

COMET IIシミュレータ・ソフトで動かしながら学習できる

シミュレータで 理解しながら習得する アセンブラ言語CASL II

見本

中村 文隆 著
Fumitaka Nakamura



はじめに

本書では、IPA (情報処理推進機構) が実施している基本情報処理技術者試験に出題される CASL II というアセンブラ言語と、CASL II を動作させるシステムである計算機 COMET II を取り上げます。CASL II と計算機 COMET II は、ともに仮想的な言語とシステムですが、現在広く使われているコンピュータの中核となる範囲を過不足なく抽象化しており、現実のさまざまなコンピュータ・システムの動作原理と、それらを制御する多様な高級言語の基礎となる思考形態を理解するためには最適な環境であると言えます。

わかりやすく、そしておおまかに表現するなら、COMET II はハードウェア、CASL II はソフトウェアということになります。

CASL II /COMET II は、基本情報処理技術者試験の資料である「試験で使用する情報技術に関する用語・プログラム言語など」の中の、「別紙1 アセンブラ言語の仕様」という項目ですべてが定義されています。わずか8ページですべてを見通すことのできる簡潔なシステムです。

このことは、CASL II /COMET II の機能が低いということの意味してはではありません。上でも述べたように、現在のコンピュータにおける情報処理の本質は CASL II /COMET II に含まれており、手間を度外視すれば、現実のコンピュータにできることで、CASL II /COMET II でできないことはない、といってよいでしょう。手間を除いて、と、ただし書きが付くのは、CASL II がアセンブラ言語で、コンピュータの処理の最小単位^{*1}を一つひとつ記述していく形式なので、実際にそんなことをすると、とても手間がかかることが予想されるからです。

実際のコンピュータ・プログラミングでは、C/C++ や Java、(各種の改良された) Basic、ウェブ (ホームページ) 用のマークアップ言語やスクリプト言語など、アセンブラ言語のように CPU に対する命令を直接並べなくともよい、高級言語と呼ばれる種類の言語を用いることが大半で、現在、コンピュータのプログラミングをアセンブラ言語を用いて行う機会は相対的に少なくなっています。

けれども、高級言語で記述された処理も最終的にはアセンブラ言語に翻訳され、アセンブラ言語から機械語の命令となってコンピュータで実行されることに、かわりはありません。高級言語は、できるだけ機械の都合を人間が意識しなくて済むように作られています。ベースとなるシステムの構造は随所に顔を出してきます。また、組み込み機器などで CPU の処理能力や主記憶装置の容量の制約が厳しい場合には、高級言語を経由せずに直接アセンブラ命令を記述することもあります。

つまり、情報処理技術を構成する要素の中で、もっともハードウェアに近い部分がアセンブラであり、アセンブラを具体的に理解しておくことは、高度な情報処理技術を学ぶ際にも、表面的な理解より一段深い理解につながっていきます。また、実際的な話として、基本情報処理技術者試験におけるプログラム言語の選択対象のうち、CASL II は言語仕様として最小のものであり、まったくプログラム言語の経験がなければ、ほかの言語に比べて学習量が少なく済むという利点もあります。

以上のような背景から、本書では CC22 というシミュレータを通して CASL II と計算機 COMET II について学び、コンピュータ・システムについて基本的な理解を培うことを目標とします。

^{*1} アセンブラの各命令のことを指すが、現在の CPU ではアセンブラ命令をさらに小さな処理単位であるマイクロコードに分解した上で処理の並列化を図り、高速化を行うようになってきている。ただし、マイクロコードをユーザが直接コーディングすることはほぼないと言ってよく、ユーザからみた処理の最小単位はアセンブラの各命令ととらえておいても問題はない。

第2章

アセンブラの基本

学習する前に準備しよう

● 準拠資料

本章の説明は、IPA（情報処理推進機構）が発行している「試験で使用する情報技術に関する用語・プログラム言語など」Ver 2.0 に準拠しています。この文書の URL は、

http://www.jitec.ipa.go.jp/1_13download/shiken_yougo_ver2_0.pdf

です。文書名を除いた http://www.jitec.ipa.go.jp/1_13download/ にアクセスして、文書名のアンカーをクリックすることで閲覧やダウンロードを行うことができます。

計算機 COMET II と CASL II については、上記文書の「別紙 1」（p.3 ～ p.10）に仕様が定義されています。以下では、必要に応じて上記の文書を「IPA 別紙 1」と略記することにします。

本書では、IPA 別紙 1 を参照しなくとも計算機 COMET II と CASL II が理解できるように解説を進めていきますが、初めての人にもわかりやすくするために、アセンブラなどの特有の用語を一般的に通じやすいと思われる表現に言い換えたりすることもあります。その際、IPA 別紙 1 の正式の用語のどれに対応するかを都度明記していくので、適宜 IPA 別紙 1 を参照し、正式な用語を確認しながら学んで行ってください。

● シミュレーション対象システムの構成

アセンブラのプログラミングを覚えるあたって、まずシミュレータ CC22 のシミュレーション対象である計算機 COMET II の基本的な構成と、計算機 COMET II を制御する機械語命令を生成するための CASL II の概要を理解しておく必要があります。

CC22 は、図 6 のようなシステムのシミュレータです。おおまかに右側がハードウェア、左側がソフトウェアに対応しています。

ハードウェアは、

- CPU (中央処理装置)

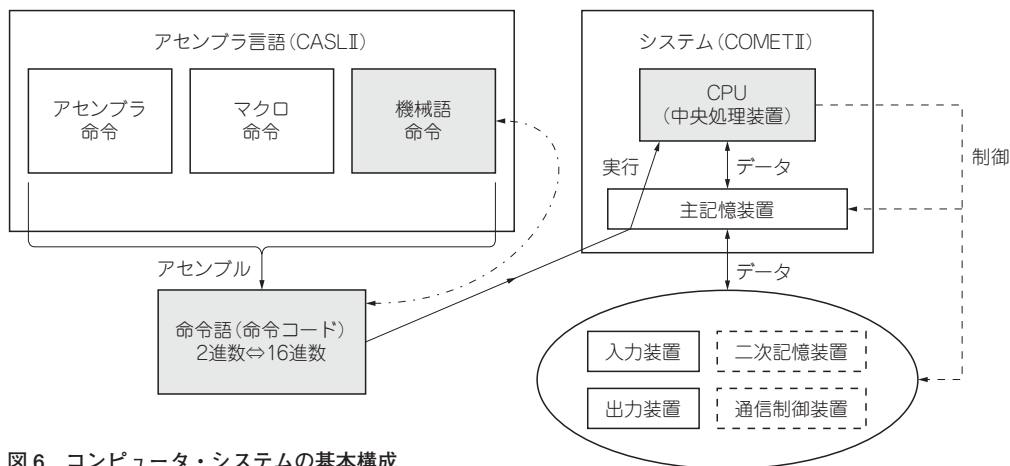


図6 コンピュータ・システムの基本構成

- 主記憶装置
- 入力装置
- 出力装置
- 二次記憶装置
- 通信制御装置

から構成されています。

計算機 COMET II では、図6の右下に点線で表記している二次記憶装置と通信制御装置は規定されていません。そこで、シミュレータ CC22 でも、これらの装置はシミュレーションの対象としていません。

システムとしての計算機 COMET II について、IPA 別紙1では、COMET II の CPU が実行できる機械語命令、主記憶装置の構成、入出力装置の取り扱いについて規定されており、CC22 はそれらに準拠した上で若干の独自拡張を施したシミュレータとして機能します。

一方、ソフトウェアは、

- アセンブラ命令
- マクロ命令
- 機械語命令

から構成されるアセンブラ言語で、このアセンブラ言語の名称が CASL II です※4。これらのうち、機械語命令は、計算機 COMET II の CPU が実行できる数値(本質的に2進数)を、人間の直観で覚えやすい

※4 IPA 別紙1の記述を読むと、OS(オペレーティング・システム)についての言及もあるが、このOSについては特に名称は指定されておらず、機能定義もごく一般的かつ抽象的な記述にとどまっている。

ような英数字に1対1で対応させた記号で表現しています。計算機 COMET II の CPU が理解して実行できるのは、基本的に機械語命令のみです。

とはいえ、プログラムのすべてを CPU に直結した機械語だけで書くことは實際上、それなりに苦勞がともないます。たとえば、プログラムの途中で条件次第で処理の流れを変更したい場合、機械語だけでプログラミングしようとするれば、処理の変更によってジャンプする個所がどこになるかを毎回手計算で求めて指定しなくてはなりません。

また、よく使われる決まりきった処理があるときも、機械語では一連の処理をすべて逐一書かねばならず、プログラムの記述が煩雑で読みにくいものとなります。

こうした事情を軽減するために、CASL II ではアセンブラ命令とマクロ命令という、アセンブラ特有の命令を用意しています。アセンブラ命令とマクロ命令は、図6の左側に示したように、最終的にはアセンブルされることによってすべて機械語に変換され、CPU が理解できる機械語プログラム^{※5}に翻訳されます。

アセンブラ命令には、

```
START, END, DS, DC
```

の四つがあり、マクロ命令には、

```
IN, OUT, RPUSH, RPOP
```

の四つがあります。これらの八つの命令は、計算機 COMET II の機械語に直接対応したものではありませんが、アセンブラでプログラムを書く際には、機械語と同列の扱いで使用できます。詳しくは後の節で説明します。

続いて、ハードウェアである COMET II の仕様をもう少し詳しく説明していきます。

CPU と主記憶装置

CPU は演算処理を行うことのできる装置で、コンピュータの中核をなすものです。CPU 内にはレジスタと呼ばれる記憶領域があります。しかし、計算機 COMET II のレジスタに記憶できるデータ量は、主記憶の記憶領域に記憶できるデータ量よりもはるかに少ないものとなります。

計算機 COMET II の仕様として、1語は16ビットで、主記憶装置の記憶容量は65,536語と規定されています。

※5 機械語命令は本質的に2進数で表され、2進数のことをbinary(バイナリ)と呼ぶため、CPU が直接理解して実行できる形のプログラムのことをバイナリ、バイナリ・ファイルと言うこともある。ただし、画像や音楽など、テキスト(文字)として解釈しても意味をなさないファイルのこともバイナリと呼ぶことがあり、機械語のプログラム・ファイルにも実行ファイルという呼び方が用いられたりする。いずれにしてもこれらの用語は慣用的なもので、あまり厳密な定義はない。

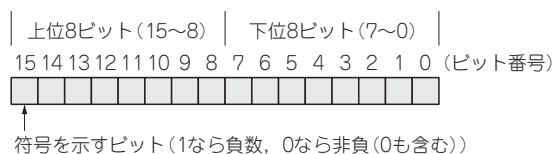


図7 COMET IIの1語のビット構成

アドレス(10進数)	アドレス(16進数)	1語=16ビット=2バイト
0番地	0000番地	
1番地	0001番地	
⋮	⋮	
65534番地	FFFE番地	
65535番地	FFFF番地	

図8 主記憶の番地

16ビットは2バイトなので、65,536語の記憶容量は、イコール131,072バイトという意味です。1Kバイトを1,000バイトとする流儀と、1,024バイトとする流儀とがありますが、1Kバイトを1,024バイトとして換算すると、COMET IIの主記憶はちょうど128Kバイトということになります。ここで、「語」というのは、一般には「ワード」と呼ばれることもあり、**COMET IIの1語=1ワードが2バイト=16ビット**であるので、**COMET IIの語長、ワード長は2バイト**であるという言い方もできます。

1語の構成は図7のように規定されています。この図の表示は2進数で、図7の1マスには2進数の0または1が入ります。2進数の桁のことをビット(bit)と呼び、1語を構成する16個のビットには右から順に0～15までのビット番号が振られています。ビット番号15はもっとも左のビットになりますが、このビットが1のときは、**算術演算**^{※6}の数値として解釈する場合は負の数(数が0より小さい、<0、ということ)、0のときは非負の数(数が正の数か、または0、つまり、 ≥ 0 、ということ)を表します。

アドレス

計算機COMET IIは、ストアプログラム方式、またはプログラム内蔵方式^{※7}と呼ばれるアーキテクチャ^{※8}で規定されています。CPUに対する命令(機械語命令)と、命令によって処理すべきデータは、どちらも数値として主記憶上に区別なく置かれます。

機械語命令には1語で表されるものと、2語で表されるものがあり、これを命令語長と呼びます。命令語長は、それぞれの機械語命令ごとにIPA別紙1で定義されています。

データとしての数値の語数は、それぞれのデータの意味によって違ってくるので、決まった語長はありません。

主記憶は65536語分の数値を格納できます。機械語命令や処理データとしての数値を参照するためには、65536語分のうちのどの語であるかを指定する必要があります。この位置指定に使われるのがアドレス(address)です。

アドレスは、図8のように0番地から65535番地(10進数での表記)まで主記憶上の各語に番号を付けて指定します。16進数でアドレスを指定するときは慣用的に4桁分を(0があっても)書くことがありま

※6 算術演算については、この後、フラグ・レジスタの項で詳しく説明するが、計算機COMET IIでは2進数の解釈が2通りあり、正負(0を含む)の数を扱う解釈が算術演算、非負(0以上の数)を扱う解釈が論理演算である。

※7 ノイマン型という呼び方をすることもある。文脈によってはノイマン型と、ストアプログラム方式、またはプログラム内蔵方式がほぼ同義に使われることもある。一部で議論の対象となることもあるので、本書ではこの呼び方は用いない。

※8 コンピュータの設計概念、構成のことをアーキテクチャという。

の事象を表すことができますが、これでは、たとえば文字としてアルファベット 26 文字を表すにも不足します。2⁸ = 256 なので、8 ビットを単位とすれば文字や数字、記号を表すにも一つのまとまった信号線で扱うことができ扱いも簡便になる、という理由はあったかもしれません。

一度 8 ビットが基本として定着してしまえば、信号線の数を増やしていく際に 2 倍ずつにすることで、既存のユニットを二つ並べるような構成で過渡期でも開発がやりやすい、などの理由から、8 ビット、16 ビット、32 ビット、64 ビットという系列ができてきている面があります。ただし、これは本質的な話ではなく、現実的な開発や運用の面での利便性、簡便性などからくる慣習的な話です。

ともあれ、実用の観点からは、図 16 の 2 進数と 16 進数の対応表を覚えてしまうのが楽でしょう。

この表は、16 通りの組み合わせしかありません。子供のころに掛け算の九九の表を暗記させられたと思いますが、九九の表は 81 通りもあるのに比べれば 1/5 以下です。九九の計算結果は暗記ではなくちゃんと計算して答えを出すことができるように、2 進数 ⇄ 16 進数の変換も、本書後半の基数変換をマスターすれば計算で答えを出すことができます。しかし、九九の暗記が計算の高速化を図るためであったのと同様に、図 16 の対照表も考えることなくノータイムで出てくるようにすることをお勧めします。そうすることで、2 進数の桁数(ビット数)が増えた場合も、右から四つずつに区切って 16 進数に変換する、あるいはその逆に、16 進数の 1 桁を 4 ビットずつに変換することで、2 進数と 16 進数の相互の変換は、計算しなくても答えが出るようになります。

例題 1

2 進数の 01011010 を 16 進数に直せ

答：4 ビットずつに区切ると、0101 1010 で、0101 = 5、1010 = A、であるから、16 進数では 5A、となる。

例題 2

16 進数の B9 を 2 進数に直せ

答：B = 1011、9 = 1001 であるから、10111001、となる。

2 進数と 10 進数、16 進数と 10 進数の変換は、本書の後半にある基礎知識で計算方法を説明します。考え方は身に付けておくべきですが、実用上は 10 進数へ、あるいは 10 進数からの変換が必要になった場合は、Windows の電卓のプログラマ電卓 (XP では関数電卓) の機能を使って変換してもかまわないでしょう。詳しい使い方は本書では説明しませんが、Windows で電卓を起動し、[表示 (V)] から [プログラマ (P)] を選ぶと図 17 のような画面になります。ここで、左下の Qword, Dword, Word, Byte, の

2進数	16進数	2進数	16進数
0000	0	1000	8
0001	1	1001	9
0010	2	1010	A
0011	3	1011	B
0100	4	1100	C
0101	5	1101	D
0110	6	1110	E
0111	7	1111	F

図 16 2進数/16進数対照表



図 17
Windows 電卓の
プログラマ表示

ラジオ・ボタンで Word を選択^{※13}すれば、上から2段目の2進数の表示が16ビットの2進数の表示になります。この状態で10進や16進を選択して数を入力すればリアルタイムで2進数での表示が更新されます。16進や2進を選んで数を入力した後、=をクリックし、その後10進に切り替えれば10進数への変換も簡単に行うことができます。

ただし、Windows 電卓のプログラマ表示では、基礎知識の章で説明している小数点のある2進数や、浮動小数点などの計算はできません。整数に限れば、計算機 COMET II で採用している2の補数表現による負数(後の項で説明します)は扱うことができます。いずれにしても、ある程度プログラマ電卓や CASL II のプログラムで慣れた後は、基礎知識の章を参照して、正しく理解しておきましょう。

練習問題 2 (解答は p.178)

次の2進数を16進数に直せ。

- (1) 0101 (2) 0110 (3) 1100 (4) 1111 (5) 1011 1010 (6) 1111 1110 (7) 0010 1001

練習問題 3 (解答は p.178)

次の16進数を2進数に直せ。(1)~(3)は4ビットで、(4),(5)は8ビットで、(6),(7)は16ビットで答えること。8ビット、16ビットの場合、4ビットごとに空白で分かち書きをしてもよい(しなくともかまわない)。

- (1) 8 (2) 1 (3) 2 (4) FF (5) 15 (6) 3A5B (7) FFFF

※13 Word の選択にすると、後述する算術演算の解釈となって15番目のビットが1のとき、10進数解釈では負の数として扱われる。論理演算として、負の数の解釈を行わずに Windows 電卓を利用したいときは Dword の選択をして32ビットの2進数のうち下位16ビットを用いるようにすればよい。本書の説明を、Windows 電卓を用いて確認してみたい場合はこの点に注意されたい。

● 計算機 COMET II

● GR0 ~ GR7, SP, PR, FR, OF, SF, ZF, 注釈

計算機 COMET II の CPU は以下のようなレジスタ (CPU 内の小さな記憶領域) を持っています。以下では、具体的なアセンブラの命令を説明する前に、レジスタについて一通りの説明をしていきます。各命令での具体的な使い方をまだ説明していないので、イメージがつかみにくいところもあると思いますが、そういうものがあるのだ、という程度で読んでおき、先に進んでアセンブラ命令の説明が出てきたときに立ち戻って読み直して理解を深めてください。

シミュレータ CC22 では、メイン・ウィンドウの右上にある **図 18** の領域にレジスタ内容を表示しています。ここには GR0 から GR7 までの 8 個のレジスタが左側に、SP と PR, FR の三つが右側に並んでおり、FR の中は OF, SF, ZF の三つが表示されています。以下では、これらのレジスタの役割について説明していきます。

GR0 ~ GR7 (汎用レジスタ)

GR0 ~ GR7 までは、汎用レジスタ (General Register) と呼ばれ、算術、論理、比較、シフトなどの演算に用いられます。GR0, GR1 …… , GR7 と GR に 0 ~ 7 までの番号が付き合計 8 個あります。

計算機 COMET II では、基本的なルールとして、

計算やデータの移動はすべて汎用レジスタ GR0 ~ GR7 を使う

ということになっています。たとえば、ある数ともう一つの数を足す、という計算をしたいときは、まず主記憶から汎用レジスタのどれかに一つ目の数を移し、もう一つの数を汎用レジスタに足した後、主記憶のどこかに足した結果を入れておく、という流れで処理をします。

主記憶に保管されている二つの数字を直接足すということはできず、まず汎用レジスタにデータを移してから足します。

図 19 に、 $2 + 3$ の計算をする場合の処理を示しました。主記憶上に 2 と 3 が入っています。その下の“?” は、計算した結果 (ここでは $2 + 3 = 5$ になるはず) を格納するための記憶場所ですが、計算する前は何が入っているかわからないため“?”としてあります。

ちょっと考えると、主記憶に既に足すべき数が入っているのだから、**図 20** のように主記憶上の 2 と 3 を直接足してしまえばよいのではないか、と思えるかもしれませんが、しかし、計算機 COMET II の機械語命令には、主記憶上の数値同士を直接足すという命令が存在しないため、それはできません。この事情は、コンピュータの回路構成上、演算装置を利用するためには汎用レジスタを経由しなければならない

レジスタ 初期化			
GR0	0000	SP	FFBF
GR1	0000	PR	FFFF
GR2	0000	FR	OF:0
GR3	0000		SF:0
GR4	0000		ZF:0
GR5	0000		
GR6	0000		
GR7	0000		

図 18 CC22のレジスタ表示

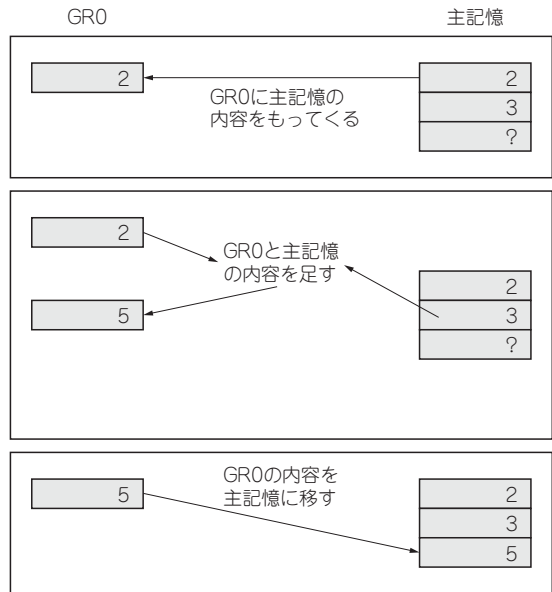


図 19 汎用レジスタを用いた計算の概念

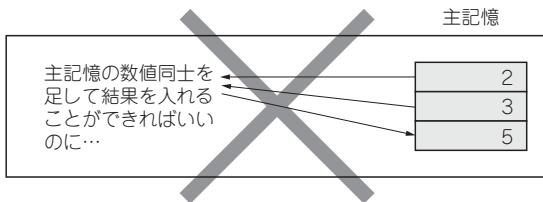


図 20 主記憶同士で演算を行うことはできない

という制約から生じています。

このため、足し算をするためには、

- まず汎用レジスタ (GR0) に、一つ目の数 (2) をもってくる
- 汎用レジスタと、もう一つの数 (3) を足すと汎用レジスタに答えが入る
- 汎用レジスタの内容を結果を保存する場所に移す

という三つの命令を続けて実行しなければなりません。

計算機 COMET II には、さまざまな機械語の命令が定義されています。機械語命令には、命令の操作の対象に何を指定ができるかということが決まっています。命令の操作の対象のことをオペランド (operand) と呼び、先ほどの例では、足し算の命令のオペランド (操作対象) は、汎用レジスタを含んでいなければならないという言い方をすることができます。

計算機 COMET II では、8 個ある汎用レジスタの用途は、1 点を除いて制限がありません。ただ 1 点、

GR0 は指標レジスタとしては使用できない

第 3 章

応用編

本章では、CASL II の命令を使った応用的なプログラムを見ていきます。

シミュレータ CC22 には、LCD 表示装置、描画装置、ボタン入力装置の三つの拡張入出力装置があります。これらは、マイクロ・コンピュータのシステムでよく使われる周辺装置の代表例として用意されており、ある程度抽象化した形で実際の周辺機器の制御の流れを体験することができます。

LCD 表示装置

● LCD 表示装置の概要

仕様

シミュレータ CC22 の LCD 表示装置は、図 130 のように 16 文字 × 2 行の表示領域と、内部に 80 文字分の文字バッファを持っています。

LCD 表示装置に文字を表示するには、

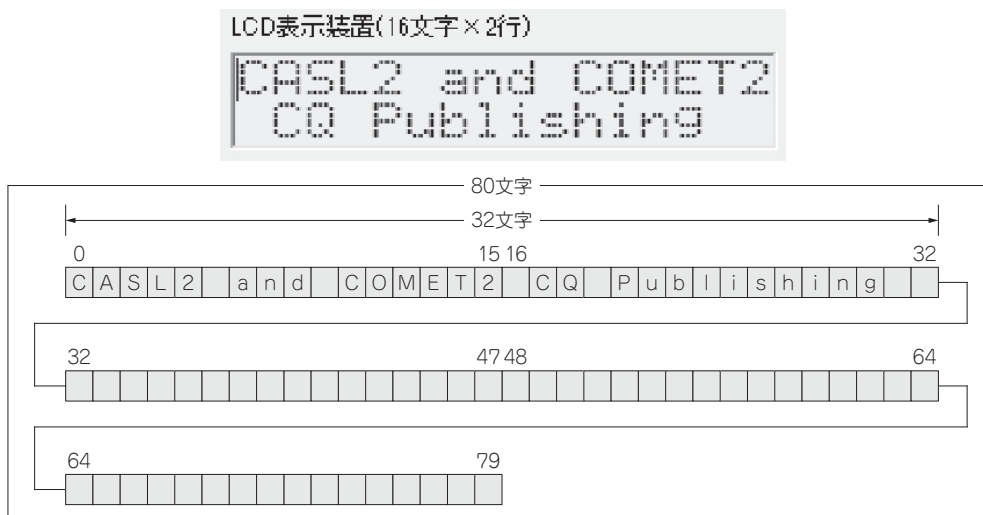


図 130 LCD 表示装置と内部バッファ

見本

ISBN978-4-7898-1630-4

C3055 ¥2400E

CQ出版社

定価：本体2,400円（税別）



このPDFは、CQ出版社発売の「シミュレータで理解しながら習得するアセンブラ言語CASL II」の一部見本です。

内容・購入方法などにつきましては以下のホームページをご覧ください。

内容 <http://shop.cqpub.co.jp/hanbai/books/16/16301.htm>

購入方法 <http://www.cqpub.co.jp/hanbai/order/order.htm>