オープンソースISAであるRISC-Vの ビジネスにIARコンパイラが不可欠な理由

IAR　　技術部FAE　　赤星 博輝

2023年6月20日

目次

- 会社紹介
- オープンソースISAであるRISC-Vのメリットは？
- RISC-Vのビジネスでのソフトの重要性
- オープンソースISAのRISC-Vのコンパイラはオープンソース？
- IARのコンパイラの特徴
- IARのコンパイラをRISC-Vで使うべき理由
- まとめ

IAR会社紹介

会社概要

設立：1983年に設立
本社：スウェーデン
拠点：世界に12のローカルオフィス
上場：NASDAQ Stockholmに上場
事業：組込みソフトウェアソリューション

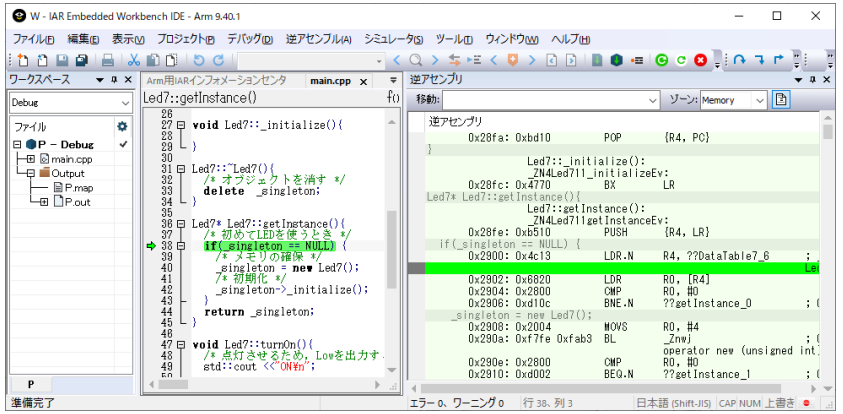
日本法人 –実績・安心サポート

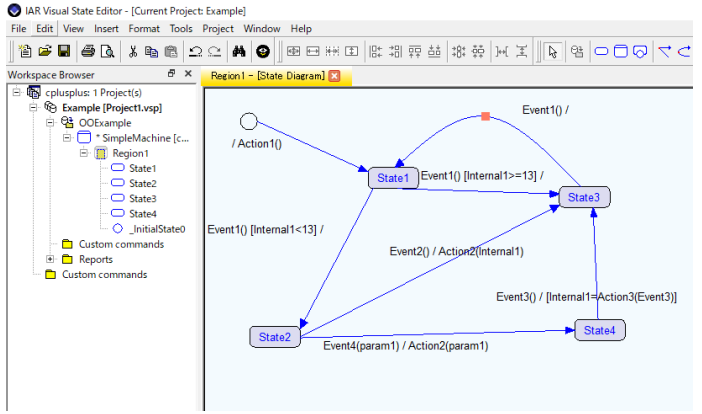
設立20年以上
現地法人の中で最大規模
日本人エンジニアでのサポート
車載ソフトに精通したエキスパート

販売製品

組込み開発ソリューション
IAR Embedded Workbench™
(C/C++コンパイラ/デバッガ)
C-STAT ™ / C-RUN ™ (静的・動的解析)
IAR Visual State ™ (グラフィカルモデリング)

組込みセキュリティソリューション
Embedded Trust™/SDP





オープンソースISAであるRISC-Vのメリットは？

綺麗な命令セット

データに関する部分を共通化して、ハード設計を容易化。

	31–25	24–20	19–15	14–12	11–7	6–0
R形式	funct7	rs2	rs1	funct3	rd	Opcode(7)
I形式	imm[11:0]		rs1	funct3	rd	Opcode(7)
S形式	imm[11:5]	rs2	rs1	funct3	imm[4:0]	Opcode(7)
B形式	imm[12],[10:5]	rs2	rs1	funct3	imm[4:1],[11]	Opcode(7)
U形式	imm[31:12]				rd	Opcode(7)
R形式	imm[20],[10:1],[11]		imm[19:12]		rd	Opcode(7)

オープンソースで誰にも支配されない

規格は、RISC-V International contributing members が the RISC-V International Technical Working Groupsで決定

モジュール構成：基本命令＋拡張命令＋カスタム命令

RV32I
RV32E
RV64I
RV64E
RV128I

基本命令

＋ M A C F D Q...

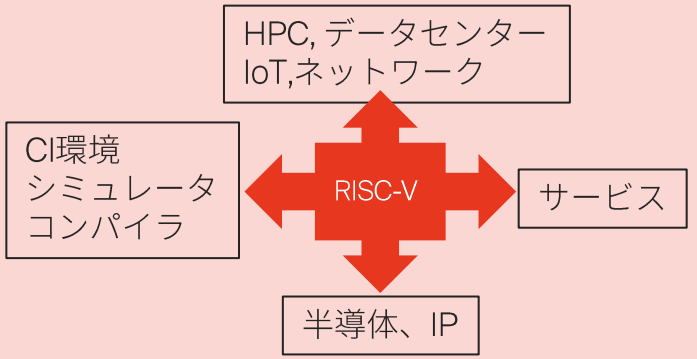
拡張命令

＋

inst[4:2] / inst[6:5]	000	001	010	011	100	101	110	111 (> 32b)
00	LOAD	LOAD-FP	custom-0	MISC-MEM	OP-IMM	AUIPC	OP-IMM-32	48b
01	STORE	STORE-FP	custom-1	AMO	OP	LUI	OP-32	64b
10	MADD	MSUB	NMSUB	NMADD	OP-FP	reserved	custom-2/rv128	48b
11	BRANCH	JALR	reserved	JAL	SYSTEM	reserved	custom-3/rv128	≥ 80b

カスタム命令で独自拡張も可能

RISC-Vエコシステムが拡大中

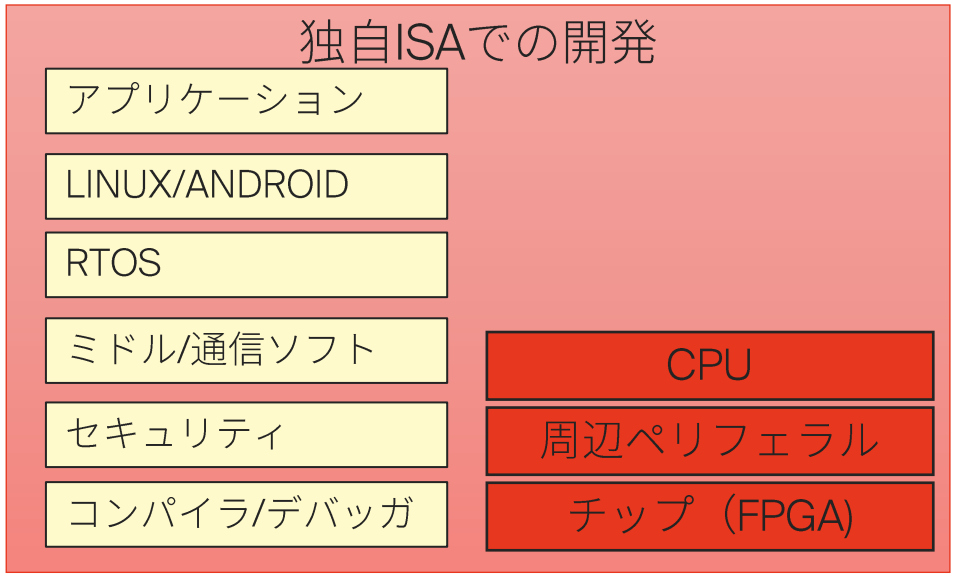


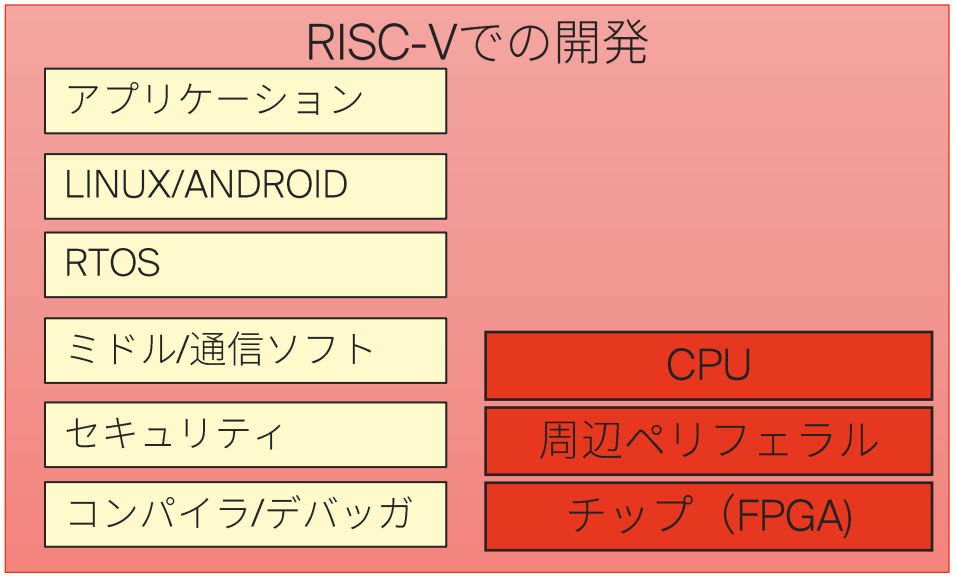
様々な会社が
様々なサービスを
提供開始

エコシステムがRISC-Vの重要なポイント

今から独自ISAではパートナーづくりが難しい。
下手すると、すべて自分達で作る必要がある。

RISC-Vの場合はエコシステムが出来ており、
必要なモジュールを調達可能。
　◎ 苦手な部分は得意なパートナーに依存出来る
また、あとから差し替えることも可能。
　例： CPU⇒最初は、既存の物を使い、
　　後からカスタム命令を実装した独自RISC-Vを設計





RISC-Vのビジネスでは、ソフトの重要性が高くなる

開発企画、ハード企画する場合には、ぜひソフトウェアについて検討をお願いします。

- 日本では、ハードが先行することが多く、ソフトが後回しになることがありますが、
- 早期のビジネス立上げをするためには、ソフト側の検討も早期にやることが重要です。

ハード側を意識すると

RV32I, RV32E, RV64I, RV128I
拡張命令
カスタム命令

パイプライン段数、
発行命令数
PMP、
マルチコア

どういったもの用意する必要があるのか？

- アプリケーション
- LINUX/ANDROID
- RTOS
- ミドルウェア/通信ソフト
- セキュリティ
- コンパイラ

独自ですべて開発は厳しい
さらに、競争力も必要

エコシステムを早く検討し、
動くことが重要です

QCD(品質、費用、納期）を
考えて選択必要です

オープンソースISAのRISC-Vのコンパイラはオープンソース？

さまざまなプロジェクトに対応するためには、商用コンパイラの対応も重要。

オープンソースISA

RV32E:32ビット命令、汎用レジスタ16本
RV32I: 32ビット命令、汎用レジスタ32本
RV64I: 64ビット命令、汎用ジスタ32本
RV128I: 128ビット命令、汎用ジスタ32本

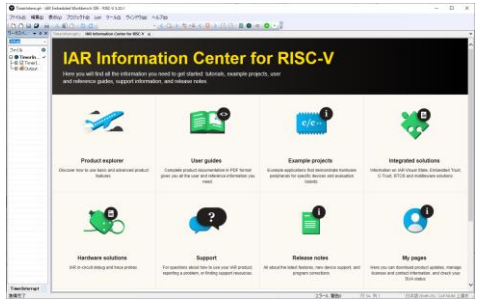
M： 乗除算
A： アトミック命令
F：浮動小数点演算
B：ビット操作
他にも多くの拡張あり

独自にカスタム命令を追加可能

オープンソースソフトウェア(OSS)

GCC

商用ソフトウェア



プロジェクトの性質

(1) Linux/Androidを使用
(2) コンパイラ自体も改造したい

(3) 機能安全対応が必要な場合
(4) コンパイラへのサポートが必要
(5) すでに資産をお持ちの方
IARの40年の歴史

IARのコンパイラは、RV32I,RV32E, RV64I、
拡張命令M,A,F,D,Bなどに加え、カスタム命令にも対応

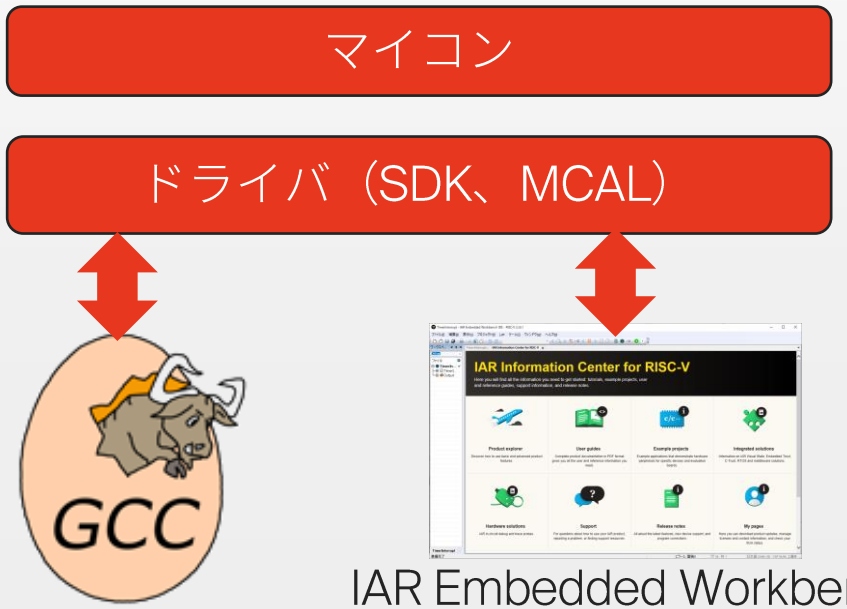
コンパイラではどういう事がおきるでしょうか？

近年のマイコンでは、複数のコンパイラに対応するのが普通。

- コンパイラはソフトウェア開発のベースなので、重要度が高いです。
  - ドライバなど開発する時には、GCCだけでなくIARの対応もお忘れなく

Linux/Androidなど

組込み系（RTOS/Baremetal/OS無し）



どちらにも
対応しておくことで
多くのユーザに対応可能

IARのコンパイラの特徴

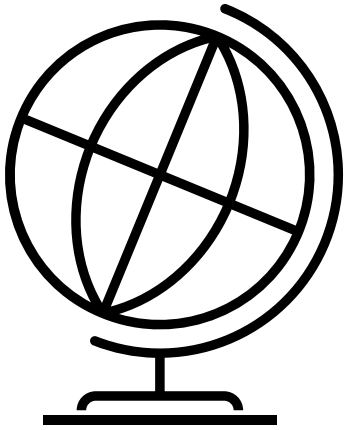
- これまでの40年での組込み向けコンパイラを開発
- 組込みシステム向けの最適化
- 使用するマイコンごとにデバイス対応し、即座に組込みシステム開発可能
  ◆デバイス設定、リンカ設定、デバッグ情報、内蔵フラッシュへのダウンローダ
- オプション機能：コード解析ツールが簡単に使用可能
- ビルドからデバッグまでAll In One
- 新しい技術への対応も実施
  ◆マルチコアデバッグ機能、VisualStudioCode対応、デバッガでのtrace対応など

本日は、RISC-Vで使うべき理由を5つほどご紹介させていただきます

理由１：すでに多くのユーザが存在

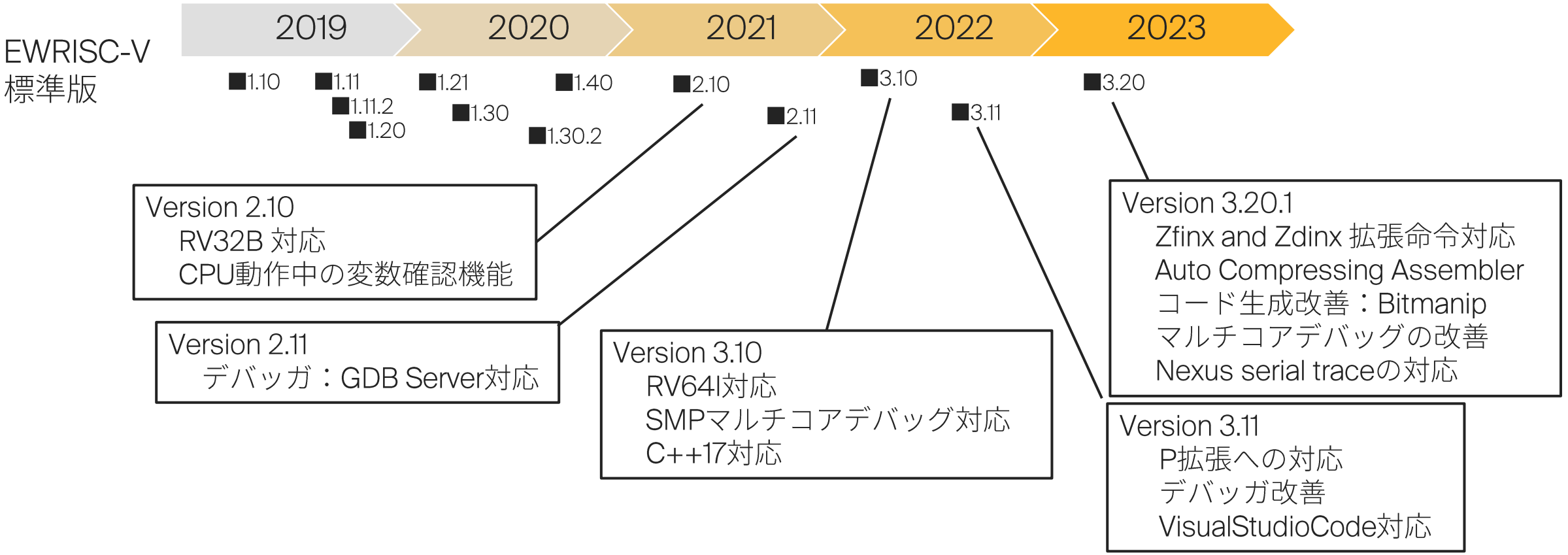
- 様々なCPUコアに対応してきて、IAR Embedded Workbenchを使っているユーザがすでに存在。
- 慣れた開発環境を使いたいユーザがかなりのボリュームで存在しています。

世界にIARのユーザが存在し、
ソフト開発を実施している。



理由２：IARコンパイラの迅速なRISC-V対応

標準版も5年で13回のリリースを実施し、新しい技術に追従し、より良いものに完成度を上げる
独自ISAで独自でコンパイラを作っていた時との差分になると思います。



理由３：機能安全対応コンパイラ

TÜV SÜDによる認証を受けているため、ユーザ認定を簡素化
不具合情報を定期レポートで提供しているため、設計やコーディングガイドラインへの反映が容易
セーフティガイドを提供しており、設計やコーディングのガイドラインの作成に活用可能
メーカーサポートによる、さらに日本語でのサポート

- 厳格な認定プロセスを経た開発ツール
  - TÜV SÜDによる認証
- 以下の機能安全規格に対応
  - IEC 61508（工業）
  - ISO 26262（車載）
  - EN 50128, EN 50657（鉄道）
  - IEC 62304（医療）
  - ISO 25119（農林業）
  - IEC 62061（機械）
  - EN ISO 13849-1（電気・機械部品）
  - IEC 61511-1（プロセス産業）
  - ICE 60730-1（家電）

■ ツール認定を省ける点がメリット
「ツール認定はソフトウェアツールを提供するベンダーではなく、
ソフトウェアツールを使用するユーザーに求められます。」
ですが、きちんと認定されたツールだと証明書が利用できます。

■ 不具合情報を定期レポートとして提供

■ 設計ガイドライン、コーディング・ガイドライン、開発ツールの操作ガイドの定義が必要。
セーフティガイドの提供
定期レポートからこうしたガイドに反映

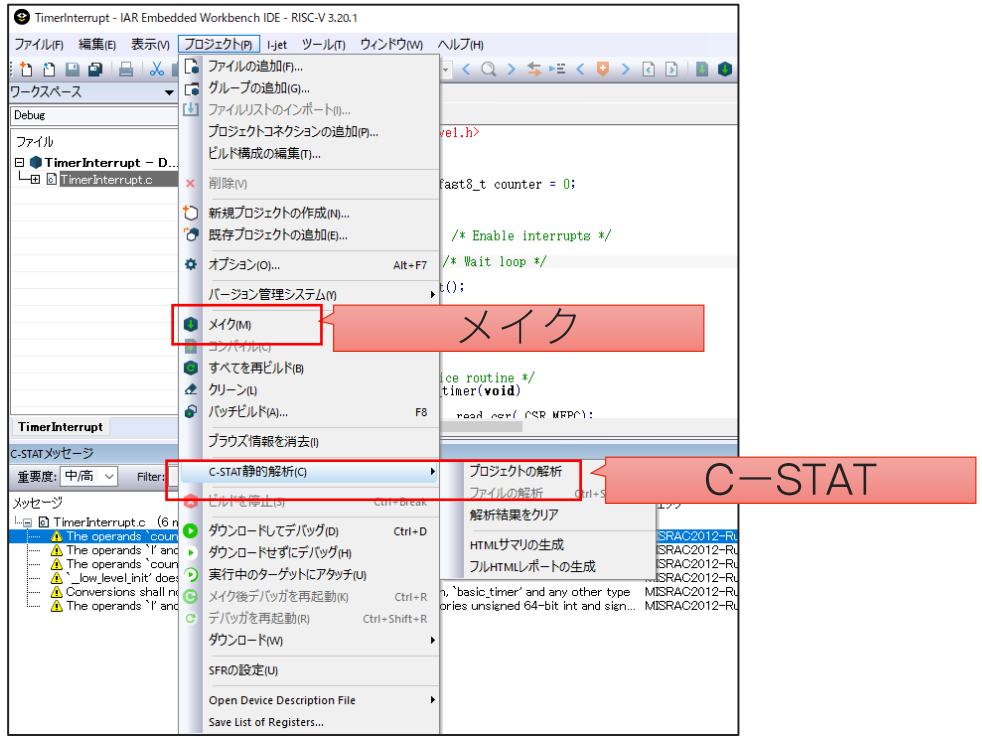
■ メーカーサポート
日本語での対応OK

理由４：安価なMISRA-C/C++チェッカ

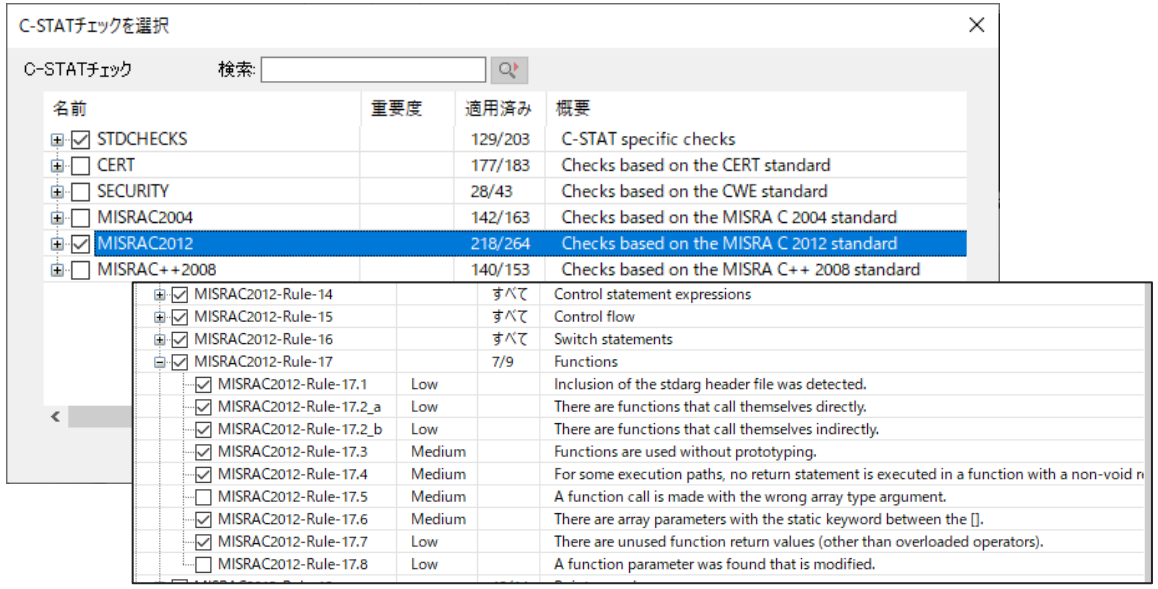
EWRISC-Vでは、C-STATを用いてMISRA-C/C++ルールを安価でかつ、容易に利用できます。

MISRA-C/C++は、安全性と移植性と信頼性を目的とするコーディングガイドです。
EWRISC-Vでは、メイクをするの同様に、容易にMISRA-C/C++チェックを実行することが可能。
　複数のコンパイラを使用する場合やデータ幅の異なる(32bit/64bit)ケースでの動作を考えると重要。

容易なオペレーション



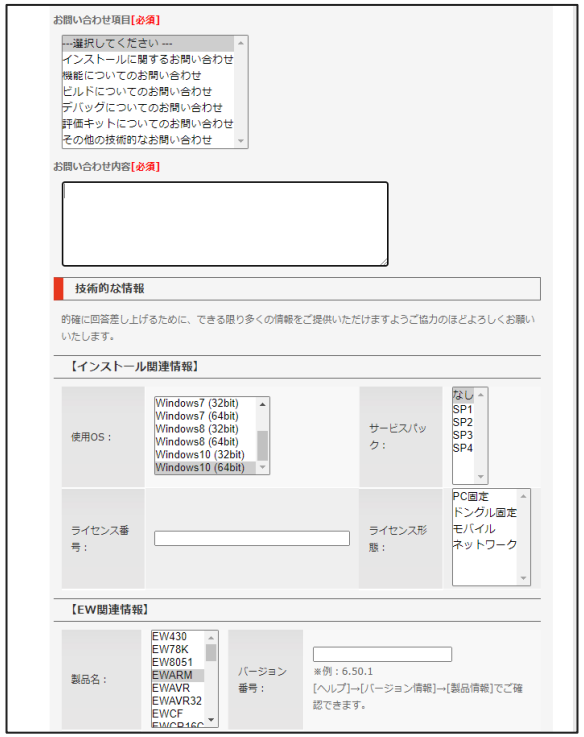
ルール選択もGUIで容易に可能



理由５：IARの技術資料＋サポート

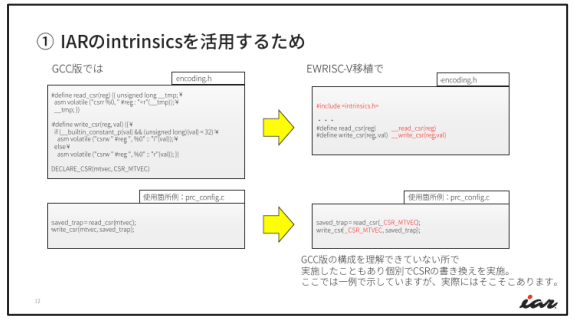
メーカーによる直接サポート、かつ、日本語でOKという点を活用頂いています。
各種技術資料を公開させていただいております。

サポートフォーム
ユーザはこちらから質問可能

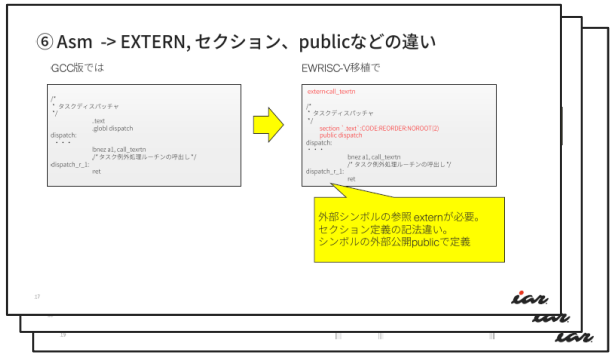


IARの学習コンテンツ
https://www.iar.com/jp/knowledge/learn/

【限定配布】TOPPERS-ASPの移植
GCC からEWRISC-Vでの変更点



• • •



まとめ

- RISC-Vを活用するポイントは、エコシステムです。
- エコシステムをうまく使う事で、よりビジネスを加速できます。
- オープンソースISAをサポートするコンパイラは、オープンソースソフトウェアだけではありません。
- IARはコンパイラの専業メーカで４０年の歴史があります。
  - 世界中で使われており、150,000ユーザが実際に使ってソフト開発をしています。
  - IARでは通常版とは別に機能安対応コンパイラを販売しています。
    - 車載、産業機械、医療などの世界では機能安全対応コンパイラが必須です。

ご清聴ありがとうございました