

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4921175号
(P4921175)

(45) 発行日 平成24年4月25日 (2012. 4. 25)

(24) 登録日 平成24年2月10日 (2012. 2. 10)

(51) Int. Cl.

F I

G 0 6 F 9/44 (2006. 01)

G 0 6 F 9/06 6 2 0 A

G 0 6 F 9/45 (2006. 01)

G 0 6 F 9/44 3 2 2 M

B 6 0 R 16/02 (2006. 01)

B 6 0 R 16/02 6 6 0 Z

請求項の数 2 (全 12 頁)

(21) 出願番号 特願2006-546575 (P2006-546575)
 (86) (22) 出願日 平成16年12月7日 (2004. 12. 7)
 (86) 国際出願番号 PCT/JP2004/018197
 (87) 国際公開番号 W02006/061884
 (87) 国際公開日 平成18年6月15日 (2006. 6. 15)
 審査請求日 平成19年4月2日 (2007. 4. 2)

前置審査

(73) 特許権者 000005108
 株式会社日立製作所
 東京都千代田区丸の内一丁目6番6号
 (74) 代理人 110000350
 ポレール特許業務法人
 (72) 発明者 成沢 文雄
 茨城県日立市大みか町七丁目1番1号 株
 式会社日立製作所 日立研究所内
 (72) 発明者 眞山 一徳
 茨城県ひたちなか市大字高場2520番地
 株式会社日立製作所 オートモティブシ
 ステムグループ内
 (72) 発明者 恒富 邦彦
 茨城県日立市大みか町七丁目1番1号 株
 式会社日立製作所 日立研究所内
 最終頁に続く

(54) 【発明の名称】 自動車用制御装置のソフトウェア作成方法

(57) 【特許請求の範囲】

【請求項 1】

自動車の制御を行うアプリケーションと、前記アプリケーションの要求を受け付けてマイクロコンピュータを制御するデバイスドライバとを有する自動車用制御装置のソフトウェアを、計算機を用いて作成するためのソフトウェア作成方法において、

複数の階層に階層化された前記デバイスドライバの各階層に、前記アプリケーションが要求する制御物理量を演算処理するための複数の上位層のモジュールと、前記マイクロコンピュータのポートとの間の入出力処理を行うための複数の下位層のモジュールとを設け、前記上位層のモジュールと前記下位層のモジュールとが処理する処理内容がソースコードで記述された第1の処理記述部、

前記上位層のモジュールのラベル名と前記下位層のモジュールのラベル名と前記マイクロコンピュータのポート名とを受取って、識別子連結機能を用いて他の文字列と連結する定義文がソースコードで記述された識別子展開部、および

他の文字列と連結された上位層のモジュールのラベル名を、どの下位層のモジュールのラベル名に置換するかと、他の文字列と連結された下位層のモジュールのラベル名を、前記マイクロコンピュータのどのポート名に置換するかと、の定義が前記自動車用制御装置の仕様に基づいてソースコードで記述された階層構成記述部、を記憶手段から前記計算機が読み出し、

前記計算機がプリプロセッサを用いて、前記上位層のモジュールのラベル名を前記階層構成記述部の記述に基づいて前記下位層のモジュールのラベル名に置換し、前記下位層の

10

20

モジュールのラベル名を前記下位層構成記述部の記述に基づいて前記マイクロコンピュータのポート名に置換し、該置換は前記識別子展開部で定義された他の文字列との連結処理を経て実行され、該置換により階層間の呼び出し処理が一つの階層内に展開されるように、前記第 1 の処理記述部の前記複数の階層を 1 つの階層にマクロ展開されたソースプログラムである第 2 の処理記述部を作成し、

前記第 2 の処理記述部に基づいて、前記計算機がコンパイラを用いて前記デバイスドライバのオブジェクト形式ファイルを作成し、

作成された前記デバイスドライバのオブジェクト形式ファイルと前記アプリケーションのオブジェクト形式ファイルに基づいて、リンカによって、前記ソフトウェアを作成することを特徴とする自動車用制御装置のソフトウェア作成方法。

10

【請求項 2】

請求項 1 において、

前記階層構成記述部は、前記上位層のモジュールのラベル名をどの下位層のモジュールのラベル名に置換するかと、前記下位層のモジュールのラベル名を前記マイクロコンピュータのどのポート名に置換するかと、の前記定義が、予め記憶された入出力端子一覧表に基づいて記述されていることを特徴とする自動車用制御装置のソフトウェア作成方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は自動車用制御装置と、自動車用制御装置を構成するマイクロコンピュータのソフトウェアの作成方法及び作成システムに関する。

20

【背景技術】

【0002】

従来より自動車エンジン制御などの制御装置として、CPU、ROM、RAM、入出力信号処理装置などを内蔵したマイクロコンピュータ（以下マイコンと記す）が用いられている。マイコンに搭載されるソフトウェアは、目的とする制御動作が行えるように、制御処理を行うアプリケーションプログラムと入出力を行うデバイスドライバによって構成されている。入出力を行うデバイスドライバは、使用するハードウェアが変更になった場合に影響を受け易く、その変更工数の大きさが問題となっている。

【0003】

30

ハードウェアの変更には、大きく分けると、制御対象であるセンサやアクチュエータの変更と、制御を行うマイコンの変更の 2 通りがある。センサやアクチュエータなどの制御対象ハードウェアの変更の場合、その特性や仕様を記載したハードウェアマニュアル等を参照し、目的とする制御動作を実現できるようにソフトウェアを開発する。また、マイコンの変更の場合には、マイコンの仕様を記載したマニュアルを参照しながら変更個所の抽出、変更仕様の決定、変更作業、及び検証を行う必要があり多大な工数を要していた。

【0004】

ハードウェアの変更に伴うソフトウェアの変更工数を削減するものとして、非特許文献 1 に開示されているように、情報系の汎用オペレーティングシステム（OS）である UNIX（登録商標）がある。ここでは、制御対象ハードウェアとのインターフェースをブロック型、メッセージチャネル型、及びキャラクタ型の 3 種類に標準化し、全ての入出力プログラムは、上記 3 種類のうちいずれかの仮想インターフェースを介する。これにより、ソフトウェアは、アプリケーション、仮想ドライバ、実ドライバプログラムの三階層構成となり、アプリケーションプログラム側は対象ハードウェア変更を意識することなく開発できる。

40

【0005】

また、特許文献 1 には、デバイスドライバを三階層化することによりデバイスの変更に伴うプログラムの変更を局所化している。

【0006】

他方、非特許文献 2 によれば、デバイスドライバを上位ドライバと下位ドライバに分割

50

して作成し、上位ドライバでは下位ドライバのインターフェースを用いて処理を行うことで、マイコンやハードウェアの変更の影響を受けない構成としている。

【0007】

さらに、インターフェースを介した処理を高速化する手法として、マクロにより展開する手法が知られており、特許文献2に開示されているように、インターフェースを介した処理呼び出しをマクロ置換により高速化している。

【0008】

【非特許文献1】S. J. Leffler 他著 中村明他訳「UNIX 4.3 BSD の設計と実装」丸善株式会社(1991)

【特許文献1】特開2000-97102号公報

10

【非特許文献2】μITRON4.0仕様研究会著「デバイスドライバ設計ガイドライン」μITRON4.0仕様研究会(1999)

【特許文献2】特開2002-287981号公報

【発明の開示】

【発明が解決しようとする課題】

【0009】

しかしながら、一般に用いられるC言語等のマクロ処理を行うプリプロセッサは、逐次処理による文字列置換処理であることから、複数の階層から構成される処理を高速化することが不可能である。このため、これらの従来技術では、階層を経るたびに発生する処理のオーバーヘッドを抑制することができない。例えば、三階層で構成されたデバイスドライバの場合、一つの入出力処理を行うために、三つの階層の呼び出し処理を行うプログラムとなり、プログラムの実行時間の遅延とサイズの増大とを招く。自動車などの制御に好適なリアルタイム制御を実現するためには、演算、入出力処理などの各処理を最適なタイミングで行う必要があり、数マイクロ秒の遅延も許されない場合も多い。階層型ソフトウェアを通常の実装方法で実装した場合、それぞれの階層間で関数呼び出しやシステムコール呼び出しが生じ、これらは無視できない遅延となる。また、これらはマイコン上のコンパイラなどを用いて機械語命令に変換すると、1~8命令からなる呼び出し処理に変換され、数マイクロ秒から数十マイクロ秒の実行時間を要する。また、階層間の接続に対応するデータが、ROM、RAM、あるいはディスクなどの記憶手段上に必要となり、必要メモリ容量が増大する。

20

30

【0010】

自動車エンジン制御装置などの組み込みシステムは、コストに直接影響のあるROM、RAM容量の削減要求や、リアルタイム制御であることから応答速度に対する要求も厳しいため、これらのオーバーヘッドは実用上の大きな障害となる。

【0011】

本発明の目的は、ハードウェアや制御方法の変更時におけるソフトウェアの変更工数を削減し、アプリケーションの実行速度と、記憶装置の必要容量を削減することである。

【課題を解決するための手段】

【0012】

本発明はその一面において、複数の階層に階層化された第1の処理記述部と、この第1の処理記述部に用いられている識別子の展開を行う識別子展開部と、第1の処理記述部の階層構成を記述する階層構成記述部とを作成し、この階層構成記述部と前記識別子展開部に記述されたデータに基づいて、第1の処理記述部の階層構成を展開して、ひとつの階層からなる第2の処理記述部を作成することを特徴とする。

40

【0013】

本発明の望ましい実施態様においては、各階層の処理を階層毎に定義したラベルを用いてマクロで記述する第1の処理記述部と、階層間の接続関係を階層毎のラベルをマクロ化し定義した階層構成記述部と、階層構成記述部に基づきマクロ展開を行う識別子展開部と、プリプロセッサ及びコンパイラを用いることにより、1つの階層に展開した第2の処理記述部に目的とする実行形式のソフトウェアを得ることができる。

50

【 0 0 1 4 】

本発明は他の一面において、マイクロコンピュータを用いて自動車の制御を行う自動車用制御装置のソフトウェアの作成システムにおいて、複数の階層に階層化された第1の処理記述部を作成する手段と、この第1の処理記述部に用いられている識別子の展開を行う識別子展開部を作成する手段と、第1の処理記述部の階層構成を記述する階層構成記述部を作成する手段と、この階層構成記述部と前記識別子展開部に記述されたデータに基づいて、前記第1の処理記述部の階層構成を展開したひとつの階層からなる第2の処理記述部を作成する手段とを備えたことを特徴とする。

【 0 0 1 5 】

本発明の望ましい実施態様においては、上記第2の処理記述部を作成するプリプロセッサと、前記第2の処理記述部をオブジェクト形式ファイルに変換するコンパイラと、前記オブジェクト形式ファイルをアプリケーションオブジェクト形式ファイルと連結して自動車用制御装置の記憶装置に書き込む制御ソフトウェアを作成するリンカを備える。

【発明の効果】

【 0 0 1 6 】

本発明によれば、複数階層で設計したソフトウェアをコンパイルした結果として、一つの階層のみのソフトウェアとすることにより、階層間オーバーヘッドのないソフトウェアを得ることができる。

【 0 0 1 7 】

本発明によるその他の目的と特徴は、以下に述べる実施例の説明で明らかにする。

【図面の簡単な説明】

【 0 0 1 8 】

【図1】本発明の一実施例によるデバイスドライバ開発時におけるデバイスドライバとアプリケーション及びハードウェアの構成図。

【図2】本発明の一実施例の対象とする自動車エンジン制御システムの構成図。

【図3】図1における処理記述部の上位層(L1)内の具体例の詳細図。

【図4】図1における処理記述部の下位層(L2)内の具体例の詳細図。

【図5】図1のデバイスドライバ中の階層構成記述部の詳細図。

【図6】図1のデバイスドライバ中の識別子展開部の詳細図。

【図7】本発明の一実施例による開発手順図。

【図8】本発明の一実施例により階層間オーバーヘッドの無いソースコードの一例図。

【図9】本発明の一実施例によるデバイスドライバ作成方法の開発環境及び開発手順図。

【図10】本発明の一実施例による入出力端子一覧表の一例を示す図。

【符号の説明】

【 0 0 1 9 】

100...アプリケーションプログラム、101...噴射制御アプリケーション、102...点火制御アプリケーション、110...オペレーティングシステム、120...デバイスドライバ、121...吸入空気量演算(IAA)、122...吸気温度演算(IAT)、130、201...マイコン(CPU)、131~133、203...入出力ポート、140...アプリケーションプログラムインターフェース(API)、150...ハードウェアインターフェース(HWI)、160...上位層(L1)、170...下位層(L2)、171...1次フィルタアナログ入力(FAI)、172...Vcc補正アナログ入力(VAI)、180...識別子展開部、191...階層構成記述部、192...識別子展開部、200...コントロールユニット、220...センサ・アクチュエータ、71...デバイスドライバプログラムソースコード、72...ビルド、73...デバイスドライバオブジェクトファイル、902...デバイスドライバリポジトリ、905...入出力端子一覧表、906...階層構成生成装置、911...プリプロセッサ、913...展開された処理記述部、914...コンパイラ、916...デバイスドライバオブジェクト形式ファイル、917...アプリケーションソースコード、918...アプリケーションオブジェクト形式ファイル、919...リンカ、921...制御ソフトウェア。

10

20

30

40

50

【発明を実施するための最良の形態】**【0020】**

以下、本発明の実施例を図面を参照して説明する。

【0021】

図1は、本発明の一実施例によるデバイスドライバ開発時におけるデバイスドライバとアプリケーション及びハードウェアの構成図である。ここでは、各種センサから得られた入力を、マイコン130の入出力ポート131～133をもとに、吸入空気量演算と、吸気温度演算を実行するデバイスドライバ120の例を示している。その詳細に入る前に、制御対象とする自動車エンジン制御システムの概要から先に説明する。

【0022】

図2は、本発明の一実施例の対象とする自動車エンジン制御システムの構成図である。コントロールユニット200は、CPU201、割込みコントローラ202、タイマ・パルスコントローラ203、AD変換器204、ROM205、RAM206を備えている。これらは、バス207で結ばれ、入出力ポート208から外部とアクセスする。これらの各要素202～208は、一つの素子上に内蔵する場合と、別素子を用いて接続する場合がある。210は、コントロールユニット200の電源部である。

【0023】

コントロールユニット200には、入出力ポート208を介して、その制御対象であるセンサ・アクチュエータ220が接続されている。センサ・アクチュエータ220は、エアフローセンサ221、電制スロットル222、インジェクタ223、点火プラグ224、LAFセンサ等の空燃比センサ225、及びクランク角センサ226等からなり、コントロールユニット200がこれらの制御を行う。すなわち、CPU201をはじめとする構成要素から、入出力ポート208内のレジスタへの読み書きにより、制御が行われる。制御の方法を記述したソフトウェアは、コントロールユニット200内のROM205及びRAM206に書き込まれる。

【0024】

本実施例は、このソフトウェアにおいて、例えば、ハードに変更が生じた場合などにも、このソフトウェアの変更が容易で、かつ、アプリケーションの実行速度が速く、メモリ容量も節約できるソフトウェアの作成方法に関する。

【0025】

図2のコントロールユニット200が実行するソフトウェアの構成を、図1に戻って説明する。ソフトウェアは、大きく分けて、アプリケーションプログラム100、オペレーティングシステム(OS)110、デバイスドライバ120の3つに分類される。アプリケーションプログラム100は、アプリケーションプログラムインターフェース(API)140を通して、OS110やデバイスドライバ120と情報や処理の受け渡しを行う。接続されるハードウェアの如何によらず、API140を標準化することにより、ハードウェアが変更されても、その影響がアプリケーションプログラム100へ及ぶことを避けることができる。OS110、デバイスドライバ120は、マイコン130の入出力ポート131～133等のハードウェアインターフェース(HWI)150を通して、これらの制御を行う。

【0026】

ここでは、マイコン130の入出力ポート131～133を通してセンサから得られた入力に基づいて、吸入空気量演算121及び吸気温度演算122を実行するデバイスドライバ120の例を示している。デバイスドライバ120は、アプリケーションプログラム100の噴射制御101や点火制御102等から、API140を通じて要求を受け付ける。その制御対象は、CPU130であり、HWI150を経て、アナログ入出力ポート131～133等を読み書きすることにより、目的とする制御を行う。

【0027】

デバイスドライバ120の開発時には、上位層(L1)160、下位層(L2)170のように、階層化することにより変更点の局所化を行う。この階層数は特に限定はなく、

10

20

30

40

50

制御対象やソフトウェアの規模に応じて適切な階層数で実現できる。ここでは、2階層の場合について説明する。

【0028】

処理記述部180において、上位層(L1)160には、吸入空気量演算121や吸気温度演算122等、アプリケーション100が必要とする入力値を求めるモジュールを設ける。また、下位層(L2)170には、入力値を取り込んで汎用的な入力処理を行うモジュールを設け、処理内容を記述する。ここでは、1次フィルタアナログ入力(FAI)171と、Vcc補正アナログ入力(VAI)172を示している。階層構成記述部191には、どの上位層がどの下位層を用いるかという階層間の具体的な構成を記述する。識別子展開部192を用いて、階層構成記述部191で定めた階層構成を、処理記述部180

10

【0029】

図3及び図4は、図1の処理記述部180の上位層(L1)160及び下位層(L2)170内の具体例の詳細図であり、これらについて説明する。

【0030】

図3は、図1における上位層(L1)160の吸入空気量演算(IAA)121の詳細図である。

【0031】

図3(A)は処理の概略を示すものである。図示するように、吸入空気量演算モジュールには、処理名1の吸入空気量更新処理と、処理名2の吸入空気量取得処理を有し、かつ、この演算のために必要な入力処理として、Vcc補正アナログ入力があることを示している。図3(B)は、上記吸入空気量更新処理の概略を示すフローチャート、図3(C)は、同じく吸入空気量取得処理の概略を示すフローチャートである。図3(D)は、図3(A)、(B)、及び(C)を実装したプログラムコードの例である。

20

【0032】

図3(B)の吸入空気量演算処理では、まず、ステップ301において、下位層L2におけるVcc補正アナログ入力(VAI)により入力値の更新を行い、次に、ステップ302において、更新した値を取得する。ステップ303では、取得した値をもとに、エアフローセンサの特性を補正するためのマップにより補正を行い、ステップ304では、前記補正によって得られた値をバッファへ保持する。

30

【0033】

また、図3(C)の吸入空気量取得の処理は、ステップ305において、上記バッファの値を返すことによって実現する。

【0034】

図3(D)は、これらの処理を、C言語のマクロを用いて記述したプログラムソースの例である。マクロ記述は、他モジュールの呼び出しを「Ln_Proc(__Ln(ID))」と記述し、データの参照を「Ls__(ID)」と記述する。但し、Lnは呼び出すモジュールの層名称、Procは呼び出す処理名、IDは自モジュール名、Lsは自モジュールの層名称である。

【0035】

40

図4は、図1における下位層(L2)170の要素であるVcc補正アナログ入力(VAI)172の詳細図である。

【0036】

図4(A)は処理の概要を示すものである。図示するように、Vcc補正アナログ入力モジュールには、処理名1のVcc補正アナログ入力値更新処理と、処理名2のVcc補正アナログ入力値取得処理を有し、かつ、この処理のために必要な入力種別として、マイコンポートからの入力があることを示している。図4(B)は、Vcc補正アナログ入力値更新処理のフローチャート、図4(C)は、同じくVcc補正アナログ入力値取得処理のフローチャートである。図4(D)は、これら図4(A)、(B)、及び(C)を実装したソースコードの例である。

50

【 0 0 3 7 】

図 4 (B) の V c c 補正アナログ入力値更新処理では、まず、ステップ 4 0 1 において、ハードウェアインターフェース (H W I) 1 5 0 を介して、マイコンポートから V c c を入力する。次に、ステップ 4 0 2 において、V c c 補正係数を乗算し、更新値を得る。ステップ 4 0 3 では、取得した更新値をバッファへ保持する。

【 0 0 3 8 】

また、図 4 (C) の V c c 補正アナログ入力値所得の処理は、ステップ 4 0 4 において、上記バッファの値を返すことによって実現する。

【 0 0 3 9 】

図 4 (D) は、これらの処理を、C 言語のマクロを用いて記述したプログラムソースの例である。マクロ記述は、図 3 (D) と同一要領である。

【 0 0 4 0 】

図 5 は、図 1 のデバイスドライバ 1 2 0 中の階層構成記述部 1 9 1 の詳細図である。

【 0 0 4 1 】

図 5 (A) は、構成の概要を示すものである。それぞれの処理モジュールのラベル名 I A T , I A A , V A I , 及び F A I と、これらの下位に属する処理モジュールのラベル名が、F A I , V A I , A N 1 , 及び A N 2 であることを示している。すなわち、吸気温度演算 (I A T) 1 2 2 の下位モジュールが 1 次フィルタアナログ入力 1 7 1 であり、そのラベルが F A I 、吸入空気量演算 (I A A) 1 2 1 の下位モジュールが V c c 補正アナログ入力 1 7 2 であり、そのラベルが V A I である。また、1 次フィルタアナログ入力 1 7 1、すなわちラベル F A I の対応ハードウェアが、アナログ入出力ポートでありラベルが A N 2、V c c 補正アナログ入力 1 7 2、すなわちラベル V A I の対応ハードウェアが、アナログ入出力ポートでありラベルが A N 1 であることを示している。

【 0 0 4 2 】

図 5 (B) は、図 5 (A) の階層構成を基に実装した階層構成記述部 1 9 1 のソースコードである。このソースコードは、マクロを用いて記述し、「# d e f i n e I D _ L n I D n」あるいは「# d e f i n e I D _ A D D R x x x x」の形式である。ここで、I D はデバイスドライバモジュールの識別子 (ラベル)、I D n は当該モジュールと関連する他のモジュールの識別子 (ラベル)、_ A D D R はマイコンのポートを示す識別子 (ラベル) であり x x x はポート名を示す。

【 0 0 4 3 】

図 6 は、図 1 のデバイスドライバ 1 2 0 中の識別子展開部 1 9 2 の詳細図である。識別子展開部は「# d e f i n e _ L n (l y) l y # # _ L n」、「# d e f i n e _ A D D R (l y) l y # # _ A D D R」、「# d e f i n e L n _ (l y) L n _ # # l y」の形式である。ここで、L n は上位下位層の名称、A D D R はマイコンのポートを示す任意の識別子を示すものであり、その他は、前述と同じである。これらは C 言語の識別子連結機能「# #」を用い、他の層のラベルと置換することにより、層間のマクロ展開を実現するものである。

【 0 0 4 4 】

図 7 は、以上で説明した処理記述部、階層構成記述部、及び識別子展開部を用いた開発手順図である。前記のように作成したデバイスドライバプログラムソースコード 7 1 の階層構成記述部 7 1 1、識別子展開部 7 1 2、処理記述部 7 1 3 を、ビルド 7 2 を通して、デバイスドライバオブジェクトファイル 7 3 を得る。すなわち、プリプロセッサ 7 2 1、コンパイラ 7 2 2 を用いて、機械語コードに変換すると、デバイスドライバのオブジェクトファイル 7 3 が出力される。このオブジェクトファイル 7 3 では、階層化して構築した処理記述部が、プリプロセッサ 7 2 1 によって一つの層に展開されるため、階層間のオーバーヘッドが除去される。

【 0 0 4 5 】

図 8 は、本発明の一実施例により階層間のオーバーヘッドが除去されたソースコードの一例図である。呼び出し側のソースコードが図 8 (A) であり、これをプリプロセッサ 7

10

20

30

40

50

21によって展開した後のコードが図8(B)である。プリプロセッサ721によるマクロ展開により、図3(B)のステップ301、302、並びに図4(B)のステップ401の階層間の呼び出し処理が1層内に展開されている。このため、他階層呼び出し処理に伴う引数や、スタックポインタの退避処理が不要となり、処理の高速化とコードサイズの削減を実現できる。

【0046】

図8(C)は、従来の手法による機械語コードであり、図8(D)が、本発明の一実施例による削減分の例である。すなわち、図3(B)のステップ301の呼び出しを行う機械語コードが図8(C)に破線枠で囲った81であり、ひとつの階層間呼び出しにつき、81の分が削減できる。

10

【0047】

図9は、本発明の一実施例によるデバイスドライバ作成方法を実現するための開発環境及び開発手順図である。作成済みのデバイスドライバ120である識別子展開部192と処理記述部180を、ステップ901にて登録する。これを保存するデバイスドライバリポジトリ902から、ステップ903にて、必要な識別子展開部192と処理記述部180を取得する。他方、ハードウェア設計の結果得られる制御装置仕様904のうち、マイコンのポートとデバイスの対応を記述した入出力端子一覧表905を入力し、階層構成生成装置906を用いて、ステップ907にて階層構成記述部908を取得する。このようにして得られた階層構成記述部908、識別子展開部909、及び処理記述部910を、プリプロセッサ911により、ステップ912にてマクロ展開を行う。これにより、階層を用いて記述された処理記述部910から、識別子展開部909と階層構成記述部908に基いて階層構造を展開された処理記述部913を得ることができる。ここで得られた展開された処理記述部913は、階層構造をもたず、ただひとつの階層からなり、階層間の処理オーバーヘッドが削減されている。次に、コンパイラ914を用いたコンパイルステップ915で、デバイスドライバオブジェクト形式ファイル916を得る。このドライバオブジェクト形式ファイル916は、アプリケーションソースコード917をコンパイルして得られたアプリケーションオブジェクト形式ファイル918とともにリンカ919を用いて、ステップ920でリンクされ、制御ソフトウェア921が得られる。

20

【0048】

図10は、入出力端子一覧表905の一例を示す図である。入出力端子一覧表905は、ハードウェアデバイスとコントロールユニットの端子の対応を示すものである。これを用い、上位層(L1)160をハードウェアデバイスに対応する層とし、下位層をハードウェアポート131~133に対応する層とすると、図10は図5(A)に対応する。階層構成生成装置906は、これを基に、図5(B)を出力するものである。階層構成生成装置906を用いることにより、ハードウェア設計結果である入出力端子一覧表905から階層構成記述部908を、計算機を用いて自動生成可能となり、生産性を向上できる。階層構成生成装置906、デバイスドライバリポジトリ902、コンパイラ914としては、キーボード、マウス、ネットワークなどの入力手段と、CRTなどの表示手段、ハードディスクなどの記憶手段を備えた計算機を用いることで実現できる。

30

【産業上の利用可能性】

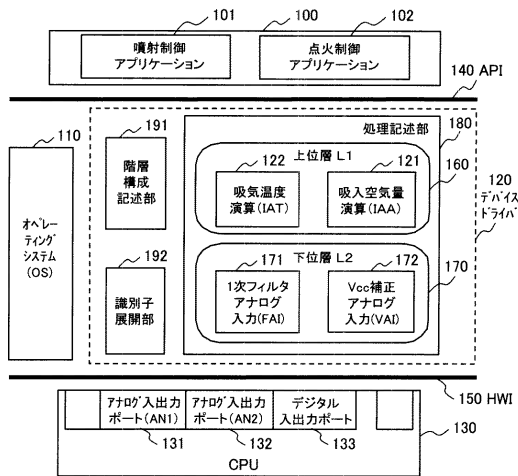
40

【0049】

本発明によるマイクロコンピュータのソフトウェアの作成方法や作成システムは、処理の迅速性と高信頼性が要求され、しかも、ハードウェアの変更頻度も激しい自動車のエンジン制御装置の電子制御ユニットに適用することによって、開発者の負担を大きく低減でき、高い利用可能性が期待される。

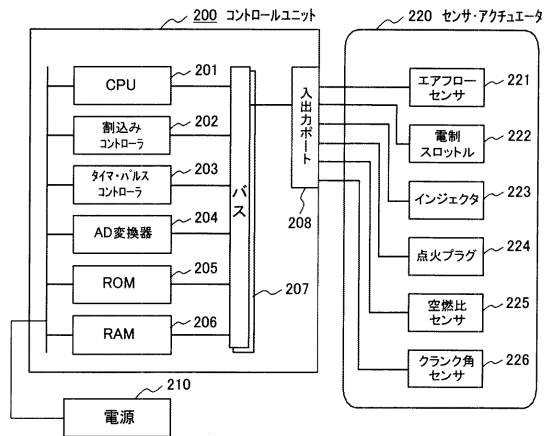
【図 1】

図 1



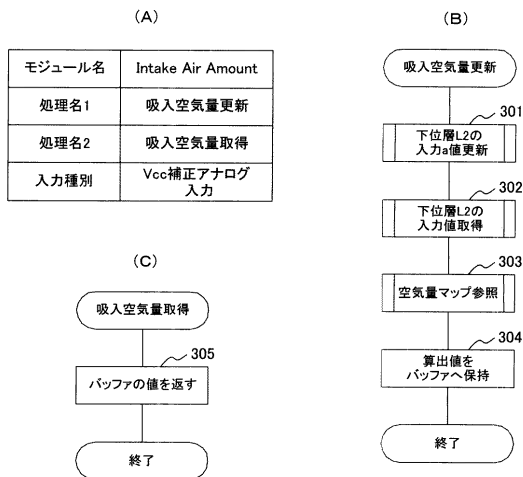
【図 2】

図 2



【図 3】

図 3



(D)

```

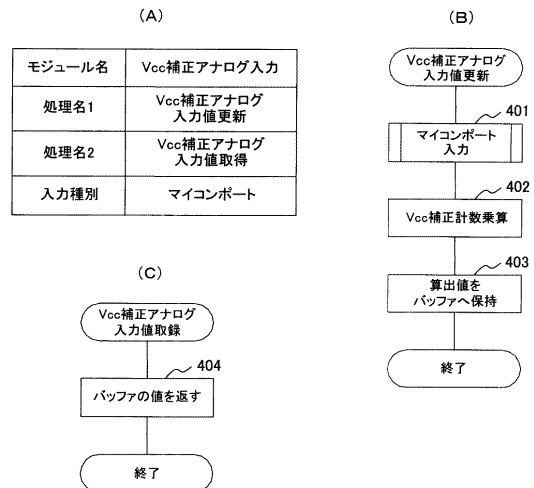
IAA.h
#define L1_IntakeAirAmount_Update(ly) ¥
L2_AIVCC_Update(L2(ly));¥
L1_(ly).result = MapComp(L2_AIVCC_Get(L2(ly)))

#define L1_IntakeAirAmount_Get(ly) ¥
(L2_AIVCC_Get(L2(ly)))

```

【図 4】

図 4



(D)

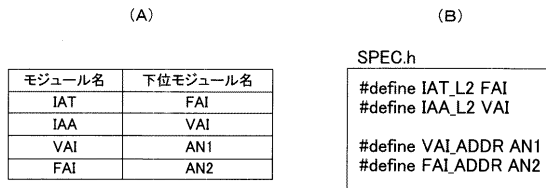
```

AIVCC.h
#define L2_AIVCC_Update(ly) L2_(ly).result = * _ADDR(ly)
#define L2_AIVCC_Get(ly) L2_(ly).result

```

【図 5】

図 5



【図 6】

図 6

MEX.h

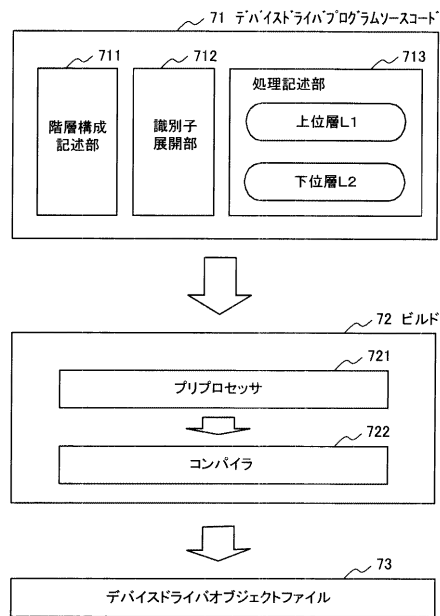
```

#define _L2(ly) ly##_L2
#define _ADDR(ly) ly##_ADDR

#define L1_(ly) L1_##ly
#define L2_(ly) L2_##ly
  
```

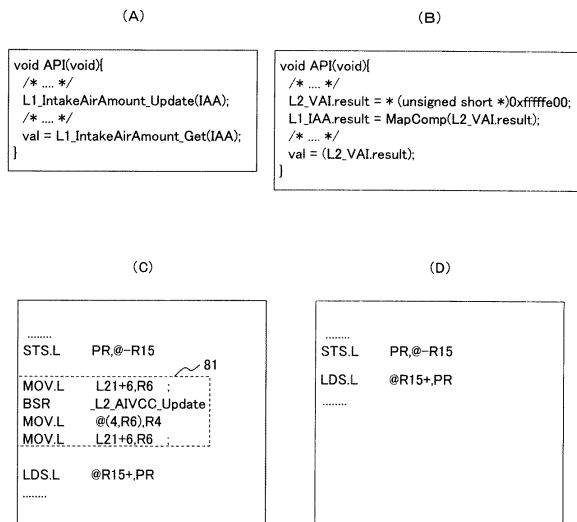
【図 7】

図 7



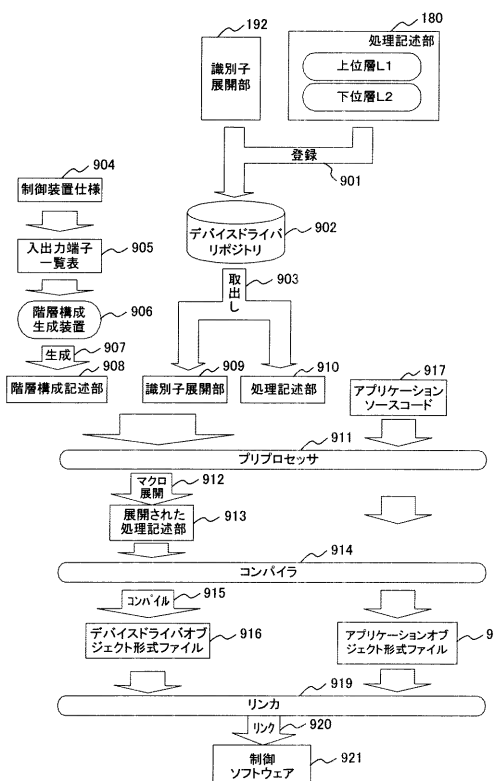
【図 8】

図 8



【図 9】

図 9



【図 10】

図 10

905

デバイス名	端子名
IAT	FAI
IAA	VAI
VAI	AN1
FAI	AN2

フロントページの続き

審査官 林 毅

(56)参考文献 特開昭63-181034(JP,A)

川口、外2名著,階層化データフロー図によるプログラムの設計と実動,電子情報通信学会論文誌,日本,社団法人電子情報通信学会,1999年10月25日,第J82-D-I巻、第10号,第1265-1275頁

(58)調査した分野(Int.Cl.,DB名)

G06F 9/44

B60R 16/02

G06F 9/45