

労働大臣賞（入選）

Windows95上で動作するCASLシミュレータ 「Turbo CASL for Windows95」

宮城県立塩釜高等技術専門学校 情報処理科 新妻 幹也

1. はじめに

3年半ほど前（本誌平成7年1号）にPC98用（DOS版）のCASLシミュレータを発表したところ、何件かのお問い合わせがあり、プログラムを提供させていただく機会がありました。また、学生の参考にとソースプログラムの提供の依頼もいただきました。

私事になりますが、2年ほど現場を離れていて久しぶりで現職場に戻ってみますと、当然のことではありますが、情報処理分野の状況変化の激しさに改めて驚かされました。数年前までパソコンといえば「PC98」、PC98用にソフトを作れば使ってもらえる機会も多く、多くのソフトがPC98用で占められていた状況がありました。

しかし、その状況は一変し、Windows95の発表以来、ソフトウェアはWindows95環境下ではメーカーや機種を問わないという状況がほぼ定着しました。逆に昔のDOS環境やPC98に依存したソフトは、それ自体動作する環境がなくなりつつあります。今や、ソフトはWindows95用が当たり前の時代になってしまいました。

そのような状況から、以前に開発したPC98用のCASLシミュレータをWindows95用にリメイクしたので紹介いたします。



Turbo CASLスタートアップ画面

2. PC98・DOS版と変わった点

以前のPC98用DOS版と変わった点で、機能アップした内容は次のとおりです。

Windows95（98）上で動作する

オペレーションは、Windowsの一般的アプリケーションとほぼ同じ

メモリの状況を示すメモリタワーが3本になり、メモリの状況を見やすくした

レジスタの値は10進、16進、2進形式で同時表示にした

スクロールなしで編集できるプログラム行数を32行にした

マクロ命令のINとOUTの表示を別にした

ヘルプ機能で画面を変えずに各命令語の学習ができるようになった

エディタ機能が強化された（カット&ペースト等）

逆に以前にあった機能を削除した内容は次のとおりです。

画面右上に表示されていた時計を削除（OSがサポートしているため）

命令語実行時の発音機能（要望があればまたつけたい）

その他の機能はほぼ前シミュレータと同じです。

3 . Turbo CASLの特徴

インターネットでCASLのシミュレータを検索するといくつかのものができます。

それらの中にもフリーソフトで提供されているものもあり、それらをお使いになっている教育訓練機関も多いと思います。それらのソフトと「Turbo CASL」の大きな違いは、シミュレータの画面構成や[プログラム入力] [アッセンブル] [シミュレート]の一連のオペレーションにあると思いますが、このシミュレータではあくまでもコンピュータの基礎知識がない学生が、アセンブリ言語という全く未知の分野で、あまり拒否反応を示さずに学習していけるように配慮しています。具体的には、入力したプログラムをアッセンブルすると、それがマシン語コードに変換されてメモリ上に配置される様子が画面上に表示されたり、また、それを実行したときに現在どの部分を実行しているのかわかるように、プログラムソースやメモリ部分にアクティブバーを点灯させて示すようになっています。

シミュレートに関しては、入力されたプログラムが、仮想のマシン語コード（独自に設定）に変換され、シミュレート時にもそのコードが解釈されて高速に動作します。例えば、かけ算のプログラムの初歩段階としてシフト命令を使わずに、足し算のループによって処理するような場合（ 2×10000 など）でも、瞬時に結果を出すことができます。

命令語の学習については、「命令語のヘルプ機能」を設けており、全命令について概要や具体的プログラム例を画面上で確認することができます。

4 . Turbo CASLの概要

4.1 インストール

このプログラムは、1.44MBタイプのフロッピーディスク2枚からお使いになるパソコン（Windows 95のOSが使用できる環境）のハードディスクにインストールする必要があります。

その手順は次のとおりです。

Turbo CASLフロッピーディスク をドライブに入れて、その中に入っているSetup.exeを起動します。

セットアッププログラムが起動しますので、画面の指示に従ってインストールを進めます。

インストール作業で特に別のディレクトリ等を設定しない場合は、図1のようにセットアップが行われます（ハードディスクがC:の場合）。

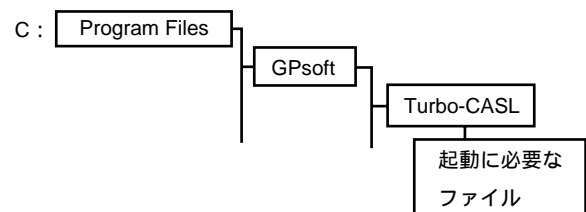


図1

セットアップ後はフロッピーディスクは使用しません。また、このプログラムを動かすために必要なファイルがWindowsディレクトリにもコピーされるため、Turbo-CASLディレクトリの内容をすべてコピーしても正常に動作しないこともありますので、プログラムを使用する場合は、必ずフロッピーディスクからセットアップを実行して行ってください。なお、このセットアップディスクの作成には、Install Shield社のInstall Shield Expressを使用しています。

4.2 このプログラムの動作環境

このプログラムはWindows95の環境下で動作しますが、十分に機能を発揮するためには、次の条件を満たす必要があります。

CPU インテル 80486 クロック
 66M以上を搭載のコンピュータ
 メモリ 32MB以上
 画面 800×600ドット以上のカラー
 65536色以上のコンピュータ
 マウス 必須
 キーボード 必須

Open
Save as
Save
Print
Print Setup
Exit

4.3 Turbo CASLの使い方

インストールをしたフォルダまたは、Windows スタートメニューからCASL.EXEを実行すると図2のような画面が表示されます。この画面上ですべての操作・学習ができるようになっています。

File

これをクリックするとさらに、

が表示されます。

Openは、すでに登録してあるソースプログラムを読み出すときに選びます。

Save asは、新規にプログラムを保存するとき、または、既存のファイルを読み出して修正を加えて別名で保存するときを選びます。

Saveは開いたファイルと同じ名称で保存するときを選びます。

Printは、現在エディタ（部分）に表示されて

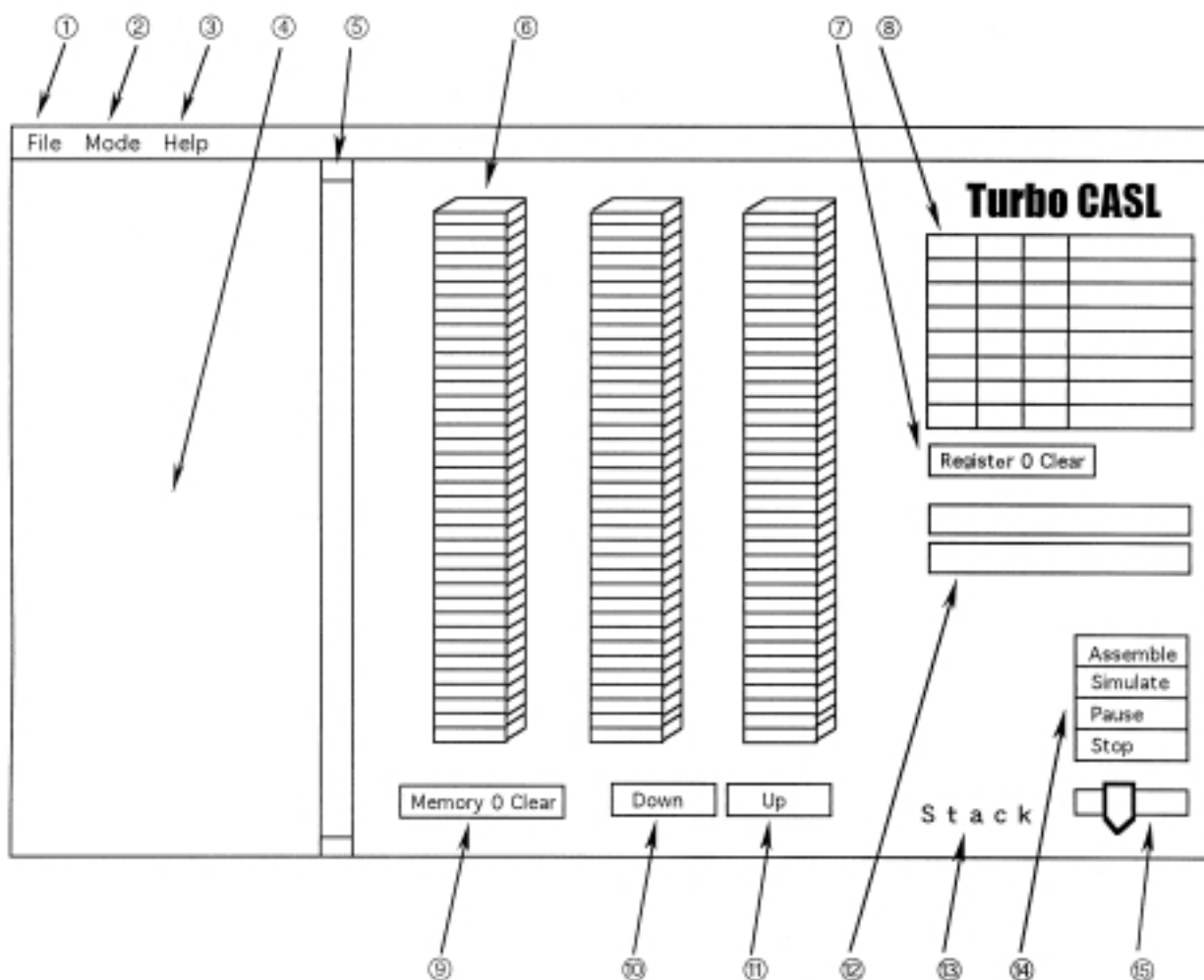


図2

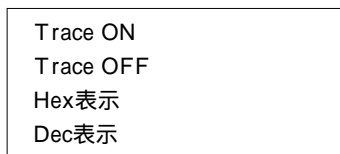
いるプログラムのリストを印刷するときを選びます。

Print Setupは、使用するプリンタのセットアップをするときに選びます。

Exitは、このシミュレータを終了するときを選びます。

Mode

これをクリックするとさらに、



が表示されます。

Trace ONは、エディタに表示されているプログラムをシミュレートするとき、現在のプログラムの実行行(ぎょう)と、メモリの実行アドレスにグリーンの実行ランプ(アクティブバー)を表示し、さらに、のスピード調整バーの調整によってシミュレートスピードを可変できるようにするときを選びます。プログラム実行時はこのモードになっています。

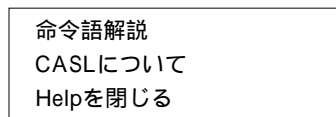
Trace OFFでは、上記の動作は省略され、実行結果のみが瞬時に出力されます。このモードでも、内部動作はTrace ON時のときと、全く同じくアッセンブルによって生成されたマシン語コードを忠実に実行しています。ループを1000回も回すようなプログラムでも瞬時に結果を出しますので、ループ回数の多いプログラムを作成して実行するときには便利です(Trace ON時の最速状態でもTrace OFF時の実行スピードには及びません)。

Hex表示は、アドレスに格納されている値の表示を16進数で行うことを指定するものです。起動時はこのモードになっています。

Dec表示は、アドレスに格納されている値の表示を10進数で行うことを指定するものです。DCなどで値を10進数で設定したものを確認するときには便利ですが、命令語で80より上のコードを持つものは、マイナスで表示が行われますから見にくくなります。

Help

これをクリックするとさらに、



が表示されます。

命令語解説を選ぶとさらに階層メニューが表示され、CASLの各命令語が表示されます。そこで、学習したい命令語を選ぶとその解説が表示されます。終わるときは「Helpを閉じる」を選びます。

「CASLについて」を選ぶとCASL全般についての解説が表示されます。終わるときは「Helpを閉じる」を選びます。

注意：CASLのサンプルソースをロードした後はディレクトリがASにあるため、そのままHelp画面を開こうとすると「ファイルが見つかりません」というエラーが表示されますので、そのときは、File Openでダイアログを開き、ディレクトリをTurbo-CASLにしてから実行してください。

プログラム編集エリア

この部分はソースプログラムを編集する部分です。全く新規にプログラムを入力していく場合はCASLの規則に従ってここに直接記入していきます。最初は、必ずこのエリアの任意の部分をマウスで左クリックしてください。そうするとカーソルが表示されます。このエディタは一通りのエディタ機能は備えています。通常はインサートモードになっていますが、キーボードの「insert」キーを押すとオーバーストライクモードになります。しかし、この場合でも、カーソルの形状は変わりませんので、注意が必要です。また、文字のコピーや削除は、マウスでその部分をドラッグした後に「CTRL+C」でバッファへコピー、「CTRL+X」でバッファへカット、「CTRL+V」でペースト等もできるようになっています。行の挿入は、インサートモードでリターンキーを押すだけです。

エディタスクロールバー

これは、プログラムが32行を超える場合に、それ

以降にあるプログラムを表示したりするときに使います。使い方は表示されている垂直のバーをマウスでドラッグします。

バーの長さは、現在表示されているプログラムが全体のどれぐらいの割合にあるかを示しています。スクロールする必要がない(32行以下)場合は、このバーは表示されません。

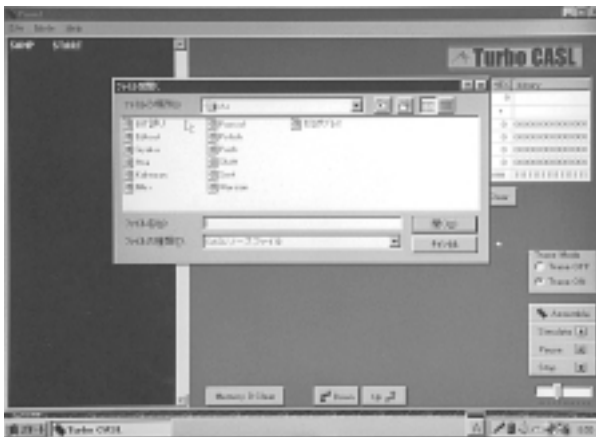
アドレス表示タワー

これは、ソースプログラムをアセンブルしたときに、それに対応するマシン語コードや各種データの設定等がメモリに格納された様子を示すものです。CASLの仕様では、プログラムの格納されるアドレス等は特に決められていませんので、このシミュレータでは、0000番地から格納されるようにしています。ただし、FFFF番地まですべてをえるよ

うな設定はしていませんので、現実には使えるのは、1000番地(10進)までですので、絶対番地指定をするときには注意が必要です。この1000番地までという制限は、このシミュレータの性質上それ以上の長いプログラムを書くことはないという想定のもとに設けています。

Register 0 Clear

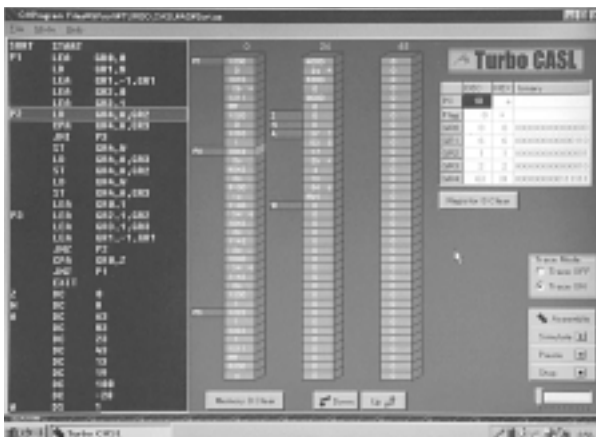
これは、レジスタを0で初期化するものです。ただし、GR 4については、-1に設定されます。これは、GR 4をスタックポインタとして使用することを想定してのためです。通常は、レジスタにどんな値が設定されていても、実行結果に影響を与えないようにプログラムすべきですが、どうしてもレジスタをクリアしてから実行したいときに使います。



既存ファイルのオープン画面



エディタのカット&ペースト機能



プログラムをシミュレート中の画面



エディタ画面のスクロールバー

レジスタ等の表示

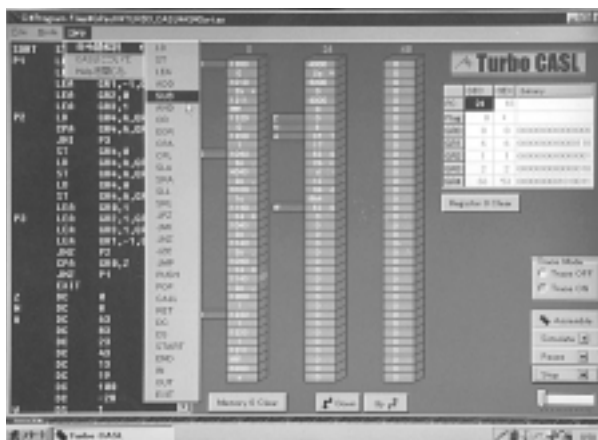
これは、プログラムカウンタ、フラグ、各レジスタの値を、10進数、16進数、および2進数で表示するためのboxです。

Memory 0 Clear

これは、メモリを0で初期化するものです。通常は、メモリにどんな値が設定されていても、実行結果に影響を与えないようにプログラムすべきですが、どうしてもメモリをクリアしてから実行したいときに使います。

Down

これは、メモリの表示アドレスを1ブロック分(36アドレス分)スクロールさせて、下位を表示させるためのキーです。



命令語参照のための選択画面



命令語参照画面

Up

これは、メモリの表示アドレスを1ブロック分(36アドレス分)スクロールさせて、上位を表示させるためのキーです。

In Box, Out Box

これは、キーボード入力受付Box(上段)と文字出力Box(下段)です。CASLにはマクロ命令として、INとOUTがありますが、それらの命令が実行されたときに機能するのがこの2つのBoxです。IN命令が実行されると上段のIn Boxがキーボードからの入力受付用に表示されますから、そこをマウスで左クリックして必要な文字列を入力してください。入力を終了したら、Returnキーを押してください。

スタック表示

PUSH命令、POP命令が実行されると、この部分にスタック領域へのデータの積み上げ、積み卸しが表示されます。スタック領域はFFFF番地から256バイト確保してありますが、この領域にビジュアルに積み上げられるスタックの数は16程度です。この量は大変に少ないと思われるかもしれませんが、このシミュレータの性質上、全く問題にはならないと考えています。

- 1 Assembleボタン

このボタンはソースプログラムをアセンブルするときに押します。文法上のエラーがなければ、中央のアドレス表示エリアにラベルも含めたプログラ



ソースプログラムの印刷機能

ムコードが表示されます。エラーがある場合はエラーの発生した行とその発生理由を表示します。

- 2 Simulateボタン

このボタンはAssembleでエラーが発生しなかったときにプログラムを実行するためのボタンです。

- 3 Pauseボタン

このボタンはSimulateを一時的に停止させるためのボタンです。再開するためには、もう一度押してください。

- 4 Stopボタン

このボタンはSimulateを完全に停止させるためのボタンです。再開するためには、もう一度Simulateボタンを押す必要があります。

トラックバー

これは、Simulate中にステップ実行しているスピードを変化させるものです。右に動かすほど高速に実行します。しかし、この場合の最高速度でも、Trace OFFのときのスピードには及びません。

5 . 開 発 環 境

今回このプログラムの作成に使用した処理系は、インプライズ社(旧ポーランド社)のC++ Builder3です。この言語は基本的にはC++ですが、Windows用のアプリケーションを作成するためのさまざまなコンポーネント(部品)があらかじめ用意されているため、画面のインターフェイスを作成するのは非常に簡単です。また、今回もそうだったのですが、以前に作成したCのコードは画面のインターフェイス部分やグラフィックス部分で手直しは必要なものの、ほぼそのまま使えますので、移植性は非常に高いといえます。また、C++コンパイラであるため、処理速度は非常に高速であり、今回のプログラムのようにシミュレートの高速度性を要求されるようなものでも、快適なアプリケーションソフトを完成させることができます。

現在多くの情報処理訓練施設で、C言語やC++をカリキュラムに取り入れていると思いますが、Windows用のアプリケーション作成実習をするのであれば、このC++ Builder 3は適当な処理系とい

えます。オブジェクト指向やイベントドリブン等の新しい概念が取り入れられていて、ときに発想の転換を余儀なくされる場面もありますが、学生にとっては、これまでの手続き型のプログラミング手法の中での経験というものもないわけですから、このことが問題になることはないのかもしれませんが。実際にプログラミングしてみると思ったほど敷居の高いものでもなく、これまでよりも短時間で見栄えのするプログラムが完成しますので、プログラム作成における学生の自信につながるかもしれません。ただ、イベントによって発生する処理については、これまでどおり手続き型のプログラム記述をすることに変わりはありませんから、入門段階においては、言語の基礎を教えることについての大きな変化はないといえます。

6 . お わ り に

21世紀を迎えようとしている現在、DOSの時代が終わって、コンピュータの環境はWindowsやMacのようなGUIの発達したOSにほぼ変わりつつあります。それに伴い開発するプログラムに要求されることは、それらのOS上で動作することです。一方で、今回のシミュレータの内容は、そのような環境からは遠い存在ともいえます。性質がだいぶ異なる両者のプログラムですが、いずれも重要な分野であると考えます。学生にはどちらもほどよく習得してもらえればいいのですが...

このプログラムは無料で配布しています。インターネット等で配信できればよいのですが、現在当校のネットワーク環境がそのようになっていませんので、お手数ですが、返信用の封筒(A4のマニュアルを入れられるもの)とフロッピーディスク2枚を同封して本校までお送りください。

〒985-0001

宮城県塩竈市新浜町2丁目3-3

宮城県立塩釜高等技術専門学校 情報処理科

Tel 022-362-3958

E-mail : mikiya@jody.svtc.ac.jp