

# BASIC、ときどき機械語

## BASIC, sometimes BINARY

田中二郎@日本橋学館大学

jiro@nihonbashi.ac.jp \*

### 概要

PaloAlto の TINY BASIC が出て、はや30年。NEC の PC-8001 も、そろそろ30年になろうとしています。カードや紙テープから脱却し、キーボードと画面でインタラクティブにプログラムするようになったのは、この頃のパソコンのおかげではないでしょうか？

多くの人が、いまだに BASIC で書け…といわれると、何かしらのものを書くことができるのではないのでしょうか。それだけ、BASIC は私たちの心に深く影響を与えてしまったのです。

一方、TINY BASIC を移植するということは、機械語の勉強にもなりました。少ないメモリという環境が、我々の創造性をふるいたたせてくれたのです。おかげで、今でも16進でプログラムを読み書きできます。

今回は、Z80 を使ったハードウェアに TINY BASIC を載せてみました。

## 1 歴史

ンジスタやダイオードを使ったデジタル回路に興味を持ちました。

### 1.1 最初の一步

その頃、NHK で HITAC-10 を使ったコンピュータ講座をやっていました。微分方程式を解いていたのを覚えています。テキストも買って、キーボードの配列や紙テープの穴を覚えたりもしました。

物心つく前から、祭りの夜店で「ドライバーセット買って」と親にねだったらしい私ですが、自作の配線をコンセントにつっこんではヒューズをとばしていたことを覚えています。小学生の頃は、真空管 (12AU7 など) やトランジスタ (2SB56、2SC372 など) を使って、ラジオを作ったりしていました。

そうこうするうちに TTL というものが出て、IC を使ったデジタル回路を作り始めます。ALU (74LS181 など) を使って、計算機らしいものを作りました。

また、電子計算機がトランジスタから IC を使ったものへの移行時期だったので、ジャンクの基板が出回り、トラ

1971 年頃のトランジスタ技術には、4004 の紹介や、TTL で作るコンピュータの記事があったことを覚えています。

---

\*Jiro TANAKA, NIHONBASHI GAKKAN UNIV., <http://gakkan.net/jiro/>

## 1.2 マイコン

中高のあいだ、オーディオやアマチュア無線にむいていた私の興味も、マイコンの登場で、またデジタル回路に戻ります。SSTV（約7秒／フレームの画像通信）を実現するのにデジタル回路が便利だったのも一因です。

大学に入ると同時に、大阪のパソコンショップに出入りするようになり、NECのPDA-80という、TK-80用の開発マシンを使うようになりました。

最初に作ったのは逆アセンブラです。これでPDA-80のモニタやアセンブラなどの開発環境を逆アセンブルし、機器構成をリバース・エンジニアリングしたのです。

同時に、自分でもマイコンを作ってみました。できれば、当時大流行だったSTAR TREKゲームを走らせてみたかったのですが、予算不足のため、メモリは256バイト、I/Oもつなげることはできませんでした。

小さいマシンなので開発環境もなく、ハンドアSEMBルするうちに機械語を覚えてしまいました。その結果、今でもZ80の機械語とクロック数は覚えていますが、紙テープをそのまま読めます。

## 1.3 パソコン

1979年になって、PC-8001が発表されました。それ以前の試作機段階からさわる機会があったのですが、スイッチを入れるといきなりBASICが起動してしまう。そのため、得意の逆アセンブルによるリバース・エンジニアリングができませんでした。

そこで、えいやっ作ったのが、BASICで書いた逆アセンブラです。これが以外に好評で、どこで聞きつけたのか、ASCII誌の方から「PC-8001用のソフトが無いので、記事にしてみない？」といわれ、機能を追加したものが1979年12月号に掲載されました。

逆アセンブラをつかって解析したら、次のステップは改造です。PC-8001のDISK-BASICのフロッピーまわりの仕様が腐っていたので、まずはそれを改良しようと、やはりBASICでエディタとアセンブラを書きました。

逆アセンブラによる解析は結構好きで、後続のパソコンたち（PC-6001、PC-8801など）の解析記事も、雑誌掲載されました。解析は、16bitマシン・PC-9801になっても続けていました。

## 1.4 インタプリタ

1982年になって、BASICインタプリタ作成の仕事をするようになりました。解析する側から、される側になってしまったわけです。要求されたのは速度と、ゲームをつくるための色々な機能でした。

しかし、部品数を減らすためにクロックが3.58MHzになり、これで1割のスピードダウン。さらにBIOS開発側からメモリ領域を要求され、そのたびにインタプリタを書き換えて対応することとなりました。バイト数を減らしながらも、クロック数になるべく増えないように数百ステップを書き換えることもありました。おかげで、一部マニアには「高速で使いやすい変態BASIC」という評価を受ける製品となりました。

## 2 再現

### 2.1 ハードウェア

秋月電子の AKI-80(12M)<sup>1</sup>は、最新 Z80 複合 CPU チップ・Z84C015-12 (東芝) を使った、クロック 12MHz で動作するマイコンボードです。CPU、PIO、CTC、SIO、CGC の機能をもち、32MB の ROM、32MB の RAM を載せています。

今回は、このマイコンボードの SIO に、シリアル-USB 変換の IC を接続し、パソコンを接続できるようにしました。パソコン側では端末ソフトを起動し、文字の入出力をおこないます。

電源は、パソコンの USB からの供給でまかないます。また、メモリのバックアップにはリチウム電池を使用します。

その他に、動作確認&趣味として、アドレス、データ、制御信号、ステータスなどのラインに LED をつないで、動作にしたがってチラチラと明滅するようにしました。

### 2.2 開発環境

ソフトウェアの開発はパソコン側でおこないました。使用したのは、村上敬司さん作の CP/M エミュレータ<sup>2</sup>と Digital Research 社のマクロアセンブラ<sup>3</sup>です。できたバイナリ (インテル HEX 形式) はエディタで整形・リンクして、ROM ライタ<sup>4</sup>で 27C256<sup>5</sup>に焼きます。

<sup>1</sup><http://akizukidenshi.com/catalog/items2.php?q=K-00122&p=1>

<sup>2</sup><http://www.vector.co.jp/soft/dl/win95/util/se378130.html>

<sup>3</sup><http://www.cpm.z80.de/download/mac-b.zip>

<sup>4</sup><http://akizukidenshi.com/catalog/items2.php?q=M-00513&p=1>

<sup>5</sup><http://akizukidenshi.com/catalog/items2.php?q=I-01486&p=1>

<sup>6</sup><http://www.cpm.z80.de/roche/PCWPATB.ASM>

<sup>7</sup><http://www.dunnington.u-net.com/public/startrek/startrek.asc>

### 2.3 ソフトウェア

ROM は 0 番地から、RAM は 8000 番地からになります。そのため、RST 命令用のジャンプテーブルを RAM 領域に用意し、将来の拡張に備えています。

モニタは、メモリ内容の表示や変更、レジスタの表示、I/O のほか、インテル HEX 形式の入出力もできます。もちろん、逆アセンブラも備えています。サイズが 3KB 余となってしまいましたが、その半分以上が逆アセンブラです。

TINY BASIC インタプリタは、Palo Alto Tiny BASIC(v3)<sup>6</sup>に最低限の変更を加えたもので、サイズは 2KB 余です。変更箇所は、入出力の部分と、メモリ配置の都合によるポインタの扱い、モニタ呼び出しなどの機能追加です。

STAR TREK ゲーム<sup>7</sup>は、その格納イメージを ROM に焼き、コマンドにより BASIC のプログラムエリアに転送するようにしました。サイズは 5.5KB 余です。

0000-00F8	初期化ルーチン
0100-0CC7	モニタ
0FD1-17FF	インタプリタ
1800-2E0F	STAR TREK イメージ
8000-801A	RST 命令用ジャンプテーブル
8020-80FF	モニタ作業域
8800-8FFF	TINY BASIC 作業域
9000-FEFF	TINY BASIC プログラム

### 3 質疑応答

Q：若い人が得られることは（川中）

A：STAR TREK ゲームを楽しんでください。今のグラフィック主体のゲームにはない面白みが、キャラクタ・インターフェースにはあります。また、短い（自分で理解可能）なプログラムを読むのも、楽しいでしょう。

Q：STAR TREK のルールを教えてください。（富野）

A：現在NHK-BSで再放送されているTVシリーズ「宇宙大作戦」が元になっています。宇宙は8×8の領域からできていて、そのひとつひとつが8×8のマスになっています。

ENTERPRISE IN Q-32 S-45

LONG RANGE SENSOR

```

2   3   1
1 214  2
2   4   6
    
```

領域32にいて、周囲を見てみました。現在いる領域には、敵が2、基地が1、星が4個あります。

COMPUTER DISPLAY OF GALAXY MAP

```

1: 0  0  0  0  0  0  0  0
2: 2  3  1 303 7  1  4  0
3: 1 214 2  6  1  6  6  0
4: 2  4  6  5  6  3  6  0
5: 0  0  0  0  0  0  0  0
6: 0  0  0  0  0  0  0  0
7: 0  0  0  0  0  0  0  0
8: 0  0  0  0  0  0  0  0
.. .. .. .. .. .. .. ..
  1  2  3  4  5  6  7  8
    
```

記録してある、銀河地図です。未踏査領域は、0表示になっています。

SHORT RANGE SENSOR

```

1 . . . . .
2 . * . . . K . .
3 . . . . B . . .
4 . . . . E . . .
5 . . . . . * . .
6 . . . . . K . *
7 . . . . . . . .
8 . . . * . . . .
  1 2 3 4 5 6 7 8
    
```

敵 (K:クリンゴン) を探し、魚雷 (発射方向を指定) やフェーザー (エネルギーを指定) でやっつけます。また、宇宙のどこかにある基地 (B) を探して、そこで補給をうけることができます。移動や魚雷の方向に、整数演算ゆえの演算誤差などがあるので、プログラムを読まないで勝ちにくいです。

Q：空間は2次元？(大駒)

A：はい。

Q：コマンド・モードで「.」（ピリオド）だけ入力すると？

A：アプリケーションで「LIST」になります。

Q：同様に「NEXT」も「.」だけでいいはずだが、何故そうならない？

A：ゲームのプログラムを書いた人に聞いてください。

