

【公報種別】特許法第 17 条の 2 の規定による補正の掲載
 【部門区分】第 6 部門第 3 区分
 【発行日】平成 20 年 1 月 17 日 (2008.1.17)

【公開番号】特開 2006-154971(P2006-154971A)
 【公開日】平成 18 年 6 月 15 日 (2006.6.15)
 【年通号数】公開・登録公報 2006-023
 【出願番号】特願 2004-341236(P2004-341236)
 【国際特許分類】

G 0 6 F 9/45 (2006.01)

【 F I 】

G 0 6 F 9/44 3 2 2 F

【手続補正書】

【提出日】平成 19 年 11 月 22 日 (2007.11.22)

【手続補正 1】

【補正対象書類名】明細書

【補正対象項目名】0 0 5 7

【補正方法】変更

【補正の内容】

【0 0 5 7】

命令ブロック 5 0 1 を実行した後、分岐ブロック 5 0 6 において y が 1 0 以上であるかどうかを判定する。y が 1 0 以上 (分岐ブロック 5 0 6 の y e s) ならば、命令ブロック 5 0 2 に進み、x に命令ブロック 5 0 1 で得られた y から 1 0 引いた値を格納する。y が 1 0 以下 (分岐ブロック 5 0 6 の n o) ならば、命令ブロック 5 0 3 に進み、x に命令ブロック 5 0 0 において得られた x の値に 1 0 足した値を格納する。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0 0 6 3

【補正方法】変更

【補正の内容】

【0 0 6 3】

本発明に係るコンパイラ装置はスレッド 6 0 0 以外に、実行頻度の高い実行経路に関して、その実行経路上の命令を実行できるように変換、生成したスレッドが図 7 ~ 図 9 に示すコード列である。

図 7 には、図 5 の実行頻度一位の実行経路 5 5 1 上の命令群をコード化したスレッド 7 0 0 を記してある。コード 7 0 1、7 1 3、7 1 9 はそれぞれラベルコードである。コード 7 0 2 ~ 7 1 2 は、図 5 のブロック 5 0 0、5 0 1、5 0 2 を他の実行経路に分岐しないようにコード化した内容になっており、ブロック 5 0 5、5 0 6 がこの実行経路を通るかどうかの 2 択に変えたコードを含んでいる。

【手続補正 3】

【補正対象書類名】明細書

【補正対象項目名】0 0 6 4

【補正方法】変更

【補正の内容】

【0 0 6 4】

コード 7 1 4、7 1 5 は実行経路 5 1 1 を通る場合に他のプロセッサエレメントで実行されているスレッドを停止させるコードである。

コード 7 1 7、7 1 8 は、実行経路 5 5 1 が実行されたときにコードを反映させる処理になる。この反映処理は、実行経路 5 5 1 の出口で生存していて、かつ実行経路 5 5 1 で

変更されているデータが対象になる。

コード 7 2 0、7 2 1はこのスレッド 7 0 0 の終了処理である。

図 8 には、実行経路 5 5 2 上の命令をアセンブラコードに変換したスレッド 8 0 0 を記してある。

【手続補正 4】

【補正対象書類名】明細書

【補正対象項目名】0 0 6 5

【補正方法】変更

【補正の内容】

【0 0 6 5】

コード 8 0 1、8 1 4、8 1 7、8 2 0はそれぞれラベルコードである。

またコード 8 0 2 ~ 8 1 3 は図 5 におけるブロック 5 0 0、5 0 1、5 0 3 の命令をコード化した内容になっている。コード 8 1 5、8 1 6 は実行経路 5 5 2 を通ることが確定した場合に、他のプロセッサエレメントで実行されているスレッドを停止させ、コード 8 2 1、8 2 2 はスレッド 8 0 0 終了処理を行っている。また、コード 8 1 8、8 1 9 は、実行経路 5 5 2 が実行されたときにコードを反映させる処理になる。

【手続補正 5】

【補正対象書類名】明細書

【補正対象項目名】0 0 6 6

【補正方法】変更

【補正の内容】

【0 0 6 6】

図 9 には、図 5 におけるブロック 5 0 0、5 0 4 を通る実行経路を通る場合の最適化されたコードであるスレッド 9 0 0 を記してある。

コード 9 0 1、9 0 9、9 1 2、9 1 4はそれぞれラベルコードである。

またコード 9 0 2 ~ 9 0 8 は図 5 におけるブロック 5 0 0、5 0 4 の命令をコード化した内容になっている。コード 9 1 0、9 1 1はこの実行経路を通ることが確定した場合に、他のプロセッサエレメントで実行されているスレッドを停止させ、コード 9 1 5、9 1 6 はスレッド 9 0 0 の終了処理を行っている。また、コード 9 1 3 は、ブロック 5 0 0、5 0 4 が実行されたときにコードを反映させる処理になる。

【手続補正 6】

【補正対象書類名】明細書

【補正対象項目名】0 0 6 7

【補正方法】変更

【補正の内容】

【0 0 6 7】

図 7 のコード 7 0 2 ~ 7 1 2、図 8 のコード 8 0 2 ~ 8 1 3、図 9 の 9 0 2 ~ 9 0 8 において同じメモリへの格納処理が生じる場合、各スレッドでの値の保証ができなくなり、プログラム作成者の望む結果が得られないことがあるため、メモリ共有型では別のメモリ領域への格納処理に変えることがある。

【手続補正 7】

【補正対象書類名】明細書

【補正対象項目名】0 0 7 6

【補正方法】変更

【補正の内容】

【0 0 7 6】

「blt (番地 3)」は、直前の比較コード cmp (番地 1),(番地 2) の比較において、(番地 2) の値が(番地 1) の値よりも小さい場合に、(番地 3) に指定されるコードに命令を飛ばすコードである。それ以外の場合には以降のコードを続行する。例えば、図 7 のコード 7 0 6 においては、その前のコード 7 0 5 の比較を受けて、レジスタ D 0

に格納されている値が 0 よりも小さい場合に、コード 7 0 7 からコード 7 1 9 までは実行されずにコード 7 2 0 に飛び、以降のコードが実行されることになる。

【手続補正 8】

【補正対象書類名】明細書

【補正対象項目名】0 0 8 1

【補正方法】変更

【補正の内容】

【0 0 8 1】

「#beginthread (番地)」は、スレッドの開始コードで、(番地)のスレッドの実行開始を促す。例えば、図 1 0 のコード 1 0 0 6 においては、L A B E L 5 0 0 - 5 0 1 - 5 0 2 で始まるスレッド、即ち図 7 のコード列で示されるスレッド 7 0 0 を実行する。

「#endthread」は、スレッドの終了コードで、現在実行しているスレッドを終了状態に設定し、スレッドが終了したことを示す情報を返す。例えば、図 7 のコード 7 2 0 においてはスレッド 7 0 0 を終了し、終了したことを示す情報をプログラムのメインに返す。

【手続補正 9】

【補正対象書類名】明細書

【補正対象項目名】0 0 8 3

【補正方法】変更

【補正の内容】

【0 0 8 3】

「#waitthread (番地)」は、スレッドの終了を待つコードで、(番地)から始まるスレッドの実行結果の終了を待つ。この終了は上記#endthreadからの情報によって知ることができる。例えば、図 1 0 のコード 1 0 1 0 においては、T H R E A D 5 0 0 - 5 0 4 の終了を待っている。

「#commit (番地 1)」は、メインプログラム、若しくはスレッドプロセスで生成した情報(番地 1)を、メインプログラムと全てのスレッドプロセスに反映させるコードである。

【手続補正 1 0】

【補正対象書類名】明細書

【補正対象項目名】0 0 9 4

【補正方法】変更

【補正の内容】

【0 0 9 4】

ターゲットハードウェア 1 3 0 のメモリの形態がメモリ共有型であった場合には、図 7 ~ 図 9 におけるコード 7 0 2、8 0 2、9 0 2 のように逐一 a の値をレジスタに格納しな
おさなければならなかったが、メモリ分散型ではそのようにする必要はなく、例えば、図 2 1 にあるメモリ分散型の場合のメインスレッドのように、メインスレッドにおいて各プロセッサエレメントの各レジスタにブロードキャストすることで、処理を省略することができる。そのためのコードが図 2 1 におけるコード 2 1 0 4 ~ 2 1 0 7 である。

【手続補正 1 1】

【補正対象書類名】明細書

【補正対象項目名】0 0 9 5

【補正方法】変更

【補正の内容】

【0 0 9 5】

コード 2 1 0 5 では、コード 2 1 0 1 ~ 2 1 0 3 で生成されたスレッドに a の値を各プロセッサエレメントのメモリのレジスタ D 0 に格納するように各スレッドを実行しているプロセッサエレメントに通達する。

コード 2 1 0 6 では、コード 2 1 0 1 ~ 2 1 0 3 で生成されたスレッドに b の値を各プロセッサエレメントのメモリのレジスタ D 1 に格納するように各スレッドを実行している

プロセッサエレメントに通達する。

コード 2 1 0 7では、コード 2 1 0 1 ~ 2 1 0 3で生成されたスレッドに c の値を各プロセッサエレメントのメモリのレジスタ D 2 に格納するように各スレッドを実行しているプロセッサエレメントに通達する。

【手続補正 1 2】

【補正対象書類名】明細書

【補正対象項目名】0 0 9 9

【補正方法】変更

【補正の内容】

【0 0 9 9】

ターゲットハードウェアのプロセッサエレメントの数を取得し、コンパイラによって生成されるスレッドの数との整合をとるコードがラベル 1 1 0 5 から始まるコード 1 1 0 6 ~ 1 1 1 8 に記されている。

まず、コンパイラによって生成されるスレッドの数 m を取得し、その数 m をレジスタ D 0 に格納する（コード 1 1 0 6）。次にターゲットハードウェアの並列実行可能なプロセッサエレメントの数 n を取得し、その値をレジスタ D 1 に格納する（コード 1 1 0 7）。そしてレジスタ D 0 に格納された m とレジスタ D 1 に格納された n の値を比較し（コード 1 1 0 8）、n = m ならばラベルコード 1 1 1 1 に飛び（コード 1 1 0 9）、n < m ならばラベルコード 1 1 1 3 に飛ぶ（コード 1 1 1 0）。

【手続補正 1 3】

【補正対象書類名】明細書

【補正対象項目名】0 1 0 0

【補正方法】変更

【補正の内容】

【0 1 0 0】

n = m の場合には、特に問題はなく、m の値をレジスタ D 1 に格納する（コード 1 1 1 2）。

n < m の場合には、生成されたスレッドの数 m の方が並列実行可能な命令数 n を上回っているため、すべてのスレッドを実行できない。

そこで、まず、レジスタ D 1 に格納されている値 n から 1 引いた数を D 1 に格納しなおす（コード 1 1 1 4）。この n - 1 の数が必要とする実行可能なスレッドの数である。一つ余るプロセッサエレメントは、元のプログラムをそのままコードにした図 6 のコードを実行する。

【手続補正 1 4】

【補正対象書類名】明細書

【補正対象項目名】0 1 0 1

【補正方法】変更

【補正の内容】

【0 1 0 1】

次に命令の番地計算を行うために、n - 1 の値に命令語長、例えば 8 bit なら 8 をかけ（コード 1 1 1 5）、P__POINTER の番地をレジスタ D 2 に格納する（コード 1 1 1 6）。レジスタ D 2 に格納された値からレジスタ D 1 に格納された値を引いて、算出された値でレジスタ D 2 を更新する（コード 1 1 1 7）。そして、レジスタ D 2 に格納されている番地の値に命令を飛ばす（コード 1 1 1 8）。この D 2 に格納されている値によって以下のどのスレッドから開始するのかを決定する。例えば、ターゲットハードウェアの並列実行可能数が 2 であった場合には、コード 1 1 2 2 から開始する。並列実行可能数が 3 の場合には、コード 1 1 2 1 から開始する。コード 1 1 2 0 ~ コード 1 1 2 2 に関しては下から順に実行頻度の高かった実行経路を実行するスレッド開始のコードになっている。

【手続補正 1 5】

【補正対象書類名】明細書

【補正対象項目名】0 1 0 2

【補正方法】変更

【補正の内容】

【0 1 0 2】

このスレッド 1 1 0 0 をメインスレッドにすることにより、ターゲットハードウェアの並列実行可能数を得ていない場合であってもこのコンパイラ装置は、実行プログラムを生成できる。なお、コード 1 1 2 4 以降のコードは、全てを図示していないが、図 1 0 におけるコード 1 0 1 2 以降のコードと同様の構成とする。

ターゲットハードウェアの性能が分からない場合に、その性能を取得する必要があり、その流れを図 1 6 のフローチャートに簡単に示しておいた。

【手続補正 1 6】

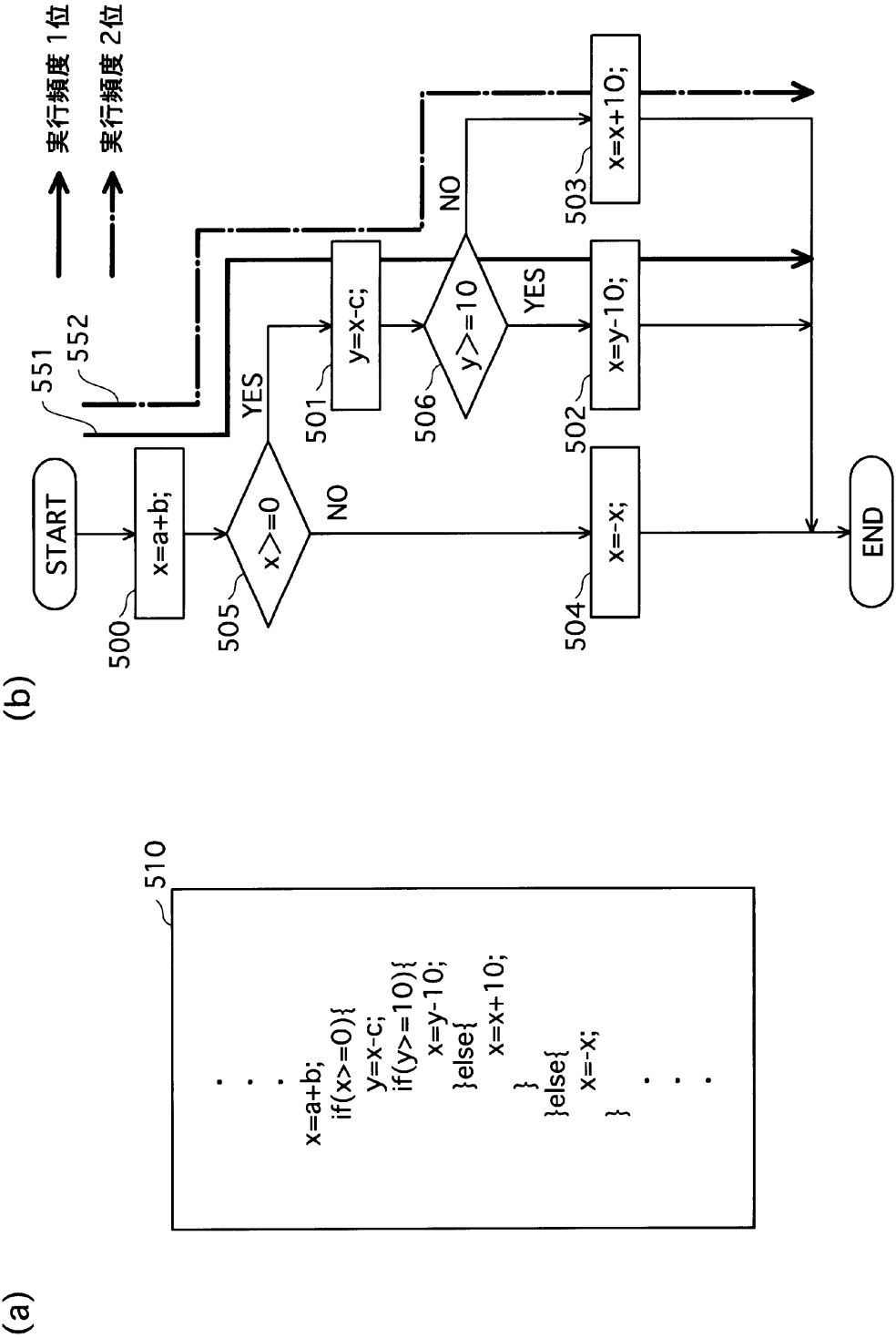
【補正対象書類名】図面

【補正対象項目名】図 5

【補正方法】変更

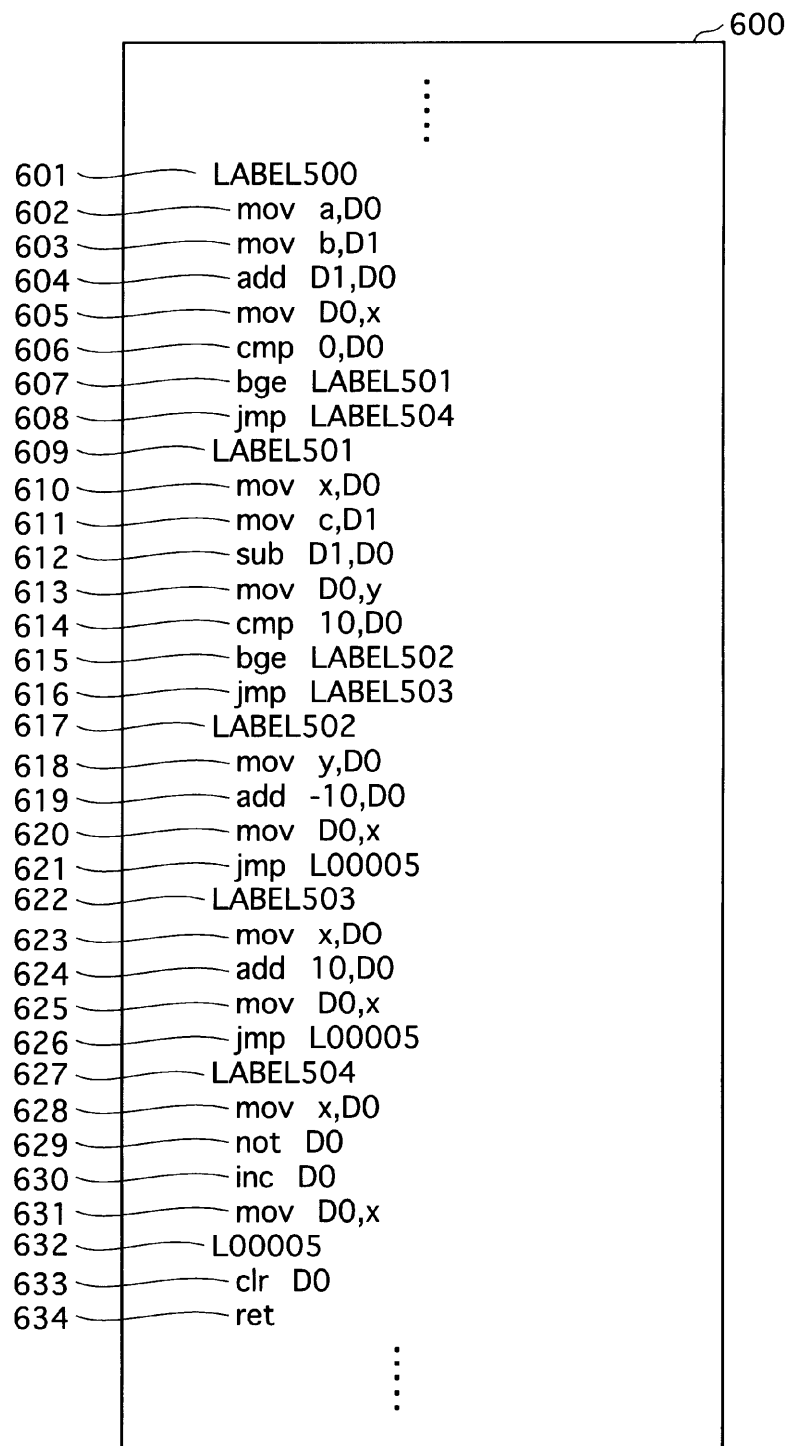
【補正の内容】

【 図 5 】



【 手続補正 1 7 】
【 補正対象書類名 】 図面
【 補正対象項目名 】 図 6
【 補正方法 】 変更
【 補正の内容 】

【 図 6 】



【 手続補正 18 】

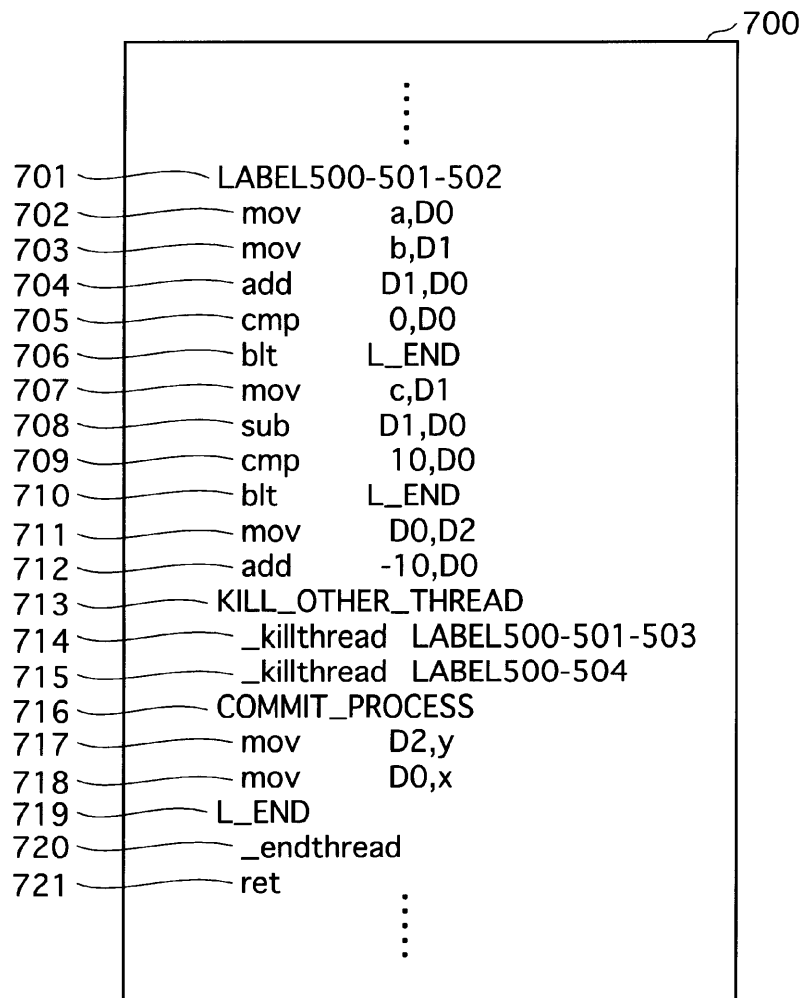
【 補正対象書類名 】 図面

【 補正対象項目名 】 図 7

【 補正方法 】 変更

【 補正の内容 】

【 図 7 】



【 手続補正 19 】

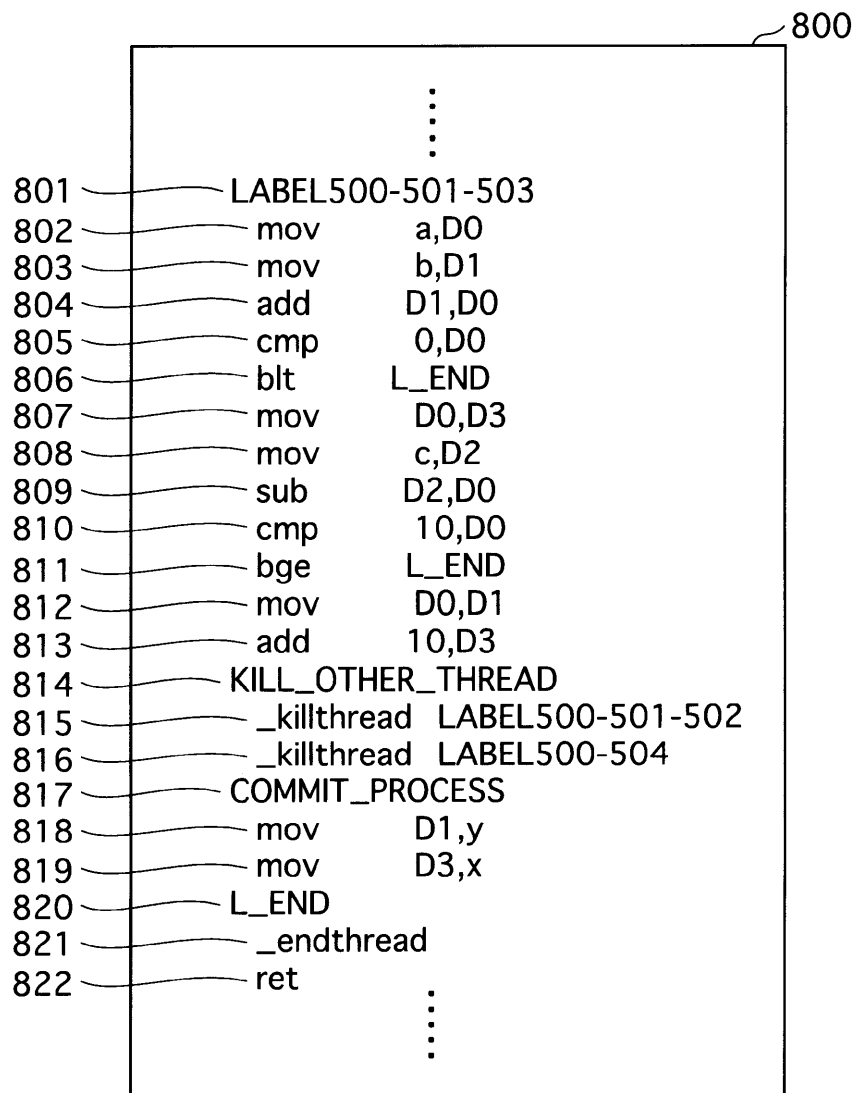
【 補正対象書類名 】 図面

【 補正対象項目名 】 図 8

【 補正方法 】 変更

【 補正の内容 】

【 図 8 】



【 手続補正 2 0 】

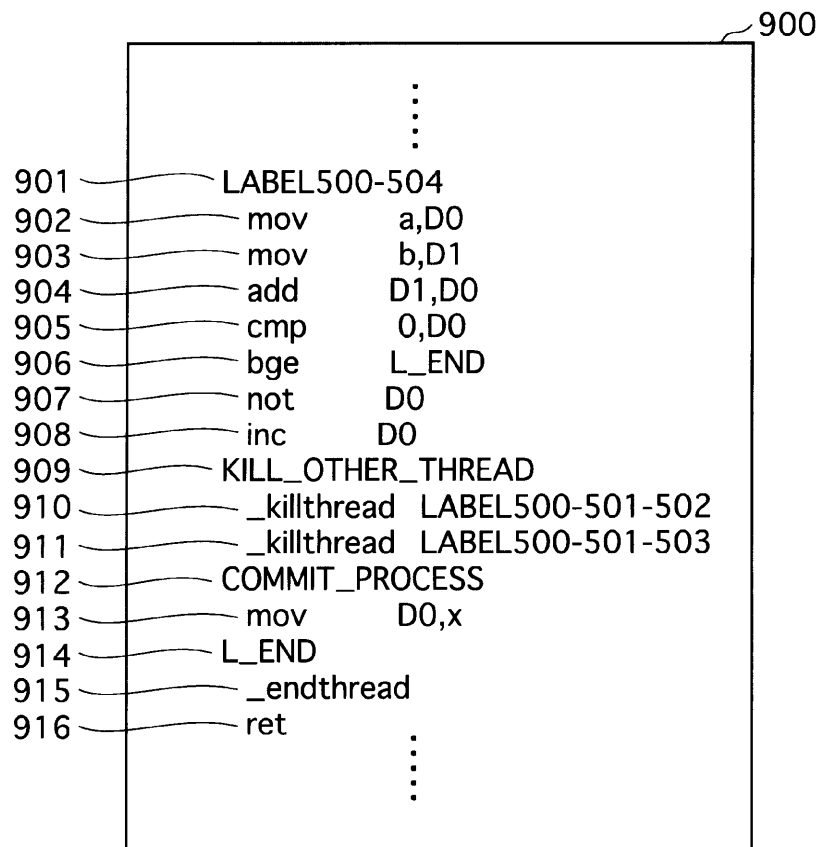
【 補正対象書類名 】 図面

【 補正対象項目名 】 図 9

【 補正方法 】 変更

【 補正の内容 】

【 図 9 】



【 手続補正 2 1 】

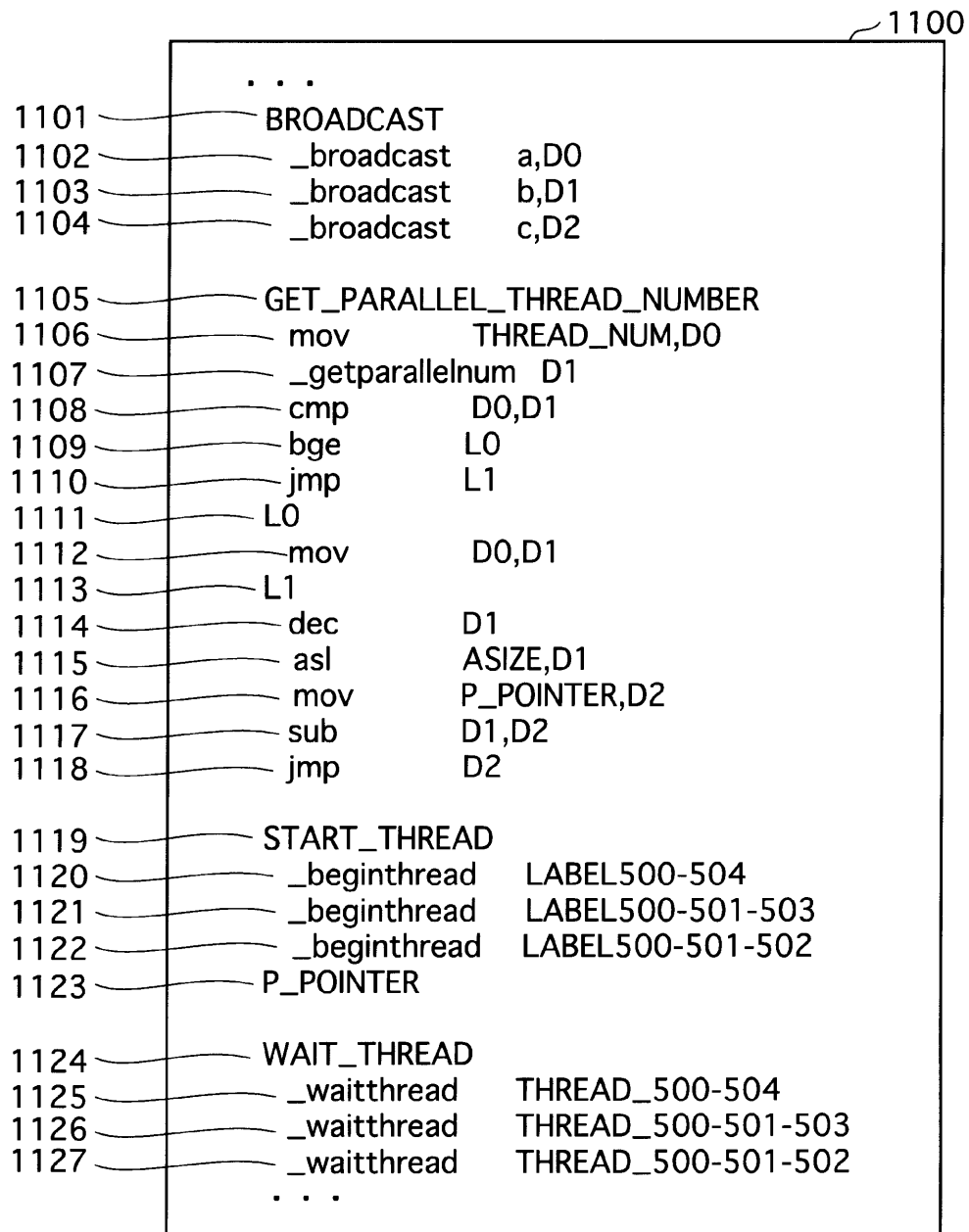
【 補正対象書類名 】 図面

【 補正対象項目名 】 図 1 1

【 補正方法 】 変更

【 補正の内容 】

【図 1 1】



【手続補正 2 2】

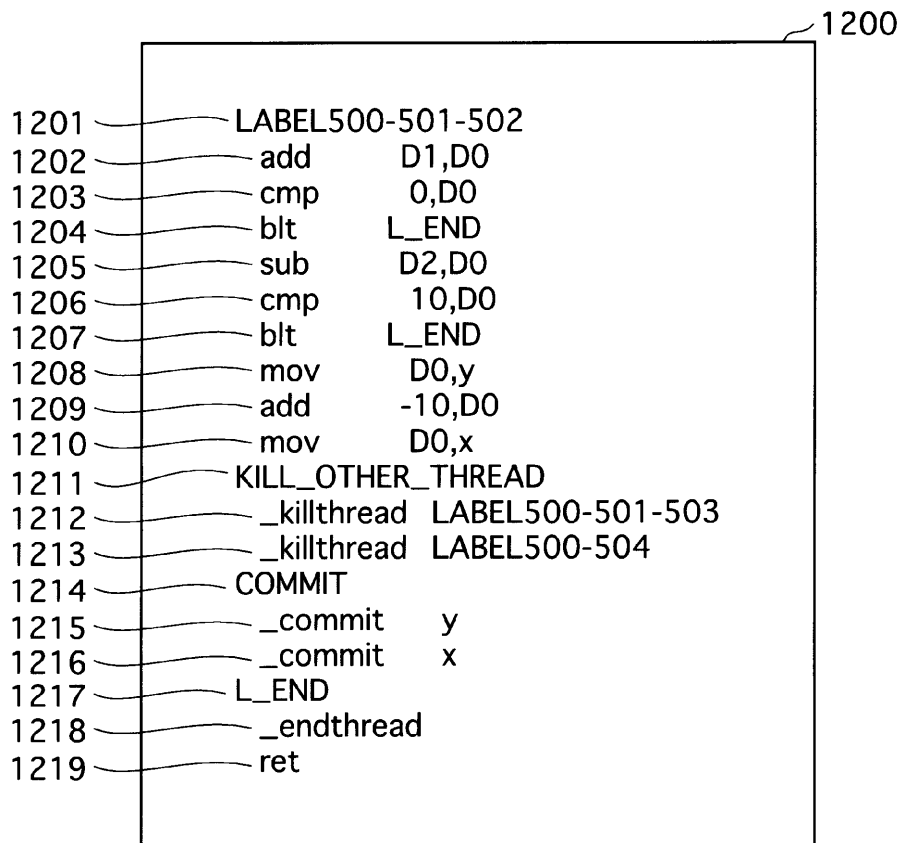
【補正対象書類名】図面

【補正対象項目名】図 1 2

【補正方法】変更

【補正の内容】

【 図 1 2 】



【 手続補正 2 3 】

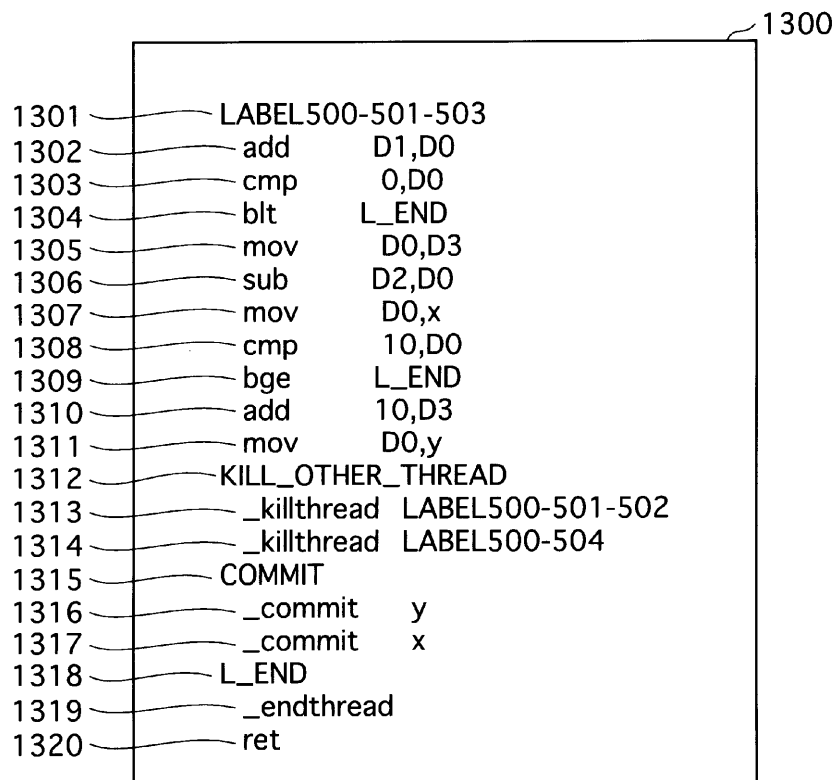
【 補正対象書類名 】 図面

【 補正対象項目名 】 図 1 3

【 補正方法 】 変更

【 補正の内容 】

【図 1 3】



【手続補正 2 4】

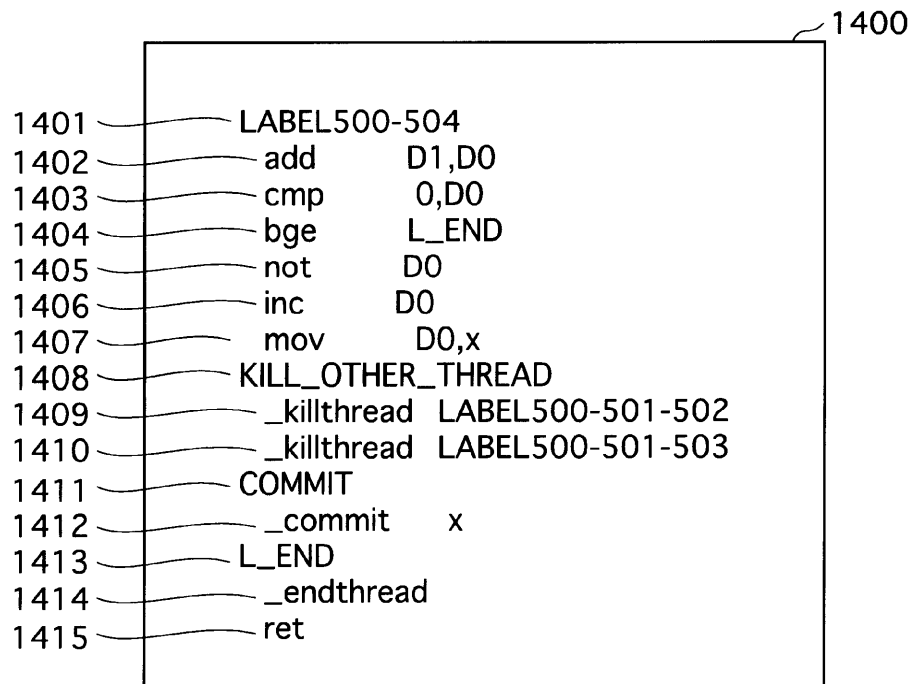
【補正対象書類名】図面

【補正対象項目名】図 1 4

【補正方法】変更

【補正の内容】

【 図 1 4 】



【 手続補正 2 5 】

【 補正対象書類名 】 図面

【 補正対象項目名 】 図 2 1

【 補正方法 】 変更

【 補正の内容 】

【 図 2 1 】

