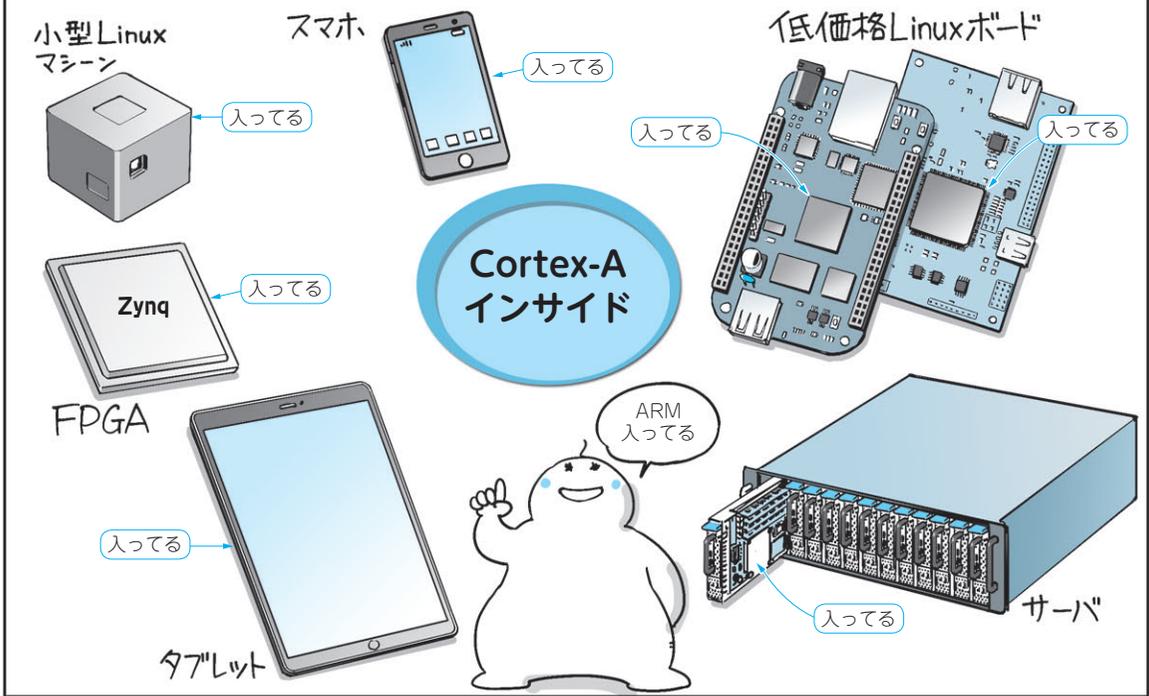


知らないうちに劇的に進化していた!

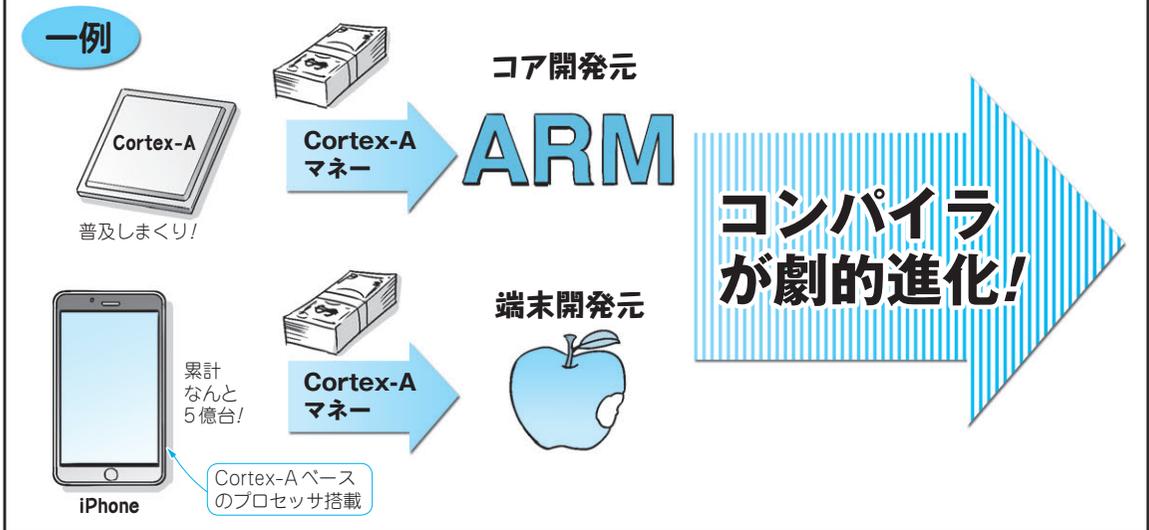
# コンパイラに新時代到来

編集部

## ちまたでは…ARM Cortex-A時代が到来!



## Cortex-Aマネーの恩恵でコンパイラが劇的進化中



# 使わない手はない!

高性能でフリー

**生成コードが高速に**

Cortex-A用コードが特に速い!

**エラー・メッセージが見やすく**

原因箇所だけをピンポイントに表示

あそこを直せばいいんだな!

**before**      コマンドがシンプルに      **after**

```
$ arm-eabi-gcc -c -mcpu=...
```

↑      ↑

そもそもコマンドが違う      オプションも言語やCPUでまちまち

➔

```
$ clang -target arm-eabi -mcpu=...
```

↑      ↑

コマンドは一つ      オプション指定することでさまざまなCPUに対応可能

**デバッグ機能が充実**

バグだらけだ!

自力では気付かなかった

デバッグ検出ツール

**最適化が使いやすい**

電力      高速      信頼性

次は消費電力重視で

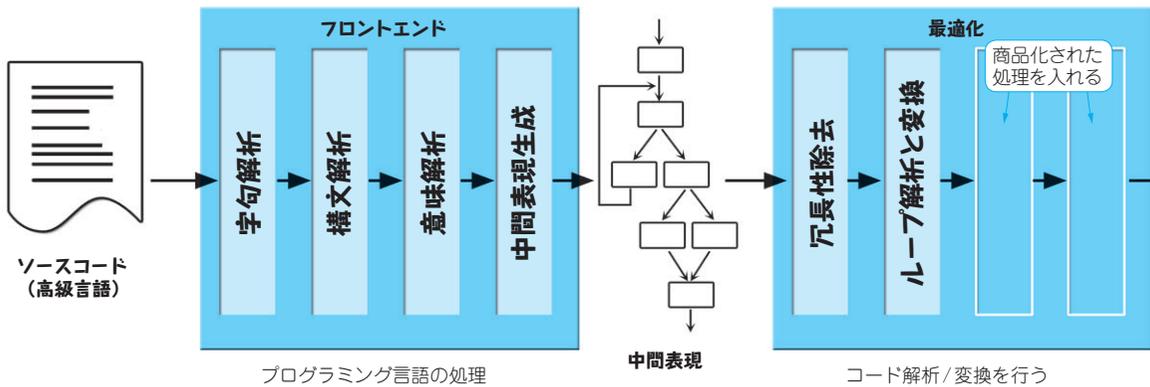
もちろん今までどおり...

**タダで使える!**

ソフト屋さんなら知らないとかっこ悪い

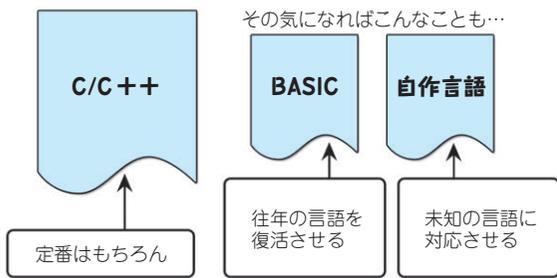
裏側で起きたこと...

# 新時代のコンパイラ登場!



## 言語対応よし

フロントエンドを用意するだけ!



部品化されているから  
ポテンシャル抜群!  
研究段階の技術も試せる!

 よくわからないが  
スゴイヤツ...

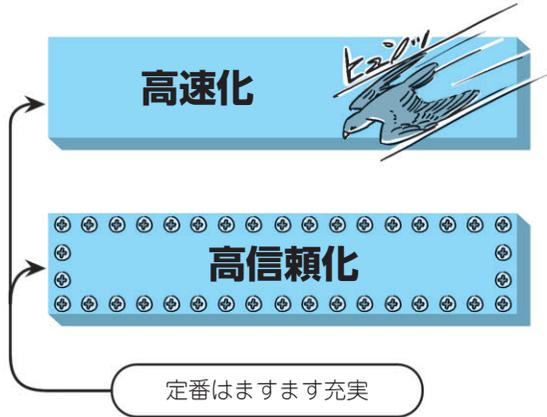
世界の英知も

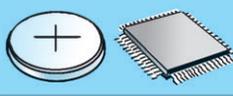
 日本発! 未知の最適化  
ガンバレ日本!

今まで考えられなかった最適化もできるかも

## 最適化対応よし

処理が一つ一つ部品化されている!

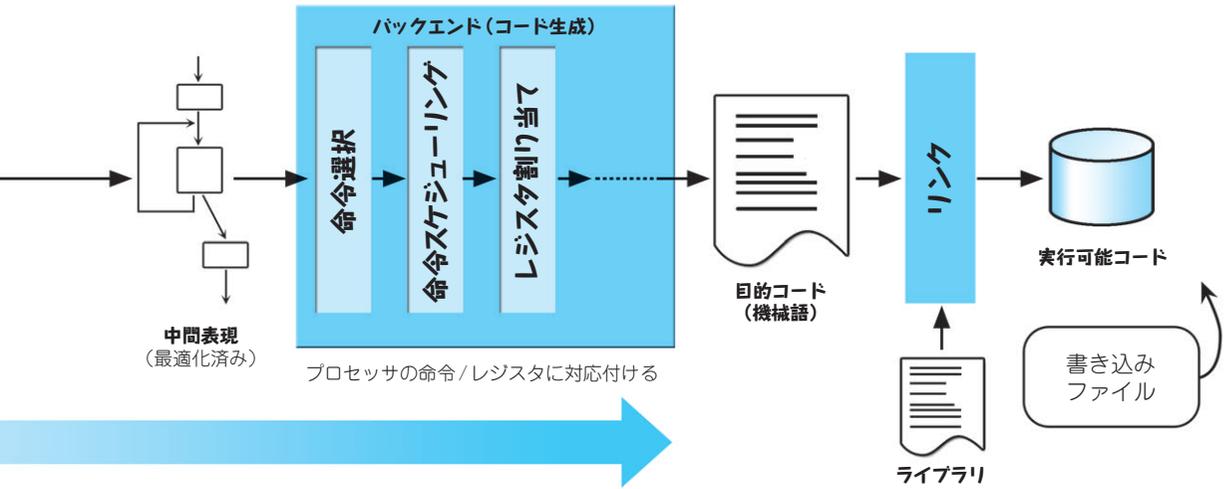


 低消費  
電力化

最新技術も

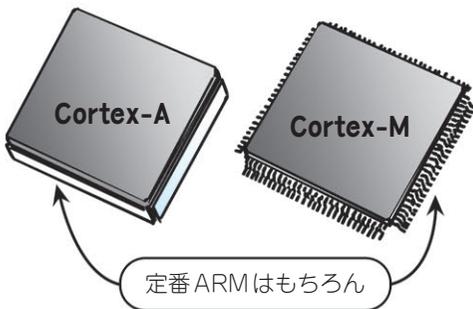
 並列化  
マルチコアやメニーコアも

# 知らないのはマズすぎる…

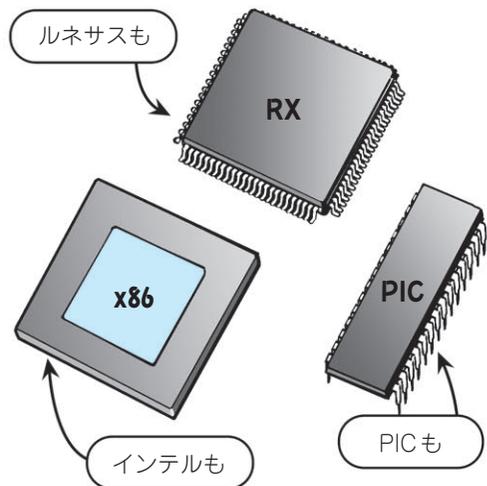
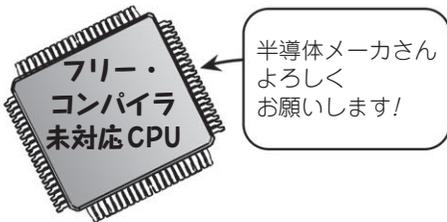


## CPU対応よし

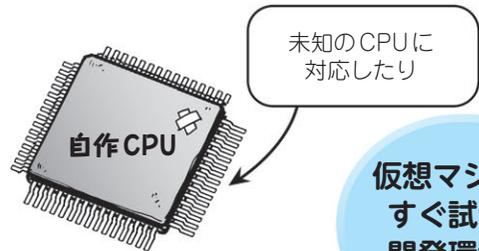
バックエンドを用意するだけ!



開発ツールも作りやすくなった



その気になればこんなことも



仮想マシンで  
すぐ試せる  
開発環境も  
用意

特集では…2大フリー・コンパイラ  
定番GCC & 新時代LLVMを紹介します!

Intro

App1

App2

1

2

App3

3

4

5

6

App4

7

8

App5

App6

9

App7

10