

次世代プログラミング言語 Rust の使用が可能に、 SOLID ソフトウェア開発環境 Ver.3.0 をリリースします

2021 年 10 月 20 日

京都マイクロコンピュータ株式会社

京都マイクロコンピュータ株式会社（以下、KMC と記載）は、本年 12 月より KMC のリアルタイム OS（以下、RTOS と記載）ベースの組込みソフトウェア開発プラットフォーム SOLID（以下、SOLID と記載）に次世代プログラミング言語 Rust に対応した SOLID Ver.3.0 をリリースすることを発表します。

1 プログラミング言語 Rust に対応

KMC では以前からオープンソースプログラミング言語である Rust に注目しており、SOLID 開発環境での Rust 対応の開発を行ってきました。2021 年 9 月末には、SOLID が Rust に Tier 3 ターゲットとして追加されました^(*)。このあと、Rust コミュニティでリリースに向けた手順を経て、SOLID-OS(TOPPERS/ASP3 および FMP3)でのプログラミングに、Rust ライブラリ群の多くを利用できるようになり、SOLID-OS/TOPPERS のタスクやハンドラが Rust 言語で記述できるようになります。

一方で KMC は、RTOS を含んだ組込みソフトウェア開発プラットフォーム“SOLID”を 2017 年より提供開始し、多くのお客様にご使用いただいています。SOLID-OS は、GCC および LLVM/Clang に対応しており、商用利用可能な RTOS および開発環境ライセンスも提供しています。

今般、SOLID-OS サポートが Rust に追加されたことにより、弊社提供の SOLID も Rust を全面的に利用できるよう、バージョンアップいたします。

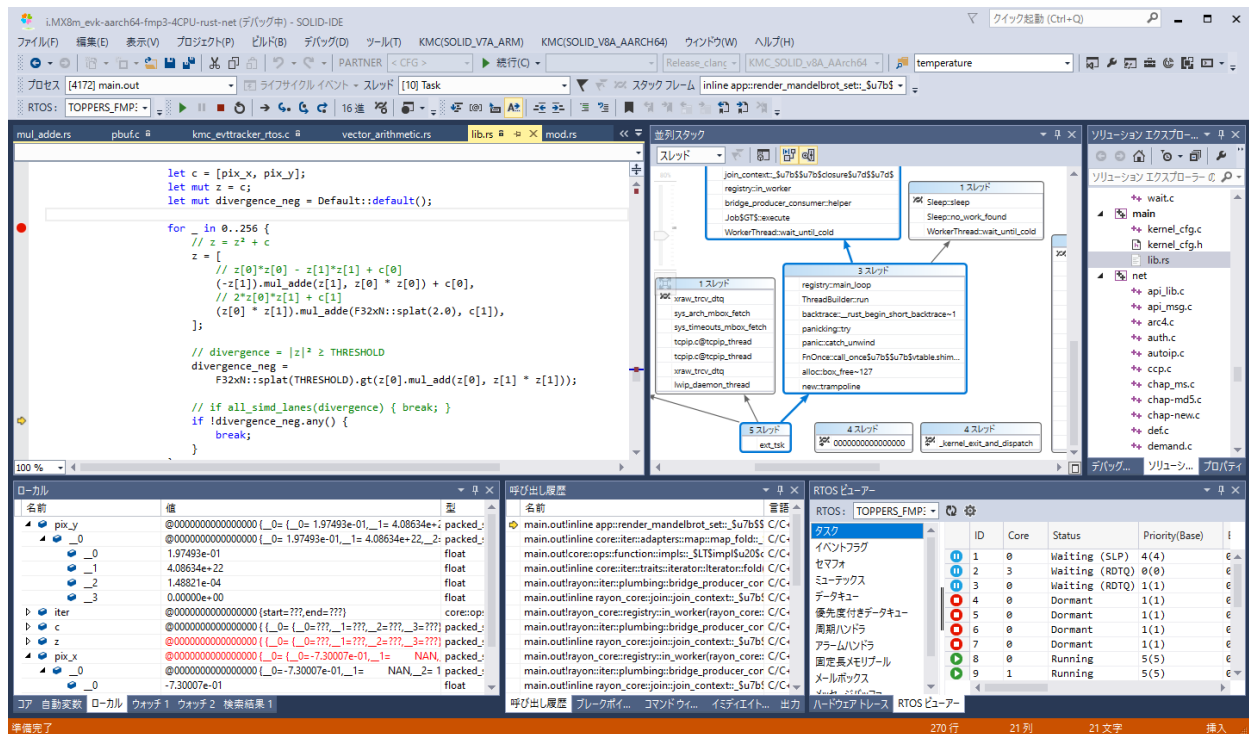
- ・ IDE ビルドツールの Rust 対応（Rust コンパイラは公式配布版のバイナリリリースを使います）
- ・ ソースコードデバッガの Rust 対応

SOLID Ver.3.0 のリリースは 2021 年 12 月頃を予定しています。

KMC は、今後も Rust コミュニティと継続的に協力し、組込みシステムでの Rust 言語の普及に努めてまいります。

(*) Tier 3 ターゲットでの対応になるので、Rust コンパイラは Nightly 版のみ使用できます。

<https://github.com/rust-lang/rust/commit/da9ca41c318aadcc0a9e8f72f86c1d36c8fe6d5d>



SOLID 開発環境 Rust 言語対応 (IDE による Rust ソースコードデバッグの様子)

2 SOLID Ver.3.0 の新機能 “スレッドサニタイザ”

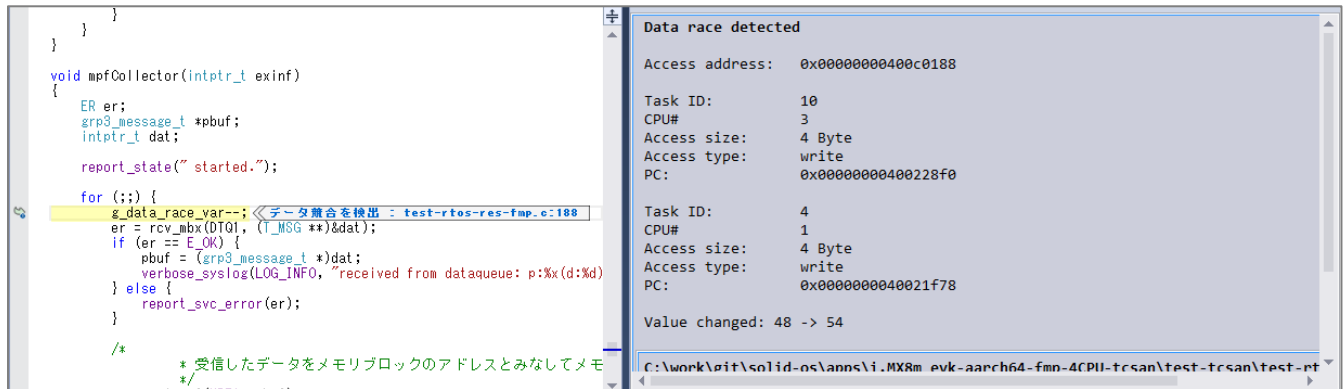
SOLID Ver.3.0 では、Rust 言語対応以外にも、新機能として LLVM/Clang 環境でスレッドサニタイザが利用可能となります。

スレッドサニタイザとは、実行時にスレッド（タスク）間でのデータアクセスの競合を検出する機能です。同期しない複数のスレッドが同じメモリをアクセスする場合に、少なくとも一つのスレッドが書き込みをしている場合の競合を検出し、デバグが競合箇所を特定してユーザーに通知します^(*)。

SOLID では大規模な RTOS システム設計が可能となるよう、DLL 機能や分散開発機能を備えていますが、プログラムが大規模化するとスレッド間のメモリ競合が発生した場合に、競合発生タイミングが不定期となりその原因を検出するのが難しいという課題がありました。スレッドサニタイザは、この問題を解決するのに大変有効な機能です。

(*) スレッドサニタイザは全ての競合検出を担保するものではありません。

変数競合検出によるパフォーマンス低下を少なくするため一定の確率での検出となりますが、検出できた競合についてはその競合が確定となります。



スレッドサニタイザによる変数競合検出の様子

3 今後の予定

今後、以下の予定で Rust 版 SOLID のご紹介、リリースを計画しています。

- ・ 2021 年 10 月 24 日～25 日に開催される、TOPPERS 開発者会議にて、当社のエンジニアが Rust について講演します。

(<https://solid.kmckk.com/SOLID/archives/5087>)

- ・ 2021 年 11 月 4 日～5 日に開催される、DSForum 2021 にて、当社のエンジニアが Rust について講演します。また、エンジニア講演の他に、弊社スポンサー講演にて、SOLID Ver.3.0 について、開発環境の、Rust 対応やスレッドサニタイザ対応について、デモンストレーション交えて紹介し、オンライン展示コーナーで、Rust 言語対応など SOLID Ver.3.0 の説明を行います。

(<https://solid.kmckk.com/SOLID/archives/5103>)

- ・ Rust 言語が利用可能な SOLID Ver.3.0 は、2021 年 12 月にリリースを予定しています。

現在 SOLID のライセンスをお持ちの方は、追加費用なしで Ver.3.0 にアップデートいただけます。また新規に SOLID ライセンス契約をされる場合も、従来とライセンス費用は変わりません。

4 名古屋大学 大学院情報学研究科

附属組込みシステム研究センター長・教授 高田広章様からのコメント

この度、京都マイクロコンピュータ株式会社の RTOS 開発プラットフォーム SOLID において、新しいプログラミング言語 Rust 対応版をリリースすることになり、大変心強く思います。

私は日頃から、情報技術ばかりでなく組込みシステム開発技術でも新しい技術を取り込んで革新させていくべき、と考えています。中でもプログラミング言語については「いつまでも C 言語でよいのか?」という問題意識を感じており、最新のオープンソースソフトウェアである Rust は TOPPERS プロジェクトでも大変注目している言語の一つです。

SOLID では、TOPPERS/ASP3 および FMP3 カーネルを Rust 対応するという事なので、この新しい開発環境に大いに期待しています。

5 京都マイクロコンピュータ株式会社

代表取締役社長 山本彰一のコメント

私達 KMC はこれまでも GCC や Linux をはじめとするオープンソースソフトウェア(OSS)開発環境関連製品をリリースしてきました。SOLID も多くの OSS 技術を利用しており、弊社エンジニアは日々 OSS に触れて開発をしています。

OSS の技術を取り込みまた貢献することで、組込みソフトウェアの開発環境をより良くする事を目標に長年ビジネスをしてきた点が KMC の大きな強みだと考えています。

Rust は、実行効率とソフトウェア安全性を言語レベルで保障可能な仕組みを持っているため、今後の肥大化するソフトウェアを支える大きな柱となっていくと考えています。この Rust を商用ツールとして市場に提供することは、KMC の使命です。また Rust 研究に取り組んできた若手エンジニアが新たに開発メンバーに加わり、開発に多大な貢献をしてくれた事も頼もしく思っています。

今後も KMC は開発環境の Rust 言語対応や、Rust の普及にも努めてまいります。

【付録 1】 開発プラットフォーム SOLID の概要

1) SOLID-IDE の特徴

SOLID-IDE は、Visual Studio®をベースに開発した統合開発環境で、エディタやビルダーそしてデバッグが含まれます。SOLID-IDE には実績が豊富な KMC 製 GCC コンパイラ(exeGCC)の他、LLVM/Clang コンパイラを含んでいます。

2) ソフトウェア品質向上と、開発効率向上に有効なデバッグ・解析支援機能

SOLID では、SOLID-OS にデバッグ支援用として強化したランタイムと、SOLID-IDE に含まれる Clang コンパイラの持つ充実した解析機能を連携させることにより、アドレスサニタイザによるバッファオーバーランなどの自動検出や、コンパイラ連携により簡単にコードカバレッジを表示すること、などを実現しています。また、他にも、スタック破壊などを実行時に自動検出したり、ソースコードの静的解析機能など、ソフトウェア品質向上と、開発効率向上に有効なデバッグ・解析支援機能を豊富にそろえています。

3) 大規模システム開発を効率よく行える支援機能

モジュール単体でのビルド・ロードおよびアンロードが行えるローダーを独自に開発しました。このローダーを利用することで、大規模ソフトウェアを多人数・多拠点で開発する場合に、他拠点のソースコードを共有することなく開発対象のソフトウェアだけのビルド・ロードをし、開発した部分だけを効率よくソースコードデバッグできます。

4) JTAG デバッガ PARTNER-Jet2 対応

当社の JTAG デバッガである PARTNER-Jet2 が使用できます。PARTNER-Jet2 はデバッグ対象の Arm®Cortex®v7-A 32bit/64bit プロセッサおよび Arm®Cortex®v8-A 64bit プロセッサに対応しています。Arm®Cortex®v8-A 64bit プロセッサのデバッグでは、A64 命令セットの解析や 64bit アドレス空間表示のほか、取得したトレース情報の 64bit アドレス空間表示が可能です。また Arm®Cortex®v8-A 64bit アーキテクチャで定義された 4 つの Exception Level(EL0～EL3)と、セキュア/非セキュア状態でのデバッグがしやすいよう、ブレーク条件やトレース結果に Exception Level とセキュア状態が設定／表示できるようにしました。さらにマルチプロセッサ構成をふまえて、ブレーク条件やトレース結果にプロセッサ ID の設定／表示が可能です。

SOLID について：

<https://solid.kmckk.com/SOLID/>

【付録 2】プログラミング言語 Rust について

1) Rust とは

Rust は「安全性」「速度」「並行性」の 3 つに焦点を合わせて開発が進められているプログラミング言語です。2006 年の開発初期はグレイドン・ホアレの個人プロジェクトでしたが、2009 年に Mozilla が開発に関わり始めてから Mozilla Research の公式プロジェクトとなり、現在では Rust Foundation を中心としてオープンソースプロジェクトとして活発に開発が行われています。2015 年にバージョン 1.0 がリリースされて以来、後方互換性を保って 6 週間間隔で新しいバージョンがリリースされています。

Rust は C++ のようなコンパイル型言語であり、例えば Windows 向けにコンパイルした場合、ネイティブコードが含まれる自己完結した exe ファイルを作成できます。C++ の特徴であるオーバーヘッドの低さを維持したまま安全なメモリ管理が容易に実現できることから、将来が期待されています。Firefox、Dropbox や Cloudflare のようなセキュリティとパフォーマンスの両立が求められるインターネット分野のソフトウェアはもちろん、組込み分野においても Rust の採用例が増えてきています。

2) Rust の特徴

・高実行効率

Rust は実行・メモリオーバーヘッドが低く、大きなランタイムやガベージコレクタを必要としないため、幅広い規模のハードウェア上で効率的に実行できます。

・高信頼性

Rust の表現力の高い型システムと所有権モデルによりメモリ安全性とスレッド安全性が保証されます。さらに様々な種類のバグをコンパイル時に排除することが可能です。

・高生産性

Rust には優れたドキュメント、有用なエラーメッセージを備えた使いやすいコンパイラ、統合されたパッケージマネージャ、ビルドツール、多数のエディタに対応するスマートな自動補完と型検査機能、自動フォーマッタといった一流のツール群が揃っています。

Rust について：

<https://www.rust-lang.org/ja>

本件に関するお問い合わせ：京都マイクロコンピュータ株式会社 solid-info@kmckk.co.jp

ここに記載されている製品名は各社の商標または登録商標です。