



Gowin_EMPU_M1 クイックデザイン リファレンスマニュアル

IPUG534-1.8J, 2021-07-16

著作権について (2021)

著作権に関する全ての権利は、**Guangdong Gowin Semiconductor Corporation** に留保されています。

GOWIN、Gowin、及びGOWINSEMIは、当社により、中国、米国特許商標庁、及びその他の国において登録されています。商標又はサービスマークとして特定されたその他全ての文字やロゴは、それぞれの権利者に帰属しています。何れの団体及び個人も、当社の書面による許可を得ず、本文書の内容の一部もしくは全部を、いかなる視聴覚的、電子的、機械的、複写、録音等の手段によりもしくは形式により、伝搬又は複製をしてはなりません。

免責事項

当社は、**GOWINSEMI Terms and Conditions of Sale** (GOWINSEMI取引条件) に規定されている内容を除き、(明示的か又は黙示的かに拘わらず) いかなる保証もせず、また、知的財産権や材料の使用によりあなたのハードウェア、ソフトウェア、データ、又は財産が被った損害についても責任を負いません。本文書における全ての情報は、予備的情報として取り扱われなければなりません。当社は、事前の通知なく、いつでも本文書の内容を変更することができます。本文書を参照する何れの団体及び個人も、最新の文書やエラッタ(不具合情報)については、当社に問い合わせる必要があります。

バージョン履歴

日付	バージョン	説明
2019/02/19	1.0J	初版。
2019/07/18	1.1J	<ul style="list-style-type: none"> ● MCU ハードウェア設計とソフトウェア設計では、周辺機器 CAN、Ethernet、SPI-Flash、RTC、TRNG、DualTimer、I2C、SPI、SD-Card の拡張をサポート。 ● MCU はオフチップ SPI-Flash によるダウンロードと起動をサポート。
2019/08/18	1.2J	<ul style="list-style-type: none"> ● MCU ハードウェア設計とソフトウェア設計では、周辺機器 DDR3 の拡張をサポート。
2019/09/27	1.3J	<ul style="list-style-type: none"> ● MCU ハードウェア設計およびソフトウェア設計では、SPI-Flash の読み出し、書き込み、および消去をサポート。 ● MCU ソフトウェア設計は、外部デバイス I2C の連続したマルチバイト読み出しおよび書き込みをサポート。 ● 既知の MCU ソフトウェア設計の AHB2 拡張インターフェースと APB2 拡張インターフェースのアドレスマッピングの問題を修正。 ● 既知の MCU ソフトウェア設計の DDR3 Memory の連続した読み出し/書き込みの問題を修正。
2019/12/06	1.4J	<ul style="list-style-type: none"> ● MCU ハードウェア設計とソフトウェア設計では、周辺機器 PSRAM の拡張をサポート。 ● MCU コンパイラ GMD V1.0 を更新。 ● RTOS リファレンスデザインを更新。
2020/03/04	1.5J	ソフトウェアのバージョンを更新。
2020/06/12	1.6J	<ul style="list-style-type: none"> ● MCU が外部命令メモリをサポート。 ● MCU が外部データメモリをサポート。 ● AHB バスインターフェースを 6 つ拡張。 ● APB バスインターフェースを 16 個拡張。 ● GPIO が複数のインターフェースタイプをサポート。 ● I2C が複数のインターフェースタイプをサポート。
2021/01/25	1.7J	<ul style="list-style-type: none"> ● C バージョンの GW1N-9C、GW2A-18C、GW2A-55C のリファレンスデザインを更新 ● リファレンスデザインで使用される Gowin ソフトウェアのバージョンを更新。
2021/07/16	1.8J	<ul style="list-style-type: none"> ● 合成ツール Synplify Pro の情報を削除。 ● FPGA および MCU ソフトウェアのバージョン情報を更新。

目次

目次.....	i
図一覧.....	iii
表一覧.....	iv
1 リファレンスデザイン	1
1.1 ソフトウェア・リファレンスデザイン	1
1.2 ハードウェア・リファレンスデザイン	1
1.2.1 Cortex-M1	1
1.2.2 AHB-Lite	2
2 ソフトウェア・リファレンスデザイン	3
2.1 ソフトウェア環境.....	3
2.2 リファレンスデザインのインポート	3
2.3 ROM の構成	3
2.4 コンパイル.....	4
2.5 ダウンロード.....	5
2.6 リファレンスマニュアル	6
3 ハードウェア設計.....	7
3.1 ハードウェア環境.....	7
3.2 ソフトウェア環境.....	7
3.3 ハードウェア・リファレンスデザインのインポート	7
3.4 合成.....	8
3.5 配置配線	9
3.6 ダウンロード.....	9
3.7 リファレンスマニュアル.....	10
4 リファレンスデザインのデバッグ方法	12
4.1 ハードウェア設計のデバッグ方法	12
4.2 ソフトウェア設計のデバッグ方法	12
4.2.1 エミュレーターデバッグ	12
4.2.2 シリアルデバッグ	12

4.3 リファレンスマニュアル..... 12

図一覧

表 1-1 リファレンスデザインでの Cortex-M1 の構成.....	1
表 1-2 リファレンスデザインでの AHB-Lite の周辺機器インターフェース.....	2
図 2-1 リファレンスデザインのインポート.....	3
図 2-2 ROM の開始アドレスと Size の設定	4
図 2-3 コンパイル.....	4
図 2-4 Device configuration.....	5
図 3-1 ハードウェア・リファレンスデザインのインポート	8
表 3-1 ハードウェア・リファレンスデザインの説明	8
図 3-2 リファレンスデザインの合成.....	9
図 3-3 配置配線	9
図 3-4 Device configuration.....	10

表一覧

表 1-1 リファレンスデザインでの Cortex-M1 の構成.....	1
表 1-2 リファレンスデザインでの AHB-Lite の周辺機器インターフェース.....	2
表 3-1 ハードウェア・リファレンスデザインの説明.....	8

1 リファレンスデザイン

1.1 ソフトウェア・リファレンスデザイン

ARM Keil MDK (V5.26 以降) および GOWIN MCU Designer (V1.1 以降) を使用した Gowin_EMPU_M1 のソフトウェアプログラミングの [リファレンスデザイン](#) :

- Gowin_EMPU_M1¥ref_design¥MCU_RefDesign¥Keil_RefDesign
- Gowin_EMPU_M1¥ref_design¥MCU_RefDesign¥GMD_RefDesign

1.2 ハードウェア・リファレンスデザイン

Gowin_EMPU_M1 ハードウェア・リファレンスデザイン :

- Gowin_EMPU_M1¥ref_design¥FPGA_RefDesign¥Debug_RefDesign
- Gowin_EMPU_M1¥ref_design¥FPGA_RefDesign¥NoDebug_RefDesign

DK-START-GW2A18 V2.0 開発ボードのリファレンスデザインを例に説明します。このリファレンスデザインは、次の 2 つの部分で構成されています。

- Cortex-M1 コア
- AHB-Lite インターフェースと拡張された AHB および APB 周辺機器インターフェース

1.2.1 Cortex-M1

Cortex-M1 の構成を表 1-1 に示します。

表 1-1 リファレンスデザインでの Cortex-M1 の構成

機能モード	構成
外部割り込みの数	32 個
拡張オペレーティングシステム	サポートする
データ保存形式	リトルエディアン形式
乗算器	標準モード
デバッガ	<ul style="list-style-type: none"> ● フルモード (4 つのブレークポイントと 2 つの観測ポイントあり)

機能モード	構成
	● デバッガを無効にする
デバッグポート	JTAG および Serial Wire インターフェースをサポート
ITCM Select	Internal Instruction Memory
ITCM Size	32KB
ITCM 初期値	サポートする
ITCM 初期値ファイルのパス	32KB Bootload
DTCM Select	Internal Data Memory
DTCM Size	32KB

1.2.2 AHB-Lite

AHB-Lite の周辺機器インターフェースは表 1-2 に示すとおりです。

表 1-2 リファレンスデザインでの AHB-Lite の周辺機器インターフェース

周辺機器インターフェース	構成
汎用非同期送受信回路 (UART0)	サポートする
汎用非同期送受信回路 (UART1)	サポートする
汎用入出力ポート (GPIO)	サポートする
タイマー (Timer0)	サポートする
タイマー (Timer1)	サポートする
ウォッチドッグ (WatchDog)	サポートする
リアルタイムクロック (RTC)	サポートする
真性乱数生成器 (TRNG)	サポートする
デュアルタイマー (DualTimer)	サポートする
内部集積回路 (I2C)	サポートする
シリアル・ペリフェラル・インターフェース (SPI)	サポートする
SD カード (SD-Card)	サポートする
イーサネット (Ethernet)	サポートする
シリアル・ペリフェラル・インターフェース・フラッシュメモリ (SPI-Flash)	サポートする
APB Master [1]	サポートする
AHB Master [1]	サポートする

2 ソフトウェア・リファレンスデザイン

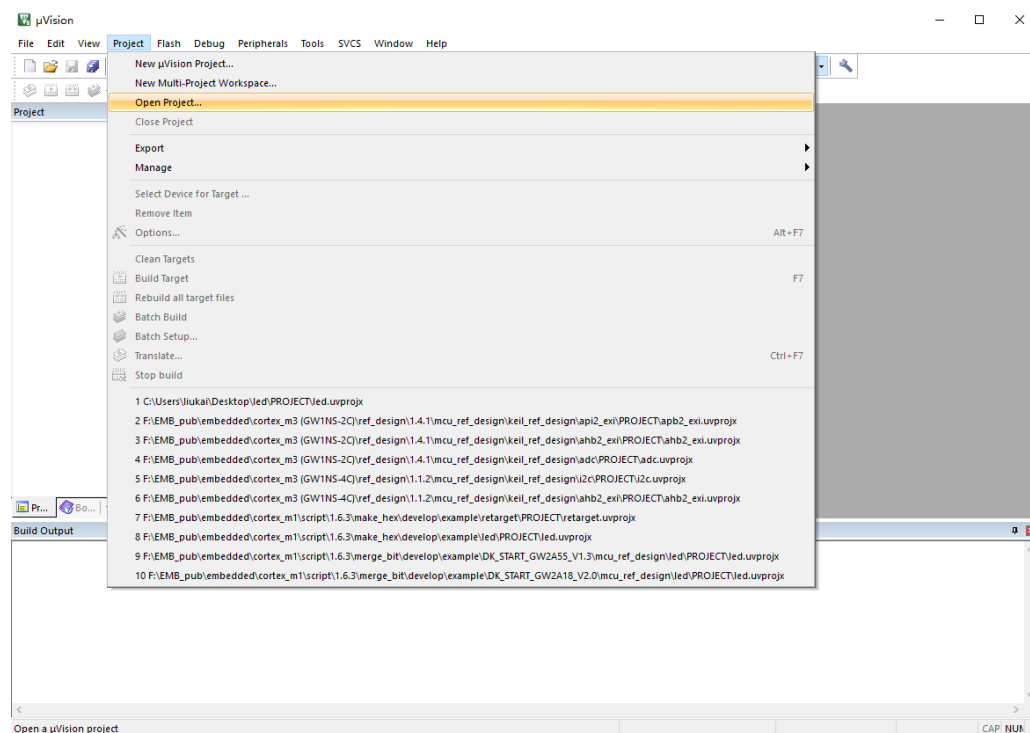
2.1 ソフトウェア環境

- ARM Keil MDK (V5.26 以降)
- GOWIN MCU Designer(V1.1 以降)

2.2 リファレンスデザインのインポート

ダブルクリックして ARM Keil MDK を開き、メニューバーの “Project > Open Project…” を選択し、ソフトウェア・リファレンスデザインをインポートします (図 2-1)。

図 2-1 リファレンスデザインのインポート

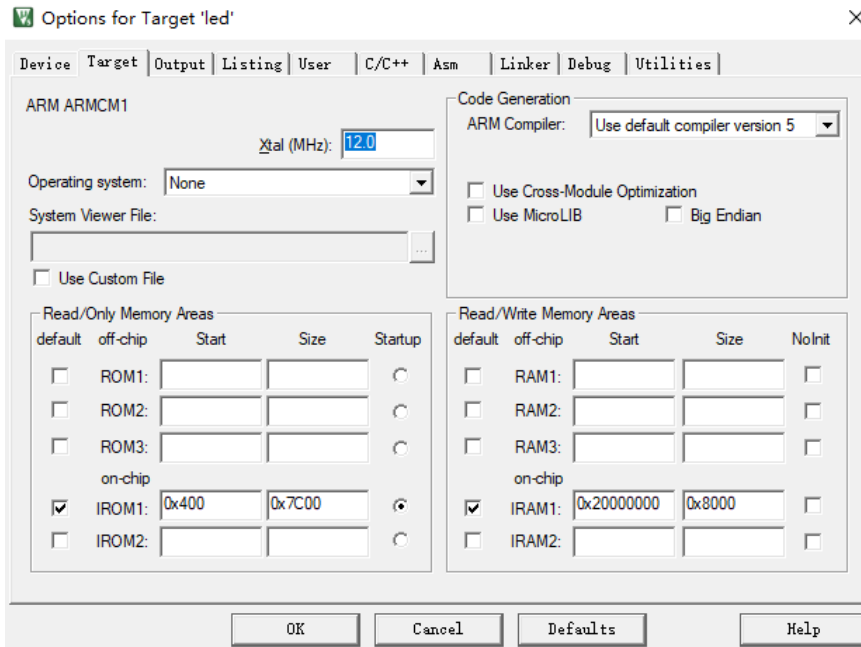


2.3 ROM の構成

DK-START-GW2A18 V2.0 開発ボードのリファレンスデザインを例に説

明します。オフチップ SPI-Flash によるダウンロードを使用する場合、ROM 開始アドレスを 0x400 に設定し、ROM Size を 0x7C00 に設定する必要があります (図 2-2)。

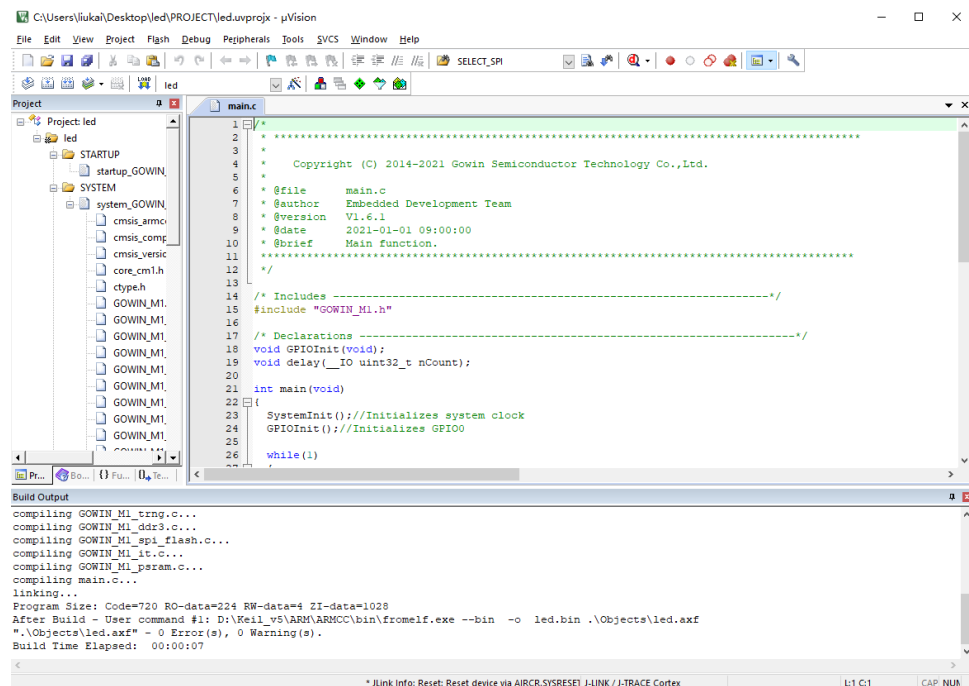
図 2-2 ROM の開始アドレスと Size の設定



2.4 コンパイル

ツールバーでコンパイルボタン “” をクリックし、リファレンスデザインをコンパイルして BIN ファイルを出力します (図 2-3)。

図 2-3 コンパイル



2.5 ダウンロード

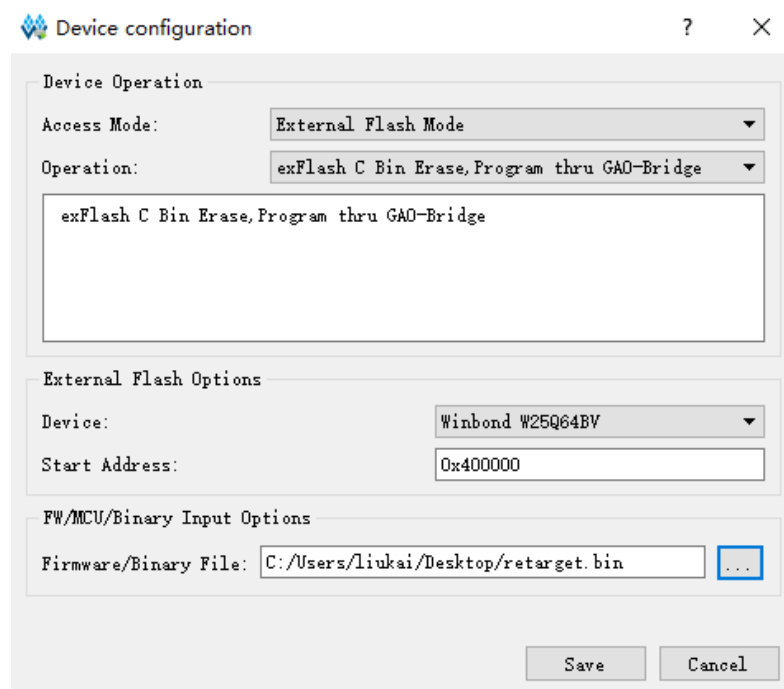
DK-START-GW2A18 V2.0 開発ボードのリファレンスデザインを例に説明します。


Programmer を使用してソフトウェア設計の BIN ファイルをダウンロードします。

Gowin ソフトウェアインターフェースまたはインストールパスで Programmer を開き、“Programmer > Edit>Configure Device” またはツールバーの “Configure Device” をクリックして、“Device configuration” ダイアログボックスを開きます。

- “Access Mode” のプルダウンリストから “External Flash Mode” を選択します。
- “Operation” のプルダウンリストから “exFlash C Bin Erase, Program thru GAO-Bridge” または “exFlash C Bin Erase, Program, Verify thru GAO-Bridge” を選択します。
- “FW/MCU/Binary Input Options > Firmware/Binary File” を選択して BIN ファイルをダウンロードします。
- オンボード Flash チップのタイプに応じて “External Flash Options > Device” を選択します。例えば、DK-START-GW2A18 V2.0 開発ボードの場合、Winbond W25Q64BV を選択します。
- “External Flash Options > Start Address” は “0x400000” に構成します。
- 最後に “Save” をクリックします (図 2-4)。

図 2-4 Device configuration



Device configuration が完了したら、Programmer のツールバーの“Program/Configure” () をクリックして BIN ファイルをダウンロードします。

2.6 リファレンスマニュアル

Gowin_EMPU_M1 のソフトウェア設計方法については、以下のマニュアルを参照して下さい：

- Gowin_EMPU_M1 ソフトウェアプログラミング リファレンスマニュアル ([IPUG533](#))
- Gowin_EMPU_M1 IDE ソフトウェア・リファレンスマニュアル ([IPUG536](#))
- Gowin_EMPU_M1 ダウンロード リファレンスマニュアル ([IPUG532](#))
- Gowin_EMPU_M1 シリアルデバッグ・リファレンスマニュアル ([IPUG535](#))
- Gowin Programmer ユーザーガイド ([SUG502](#))

3 ハードウェア設計

3.1 ハードウェア環境

DK-START-GW2A18 V2.0

- GW2A-LV18PG256C8/I7
- GW2A-18C (C バージョン)

3.2 ソフトウェア環境

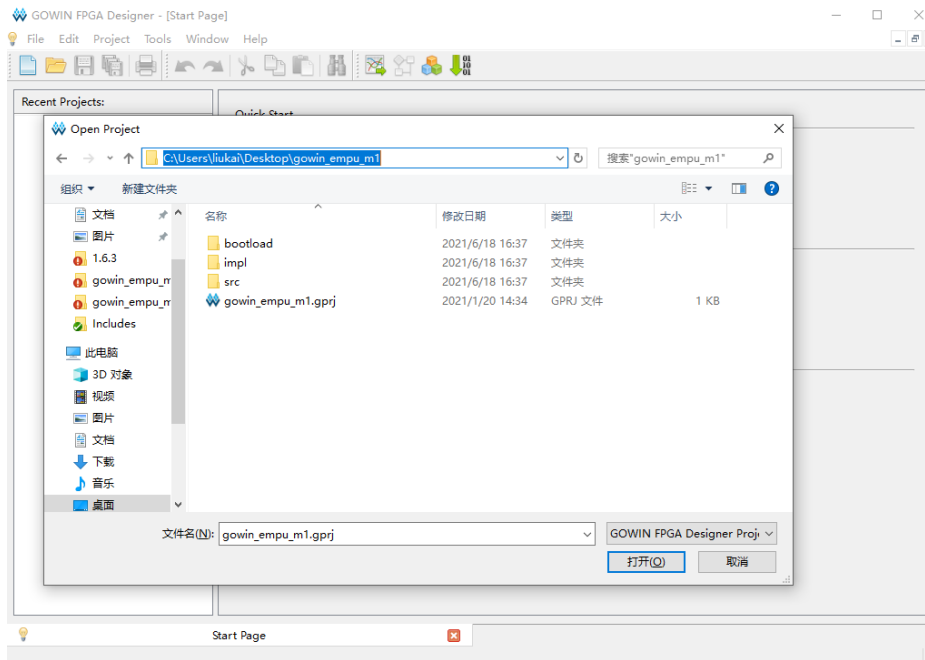
Gowin_V1.9.8 Beta 以降

3.3 ハードウェア・リファレンスデザインのインポート

ダブルクリックして Gowin ソフトウェアを開き、メニューバーで“File > Open…”を選択してハードウェア・リファレンスデザインをインポートします (図 3-1)。

または、IP Core Generator を開いて必要に応じて Gowin_EMPU_M1 ハードウェア設計を生成します。

図 3-1 ハードウェア・リファレンスデザインのインポート



DK-START-GW2A18 V2.0 開発ボードのリファレンスデザインを例に説明します。ハードウェア・リファレンスデザインのファイルの説明を表 3-1 に示します。

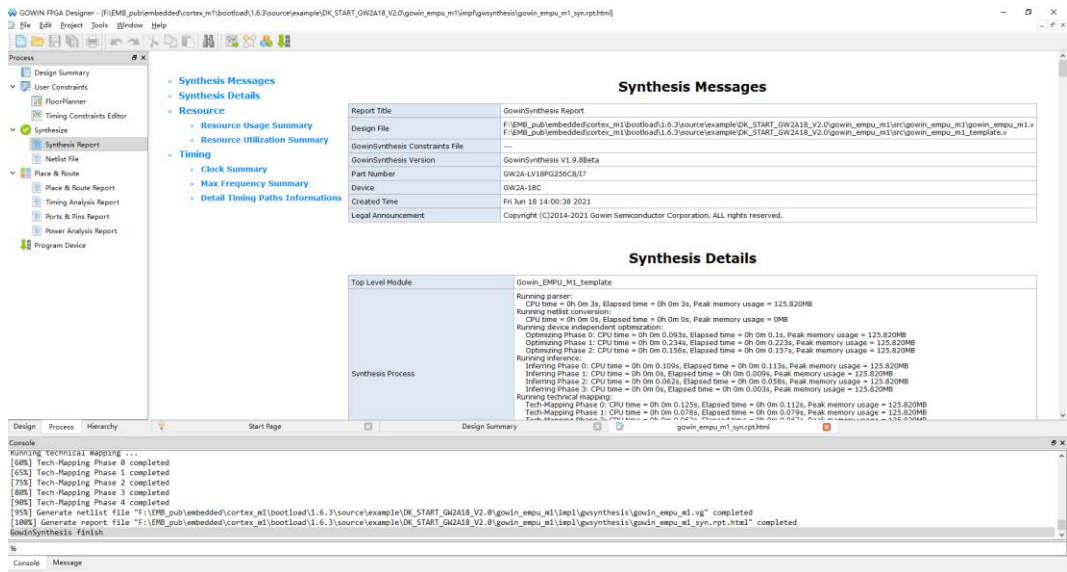
表 3-1 ハードウェア・リファレンスデザインの説明

ファイル	説明
gowin_empu_m1.v	Gowin_EMPU_M1 ハードウェア設計
gowin_empu_m1_template.v	Gowin_EMPU_M1 Top Module のインスタンス化とユーザーデザイン
ahb_multiple.v	AHB Master [1]拡張
apb_multiple.v	APB Master [1]拡張
gowin_empu_m1.cst	物理制約

3.4 合成

図 3-2 に示すように、合成ツール GowinSynthesis を実行することで、ハードウェア・リファレンスデザインを合成して Netlist File を生成します。

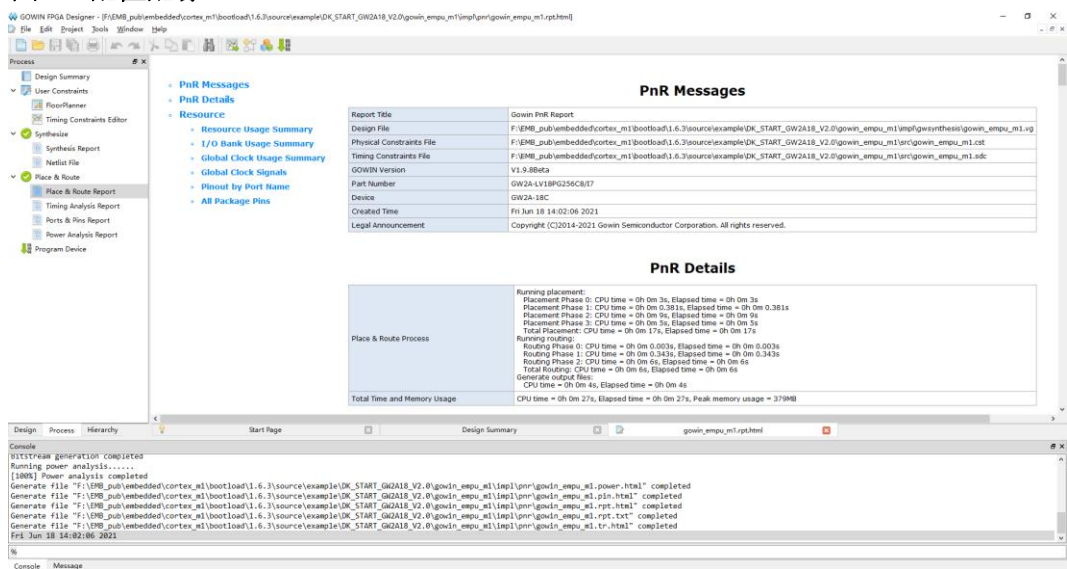
図 3-2 リファレンスデザインの合成



3.5 配置配線

合成が完了したら、配置配線ツールの Place & Route を実行してビットストリームファイルを生成します（図 3-3）。


図 3-3 配置配線



3.6 ダウンロード

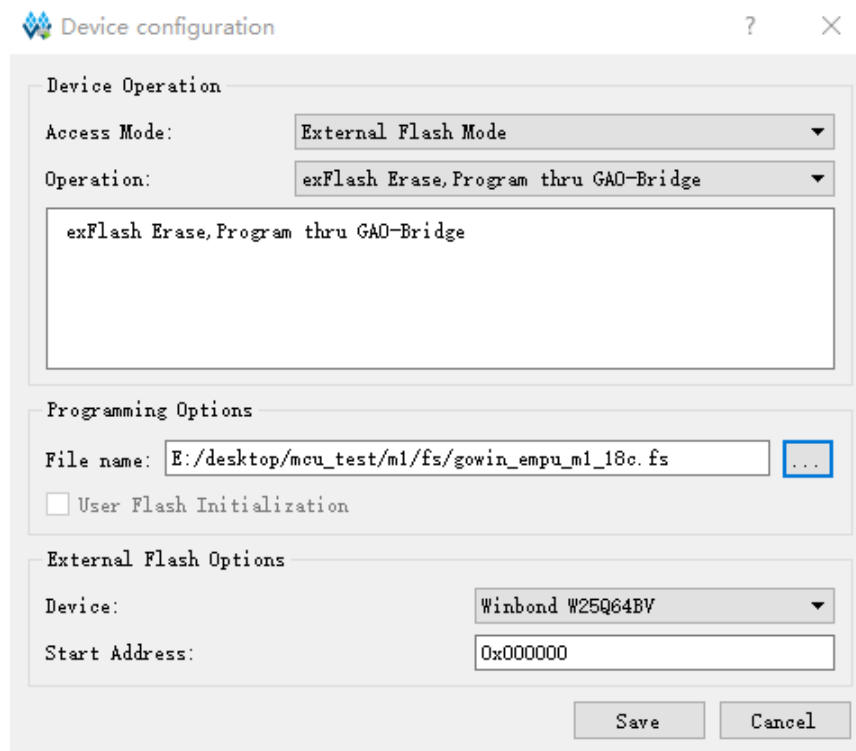
ソフトウェア開発キット DK-START-GW2A18 V2.0 内のリファレンスデザインを例に説明します。


ダウンロードツールである Programmer を使用してビットストリームファイルをダウンロードします。

“Programmer > Edit > Configure Device” またはツールバーの “Configure Device ” をクリックして、“Device configuration” ダイアログボックスを開きます。

- “Access Mode” のプルダウンリストから “External Flash Mode” を選択します。
- “Operation” のプルダウンリストから “exFlash Erase, Program thru GAO-Bridge” または “exFlash Erase, Program, Verify thru GAO-Bridge” を選択します。
- Programming Options > File name を選択してダウンロードされるビットストリームファイルをインポートします。
- オンボード Flash チップのタイプに応じて “External Flash Options > Device” を選択します。例えば、DK-START-GW2A18 V2.0 開発ボードの場合、Winbond W25Q64BV を選択します。
- “External Flash Options > Start Address” は “0x000000” に構成します。
- 最後に “Save” をクリックします (図 3-4)。

図 3-4 Device configuration



Device configuration が完了したら、Programmer のツールバーの Program/Configure “” をクリックしてビットストリームファイルをダウンロードします。

3.7 リファレンスマニュアル

Gowin_EMPU_M1 ハードウェア設計については、以下のマニュアルを参照して下さい：

- Gowin_EMPU_M1 ハードウェア設計リファレンスマニュアル ([IPUG531](#))
- Gowin ソフトウェア ユーザーガイド ([SUG100](#))
- Gowin 設計制約ユーザーガイド ([SUG101](#))

- Gowin Programmer ユーザーガイド ([SUG502](#))

4 リファレンスデザインのデバッグ方法

4.1 ハードウェア設計のデバッグ方法

GAO アナライザオシロスコープを使用して Gowin_EMPU_M1 ハードウェア設計をデバッグできます。

4.2 ソフトウェア設計のデバッグ方法

2つの Gowin_EMPU_M1 ソフトウェアプログラミング・デバッグ方法をサポートしています。

- エミュレーターのオンラインデバッグ
- シリアルデバッグ

4.2.1 エミュレーターデバッグ

エミュレータータイプ

Gowin_EMPU_M1 は、次の 2つのエミュレーターによるシングルステップ・デバッグをサポートしています。

- J-LINK エミュレーター
- U-LINK エミュレーター

デバッグポート

Gowin_EMPU_M1 は以下のデバッグポートをサポートしています：

- JTAG
- Serial Wire

4.2.2 シリアルデバッグ

シリアルポートおよびシリアルデバッグ・アシスタントを使用して、実行ステータスを印刷・出力します。

4.3 リファレンスマニュアル

Gowin_EMPU_M1 のデバッグ方法に関するリファレンスマニュアル：

- Gowin アナライザオシロスコープ ユーザーガイド([SUG114](#))

- Gowin_EMPU_M1 IDE リファレンスマニュアル ([IPUG536](#))
- Gowin_EMPU_M1 シリアルデバッグ リファレンスマニュアル ([IPUG535](#))

