

Green Hills Software

- fastest time-to-market
- total reliability
- lowest production cost

- maximum performance

```
if (flags & INDRT_CALC_CPU_RESET_TIMERS)  
{  
    indrt_system_entity->CpuTime[core] = 0;  
}  
if (flags & INDRT_CALC_CPU_RECALCULATE)  
{  
    INDRT_ASSERT(measurement <= elapsed_time);  
    cpu_usage = (uint16_t)((10000*measurement) / elapsed_time);  
    indrt_system_entity->Residual[core] = cpu_usage;  
    indrt_system_entity->CpuUsage[core] = 10000 - cpu_usage;  
}  
}  
for (; domain; domain=domain->NextSibling)  
{  
    uint64_t    accum_domain_time[INDRT_CONFIG_MAX_CORES];  
    for (core = 0; core < max_cores; core++)  
    {  
        measurement = domain->CpuTime[core];  
        accum_domain_time[core] = measurement;  
        if (flags & INDRT_CALC_CPU_RESET_TIMERS)  
            indrt_system_entity->CpuTime[core] = 0;  
    }  
}  
for (task; task; task=task->NextSibling)  
{  
    task->CpuTime[core] = 0;  
}
```

InitialOx2 - TaskID=0x4ff15000 - MULTI Debugger

Target: gHS_Probe (ghprobe20599)

Initial: Executing on Core 4

OSA_myproject_as1 / Initial: Running (on Core 3)

OSA_myproject_as2 / Initial: Running (on Core 2)

Source: w:\file\src\myproject_as1.c

running 'C:\projects\linux_demo\myproject'

Stopped by breakpoint

MULTI> b main

main#2: 0x10048 AT count: 1

MULTI> c

Stopped by breakpoint

MULTI> |

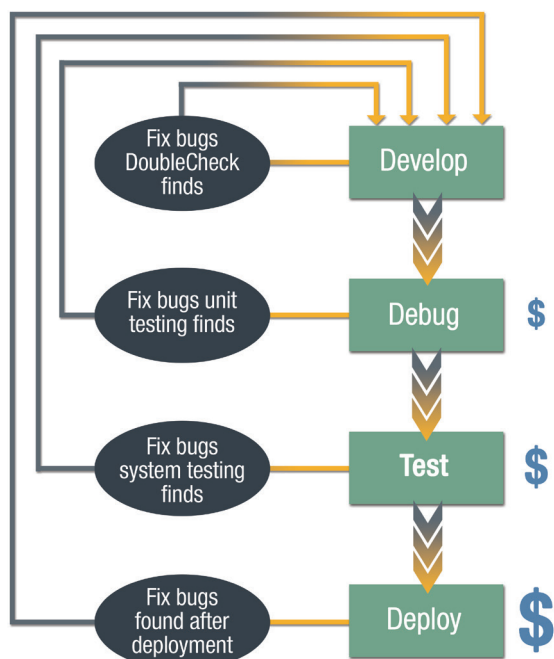
MULTI[®]

統合開発環境



バグを自動検出

GHSの静的ソースコード解析ツールである **DoubleCheck™**は、プログラムの実行前でもプログラミングエラーを特定できるツールで、時間と費用の節約に役立ちます。DoubleCheckは、最適化コンパイラである Green Hills Compiler の一部のため、個別にセットアップする必要もありません。オプションを設定するだけで、DoubleCheckがコンパイルのたびにソースコードをチェックします。さらに、DoubleCheckは従来の静的解析ツールよりも高速なので、開発者はいつでも使うことができます。



DoubleCheckはコードをハードウェアやシミュレータで実行する前に問題を特定し、開発コストの削減に貢献します。

DoubleCheckは従来のコードレビューよりも信頼性が高く、通常のシステムテストでは決して出現しないようなバグも検出できます。あるお客様は、「コードベースを DoubleCheck にかけたところ、隠れていた問題点が発見されました。1日たらずでこの問題を修正して 100%クリーンなコンパイルができるようになりました」と報告しています。

ランタイム・エラーチェックは Double Check の補完機能で、静的解析だけでは特定できないバグを検出します。特に悪質なバグというのは、長期間経過した後の故障で初めて姿を現すものです。ランタイム・エラーチェックは、問題が発生したときにその原因を警告で知らせます。バグも直接見れば、修正するのは簡単なことです。

ランタイム・エラーチェックで特定できる問題は、ポインタ経由の不正なメモリアクセスや、範囲外の値の代入、switch 文で未処理のケース、ゼロ除算などです。

複雑なシステムの理解

複雑なシステムのデバッグも、システムのステータスを知るのに必要な情報がすぐ手に入れば、ずっと簡単な作業になります。GHSのツールは重要なシステム情報を直感的な方法で表示することで、問題をより早期に発見できます。

- ▲ **OSA エクスプローラ**はカーネルのオブジェクト、タスク、リソースの状態を網羅的に表示します。
- ▲ **デバッガ**は完成したアプリケーションを完全にデバッガができるツールで、スタックの詳細分析、オブジェクト閲覧、メモリやレジスタの表示などが可能です。
- ▲ **EventAnalyzer™**は、システムイベントの時間をマイクロ秒間隔で表示します。

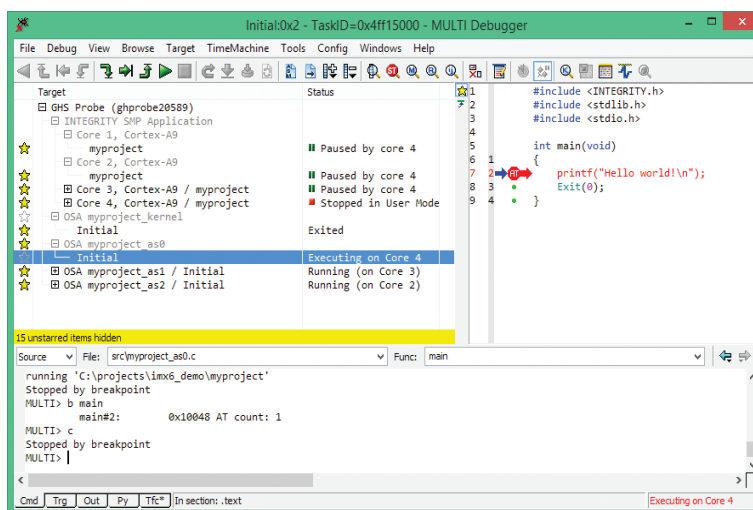
マルチコアの同時実行制御

マルチコアデバッグ機能としてプローブとデバッガのソリューションを最初に提供した会社の1つがGHSです。MULTIは現在、シングルコアシステムと同じぐらい容易にマルチコアシステムをデバッグできる機能を提供しています。

中でも重要な機能が同時実行制御です。この機能では、いずれのコアがデバッグ状態になっても、すべてのコアが1つのユニットとして停止されます。たとえば、コアがブレークポイントに達すると、ターゲットリストは以下を明確に示します。

- ▲ ブレークポイントに達したコア (1つまたは複数)
- ▲ その時点においてコア上で実行されていたスレッド
- ▲ システム内のその他のコアが行っていた内容

マルチコアのターゲットにコードをダウンロードする作業は、複雑で面倒なプロセスです。MULTIなら、これらのターゲットへINTEGRITYのダウンロードを自動で処理することができます。他のタイプのオペレーティングシステムやアプリケーションについては、簡単な設定ファイルを使用してダウンロードやセットアップのプロセスを自動化できます。



MULTIの同時実行制御は、マルチコアシステムのすべてのコアを同時に停止します。このため、あるコアをデバッグするときに、別のコア上で実行中の演算が共有メモリに影響することを心配する必要はありません。

新たな問題発生防止

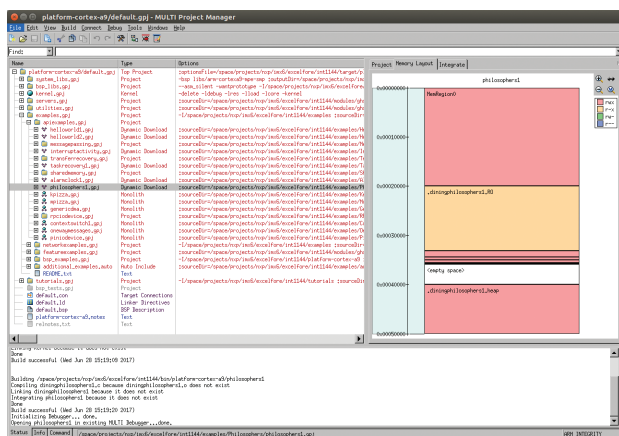
クリーンなコードはエラーを混入しにくく、またテストや理解、修正も簡単です。こうした性質はすべてバグの削減や信頼性向上にとってプラスとなります。Green Hills Compiler は、業界規格によるクリーンなコーディング規約を適用することができます。たとえば **MISRA 2004 コーディング規格**には、安全なプログラミングを実現するための 100 以上のルールがあります。こうしたルールの一部を固有の要件に合わせてカスタマイズして適用することも可能です。

GHS スタンダードモードは、通常の C/C++ よりも厳格なコーディング基準を強制するコンパイラ警告とエラーを集めたものです。このモードの土台となっているのは、長年にわたって GHS がソフトウェアを導入してきた経験と、同じ高水準の信頼性を達成したお客様との協力です。GHS でも、自社のツール開発の際に社内で GHS スタンダードモードを適用しています。

多くの設計時間を確保

MULTI の構造を単純化したことにより、貴重な時間を製品設計に回せるようになりました。GHS のビルダを使用すると、複雑なビルド基盤を維持する必要もなく、ビルドの設定も簡単に変更できます。ビルダは自動でプロジェクト全体の依存性を解析し、できる限り多くのファイルを並列にコンパイルしてリンクし、最新のマルチコアシステムを十分に活用します。

ビルダなどの高度な機能は、プロジェクト開発のオーバーヘッドを大幅に削減できます。**プロジェクトマネージャ**、**エディタ**、**フラッシュプログラマ**、**命令セットシミュレータ**などがシームレスに統合されているため、開発が後押しされ、作業効率が向上できます。



MULTI のプロジェクトマネージャは、アプリケーションのビルドやアップデート方法をより単純化するツールです。タブで分けられたブロック図やメモリレイアウトの表示など、一目でアプリケーション構造が分かる仕組みが用意されています。

生産性の向上

GHS 開発者たちは、毎日 MULTI ツールを使っています。GHS の生産性エキスパートは豊富な専門知識と実体験から、お客様が GHS 製品の機能を最大限活用できるようにアドバイスします。お客様の要件に合わせたツールの設定、製品の最適な使用方法についてのトレーニング、使用法を点検し、さらに生産性を高める方法の提案などをサポートします。

組込み Linux のデバッグ

組込み Linux システム用のソフトウェアを開発するエンジニアにとって、MULTI for Linux は、飛躍的な生産性向上に必要な総合的なデバッグ機能を備え、より信頼性の高い製品を素早く市場にもたらしめます。

従来、Linux エンジニアは、視認性と実用性が限られたデバッグツールに依存するほかありませんでした。GDB、および GDB 関連のフロントエンド (Eclipse など) は、設定が面倒で使いにくく、必ずしも信頼のおけるツールではありません。このため、開発者は多くの場合、有効性がなりゆき任せの **printf** や **printk** コマンドを使用しています。残念なことにこれらのコマンドは、複雑なシステムの実行においても非常に小さなウィンドウしか表示されません。

MULTI for Linux を組み込むことで、エンジニアは組込み Linux システムの全側面 (Linux カーネル、カーネルスレッド、割り込みサービスルーチン、ユーザーモードのスレッドとプロセスなど) を以前に比べて素早く簡単に視覚化してデバッグすることができます。

Linux デバッグ機能

- ▲ Linux カーネルの総合的な視覚化 - 共有メモリ領域、セマフォ、メッセージキュー、カーネルモジュールなど
- ▲ SMP マルチコアデバッグのサポート
- ▲ 1 つのデバッグでカーネルからユーザー空間までをデバッグ
- ▲ ネイティブ Linux と仮想化 Linux のデバッグ (INTEGRITY® Multivisor™ 仮想化技術上)
- ▲ ロード可能なカーネルモジュールとデバイスドライバのデバッグ
- ▲ プロセススタートアップコードを容易にデバッグ
- ▲ カーネルコードでのセットおよびスレッド固有のブレークポイント
- ▲ 仮想アドレス空間での共有オブジェクトのデバッグ
- ▲ パフォーマンスコストを最小限に抑えたブレークポイントの処理
- ▲ プロセス間のマルチスレッド対話のデバッグ

機能安全の信頼性

MULTI ツールチェーンは、最高水準のツール適格性と C/C++ ランタイム認証要件に適合しています。

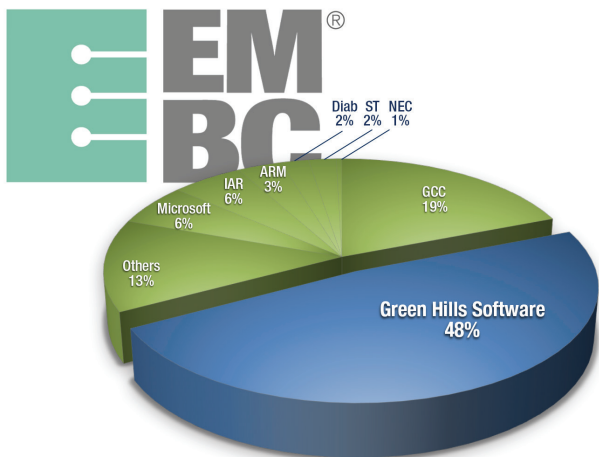
- ▲ IEC 61508:2010 (工業)、EN 50128:2011 (鉄道)、および ISO 26262:2011 (自動車) の機能安全規格
 - ▲ TÜV NORD および exida からの証明書を取得
 - ▲ SIL 4 (安全度水準) と ASIL D (自動車安全水準) のツール適合性および C/C++ ランタイム認証要件 - 最高水準の達成度
 - ▲ 広範囲のターゲットプロセッサセット上でのサポート
- お客様のメリットは、次のようにシンプルで徹底しています。
- ▲ コストを低減し認証までの時間を短縮
 - ▲ 製品の市場投入までの時間を短縮
 - ▲ 製品リリース後の認証メンテナンスを削減
 - ▲ 最高水準の SIL/ASIL でアプリケーションの開発をサポート

パフォーマンスの最大化

最大限のパフォーマンスが要求されるアプリケーションのため、統合開発環境 MULTI は、最適なコンパイラと最高のパフォーマンス解析ツールを用意しています。最高パフォーマンスを達成するには、GHS のパフォーマンスチューニングのエキスパートにお問い合わせください。お客様のデバイスに合わせてカスタマイズしたソリューションを実現します。

生成コードの高速化・小型化

優れたコンパイラは、プログラムのパフォーマンスを最大限に発揮するのに不可欠です。GHS の最適化コンパイラである Green Hills Compiler は業界でも屈指のもので、EEMBC ベンチマーク（組込み業界で最も認められているベンチマーク）では、**Green Hills Compiler** が常に競合他社のコンパイラ性能を上回っており、32 ビット・64 ビットプロセッサ用では、最高速かつ最小のサイズのコードを生成しています。



シリコンベンダーが公表する EEMBC 認定ベンチマークでは、広範囲に及ぶ実環境アプリケーションにおけるプロセッサのパフォーマンスが示されています。これらベンダーが公表するスコアの生成には Green Hills Compiler が、他社のコンパイラと比べて倍以上の割合で選ばれています。

Green Hills Compiler は、何百もの最先端の**コンパイラ最適化**を使用して、プログラムの実行速度を大きく向上させ、サイズを削減しています。たとえば、高度なプログラムの全体の最適化は、関数呼び出しの境界を超えて、Green Hills Compiler がソース全体を処理できるようにします。その結果、コンパイラは不要なコードや変数の削除、モジュールをまたいでのインライン化、プロシージャ間のエイリアスを、より詳細に解析することができるようになります。

Green Hills Compiler は、デフォルトの最適化と設定の最も優れた基準となっていますが、アプリケーションにはそれぞれ独自性があることも理解しています。さまざまな最適化を組み合わせて、プロジェクト単位、ファイル単位、関数単位、さらにはソース行単位で適用することで、コンパイラの出力を細かく調節できます。柔軟性の高い Green Hills Compiler の設定を精密に調整すると、プロジェクトに関係なく最高のパフォーマンス結果を生成できます。ほとんどのプログラムで、Green Hills Compiler は GNU コンパイラよりも 20% 高速なコードを生成します。

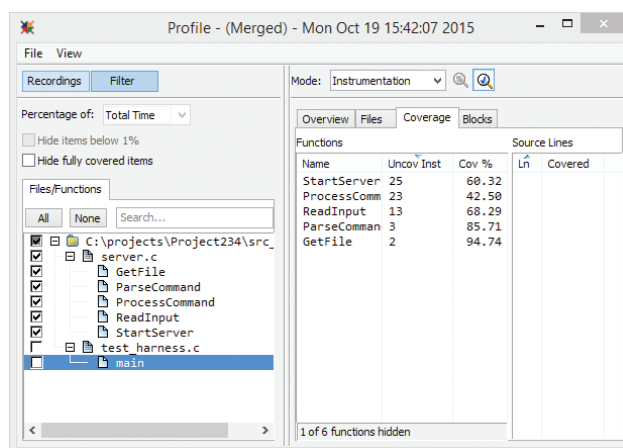
パフォーマンスのボトルネックを正確に突き止める

性能を最大化させるための第 2 の要素は最高のパフォーマンス解析ツールです。GHS の優れた PathAnalyzer、プロファイラ、EventAnalyzer をうまく活用すれば、コード性能のネックになっている部分を簡単に特定して排除できます。つまり、より高速ですっきりしたアプリケーションを開発できます。

最高性能を求めるなら、時間が最も重視すべき要素となります。これはまた、ほとんどのプロファイリングツールが投出しがちな要素でもあります。革新的な **PathAnalyzer** は、プログラム内部で実行される各関数を時系列で分かりやすく表示して、プログラムフローのパターン特定や、時間がかかっている場所の見極めが簡単にできるようにします。PathAnalyzer は単にプログラムの実行の痕跡を表示させるだけでなく、その全体像を表示します。

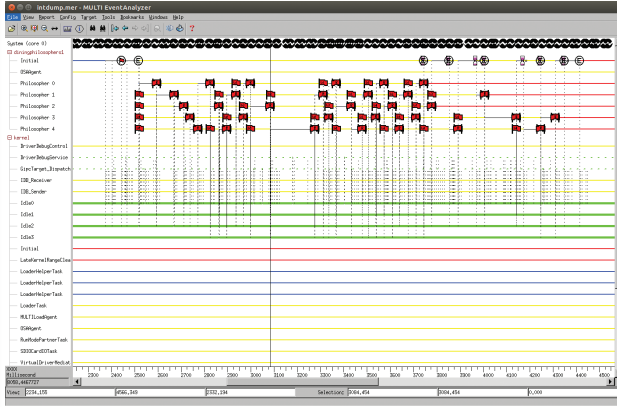
システムの最適化と解析については、MULTI のパワフルな**プロファイラ**を使用することでプログラムの挙動を最も詳細に把握することができます。プロファイラはタスク、関数、ソース行、アセンブリ言語の命令ごとに、プロセッサ時間の消費量を示し、以下が容易に行えるようになります。

- ▲ パフォーマンスの問題点を見つける
- ▲ カバレッジギャップを見つけて特定コードのテストカバレッジを追跡する
- ▲ ユーザー間でプロファイルデータを共有する
- ▲ いくつかのテストまたプロファイルの記録からプロファイルデータを収集して統合する



プロファイラのインターフェースは解析を簡素化し、さまざまなタスク、関数、命令などを呼び出した頻度および時間を表示するため、最適化とテスト作業を集中して行う場所を突き止めることができます。

EventAnalyzer は INTEGRITY® RTOS イベントの時間と頻度を表示します。それを元に、最も時間がかかっている処理がどこで、どこに最適化を集中すればよいかを素早く判断できます。EventAnalyzer を活用すれば、高レベルな性能の問題、たとえば不適切なタスク優先順位、過剰な割込みレベル処理、オペレーティングシステムへのコールの過多、予期しないタスクのコンテキストスイッチなどに取り組みやすくなります。



EventAnalyzer は INTEGRITY RTOS イベントの時間と頻度を表示するので、最適化作業をどこに集中すべきか簡単に判断できます。

豊富な知識によるメリット

可能な限り最高のパフォーマンスを達成するには、GHS のパフォーマンスチューニングのエキスパートにお問い合わせください。この 30 年間、GHS は豊富な知識と最高級のツールを活用することで、公表されるプログラム速度のベンチマークで競合を上回ってきました。

GHS が内部で使っている同じ技法をお客様のアプリケーションに適用できます。総合的な知識を活用して、プログラムの部分ごとにコンパイラ最適化オプションを設定していきます。たとえば最近では、GHS のコンパイラチームが ARM アーキテクチャのスペシャリストと協力して複数の最適化を新たに追加しました。その結果、Green Hills Compiler は Cortex-R5 プロセッサで 1.01 EEMBC Automarks/MHz レーティングのパフォーマンススコアを達成することができました。これは、以前のパフォーマンススコアに対して 30% の向上になります。

GHS のパフォーマンスチューニングサービスは商品ではなく、カスタムサービスです。お客様のアプリケーションを調査し、それに基づいてコンパイラ最適化オプションを特別に作成します。そして組込みシステムに関する GHS の専門知識を活用してプログラムを高速化します。GHS のサービスで、お客様のアプリケーション性能が 300% 以上アップすることも珍しくありません。

MULTI のサポート

新しいプロセッサとオペレーティングシステムに対する最新サポートについては GHS までお問合せください。

ターゲットプロセッサ

- ▲ ARM (AArch32 および AArch64)
- ▲ Power アーキテクチャ/VLE ▲ TriCore
- ▲ RH850/V850 ▲ Intel (x86 および x64)
- ▲ 68K/ColdFire ▲ MIPS

開発ホスト

- ▲ Windows ▲ Linux

オペレーティングシステム

- ▲ INTEGRITY ▲ Linux ▲ OSE
- ▲ μ-velOSity ▲ ThreadX ▲ 独自 OS
- ▲ Windows ▲ VxWorks ▲ ベアボード

サードパーティとの統合

- ▲ Eclipse ▲ Subversion
- ▲ Rhapsody ▲ ClearCase
- ▲ Emacs ▲ CVS
- ▲ vi ▲ SourceSafe
- ▲ DWARF や Stabs のデバッグ情報を生成するコンパイラ

言語

- ▲ C ▲ EC++
- ▲ C++ ▲ Ada (AdaMULTI)
- ▲ GNU C/C++ extensions



30年にわたって、ツールの開発と使用を通じてお客様の問題を解決してきた経験から、熟練のエキスパートが、あなたのアプリケーション性能と信頼性を最大限高めのお手伝いをします。

開発コストの削減

大量生産においては、単価を数円下げただけでも全体的なコスト削減額は莫大なものになります。GHS のツールでメモリとプロセッサの要件を大きく引き下げ、最終的な収支を大きく改善しましょう。

プロセッサコストの削減

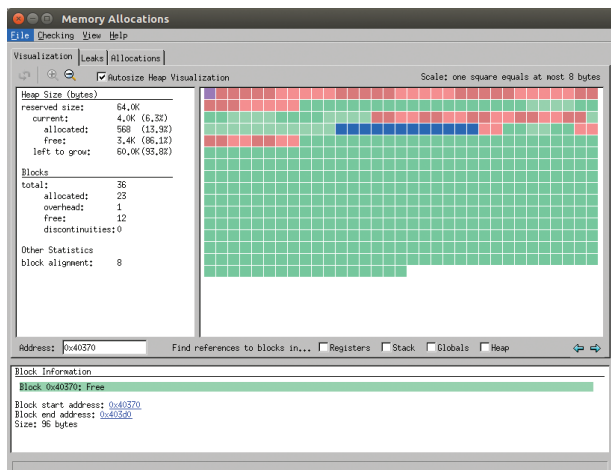
性能要件の増加分を達成するために、必ずしも前回より高速なプロセッサが必要というわけではありません。GHS の優れた**コンパイラ最適化機能**を調整すれば、どんなチップを採用するにしてもその最高性能を引き出すことができます。さらに、GHS の**PathAnalyzer**、**プロファイラ**、**EventAnalyzer** といった性能解析ツールを導入すれば、プログラムを俯瞰的に見てどこで時間がかかっているかを把握することができ、最適化を集中させて、アプリケーションを高速化できるようになります。

同じリアルタイム要件を実現するのに、安いプロセッサを使えば、いくら節約できるでしょうか。または、新製品をリリースするのに最新型のハードウェアのプラットフォームを用意する必要がなくなればどうでしょうか。

アプリケーション性能を改善すれば、プロセッサ自体にかかる費用を節約できるだけでなく、クロック速度の低い安価なプロセッサで、消費電力と発熱を抑えることで、バッテリーが長持ちし、製品はより良く低価格なものとなります。

メモリコストの削減

Green Hills **Compiler** の**最適化**で実行ファイルサイズを小型化すれば、メモリコストも削減できます。ほとんどのプログラムで、GNU コンパイラよりも 10%以上改善します。さらに、**DoubleCheck** でコンパイル時にメモリリークを検出できる上、MULTI の **Memory Allocation** ツールでランタイムでも動的にメモリリークを探し出します。



MULTI の **Memory Allocation** ツールでは、ランタイムでも動的にメモリリークを一掃できます。

最大のコスト削減を達成

GHS のコスト削減エキスパートチームが、機能はそのままメモリやプロセッサの要件を下げる手助けをします。GHS のチームは、お客様のアプリケーションのメモリ使用量を 30%以上削減し、パフォーマンスを 300%以上改善させた実績があります。

Green Hills のトレーニングとコンサルティング

Green Hills エキスパートのトレーニングとコンサルティングのサービスは、統合開発環境 MULTI の全潜在能力を理解するための一番の近道であり、短時間で最大の生産性に到達することができます。

インプリメンテーションプログラムの管理

このプログラムは、開発チームが素早く GHS の製品環境に慣れ親しんで生産性を向上できるように設計されています。提供されるサービスは次のとおりで、内容および期間は完全にカスタマイズされています。

- ▲ コエキスパートトレーニング
- ▲ 環境セットアップ
- ▲ アプリケーション設計指導

GHS の技術エキスパートは、設計段階から、お客様のニーズに最適に合致する製品の機能に集中できるようにします。カスタマイズされたトレーニングに加えて、管理されたインプリメンテーションプログラムでは、先を見越した継続的な指導を行うことで、お客様が GHS 環境を最適に使用する方法を理解し、また GHS のエキスパートを定期的に利用できるようにします。

カスタマイズされたトレーニング

GHS のトレーニングカリキュラムには、リアルタイム INTEGRITY プログラミングから高機能トレースおよび TimeMachine デバッグに至るまで広範囲の教材が用意されています。コースはエキスパートによって教えられ、GHS ツールを用いた開発を実地で体験します。クラスはお客様の施設でお客様のスケジュールに従って教えられ、中断を最小限に抑えて最大限の価値が得られるようにします。

自由入学トレーニング

トレーニング予算が少ないチームは、事前に選ばれた世界中の場所で行われる一般向けの自由入学コースに出席していただけます。このクラスは、すでに GHS トレーニングクラスを終えて、チームに加わったばかりの新採用者にとっても欠かせないクラスとなります。



Corporate Headquarters

30 West Sola Street ▲ Santa Barbara, CA 93101
ph: 805.965.6044 ▲ fax: 805.965.6343 ▲ email: info@ghs.com ▲ www.ghs.com

Green Hills Software GK

〒150-0001東京都渋谷区神宮前1-5-8神宮前タワービルディング 13階
電話番号: 03-6741-7168 ▲ お問い合わせ: jpsales@ghs.com



〒101-0045東京都千代田区神田鍛冶町3-4oak神田鍛冶町
電話番号: 03-3251-3170(代) ▲ www.adac.co.jp/