:	Secure I/O Pin	~
USER_WORD_4.NONSECA.PM:	0	PA03:	Secure I/O Pin	v
USER_WORD_4.NONSECA.MCLK:	0	PAD4:	Secure I/O Pin	v
USER_WORD_4.NONSECA.RSTC:	0	PA05:	Secure I/O Pin	v
USER_WORD_4.NONSECA.OSCCTRL	. 0	PAD6:	Secure I/O Pin	v.
USER WORD 4.NONSECA.OSC32KC	O TRL:	PA07:	Secure I/O Pin	~
USER_WORD_4.NONSECA.SUPC:	0	PA08:	Secure I/O Pin	~
USER_WORD_4.NONSECA.GCLK:	0	PA09:	Secure I/O Pin	~
USER_WORD_4.NONSECA.WDT:	0	PA10:	Secure I/O Pin	~
USER_WORD_4.NONSECA.RTC:	0	PA11:	Secure I/O Pin	Ý
USER_WORD_4.NONSECA.EIC:	0	PA12	Secure I/O Pin	~
USER_WORD_4.NONSECA.FREQM:	0	P413	Secure I/O Pin	101
USER_WORD_4.NONSECA.PORT:		Pa14-	Secure I/O Pin	
USER_WORD_4.NUNSECA.AC!			Secure I/O Pill	~

8. 安全プロジェクトは上の構成設定に基づいて2つのリンカ スクリプト ファイルを作成し、1つは安全プロジェクトで使うためで、他は後で作成される非安全プロジェクトで使うため用です。

5.2. 非安全プロジェクトの作成

1. TrustZone管理部ミドルウェアを追加せずにSAM L11デバイスを使ってプロジェクトを作成してください。非安全プロジェクトに対してツール チェーン設定は表示されません。

5.3. Atmel Studioでの安全プロジェクトのインポート

- 1. 「6.1.2. Atmel StudiolこAtmel START出力をインホートする手順」で記述される手順を使ってAtmel Studio IDEに安全プロジェクトのAT ZIPをインホートしてください。
- 2. 解決策ェクスプローラ(Solution explorer)でプロジェクト名を右クリックすることによってプロジェクトプロハプティ(Project Properties)へ行ってツー チェーン(Toolchain)を選んでください。コンハプラ構成設定(ARM/GNU Compiler)下でその他設定(Miscellaneous)へ行ってSTARTからのコンハプラ設定が更新されているのを見てください。

SecureProject + × Build	
Build Events	Configuration: Active (Debug) Platform: Active (ARM)
Toolchain	Configuration Manager
Device	
Tool	ARM/GNU Common
Packs	Output Files Other flags: -std=gnu99 -mcmse
Advanced	General Verbose (-v)
	 Verbal (V) Preprocessor Symbols Directories Optimization Debugging Miscellaneous ARM/(SNU Linker General Debugging Miscellaneous ARM/(SNU Assembler General Debugging ARM/(SNU Preprocessing Assemble General Debugging ARM/(SNU Preprocessing Assemble General Miscellaneous

3. リンカ構成設定(ARM/GNU Linker)下でその他設定(Miscellaneous)へ行って安全なリンカ スクリプト パスと共にSTARTからのリンカ構成 設定が更新されているのを見てください。

Configuration: Active (Debug)	▼ Platform: Activ	re (ARM)
Configuration Manager		
ARM/GNU Common	ARM/GNU Linker 🔿 Mi	scellaneous
General Output Files	Linker Flags:	-WI,out-implib=libsecure_gateway_veneer.lib -WI,cmse-implib -T"/Device_Startup/saml11_secure.ld"
General Preprocessor	Other options (-Xlinke	er [option])
Symbols Directories		
Debugging Warnings		
ARM/GNU Linker	Other objects	
General Libraries	Other objects	
Memory Settings		
ARM/GNU Assembler		
Debugging ARM/GNU Preprocessing Asser	- F.C.	
Symbols		
4 [] >		

5.4. Atmel Studioでの非安全プロジェクトのインポート

- 1. 非安全プロジェクトは新しい解決策(Solution)を作成するか、または存在するならば安全プロジェクト解決策にそれを直接インポートする かのどちらかによってインポートすることができます。
- 2. 安全プロジェクト解決策に直接インポートするには、「6.1.2. Atmel StudiolこAtmel START出力をインポートする手順」で記述されるように 安全プロジェクト解決策を開いて解決策に非安全プロジェクトをインポートしてください。
- 3. 非安全プロジェクトをAtmel Studioにインボートしている間に使用者はインボート処理を完了するために被覆層(Veneer)ヘッダ ファイル、安 全プロジェクト エクスボート ライフ・ラリ、リンカ スクリプト ファイルを検索することを求められます。予めコンハーイルされた安全プロジェクトからこれらの ファイルを得ることができます。
- 4. ファイル選択中に使用者は以下の2つの任意選択を持ちます
- リンクとして追加: プロジェクトにファイルを複写する代わりにファイルが検索されたパスへのリンクを追加します。普通、単一の解決策に安全と非安全の両プロジェクトを組み込む場合に選ばれます。
- **ファイルとして追加**: プロジェクトにファイルを複写して追加します。通常、上で必要とされたファイルが別の使用者供給元によって与えられ、使用者側で安全プロジェクトが利用可能でない場合に使われます。
- 5. インポートを完了するためにファイルを検索してOKをクリックしてください。

onSecureProject Importer			
Import Atmel Start I	Project	PPPPPPPPPPPPP	
Atmel Start Project(.atzip):	E:\Studio\NonSecureProject.atzip	F	Browse
View project summary (CMSIS	package information)		
Project Name:	NonSecureProject1		
Location:	E:\Studio\SecureProjectSAML11SoIn		Browse
Solution:	Add To Solution	•	
Solution Name:	SecureProjectSAML11SoIn		
View project import summary			
cure Project Veneer Header File	E:\Studio\SecureProjectSAML11SoIn\SecureProject\trustzone\trustzone_veneer.h	Add As Link 🔻	Browse
cure Project Export Library:	$\label{eq:expectation} E:\Studio\SecureProjectSAML11Soln\SecureProject\Debug\libsecure_gateway_veneer.lib$	Add As Link 🔻	Browse
on-Secure Project Linker Script:	$\label{eq:constraint} E: \times the second second$	Add As Link 🔻	Browse
Device Pack Manager	Help	ОК	Cancel

6. 解決策ェクスプローラ(Solution explorer)でプロジェクト名を右クリックすることによってプロジェクトプロハティ(Project Properties)へ行ってツー チェーン(Toolchain)を選んでください。コンハイラ構成設定(ARM/GNU C Compiler)下でディレクトリ設定(Directories)へ行ってインクルート パスが更新されているのを見てください。



7. リンカ構成設定(ARM/GNU Linker)下でライブラリ(Libraries)へ行って検索したライブラリとそれのパスが更新されているのを見てくださ

 γ°

NonSecureProject1 +> Build Build Events	X Atmel START SecureProject Configuration: Active (Debug)	▼ Platform: Active (ARM) ▼
Toolchain Device	Configuration Manager	
Tool	ARM/GNU Common	ARM/GNU Linker Libraries
Packs	Output Files	Libraries (-I)
Advanced	General General Symbols Directories Coptimization Cobeugging	libsecure_gateway_veneer.lib libm
	Warnings Miscellaneous	Library search path (-L)
	 ARM/GNU Linker General Libraries Optimization Memory Settings Miscellaneous ARM/GNU Assemblar 	\\SecureProject\Debug \${ProjectDir}\Device_Startup
	 AKM/GNU Assembler General Debugging ARM/GNU Preprocessing Assemble General Symbols Debugging ARM/GNU Archiver General 	
	۲. III. ا	

8. その他設定(Miscellaneous)へ行って検索したリンカスクリプトパスが更新されているのを見てください。

oolchain Device Tool Jacks	Configuration Manager Configuration Manager ARM/GNU Common C General]		10
ool Packs	 ARM/GNU Common General 			
ldvanced	Output Files ARM/GNU C Compiler General Preprocessor Symbols Directories	ARM/GNU Linker Linker Flags: Other options (-Xlin	Aiscellaneous -T"//SecureProject/Device_Startup/saml1 ker [option])	1_nonsecure.ld"
	 Optimization Debugging Warnings Miscellaneous ARM/GNU Linker General Libraries Optimization Memory Settings Miscellaneous 	Other objects	A 1	
	 ARM/GNU Assembler General Debugging ARM/GNU Preprocessing Assemble General Symbols Debugging ARM/GNU Archiver General 			

助言: 非安全プロジェクトは新しい解決策の作成によってインポートすることもできます。一旦解決策が作成されると、下で示される ように、追加(Add)⇒既存プロジェクト(Existing Project)を使って安全プロジェクトに追加することができます。

Search Solution Explorer (Ctrl+;) > Search Solution Explorer (Ctrl+;) > Rebuild Solution F7 Rebuild Solution Ctrl+Alt+F7 Clean Solution Ctrl+Alt+F7 Clean Solution Ctrl+Alt+F7 Configuration Manager Config Copy Full Path examples Collapse hal New Project Add Existing Project Set StartUp Projects Existing Project Set StartUp Projects Existing Item Shift+Alt+A New Solution Folder Paste Veroperties Ctrl+V Properties Copen Folder in File Explorer Properties Close Solution	Search Solution Explorer (Ctrl+:): Search Solution Explorer (Ctrl+:): Build Solution F7 Rebuild Solution Ctrl+Alt+F7 Dependencies Output Files Batch Build Configuration Manager Copy Full Path Collapse hal hpl hri Collapse Add Existing Project Example Project Ctrl+Shift+E	. م
New Project Add Point Point New Project Add Point Point New Project Add Point Point New Project Set StartUp Projects Point Point Existing Project Ctrl+Shift+A Point Point Point New Rem Ctrl+Shift+A Point Point Point Point New Solution Folder Point Point <td< th=""><th>New Project Ctrl + Shift+E Example Project Ctrl + Shift+E</th><th></th></td<>	New Project Ctrl + Shift+E Example Project Ctrl + Shift+E	
New Project Add hpl Existing Project Add toolchain Existing Project Set StartUp Projects trustzone Existing Project Ctrl+Shift+E Export Solution as Extension atmel_start.c New Item Ctrl+Shift+A Paste Ctrl+V atmel_start.d Existing Item Shift+Alt+A Rename F2 driver_init.c Vew Solution Folder Vopen Folder in File Explorer main.c main.c	Image: New Project New Solution Explorer View hpl New Project Add toolchain Existing Project toolchain trustzone Image: Description of the project Ctrl+Shift+E Export Solution as Extension	
New Project Add toolchain Existing Project Set StartUp Projects toulchain Example Project Ctrl+Shift+E Export Solution as Extension atmel_start.c New Item Ctrl+Shift+A Paste Ctrl+V atmel_start.pins.h Existing Item Shift+Alt+A Rename F2 driver_init.c New Solution Folder Open Folder in File Explorer main.c	New Project Add toolchain Existing Project Set StartUp Projects trustzone Example Project Ctrl+Shift+E Export Solution as Extension atmel_start.c	
Example Project Ctrl+Shift+E Export Solution as Extension atme_start.h New Item Ctrl+Shift+A Paste Ctrl+V atme_start.h Existing Item Shift+Alt+A Rename F2 driver_init.c New Solution Folder Open Folder in File Explorer driver_init.h main.c Properties Close Solution Close Solution driver_init.h	Example Project Ctrl+Shift+E Export Solution as Extension	
Properties Close Solution Main.c	Image: Solution Folder Ctrl+Shift+A Paste Ctrl+V atme_start.n Image: Solution Folder Shift+Alt+A Image: Solution Folder F2 driver_init.c Image: Solution Folder Open Folder in File Explorer driver_init.h driver_init.h	
	Properties main.c	

5.5. Keil[®]からのµVision[®]での安全プロジェクトのインポート

- 1. 同じ作業空間で安全と非安全の両プロジェクトをインポートするには、
 - フォルダを作成して同じフォルダに安全と非安全のプロジェクトを解凍(unzip)してください。
 - 混乱を避けるために両ファイルで*.gpdscファイルを改名してください(例えば、AtmelStart_S.gpdscとAtmelStart_NS.gpdsc)。
 - これらの*.gpdscファイルを個別に選ぶことによってkeilプロジェクトを作成してください。
 - 1.1. ファイル(File)⇒開く(Open)⇒安全プロジェクト フォルダ⇒(全ファイル形式を表示(Show All File Types)選択後に)AtmelStart_S.gpd scを選択で、プロジェクトが作成されます。プロジェクトを保存して閉じてください。
 - 1.2. 非安全プロジェクトに対して上の手順を繰り返してください。
 - Keil MDKで複数プロジェクト作業空間を作成して両プロジェクトファイル(安全用と非安全用の.uvprojxファイル)を追加してください。
 - 下で記述されるように安全なプロジェクトを構成設定してください。
- 作業空間でプロジェクト名を右クリックすることによって任意選択(Options)へ行ってください。目的対象(Target)タブ下でコート、生成(Cod e Generation)部分へ行ってARMコンハペイラ(ARM Compiler)が"既定コンハペイラ6版を使用(Use default compiler version 6)"であるのを 確実にし、ソフトウェア様式(Software Model)を安全プロジェクト用の"安全形態(Secure Mode)"に変更して"MicroLIB使用(Use MicroLI B)"チェック枠を選んでください。

evice	arget C	output Listing	User C/C	C++ (AC6) A	sm Lin	iker Deb	oug Utilities		
Microchip	ATSAML	.11E16A	Xtal (MHz):	12.0	-Code C ARM	Generation Compiler:	Use default	compiler version	on 6 💌
Operation	evetem:	Nana	=	-	Softw	rare Model	: Secure Mod	le	-
System V	iewer File	Tractic		<u> </u>	ν υ	se MicroL	ів Г	Big Endian	
ATSAMI	_11E16A.	svd			-			and and a	
- Read/(default	Only Mem off-chip ROM1:	ory Areas	Size	Startup	Read/ default	Write Men off-chip RAM1:	ory Areas Start	Size	Nolnit
F	ROM3:		-	- c	—	RAM3:			Г
•	on-chip IROM1:	0x0	0x10000	e	2	on-chip IRAM1:	0x20000000	0x4000	Г
Г	IROM2:	0x400000	0x800	0	Г	IRAM2:			

3. C/C++タブへ行ってください。言語/コート'生成(Language/Code Generation)部分下でC言語(Language C)を"C99"に変更してください。

Device Target Output Listing O	ser <u>C/C++ (AC6)</u> Asm Linker Debug Utilities	
Preprocessor Symbols		
Define:		
U <u>n</u> define:		_
- Language / Code Generation		
Execute-only Code	Wamings: AC5-like Wamings 💌 Language C: c99	-
Optimization: -00	Tum Warnings into Errors Language C++: C++98	•
Link-Time Optimization	☐ Plain Char is Signed	ar
Split Load and Store Multiple	🔲 Read-Only Position Independent 🛛 🗍 use RTTI	
✓ One ELF Section per Function	Read-Write Position Independent IN No Auto Includes	
Include		
Paths		
Misc Controls		
Compiler -xc -std=c99 -target=a	am-am-none-eabi-mcpu=cortex-m23-c	-
control fno-rtti funsigned-cha	ar fshort-enums fshort-wchar	-

注: 警告に対する値を"AC-5様の警告(AC-5 like Warnings)"として選ぶのは任意選択です。

4. リンカ(Linker)タブ下で"目的対象ダイアログ ボックスからメモリ配置を使用(Use Memory Layout from Target Dialog)"のチェックを外して安全プロジェクトから分散ファイル(Scatter File)を検索してください。

vice Targ	et Output Listing User C/C++ (AC6)	Asm Linker De	bug Utilities	
Use Mer Make Make Don Repo	nory Layout from Target Dialog e RW Sections Position Independent e RO Sections Position Independent : Search Standard Libraries ort 'might fail' Conditions as Errors	<u>X</u> /O Base: <u>R</u> /O Base: R/ <u>W</u> Base <u>d</u> isable Warnings:	0x00000000 0x20000000	
Scatter File	\toolchain\keil\linker\saml11_secure.sct		_]	Edit
Scatter File <u>M</u> isc controls	.\toolchain\keil\linker\saml11_secure.sct		.	Edit
Scatter File <u>Misc</u> controls Linker control string	.\toolchain\keil\linker\saml11_secure.sct -cpu=Cortex-M23 *.o -library_type=microlib -strict -scatter ".\tool	lchain\keil\linker\saml'	I1_secure.sct"	Edit

5. デバッグ(Debug)タブ下で使用(Use)ラジオ釦に対してCMSIS-DAP値を選んでください。設定(Settings)をクリックして一括(Pack)タブへ 行ってください。Toolchain¥Keil¥debugConfig¥SAMl11.dbgconfで利用可能な新しいUROW設定で既定dbgconfファイルを変更して ください。

Debug.allni 1 LOAD "\\TZ-MixSecureEIC-S\\UVBuild\\AtmelStart_S.ax 2 LOAD "\\TZ-MixSecureEIC-NS\\UVBuild\\AtmelStart_NS.d 3 RESET 4 5 g, main	f" incremental axf" incremental
Options for Target 'ATSAMLI1E16A' Device Target Output Listing User C/C++ (AC6) Aem Linker Debug Ukites Use Smulator with methodoons Settings Use: [CMSIS-DAP ARM/v8-M Debugg • Settings Umit Speed to Real-Time	CMSIS-DAP ARM:6-M Target Driver Setup
Iver Duck Application at Startup Iver Number of Application at Startup Iver Number of Application at Startup Instaltation File: Eduction at Startup Iver Number of Application at Startup Restore Debug Session Settings Iver Debug Session Settings Iver Debug Session Settings Iver Debug Session Settings Iver Debug Session Settings Iver Duck Performance Analyzer Iver DLL: Parameter: Dalog DLL: Parameter: Dalog DLL: Parameter: IVEM of outdated Executable is loaded Iver Duck Manage Component Vewer Deception Files	CMSISDAP - JTAG/SW Adapter Serial No: [ATML3138011800] Firmware Version: (33 25 0186 IF SWJ Port: SW • Max Cock: 10MHz • Debug Connect & Reset Options Connect: (under Reset • Reset: SYSRESETIREQ • Cache Options Cache Code IF Cache Code IF Connect & Reset Options Connect & Reset Options Cache Code IF Cache Code IF Cache Code IF Cache Code IF Cache Memory Cache Memory IF Cache M
OK Cancel Defaults Hep	OK Cancel Heb

- 6. 加えて、安全プロジェクトに対する任意選択(Options)でデベック(Debug)タブへ行って初期化ファイルの内容を次のように編集してください。
 - LOAD "Specify the Path of .axf file of non-secure project" incremental
 - LOAD "Specify the Path of .axf file of secure project" incremental
 - RESET
 - g, main
- 7. ユーティリティ(Utillities)タブ下でフラッシュ構成設定(Configuration Flash)メニューへ行って"フラッシュ プログラミング用使用者目的対象トライハ (User Target Driver for Flash Proguramming)"下のLoad_ns.iniとしてInitファイルを検索してください。Initファイルの内容を次のように 編集してください。
 - LOAD "...¥¥Specify Non-secure project path¥¥UVBuild¥AtmelStart_NS.axf" incremental
- 8. 安全プロジェクトをコンパイルしてください。
- 9. 非安全プロジェクトの構成設定で記述されるように非安全プロジェクトを構成設定してください。

5.6. Keil[®]からのµVision[®]での非安全プロジェクトのインポート

- 1.「Keil[®]からのµVision[®]での安全プロジェクトのインホート」で記述されるように複数プロジェクト作業空間を作成してください。
- 作業空間でプジェクト名を右クリックすることによって任意選択(Options)へ行ってください。目的対象(Target)タブ下でコート、生成(Code Generation)部分へ行ってARMコンハ。イラ(ARM Compiler)が"既定コンハ。イラ6版を使用(Use default compiler version 6)"であるのを確 実にし、ソフトウェア様式(Software Model)を安全プロジェクト用の"非安全形態(Non-Secure Mode)"に変更して"MicroLIB使用(Use Mi croLIB)"チェック枠を選んでください。

/licrochic	ATSAML	11E16A			-Code C	Generation			
			Manager N	2.0	ARM	Compiler:	Use default	compiler versi	on 6 💌
				12.0	Softw	rare Model	Non-Secure	Mode	-
Operating	g system:	None		-					
System \	/iewer File:				U 🟹	se MicroLl	ВГ	🗌 Big Endian	
ATSAM	L11E16A.s	svd							
Use Use	Custom Fi	e							
-Read/	Only Mem	ory Areas			-Read/	Write Mem	iory Areas	2155C	
default	off-chip	Start	Size	Startup	default	off-chip	Start	Size	NoInit
Γ	ROM1:			C	Г	RAM1:			Г
Γ	ROM2:			С	Г	RAM2:			Г
Г	ROM3:		-	C		RAM3:		1	Г
	on-chip	*:	1			on-chip		2. 	
V	IROM1:	0x0	0x10000	œ	V	IRAM1:	0x20000000	0x4000	Г
		-	-		_				

3. C/C++タブへ行ってください。言語/コート'生成(Language/Code Generation)部分下でC言語(Language C)を"C99"に変更してください。 インクルート[・] パス(Include Paths)でTrustzone_venneer.hファイルのパスを検索してください。

Options for Target 'ATSAML11E16A'
Device Target Output Listing User C/C++ (AC6) Asm Linker Debug Utilities Preprocessor Symbols
Define:
Language / Code Generation Execute-only Code Optimization: -00 Image: Interview of the secure only Code Image: Imag
Include Paths Misc Controls Trustzone_venneer.hファイルのインクルート、ハ°ス
Compiler xc -std=c99 -target=arm-arm-none-eabi -mcpu=cortex-m23 -c control fino-tti -funsigned-char -fshort-enums -fshort-wchar
OK Cancel Defaults Help

注: 警告に対する値を"AC-5様の警告(AC-5 like Warnings)"として選ぶのは任意選択です。

4. 非安全プロジェクトに完全被覆層ライフブラリ(Workspace ¥UVBuild¥AtmelStar t_S_CMSE_Lib.o)ファイルを追加してください。



5. リンカ(Linker)タブ下で"目的対象ダイアログ ボックスからメモリ配置を使用(Use Memory Layout from Target Dialog)"のチェックを外して非 安全プロジェクトから分散ファイル(Scatter File)を検索してください。

Use Memory Layout from Target	-			
Make RW Sections Position	Dialog Independent ndependent ies	<u>X</u> /O Base: <u>R</u> /O Base: R/ <u>W</u> Base	0x00000000 0x20000000	
Scatter [\TZ-MixSecureEIC-S File	\toolchain\keil\linker\saml	11_nonsecure.scl	_	Edit
Misc controls				Ŷ

6. デハッグ(Debug)タブ下で使用(Use)ラジオ釦に対してCMSIS-DAPとして値を選んでください。設定(Settings)をクリックして一括(Pack)タ ブへ行ってください。Toolchain¥Keil¥debugConfig¥SAMl11.dbgconfで利用可能な新しいUROW設定で既定dbgconfファイルを変更 してください。

Options for	Target 'ATSAML11E16A'	
Device Target	Output Listing User C/C++ (AC6)	Asm Linker Debug Utilities
C Use Simulat	or <u>with restrictions</u> Settings to Real-Time	Setting
Load Applic	ation at Startup 🔽 Run to main()	Load Application at Startup Initialization File:
	Edit	Edit
Restore Debu	g Session Settings ints 🔽 Toolbox Windows & Performance Analyzer Display 🔽 System Viewer	Restore Debug Session Settings Breakpoints Toolbox Watch Windows Memory Display System Viewer
CPU DLL:	Parameter:	Driver DLL: Parameter:
		SARMV8M.DLL -MPU
Dialog DLL:	Parameter:	Dialog DLL: Parameter:
		TCM.DLL PCM23
_		

7. 構成設定(Configuration)ウィザートで非安全プロジェクトに対してチップ消去(Chip Erase)値をCE0に変更してください。

ATSAML11E16A_ATSAML11E16A.dbgconf					
Expand All Collapse All Help Show Grid	1				
Option	Value				
 Debugger Setup Chip Erase Chip Erase Select CE0 Erase Key CE1 Erase Key CE2 Erase Key CE2 Erase Key Debug Access Level Program User Configuration Row (UROW) Program Boot Configuration Row (BOCOR) 	CEO (Erase Non-Secure Application)				
Text Editor Configuration Wizard					

- 8. デバッグ(Debug)タブでDebug_ns.iniとして初期化ファイルを検索して以下の内容を追加することによって編集してください。
 - LOAD "Specify the Path of .axf file of non-secure project" incremental
 - RESET
 - g, main
- 9. 非安全プロジェクトをコンハプイルしてください。

5.7. IAR Embedded Workbenchでの安全プロジェクトのインポート

IAR Embedded Workbench支援はまもなく追加されます。

5.8. IAR Embedded Workbenchでの非安全プロジェクトのインポート

IAR Embedded Workbench支援はまもなく追加されます。

5.9. Makefileを使う安全プロジェクトのインポート

「6.4.2. GNU C Makefileの実行」で記述される手順に従ってMakefileを使ってプロジェクトを構築してください。

5.10. Makefileを使う非安全プロジェクトのインポート

- 1.「6.4.2. GNU C Makefileの実行」で記述される手順に従うことにより、Makefileを使ってプロジェクトを構築してください。
- 2. それは以下の警告メッセーシを表示します。
 - 2.1. SAM L11非安全プロジェクトは安全プロジェクトからの入力ファイルが必要です。ここは次のとおりの手順例です。
 - 2.1.1. 安全プロジェクトから'Device_Startup¥saml11_nonsecure.ld'、'trustzone¥trustzone_veneer.h'、'gcc¥libsecure_gateway_ veneer.lib'のファイルを得てください。
 - 2.1.2. 'saml11_nonsecure.ld'を非安全プロジェクトの'gcc¥gcc'フォルタ」に複写してください。
 - 2.1.3. 'trustzone_veneer.h'と'libsecure_gateway_veneer.lib'を非安全プロジェクトのルートフォルダに複写してください。
 - 2.1.4. 新しいリンカ スクリプトの -T"../gcc/gcc/saml11_nonsecure.ld"を使うようにMakefileを更新してください。
 - **2.1.5**. 安全交換ライブラリをインクルードするよう、-W1, --start-group -lm -1:libsecure_gateway_veneer.lib -L"... /" -W1, --end-group でmakefileを更新してください。
- 3. 警告メッセージで記述される手順に従ってください。

6. 外部ツールでのAtmel START出力の使用

Atmel STARTであたなたのプロジェクトが構成設定されると、Cコンハペイラでそれを使うためにそれをエクスホートしなければなりません。 関連リンク 2.2. 支援するIDEとコンハペイラ

6.1. Atmel Studio

以下の段階はAtmel StudioでAtmel STARTからの出力を使う方法を記述します。

6.1.1. Atmel STARTからプロシェクトをエクスホート

- 1. Atmel STARTウェブサイトで(例または基板の)新しいプロジェクトを作成してください。
- 2. ソフトウェア構成部品ェクスポート(Export Software Componen t)釦をクリックしてください。Atmel StudioとSomonium IDE 用のプロジェクトをェクスポートするためにAtmel Studioを選 んでください。



3. 一括ダウンロード(DOWNLOAD PACK)をクリックしてください。'atmelstart.atzip'一括ファイルがダウンロードされます。

6.1.2. Atmel START出力をAtmel Studioにインホートする手順

- 1. Atmel Studioを開始してください。
- 2. ファイル(File)⇒インホート(Import)⇒Atmel STARTプロジェクト(Atmel Start Project)を選んでください。

*	AtmelStudio (Administrato	r)				
File	Edit View ASF Project	Debug	Tools	Wi	ndow Help	
	New			9	- C - 🔝 🔍 🕨 🕅	1
	Open		•	ex 7	8 🖸 - 🛯 📾 🛱	
	Close					
23	Close Solution					
	Import		•		AVR32 Studio Project	Ctrl+3
	Save Selected Items	Ctrl+S			AVR Studio 4 Project	Ctrl+4
	Save Selected Items As				Atmel Start Project	
e Pa	Save All	Ctrl+Shift	+S		Project Template	Ctrl+T
	Export Template				,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	
	Page Setup					
	Print	Ctrl+P				
	Recent Files		•			
	Recent Projects and Solutions		•			
×	Exit	Alt+F4			_	

- 3. ダウンロードした'atmelstart.atzip'ファイルを検索して選んでください。
- 4. "Atmel STARTインホート部(Atmel Start Importer)"ダイアログボックスが開きます。以下のプロジェクト詳細、プロジェクト名(Project name)、 場所(Location)、解決策名(Solution name)を入力してください。OKをクリックしてください。

nport Atmel Start	Project		
tmel Start Project(.atzip):	C:\Users\m43934\Downloads\My817Pro.atzip		Browse
iew project summary (CMS	IS package information)		
Project Name:	My817Pro1		
Location:	C:\Users\m43934\Documents\Atmel Studio\7.0		Browse
Solution:	Create New Solution	×	
Solution Name:	My817Pro1		
ew project import summar	¥.		

新しいAtmel Studioプロジェクトが作成され、ファイルがインポートされます。

\delta My817Pro0 - AtmelStudio	
File Edit View VAssistX ASF Project Build Debug Tools Window Help	
🕴 😋 🕶 🕲 🔞 - 😩 🔛 🎥 👗 🗗 合 🛛 フィマ - 🗐 🔍 🕨 Debug 🛛 -	Debug Browser + 👘 full
M & = → U ▶ ↔ ‡ ♀ ↑ ▶ ∓ Hex ※ 3 +	📔 🚽 🗰 ATtiny817 🎁 No Tool 🖕
driver_isr.c atmel_start.c 🗢 🗙	Solution Explorer
→ atmel_start.c → C:\Users\m43934\Documents\Atmel Studio\7.0\My817Pr · CGo	o o 🔂 'o - a 🕼 🕨 🗕 🚳 D
<pre>#include <atmel_start.h></atmel_start.h></pre>	Search Solution Explorer (Ctrl+ ")
<pre>P'** * Initializes MCU, drivers and middleware in the project **/ void atmel_start_init(void) { system_init(); }</pre>	Solution 'My817Pro0' (1 project) My817Pro0 Dependencies Output Files Libraries Libraries Config Govygen Govygen

6.1.3. Doxygen出力生成

いくつかのプロジェクトはDoxygen用に形式化された文章を含みます。

 注: http://www.doxygen.orgからそれをダウンロードしてDoxygenがインストールされなければなりません。Doxygen実行可能物を位置 付けるためにAtmel Studioを構成設定するよう尋ねられるでしょう。これは既定で'C:¥Program Files¥doxygen¥bin¥doxygen.exe' です。

資料を生成するにはDoxygen釦をクリックしてください。Doxygenが走行して生成した資料がウィンドウで開きます。



助言: 生成したファイルは'doxygen¥generator¥html'と'doxygen¥generator¥xml'のフォルダ下で見つけることができます。これ らはプロジェクトに追加されません。もっと制御が必要とされる場合に使うことができる独立したAtmel Studio用Doxygen 拡張があります。これはhttps://gallery.atmel.comで見つけることができます。

6.1.4. Atmel STARTを使ってプロジェクトを再構成設定する手順

1. 再構成設定(Re-Configure)釦をクリックするか、または解決策エクスプローラ(Solution Explorer)ウィンドウでプロジェクト節点を右クリックしてそのメニューからAtmel STARTプロジェクトの再構成設定(Re-configure Atmel Start Project)を選んでください。

Solution Explorer
◎ ◎ ☆ 'o - @ @ ⊁ 🥸 D
Search Solution Explorer (Ctrl+")
Solution 'My817Pro0' (1 project)
▲ 📙 My817Pro0 再構成設定(Re-Configure)
Dependencies
📴 Output Files
Libraries
Config
🕨 🛅 doxygen
i examples
include
Image: Second
tils
c atmel_start.c
atmel_start.h
c driver_isr.c
C main.c

Atmel STARTはAtmel Studio内部のウィントウで利用可能になります。

2. プロジェクトを変更してください。Atmel STARTウィンドウの下部でプロジェクトを生成(GENERATE PROJECT)釦をクリックしてください。

My817Ped/-AtmrtShuke File Edit Verw WastetX ASP Project Build Debug Tools Window Help O 10 No 10	Standard Mode: T2 Quald Launch (Chi+O) P - C ×
※MI会=→Ⅱ▶ @ まままたで Hex 76 幅・200 単位目間 24 出 除。 ● AltingST7 1	No Teel
Atmel/START •• × addres/start.e MY SOFTWARE COMPONENTS	Skutico topier • Skutico topier • • • • • • • • • • • • • • • • •
Application Application Add software component Middleware Middleware MYS17PR0 Acc.0 Acc.0 Acc.0	Show system drivers (2) Show hardware (2) Show hardware (2) 12C,0
SELECTED BOARD: ATTINY817 XPLAINED PRO	Properties - 0 × attret_strict. Fit Properties - 9(0, 1, p)
Grease Armsi The Armel ATtiny817 Xplained Pro evaluation kit is a hardware platform to evaluate the Arm Armel Studio integrated development platform, the kit provides easy access to the features integrate the device in a customer design.	el ATtiry\$17 microcontroller. Supported by the of the Atmel ATtiry\$17 and explains how to Bit Mice False
GENERATE PROJECT	

古い構成設定は取り去られて新しい構成設定が適用されます。

3. 何れかの変更されたファイルがある場合に変更されたファイルの一覧を持つウィンドウを得ます。どのファイルが上書きされるべきか選んで ください。任意選択で、上書きされるファイルの控え保存(Backup Current Project)を選んでください。

Project Summary	×
There are differences between some of the imported Atmel Start files and t project.	he corresponding files in your
Please select the files you would like to overwrite from the list below.	
Files Modified	~
include/port.h	View Diff
Files Added	
Import File : doxygen/generator/DoxygenLayout.xml	
Import File : doxygen/generator/doxyfile.doxygen	
Import File : doxygen/mainpage.dox	
C Import File : main.c	
C Import File : src/driver_init.c	
h Import File : include/driver_init.h	
h Import File : include/atmel_start_pins.h	
C Import File : driver_isr.c	¥.
Backup Current Project File Name : My817Pro0_2	
	K Cancel

6.1.5. Atmel STARTを使って新しいプロジェクトを作成

Atmel STARTはプロジェクトを生成する、または例プロジェクトを読み込みためにAtmel Studio内部ウィントウで開くことができます。

1. ファイル(file)⇒新規(New)メニューからAtmel STARTプロジェクト(Atmel Start Project)またはAtmel START例プロジェクト(Atmel Start Exam ple Project)を選んでください。

ĕ	AtmelStudio				
File	Edit View VAssistX ASF	Project Deb	ug	Tools Window Help	
	New	•	わ	Project	Ctrl+Shift+N
	Open	•	*b	File	Ctrl+N
	Close		As	Atmel Start Project	
×	Close Solution		As	Atmel Start Example Project	
	Import	•	₽)	Example Project	Ctrl+Shift+E
E.	Save Selected Items	Ctrl+S			
	Save Selected Items As				
e ^{les}	Save All	Ctrl+Shift+S			
	Export Template				
₽	Page Setup				
-	Print	Ctrl+P			
	Recent Files	•			
	Recent Projects and Solutions	•			
×	Exit	Alt+F4			



6.1.6. 設定

設定はツール(Tools)→任意選択(Options)→Atmel START(Atmel Start)メニューからアクセスされます。

1. Doxygen実行可能物の位置を設定してください。

ptions		? >
earch Options (Ctrl+E)		
Environment Projects Source Control Text Editor Debugger Atmel Software Framework Atmel Start Doxygen File Compare Settings Atmel Studio Feedback	Path of the Doxygen application: C:\Program Files\doxygen\bin\doxygen.exe	
Builder Device and Tool libraries Documentation Extensions Programming Dialog Status Management		

2. ファイル比較用ツールを定義してください。

Search Options (Ctrl+E)	Q	Configure application path for Comparing Files		
Environment	^	Path of the application used for comparing files:		
Projects		C:\Program Files (x86)\WinMerge\WinMergeU.exe		
 Source Control Text Editor 		Command line arguments to be used for file comparision:		
 Debugger Atmel Software Framework 		%original %mine /s /u		
 Atmel Start Doxygen 		%original - Path of the original Software Framework file.	^	
File Compare		 %mine - Path of the modified file in the local project. 		
Settings Atmel Studio Feedback Builder Device and Tool libraries Documentation Extensions Programming Dialog		If the command line for the configured file compare application is FileCompar filepath1 filepath2, specify Soriginal for filepatha and Somine for filepath2 For example, if configuring WinMerge as the compare application, specify the lowing command line arguments: Storiginal Smine /s /u	e.exe t. e fol-	

3. 一般設定。



<u>6.1.7.</u> 控え保存からプロジェクトを復元

プロジェクトの控え保存はAtmel STARTで再構成設定する時に取られます。控え保存はプロジェクトフォルタ、の'.atmel-start-backup'フォル タ、に格納されます。

- 1. Atmel STARTインホート部(Atmel Start Importer)を使うことによってAtmel Studioで'.atzip'形式の控え保存が開かれます。
 - 注: これらの控え保存はハグを持ちます。使用者が控え保存されたプロジェクトに追加や改名されたファイルとフォルダを持つ場合、Atm el STARTで復元された控え保存が再構成設定された場合にそれらが取り去られます。これが起きた場合、プロジェクトを再びイ ンポートして再構成設定するためにAtmel STARTを使うのを避けてください。
- 2. '.zip'形式の控え保存
 - 2.1. フォルタ に解凍(unzip)してください。
 - 2.2. Atmel Studioでファイル(File)⇒新規(New)⇒プロジェクト(Project)でAtmel Studio解決策(Atmel Studio Solution)⇒空の解決策 (Blank Solution)形式の新しいプロジェクトを作成してください。
 - **2.3.** 右側の解決策ェクスプローラ(Solution Explorer)で解決策(Solution)を選び、右クリックして既存プロジェクトを追加(Add Existing Project)を選び、解凍されたフォルダを検索してAtmel Studioプロジェクトファイルを選んでください。

6.2. IAR Embedded Workbench®

以下の手順はIAR Embedded WorkbenchでAtmel STARTからの出力を使う方法を記述します。インストールされた最新更新のIAR Embe dded Workbenchを持つことを確実にしてください。

6.2.1. Atmel STARTからプロシェクトをエクスホート

- 1. Atmel STARTウェブサ小で(例または基板の)新しいプロジェクトを作成してください。
- 2. ソフトウェア構成部品エクスホート(Export Software Component)釦をクリックしてください。

3. "IAR Embedded Workbench"チェック枠がチェックされる ことを確実にしてください。	DOWNLOAD YOUR CONFIGURED PROJECT
	Download a generated pack containing all your configured software components.
	Select which IDE or command line tool you want the pack to include support files for:
	Atmel Studio:
	W µVision from Keil:
	IAR Embedded Workbench:
	Somnium DRT. (Atmel Studio plugin):
	Makefile (standalone):
	Specify file name (optional): My Project

- 4. 一括ダウンロート(DOWNLOAD PACK)をクリックしてください。'atmelstart.atzip'一括ファイルがダウンロート'されます。
- 5. ダウンロードしたファイルの拡張子を'.zip'に改名('.atzip'から'.zip'に変更)してください。
- 6. 結果の'atmelstart.zip'ファイルをIARプロジェクトを作成しそうなあなたが選ぶフォルタに解凍(unzip)してください。

6.2.2. IAR Embedded Workbench®でプロジェクトを開く

プロジェクトをインポートする前に、IAR Embedded Workbenchでプロジェクト接続が許可されていることを確実にしてください。ツール(Tools)⇒ 任意選択(Options)⇒プロジェクト(Project)へ行き、ここで"プロジェクト接続許可(Enable project connections)"枠をチェックします。

•	IDE Opt	ions			×
Common Fonts Key Bindings Language E-Editor Messages Project Source Code Control	Stop build operation on: Save gditor windows before building: Save workspace and projects before building: Make before debugging: □ Beload last workspace at startup ✓ Play a sound after build operations ✓ Generate browse information □ No source browser and build status up ✓ Enable project connections	Never Always Always Always	v v v	1 process	
		ОК	Cancel	Apply	Help

- プロジェクト接続を使って開始するには、空で有り得る開いているプロジェクトを持つことが必要です。新しい空のプロジェクトを作成する にはプロジェクト(Project)⇒新規プロジェクト作成...(Create New Project...)⇒空プロジェクト(Empty Project)へ行ってください。
 注: IARプロジェクト(.ewpファイル)はダウンロートしたatmelstart.zipが解凍されたのと同じフォルダに作成されなければなりません。
- 2. 開いているIARプロジェクトを持つと、それにAtmel STARTプロジェクトを繋げることができます。プロジェクト(Project)メニューからプロジェクト 接続追加(Add Project Connection)を選ぶことができます。これは接続を構成設定したい外部ツールを選ぶことができるタイアログ ボックスを開きます。IARプロジェクト接続(IAR Project Connection)を選んでOKをクリックしてください。
- 3. 検索ウィンドウを開いてください。「6.2.1. Atmel STARTからプロジェクトをエクスホート」項でのように作成され、解凍されたフォルダでiar-pro ject-connection.ipcfファイルを選んでください。.ipcfファイルのインホート後、生成された全てのコートがIAR Embedded Workbenchプロジェ 外へ自動的に追加されます。



必要とされるインクルート、ディレクトリ(任意選択(Oprions)⇒C/C++コンハ゜イラ(C/C++ Compiler)⇒前処理部(Preprocessor))もプロシュクトに 追加されます。

General Options Runtime Checking	Multi	-file Compilation Discard Unused	l Publics			Factory Se	ttings
Assembler	Code	Optimizations	Output	List	Preprocessor	Diagnostics	• •
Custom Build Build Actions	<u>A</u> dditi	nore standard in onal include dire	clude dire ectories: (ctories	line)		s
Linker Debugger Simulator Angel	\$PROJ_DIR\$\utils\include \$PROJ_DIR\$\hal\hpl\gclk \$PROJ_DIR\$\CMSIS\Include\ \$PROJ_DIR\$\CMSIS\Include \$PROJ_DIR\$\hal\include \$PROJ_DIR\$\hal\include						
GDB Server IAR ROM-monitor	Preinclude file:						
I-jet/JTAGjet	Define	ed symbols: (one	e per line)				(const)
TI Stellaris Macraigor PE micro	AT	SAML22N18A		< >	Preprocesso Preserve <u>G</u> enerate	r output to file <u>c</u> omments #line directive	s

4. プロジェクトが8ビットAVRマイクロコントローラを使う場合、Cライブラリが変更されなければなりません。プロジェクト(Project)⇒任意選択(Options)⇒全般任意選択(General Options)⇒ライブラリ構成設定(Library Configuration)を選んでください。"ライブラリ(Library)"を"標準 DLIB(Normal DLIB)"に変更してください。

Category: General Options	
C/C++ Compiler Assembler	Target Output Library Configuration Library Options Heap Configur
Custom Build Build Actions	Library: Description:
Linker Debugger Atmel-ICE AVR ONE! CCR ICE200	Use the normal configuration of the C/C++ runtime library. No locale interface, C locale, no file descriptor support, no multibytes in printf and scarf, and no hex floats in strtod.
JTAGICE3	STOOLKIT_DIR\$\LIB\DLIB\dIAVR-3s-ec_mul-n.r90
JTAGICE mkII	Configuration file:
Simulator Third-Party Driver	STOOLKIT_DIR\$\LIB\DLIB\dlAVR-3s-ec_mul-n.h

注: AVRデバイス用プロジェクトに対するプロジェクト接続の追加はIAR IDEがリンカのマップとログのファイルを許可することができないことに ついての警告を生成するかもしれません。これらのメッセージは既定リンカの使用についての通知で安全に無視することができます。"再び表示しない(Don't show again)"チェック枠をチェックすることによってこれらの通知を禁止することが推奨されます。 5.7

゜ロシ゛	ェクトがQTouchライン	ブラリを使う場合、	既定によっ	て設定される	メモリが充分	ではなく、	CSTACK	とRSTACKを	変更して	ください。	с
------	---------------	-----------	-------	--------	--------	-------	--------	----------	------	-------	---

Options for node "MyQTouch	Project"			×
Category: General Options Static Analysis C/C++ Compiler Assembler	Target Output	Library Configuration	Library Options	Heap Configur
Custom Build Build Actions Linker Debugger Atmel-ICE AVR ONE! CCR ICE200	Data stack (CST Size (bytes): Data stack in ext Data stack in ext Data stack Data stack in extension Data stack in extension	ACK) 0x50 emal memory / Configuration mal memory bus	Betum address Depth (levels)	stack (RSTACK) 20 demal memory
JTAGICE3	Add one wa	RAM	mory accesses R <u>O</u> M	Non-Volatile
JTAGICE mkII	<u>B</u> ase address	0x0	0x0	0x0
Simulator Third-Party Driver	Memory size	0×0	Dx0	0x0
	Initialize unuse ☐ Ena <u>b</u> le bit defir	d interrupt <u>v</u> ectors wit iitions in I/O-Include f	h RETI instruction illes	Cancel

6. プロジェクト(Project)→任意選択(Options)⇒デ`ハ`ッカ`(Debugger)→構成設定(Setup)を選んでください。"ト`ライハ`(Driver)"を"CMSIS D AP"に変更してください。

Lategory:	Factory Settings
General Options Runtime Checking C/C++ Compiler Assembler	Setup Download Images Extra Options Multicore Plugins
Output Converter	
Custom Build Build Actions	Unver VI Run to
Linker	CMSIS DAP V main
Debugger	Setup macros
Simulator	Use macro file(s)
Angel	
CMSIS DAP GDB Server	
IAR ROM-monitor	
I-jet/JTAGjet J-Link/J-Trace	Devi <u>c</u> e description file
TI Stellaris	
Macraigor	
RDI	
ST-LINK	
Third-Party Driver	

7. ダウンロート'(Download)タブをクリックしてください。"フラッシュ読み込み部使用(Use flash loader(s))"がチェックされているのを確実にしてく ださい。

General Options							Factory Settings
Runtime Checking C/C++ Compiler Assembler	Setup	Download	Images	Extra Options	Multicore	Plugin	s
Custom Build Build Actions Linker Debugger Simulator Angel CMSIS DAP GDB Server IAR ROM-monitor I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor PE micro		ach to runnir rify download ppress down e flash loade Qverride def STOOLKIT Edit	ng target load r(s) ault boar _DIR\$\cc	d file nfig\flashloader	λ		

8. CMSIS DAP区分を選び、JTAG/SWDタブへ行き、インターフェース(Interface)を"SWD"に設定してください。OKをクリックしてください。

Category: General Options			Factory Settings
Runtime Checking C/C++ Compiler	Setup J	TAG/SWD	Breakpoints
Assembler Output Converter Custom Build Build Actions Linker Debugger Simulator	Probe c Auto Fron Exp	onfig o m file licit	Probe configuration file Override default CPU: Select
Angel CMSIS DAP GDB Server IAR ROM-monitor I-jet/JTAGjet J-Link/J-Trace TI Stellaris Macraigor PE micro	JTAG/SV Auto del	e VG VD speed tect v	Explicit probe configuration Multi+arget debug system Iarget number (TAP or Multidrop ID): Target with multiple CPUs CPU number on target:

Atmel START

 プロジェクトがQTouchライブラリを使う場合、リンカ出力形式がIntel hexに設定されなければなりません。プロジェクト(Project)⇒任意選択 (Options)⇒リンカ(Linker)⇒出力(Output)に行ってください。"既定無効(Override default)"チェック枠をクリックして出力ファイルの拡張子 を'.hex'に変更してください。形式(Format)下でその他(Other)を選んで引き落としメニューから'拡張Intel(intel-extended)'を選ん でください。



6.3. Keil[®] µVision[®]

以下の段階はKeil µVisionでAtmel STARTからの出力を使う方法を記述します。

6.3.1. Atmel STARTからプロジェクトをェクスポート

- 1. Atmel STARTウェブサイトで(例または基板の)新しいプロジェクトを作成してください。
- ソフトウェア構成部品エクスポート(Export Software Compon ent)釦をクリックしてください。"µVision from Keil"チェック 枠がチェックされていることを確実にしてください。

DOWNLOAD YOUR CONFIGURED PROJECT Download a generated pack containing all your configured software components. Select which IDE or command line tool you want the pack to include support files for: Image: Atmel Studio: Image: Atmel Studio: Image: Project Image: Atmel Studio: Image: Project Image: Project Image: Project

- 3. 一括ダウンロート(DOWNLOAD PACK)をクリックしてください。'atmelstart.atzip'一括ファイルがダウンロート'されます。
- 4. ダウンロードしたファイルの拡張子を'.zip'に改名('.atzip'から'.zip'に変更)してください。
- 5. 結果の'atmelstart.zip'ファイルをプロジェクトを作成しそうなあなたが選ぶフォルタに解凍(unzip)してください。

4

DOWNLOAD PACK

6.3.2. Keil[®] µVision[®]でプロジェクトを開く

Keil µVisionを開始し、ファイル(File)⇒開く(Open)を選んでください。生成部一括記述ファイル(Generator Pack Description files,(*.gp dsc))を示すようにファイル形式選別部を変更してください。「6.3.1. Atmel STARTからプロジェクトをエクスホート」項で記述されるようなフォ ルダから'AtmelStart.gpdsc'ファイルを選んでください。開く(Open)をクリックしてください。

Software Component	Sel.	Variant		Version	Description
🗉 💠 AtmelStart					Configuration Files generated by Atmel S
					Cortex Microcontroller Software Interface
CORE)		5.0.0	CMSIS-CORE for Cortex-M, SC000, SC300
OSP				1.4.6	CMSIS-DSP Library for Cortex-M, SC000,
🗉 🚸 RTOS (API)				1.0	CMSIS-RTOS API for Cortex-M, SC000, ar
🗄 🚸 RTOS2 (API)				2.0	CMSIS-RTOS API for Cortex-M, SC000, ar
🕀 🚸 CMSIS Driver					Unified Device Drivers compliant to CMS
🗄 🗇 Compiler					ARM Compiler Software Extensions
🖃 💠 Device					Startup, System Setup
Startup				1.0.1	System Startup for Atmel SAM D21 Devic
🗉 🚸 File System		MDK-P	·o 🔻	6.8.0	File Access on various storage devices
🗉 💠 Graphics		MDK-P	·o 🔽	5.32.2	User Interface on graphical LCD displays
🕀 🗇 Network		MDK-P	·o 🔻	7.2.0	IPv4/IPv6 Networking using Ethernet or S
🗄 💠 USB		MDK-P	·o 💌	6.8.0	USB Communication with various device
•)
Validation Output			Description		
Atmel::AtmelStart:Framework			Additional software components required		
require Device:Startup:1.1.0			Install missing component		

2. CMSIS⇒CoreとDevice⇒Startupがチェックされているのを確実にしてください。

GPDSCがプロジェクト設定を含まず、生成したコート、はコンハペイルするのにKeil C/C++の"C99 Mode"が必要な(コート、書きでCMSIS例外が許される)ため、KeilでGPDSCを開いた後でこれが許可されるべきです。

- 3. プロジェクト(Project)⇒目的対象用任意選択(Options for target...)⇒C/C++(C/C++)を選んでください。
- 4. "C99動作(C99 Mode)"チェック枠がチェックされているのを確実にしてください。OKをクリックしてください。

evice Target Output Listing Use	er C/C++ Asm Linker Debug Utilities	5
Preprocessor Symbols		
Define:		
U <u>n</u> define:		
- Language / Code Generation		W
Execute-only Code	Strict ANSI C	vvamings:
Optimization: Level 0 (-00) 💌	Enum <u>C</u> ontainer always int	
Optimize for Time	Plain Char is Signed	🔽 Thum <u>b</u> Mode
Split Load and Store Multiple	Read-Only Position Independent	No Auto Includes
✓ One ELF Section per Function	Read-Write Position Independent	C99 Mode
Include Paths		
Misc Controls		
Compiler control string	+ -D_EVAL -g -O0apcs=interworksplit_sectio wnloads\atmelstart (13)\RTE"	ons

Xplianed基板でプロジェクトをディックする前に、CMSIS-DAPディッカを使うように設定を変更してください。

- 5. プロジェクト(Project)⇒目的対象用任意選択(Options for target...)⇒デ`ハ`ック`(Debug)を選んでください。
- 6. 使用(Use)一覧からCMSIS-DAPディッガ(CMSIS-DAP Debugger)を選んでください。OKをクリックしてください。

◯ Use <u>S</u> imulato □ Limit Speed t	r <u>with restrictions</u> Settings	Settings
Load Applica	tion at Startup 🔽 Run to main()	✓ Load Application at Startup ✓ Run to main() Initialization File:
	Edit	
-Restore Debug	Session Settings	Restore Debug Session Settings
Breakpoir	nts 🔽 Toolbox	Breakpoints I Toolbox
Watch W	indows & Performance Analyzer	Watch Windows
Memory D)isplay 🔽 System Viewer	Memory Display Vistem Viewer
CPU DLL:	Parameter:	Driver DLL: Parameter:
SARMCM3.DLL		SARMCM3.DLL
Dialog DLL:	Parameter:	Dialog DLL: Parameter:
	1	

7. 任意選択: プロジェクトが "sam.h" ソース入力ファイルを開けません。(cannot open source input file "sam.h") の異常と共にコンハイルす る場合、SAM D20デバイスに対して '.¥samd20¥include'のようなデバイス特有インクルート パスをC/C++入力パスに追加してください。

Device Target Output Listing User	C/C++ Asm Linker Debug Utilities		
Preprocessor Symbols			
Define:			_
U <u>n</u> define:			
Language / Code Generation			
Execute-only Code	Strict ANSI C	Warnings:	
Optimization: Level 0 (-00)	Enum Container always int	All Warnings 💽	
Optimize for Time	Plain Char is Signed	🔲 Thum <u>b</u> Mode	
Split Load and Store Multiple	Read-Only Position Independent	No Auto Includes	
✓ One ELF Section per Function	<u>Read-Write Position Independent</u>	C99 Mode	
Include Paths]
Misc Controls			
Compiler control string	D_EVAL -g -O0apcs=interworksplit_section nloads/My Project/RTE/AtmelStart''	s -l ./samd20/include	* *
string	indus ny majourne stanistan		Ŧ

6.4. GNU C Makefile

Atmel STARTからの出力はMakefileを使ってGNU Cで使うことができます。

6.4.1. Atmel STARTからプロジェクトをエクスホート

- 1. Atmel STARTウェブサイトで(例または基板の)新しいプロジェクトを作成してください。
- 2. ソフトウェア構成部品エクスポート(Export Software Component)釦をクリックしてください。
- 3. "Makefile"チェック枠がチェックされているのを確実にして ください。

Download a generated pack containing all you	r configured software components.
Select which IDE or command line tool you wa	nt the pack to include support files fo
Atmel Studio:	\checkmark
🕎 µVision from Keil:	\checkmark
lAR Embedded Workbench:	\checkmark
Somnium DRT. (Atmel Studio plugin):	\checkmark
😚 Makefile (standalone):	
Specify file name (optional): My Project	
DOWNLOAD PACK	

- 4. 一括ダウンロート(DOWNLOAD PACK)をクリックしてください。'atmelstart.atzip'一括ファイルがダウンロートされます。
- 5. ダウンロードしたファイルの拡張子を'.zip'に改名('.atzip'から'.zip'に変更)してください。
- 6. 結果の'atmelstart.zip'ファイルをGNU Cプロジェクトを作成しそうなあなたが選ぶフォルタに解凍(unzip)してください。

6.4.2. GNU C Makefileの実行

選んだデバイスを支援するGNU Cコンパイラがインストールされているのを確実にしてください。CコンパイラはPATHで示されなければなりません。また、'make'ユーティリティもインストールされてPATHで示されるべきです。

Atmel STARTからプロジェクトをエクスポートする時に"Makefileを含む(Include Makefile)"チェック枠をチェックした場合、解凍された'atmelstar t.atzip'ファイルは'gcc'フォルダ内側にMakefileを含むべきです。

1. 解凍した'atmelstart.atzip'ファイルのフォルダで'gcc'副フォルダへ行ってコマント、プロンプトを開いてください。

2. makeを走らせてください。これがプロジェクトを構築します。

6.5. SOMNIUM DRT

以下の段階はSOMNIUM DRTでAtmel STARTからの出力を使う方法を記述します。

6.5.1. Atmel STARTからプロジェクトをエクスホート

1. Atmel STARTウェブサイトで(例または基板の)新しいプロジェクトを作成してください。

2. ソフトウェア構成部品エクスホート(Export Software Componen t)釦をクリックしてください。Atmel StudioとSomonium IDE	DOWNLOAD YOUR
用のプロジェクトをエクスホートするためにAtmel Studioを選 んでください。	Download a generated Select which IDE or con
	Atmel Studio:

Download a generated pack containing all your	configured software components.
Select which IDE or command line tool you war	nt the pack to include support files for
Atmel Studio:	
W pVision from Keil:	
IAR Embedded Workbench:	
💬 Makefile (standalone):	
Specify file name (optional): My Project	
DOWNLOAD PACK	

CONFICUEED DEOLECT

3. 一括ダウンロート(DOWNLOAD PACK)をクリックしてください。'atmelstart.atzip'一括ファイルがダウンロートされます。

6.5.2. Atmel START出力をSOMNIUM DRTにインホートする手順

- 1. SOMNIUM DRTを開始してください。
- 2. インボート ダイアログを開くためにファイル(File)⇒インボート(Import)を選んでください。



3. 全般(Gereral)フォルダで既存プロジェクトを作業空間へ(Existing Projects into Workspace)を選んで次へ(Next)をクリックしてください。

S Import	– 🗆 X
Select Create new projects from an archive file or directory.	Ľ
Select an import source:	
type filter text	
 General Archive File Existing Projects into Workspace File System Preferences C/C++ CVS Git Install Remote Systems Run/Debug Team 	
(?) < <u>B</u> ack <u>N</u> ext >	<u>F</u> inish Cancel

4. 書庫ファイル選択(Select Archive File)ラジオ釦を選んでAtmel STARTからエクスポートした'.atzip'ファイルを検索してください。

5 Import					×
mport Projects Select a directory to sear	ch for existing Eclipse p	rojects.		T	
○ Select roo <u>t</u> directory:			~	B <u>r</u> owse	
Select <u>a</u> rchive file:	C:\atmel-start\SAMD2	21_START.atzip	~	Browse	
Projects:					
type filter text				<u>S</u> elect All	É I
Project Name	Project Type	Path		Deselect A	11
SAMD21_STAR	Atmel START			R <u>e</u> fresh	
Options Search for nested pro Copy projects into w Hide projects that all SOMNIUM advanced	ojects orkspace ready exist in the worksp I import	bace			
Working sets	ing sets				
W <u>o</u> rking sets:			~	S <u>e</u> lect	

5. 次へ(Next)、その後に終了(Finish)をクリックすると、インポートされたファイルと共に新しいDRTプロジェクトが作業区間で作成されます。

S DRT C/C++ - SAMD21_START/main.c - Si File Edit Source Refactor Navigate S Source DRT SAMD2	MNIUM DRT Cortex-M IDE - □ X egrch Project Run Window Help
Project explorer 23	<pre>man.c 23 1 #include <atmel_start.h> 2 3@ int main(void) 4 { 5 /* Initializes MCU, drivers and middleware */ 6 atmel_start_init(); 7 8 /* Replace with your application code */ 9 while(1) { 0 } 1 } 2 </atmel_start.h></pre>
	< >
Problems 🔀 🥫 Tasks	$\bigtriangledown \Box \Box \Box \Box \Box \bullet \Box $
Description	QF Resource F Image: Second s
DRT license: OK	Writable Smart Insert 2:1

6.6. MPLAB™ X IDE

6.6.1. MPLAB[™] Xプロジェクトの作成

- 1. Atmel STARTウェブサイト(http://start.atmel.com/)で(例または基板の)新しいプロジェクトを作成してください。
- 2. 一旦プロジェクトが作成されると、プロジェクト エクスポート(EXPORT PROJECT)頁へ行って"MPLAB X IDE"枠をチェックしてください。

Atmel START ATTINY817	← Return To Front Page Help
{ } VIEW CODE	CONFIGURATION EXPORT PROJECT
EXPORT PROJECT	
DOWNLOAD YOUR CONFIGURED PROJECT	WHAT TO DO NEXT?
Download a generated pack containing all your configured software components. Select which IDE or command line tool you want the pack to include support files for:	Use one of the selected IDEs and import your project as described in the u If you do not have any IDEs installed you can download and install <u>Atmel S</u> free.
MPLAB X IDE:	Note: The exported pack can also be used later if you want to import a co Atmel START
AR Embedded Workbench:	How to open in IDEs
Specify file name (optional): My Project	
DEVICE PACK INFORMATION	
Device: ATtiny817	
Device Pack: ATtiny_DFP	

6.6.2. MPLAB[™] XでAtmel STARTプロジェクトをインポート

1. 下で示されるようにファイル(File)⇒インボート(Import)⇒START MPLABプロジェクト(START MPLAB Project)を選んでください。



2. 下で示されるように'.atzip'ファイルを検索してください。

2	Import Start Project	×
Steps	Locate *.atzip File	
 Locate *.atzip File Select Device Select Tool Select Compiler Select Project Name and Folder Summary 	President Eilenemen C (MDI AB/MDI AB atria	
MPLAB X IDE	Project Hiename: C: WPLAB WPLAB, atzip	NSC
	< Back Next > Finish Cancel	Help

3. ウィザートで記述される手順に従ってください。

4. MPLAB Xプロジェクトが作成されます。



7. Atmel STARTでの内容

Atmel STARTはMicrochipのAVRとARMのデバイス用のデバイストライバ、ミドルウェア、実演応用を意味する内容を構成設定して配給するためのウェブに基づく基盤です。AVRデバイス用の内容はAVR Codeと呼ばれる枠組みで、ARMデバイス用の内容はASF4と呼ばれる枠組みです。

Atmel STARTのウェブ インターフェースは新しい使用者でさえ短い時間でシステムを構成設定してシステムを整えて動かすことができるようにト ライバと応用の構成設定を簡単にします。Atmel STARTでプロジェクトを生成する初期段階で使用者はデバイスを選びます。Atmel STAR Tは選んだデバイスに応じて自動的に適切な枠組みを選びます。

7.1. ARM内容

Atmel STARTでは次のASF4のソフトウェア開発の一部としてARM用のドライバとソフトウェア階層が提供されます。この版は旧ASF版の使用 者と投稿者によって報告された問題を解決するためと、Atmel STARTウェブUIとのより良い統合のために0から構築され、枠組み全体 の完全な再設計と再実装です。経験豊かなASF使用者に対して親近感を保つことが未だ目標ですが、新しい使用者に対する開始 にも未だ容易です。ASF4でのいくつかの変更はこの版に対する必要条件に合うことが必要で、最も重要な変更はASF API手引書で 一覧にされます。

ASF4はAtmel STARTに密接に統合され、これは以前よりもASF4コードを使用者の仕様に対してより一層誂えることができることを意味 します。例えば、コードの塊を許可/禁止するのにC前処理部条件式を使うことに代わると、禁止されたコードの塊はプロジェクトのソースから 完全に取り去られ、より綺麗で容易なコード読みに帰着します。Atmel STARTへの統合はソフトウェア構成設定がより一層使い易い環境 で行われ、デバイスに積み込まれた構成設定情報が生の周辺機能レジスタ内容だけであることを意味し、これはファームウェアイメージをより 一層簡潔にします。

我々が本気で取り掛かった1つの重要な問題はASF4に基づくコート、のメモリ占有量と性能です。ASFv3に基づくコート、を走行するための フラッシュメモリ必要条件は多くの使用者によって高すぎると思われています。これはコート、生成を使って周辺機能が初期化される方法を 変更することによって処理されました。報告された性能の問題は代表的に高い割り込み遅延/遅いコート、実行で、これは割り込み処理 部をより小さくてより簡単にすることによって解決されました。ASF4とASFv3を比較するいくつかのヘンチマーク値はASF4 API参照基準 手引書で見つけることができます。

ASF4の詳細な記述はASF4 API参照基準手引書で見つけることができます。

7.2. AVR[®]内容

Atmel STARTはAVRデバイス用のコードを生成するのに使うことができます。

AVRデバイス用のハードウェアドライバ、ミドルウェアドライバ、実演応用がAVR Code枠組みに収集されています。AVR CodeはAtmel START に密接に統合され、これは使用者の求めにコードを誂えることができることを意味します。例えば、コードの塊を許可/禁止するのにC前 処理部条件式を使うことに代わると、禁止されたコードの塊はプロジェクトのソースから完全に取り去られ、より綺麗で容易なコート読みに帰 着します。Atmel STARTへの統合はソフトウェア構成設定がより一層使い易い環境で行われ、生成したコートがより密接に生のハードウェア に接続されることを意味し、これはファームウェアイメージをより一層簡潔にします。

AVR CodeはARMデバイス用のASF4のAVR等価物です。AVRデバイスは一般的にARMデバイスよりももっと制限された資源を持ち、これがAVR Codeに反映されます。応用を構成設定してそれを選んだIDEにエクスポートするのにAtmel STARTを使うと、AVRとARMの両デバイスに対して同じ方法で動きます。

AVR Code枠組みは以下の優先順によって先導されます。

- 1. コードの大きさ
- **2**. コート 速度
- 3. コードの簡潔性
- 4. コードの信頼性/理解性

AVR Codeの詳細な記述はAVR Code参照基準手引書で見つけることができます。

8. キーホ゛ート゛ ショートカット

以下のキーボート ショートカットがAtmel STARTによって支援されます。

- ・<A> ソフトウェア構成部品追加(Add Software Component)ダイアログを開く
- <D> 主計器盤へ切り替え
- ・ <P> PIN/MUX構成設定部へ切り替え
- ・ <C> クロック構成設定部へ切り替え

9. Atmel STARTへのリンク

他のウェブ頁からAtmel START内のデバイス、基板、プロジェクトへ直接リンクすることができます。

計器盤(Dashboard)

利用可能なら、ブラウザで最後に使ったプロジェクトで計器盤を開きます。

・リンク:http://start.atmel.com/#dashboard

指定したデバイスで新しいプロジェクトを開く

指定したデバイスに設定した検索領域で新しいプロジェクト頁を開きます。デバイス識別名やもっと一般的な検索用語を指定することができます。検索結果の一覧は1つ以上の一致するデバイスを示します。

- ・形式1:http://start.atmel.com/#device/[デバイス識別名]
- ・形式2:http://start.atmel.com/#device/[デバイス検索]
- 例1:http://start.atmel.com/#device/ATSAMD21J18A
- ・ 例2:http://start.atmel.com/#device/SAMD21

デバイス識別名の完全な一覧についてはhttp://start.atmel.com/api/v1/devicesをご覧ください。

指定した基板で新しいプロジェクトを開く

指定した基板に設定した検索領域で新しいプロジェクト頁を開きます。デバイス用と同じように、多かれ少なかれ正確な検索用語を指定 することができます。

- ・形式:http://start.atmel.com/#board/[基板識別名]
- 例:http://start.atmel.com/#board/SAML21XplainedPro

基板識別名の完全な一覧についてはhttp://start.atmel.com/api/v1/boardsをご覧ください。

特定の例プロジェクトを開く

指定した例で計器盤を開きます。

- ・形式:http://start.atmel.com/#example/[例識別名(符号化したURL)]
- 例:http://start.atmel.com/#example/Atmel%3AApplication%5FExamples%3A0%2E0%2E1%3A%3AApplication%3ALED%5Fswitcher %3A

それを開くことによって与えられた例用のリンクを得ることができます。その後に例構成設定を開いて(計器盤(DASHBOARD)タブでそれ をクリック)全般(Gerenal)部分からリンク(Link)を選んでください。

予め定義した検索で例閲覧部を開く

指定した文に設定した検索領域で例閲覧頁を開きます。

- ・形式:http://start.atmel.com/#examples/[検索文]
- 例:http://start.atmel.com/#examples/LED

10. 改訂履歴

資料改訂	日付	注釈
А	2018年8月	初版資料公開
В	2018年9月	 「Keil[®]からのµVision[®]での安全プロジェクトのインポート」を更新。画像を整列 「Keil[®]からのµVision[®]での非安全プロジェクトのインポート」を更新。画像を整列 「MPLAB[™] X IDE」を更新。SAMデバイス用MPLAB Xプロジェクトのエクスポートに対して拡張された支援

Microchipウェフ゛サイト

Microchipはhttp://www.microchip.com/で当社のウェブ サイト経由でのオンライン支援を提供します。このウェブ サイトはお客様がファイルや情報を容易に利用可能にする手段として使われます。お気に入りのインターネット ブラウザ を用いてアクセスすることができ、ウェブ サイトは以下の情報を含みます。

- ・製品支援 データシートと障害情報、応用記述と試供プログラム、設計資源、使用者の手引きとハートウェア支援資料、最新ソフトウェア配布と 保管されたソフトウェア
- ・全般的な技術支援 良くある質問(FAQ)、技術支援要求、オンライン検討グループ、Microchip相談役プログラム員一覧
- ・Microchipの事業 製品選択器と注文の手引き、最新Microchip報道発表、セントとイベントの一覧、Microchip営業所の一覧、代理 店と代表する工場

お客様への変更通知サービス

Microchipのお客様通知サービスはMicrochip製品を最新に保つのに役立ちます。加入者は指定した製品系統や興味のある開発ツール に関連する変更、更新、改訂、障害情報がある場合に必ず電子メール通知を受け取ります。

登録するにはhttp://www.microchip.com/でMicrochipのウェブ サイトをアクセスしてください。"Support"下で"Customer Change Notificati on"をクリックして登録指示に従ってください。

お客様支援

Microchip製品の使用者は以下のいくつかのチャネルを通して支援を受け取ることができます。

- ・代理店または販売会社
- ・最寄りの営業所
- ・現場応用技術者(FAE:Field Aplication Engineer)
- ・技術支援

お客様は支援に関してこれらの代理店、販売会社、または現場応用技術者(FAE)に連絡を取るべきです。最寄りの営業所もお客様の手助けに利用できます。営業所と位置の一覧はこの資料の後ろに含まれます。

技術支援はhttp://www.microchip.com/supportでのウェブ サイトを通して利用できます。

Microchipデバイスコート、保護機能

Microchipデバイスでの以下のコード保護機能の詳細に注意してください。

- ・Microchip製品はそれら特定のMicrochipデータシートに含まれる仕様に合致します。
- ・Microchipは意図した方法と通常条件下で使われる時に、その製品系統が今日の市場でその種類の最も安全な系統の1つである と考えます。
- コード保護機能を破るのに使われる不正でおそらく違法な方法があります。当社の知る限りこれらの方法の全てはMicrochipのデータシートに含まれた動作仕様外の方法でMicrochip製品を使うことが必要です。おそらく、それを行う人は知的財産の窃盗に関与しています。
- ・Microchipはそれらのコードの完全性について心配されているお客様と共に働きたいと思います。
- ・Microchipや他のどの半導体製造業者もそれらのコートの安全を保証することはできません。コート、保護は当社が製品を"破ることができない"として保証すると言うことを意味しません。

コート、保護は常に進化しています。Microchipは当社製品のコート、保護機能を継続的に改善することを約束します。Microchipのコート、保護機能を破る試みはデジタルシニアム著作権法に違反するかもしれません。そのような行為があなたのソフトウェアや他の著作物に不正なアクセスを許す場合、その法律下の救済のために訴権を持つかもしれません。

法的通知

デバイス応用などに関してこの刊行物に含まれる情報は皆さまの便宜のためにだけ提供され、更新によって取り換えられるかもしれま せん。皆さまの応用が皆さまの仕様に合致するのを保証するのは皆さまの責任です。Microchipはその条件、品質、性能、商品性、 目的適合性を含め、明示的にも黙示的にもその情報に関連して書面または表記された書面または黙示の如何なる表明や保証もし ません。Microchipはこの情報とそれの使用から生じる全責任を否認します。生命維持や安全応用でのMicrochipデバイスの使用は完 全に購入者の危険性で、購入者はそのような使用に起因する全ての損害、請求、訴訟、費用からMicrochipを擁護し、補償し、免責 にすることに同意します。他に言及されない限り、Microchipのどの知的財産権下でも暗黙的または違う方法で許認可は譲渡されま せん。

商標

Microchipの名前とロゴ、Mcicrochipロゴ、AnyRate、AVR、AVRロゴ、AVR Freaks、BitCloud、chipKIT、chipKITロゴ、CryptoMemory、Cr yptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KeeLoq、KeeLoqロゴ、Kleer、LANCheck、LINK MD、maXStylus、maXTouc h、MediaLB、megaAVR、MOST、MOSTロゴ、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32ロゴ、Prochip Designer、QTou ch、SAM-BA、SpyNIC、SST、SSTロゴ、SuperFlash、tinyAVR、UNI/O、XMEGAは米国と他の国に於けるMicrochip Technology Incor poratedの登録商標です。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTou ch、Precision Edge、Quiet-Wireは米国に於けるMicrochip Technology Incorporatedの登録商標です。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、CodeGuard、CryptoAut hentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherG REEN、In-Circuit Serial Programming、ICSP、INICnet、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet^{DD*}、memBrain、 Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certified^{DD*}、MPLAB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、SAM-ICE、 Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、View Sense、WiperLock、Wireless DNA、ZENAは米国と他の国に於けるMicrochip Technology Incorporatedの商標です。

SQTPは米国に於けるMicrochip Technology Incorporatedの役務標章です。

Silicon Storage Technologyは他の国に於けるMicrochip Technology Inc.の登録商標です。

GestICは他の国に於けるMicrochip Technology Inc.の子会社であるMicrochip Technology Germany II GmbH & Co. KGの登録商 標です。

ここで言及した以外の全ての商標はそれら各々の会社の所有物です。

© 2018年、Microchip Technology Incorporated、米国印刷、不許複製

DNVによって認証された品質管理システム

ISO/TS 16949

Microchipはその世界的な本社、アリゾナ州のチャント・ラーとテンヘ。、オレゴン州グラシャムの設計とウェハー製造設備とカリフォルニアとイント・の設計センターに対してISO/TS-16949:2009認証を取得しました。当社の品質システムの処理と手続きはPIC® MCUとdsPIC® DSC、KEELOQ符号飛び回りデバイス、直列EEPROM、マイクロ周辺機能、不揮発性メモリ、アナログ製品用です。加えて、開発システムの設計と製造のためのMic rochipの品質システムはISO 9001:2000認証取得です。

日本語© HERO 2020.

本使用者の手引きはMicrochipのAtmel START使用者の手引き(DS50002793B-2018年9月)の翻訳日本語版です。日本語では不自 然となる重複する形容表現は省略されている場合があります。日本語では難解となる表現は大幅に意訳されている部分もあります。 必要に応じて一部加筆されています。頁割の変更により、原本より頁数が少なくなっています。

必要と思われる部分には()内に英語表記や略称などを残す形で表記しています。

青字の部分はリンクとなっています。一般的に赤字の0,1は論理0,1を表します。その他の赤字は重要な部分を表します。



世界的な販売とサービス

本社

2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 技術支援: http://www.microchip.com/ support ウェブ^{*}アトレス: www.microchip.com

米国

7h7>9 Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455

オースチン TX Tel: 512-257-3370

ボストン Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088

シカゴ Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075

\$`7X Addison, TX Tel: 972–818–7423 Fax: 972–818–2924

7[°]トロイト Novi, MI Tel: 248−848−4000

ヒューストン TX Tel: 281-894-5983

1)7^{*}7†π[°]リス Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380

ロサンセ゛ルス

Mission Viejo, CA Tel: 949–462–9523 Fax: 949–462–9608 Tel: 951–273–7800 **D–IJ–** NC Tel: 919–844–7510

____9 NY Tel: 631–435–6000

サンホセ CA Tel: 408-735-9110 Tel: 408-436-4270 カナダ[・] - トロント

Tel: 905-695-1980 Fax: 905-695-2078 オーストラリア - シト・ニー Tel: 61-2-9868-6733 中国 - 北京 Tel: 86-10-8569-7000 中国 - 成都 Tel: 86-28-8665-5511 中国 - 重慶 Tel: 86-23-8980-9588 中国 - 東莞 Tel: 86-769-8702-9880 中国 – 広州 Tel: 86-20-8755-8029 中国 – 杭州 Tel: 86-571-8792-8115 中国 - 香港特別行政区 Tel: 852–2943–5100 中国 - 南京 Tel: 86-25-8473-2460 中国 - 青島 Tel: 86-532-8502-7355 中国 - 上海 Tel: 86-21-3326-8000 中国 - 瀋陽 Tel: 86-24-2334-2829 中国 - 深圳 Tel: 86-755-8864-2200 中国 – 蘇州

亜細亜/太平洋

Tel: 86-186-6233-1526 中国 - 武漢 Tel: 86-27-5980-5300 中国 - 西安 Tel: 86-29-8833-7252 中国 - 廈門 Tel: 86-592-2388138 中国 - 珠海 Tel: 86-756-3210040

亜細亜/太平洋

イント - ハンガロール Tel: 91-80-3090-4444 イント - ニューデリー Tel: 91-11-4160-8631 イント・フネー Tel: 91-20-4121-0141 日本 - 大阪 Tel: 81-6-6152-7160 日本 - 東京 Tel: 81-3-6880-3770 韓国 - 大邱 Tel: 82-53-744-4301 韓国 - ソウル Tel: 82-2-554-7200 マレーシア – クアラルンプール Tel: 60-3-7651-7906 マレーシア ー ヘ・ナン Tel: 60-4-227-8870 フィリピン ー マニラ Tel: 63-2-634-9065 シンガポール Tel: 65-6334-8870 台湾 - 新竹 Tel: 886-3-577-8366 台湾 - 高雄 Tel: 886-7-213-7830 台湾 - 台北 Tel: 886-2-2508-8600 タイ ー バンコク Tel: 66-2-694-1351 ベトナム ー ホーチミン Tel: 84-28-5448-2100

欧州 オーストリア – ウェルス Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 テンマーク - コヘンハーケン Tel: 45-4450-2828 Fax: 45-4485-2829 フィンラント – エスホー Tel: 358-9-4520-820 フランス – パリ Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 トイツ – ガルヒング Tel: 49-8931-9700 ドイツ – ハーン Tel: 49-2129-3766400 トイツ – ハイルフロン Tel: 49-7131-67-3636 ドイツ – カールスルーエ Tel: 49-721-625370 ドイツ - ミュンヘン Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 ドイツ - ローセンハイム Tel: 49-8031-354-560 イスラエル - ラーナナ Tel: 972-9-744-7705 イタリア ー ミラノ Tel: 39-0331-742611 Fax: 39-0331-466781 イタリア ー パドバ Tel: 39-049-7625286 オランダーデルーネン Tel: 31-416-690399 Fax: 31-416-690340 ノルウェー - トロンハイム Tel: 47-72884388 ポーラント゛ー ワルシャワ Tel: 48-22-3325737 ルーマニア – ブカレスト Tel: 40-21-407-87-50 スペイン - マドリート Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 スウェーデン – イェーテホリ Tel: 46-31-704-60-40 スウェーデン – ストックホルム Tel: 46-8-5090-4654 イキ・リス - ウォーキンガム Tel: 44-118-921-5800 Fax: 44-118-921-5820