

情報工学実験 2
コンピュータアーキテクチャと命令セット

学籍番号 045718D : 翁長絵美

実地日 : 平成 17 年 10 月 17 日
提出日 : 平成 17 年 10 月 24 日

共同実験者
045751F : 又吉 美奈

1 実験目的

- (1) 教育用ワンボードコンピュータ KUE-CHIP2 を用いて例題プログラムを実行することにより、KUE-CHIP2 の操作方法を習得する。
- (2) また、簡単な機械語 (マシン語) プログラムを作成し、機械語プログラムの構造を理解することを目的とする。

2 実験概要

上記の実験目標である (1) を達成するためにまず「KUE-CHIP2 教育用ボード・リファレンスマニュアル (抜粋)」をよく読み、例題プログラムや各操作手順を調べ、実際に自らの手で入力・実行を確認し操作方法の取得を目指した。また、(2) を達成するために学籍番号の 6 個の数を足してその合計を求めるプログラムを作成し、実際に入力・実行し機械語プログラムの構造を理解するようにする。

3 実験結果

- ### 3.1 「KUE-CHIP2 教育用ボード・リファレンスマニュアル (抜粋)」第 1 章、第 2 章をよく読み、KUE-CHIP2 の各表示の意味、スイッチ類の操作方法を調べよ。

本実験の結果に関しては、報告を省略する。

- ### 3.2 「KUE-CHIP2 教育用ボード・リファレンスマニュアル (抜粋)」第 3 章 3.2(P.18) および 3.3(P.24) の内容に従って、例題プログラムを入力し、実行せよ。

本実験の結果に関しては、報告を省略する。

- ### 3.3 「KUE-CHIP2 教育用ボード・リファレンスマニュアル (抜粋)」第 3 章 3.1(P.17) の例題プログラムを理解し、C 言語によるプログラムに書き換えよ。

例題のプログラム (表 1) を KUE-CHIP2 に実際に入力し、実行すると結果は想定通り 20 という値を示した。

このプログラムは、実行する前に ACC に 5 という値、IX に 4 という値を設定する。そして実行すると、

ADRS 00,01 で ACC のデータを 03 番地に格納
02 で ACC を初期状態に戻す
03,04 で ACC と 03 番地のデータを足した値を ACC に格納
05,06 で IX-1 の値を IX に格納
07,08 で IX-1 が 0 でなかったら 03 に戻り、0 ならばループから抜ける
09 で停止

という処理がなされる。(表 1)

図 1 に実行したときのフローチャートを示す。

フローチャートから、このプログラムが実際には $5*4$ ではなく、5 を 4 回足す事を繰り返しており、IX が C 言語などでいう for 文の継続条件式の様な働きをしていることがわかる。

例題のアセンブラプログラムに基づいて、同様の動作をする C 言語プログラムを作った。以下がそのソースである。

```
#include<stdio.h>
main()
{
  int n,i,l,o;
  n=5;
  i=4;
  o=0;
  for(l=i; l>0; l--){
    o=o+n;
  }
  printf("5*4=%d\n",o);
}
```

実行結果は $5*4=20$ となった。

表 1:例題のアセンブラプログラム

ADRS	DATA	OPECODE
00	75	ST ACC,(03H)
01	03	
02	C0	EOR ACC,ACC
03	B5	ADD ACC,(03H)
04	03	
05	A4	SUB IX,1
06	01	
07	31	BNZ 03H
08	03	
09	0F	HLT

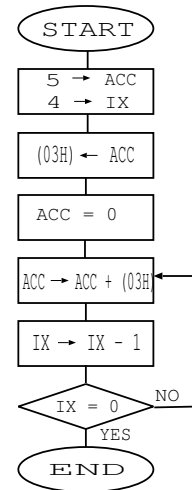


図 1: 例題プログラムのフローチャート

3.4 以下の各項目について、その操作方法 (操作手順) を調べ、実際に KUE-CHIP2 を操作し、動作を確認せよ。

(a) ACC の内容を見る方法 (手順)

SEL 3~0(4 ビットグル SW)(以下 SEL) を SEL=4 つまり SEL の 2 のスイッチを "0" から "1" にする。その結果 KUE-CHIP2 の DATA の 7seg LED,LED にそれぞれ 16 進数と 2 進数で ACC の値が表示される。

(b) IX の内容を見る方法 (手順)

(a) と同様に SEL を SEL=5 つまり SEL の 2 と 0 のスイッチを "0" から "1" にする。その結果、KUE-CHIP2 の DATA の 7seg LED,LED にそれぞれ 16 進数と 2 進数で IX の値が表示される。

(c) PC の内容を見る方法 (手順)

SEL を SEL=2 つまり SEL の 1 のスイッチを "0" から "1" にする。その結果、DATA の 7seg LED,LED にそれぞれ 16 進数と 2 進数で PC の値が表示される。

(d) ACC に 0x10 をセットする方法 (手順)

SEL を SEL=4(ACC) に設定し、DATA に DATA=0x10 つまり $16_{(10)}$ となるように入力し、SET スイッチを押す。その結果、DATA の 7seg LED に 10、LED は 4 番が光る。

(e) IX に 0x10 をセットする方法 (手順)

SEL を SEL=5(IX) に設定し、(d) と同様に DATA=0x10 を入力し、SET スイッチを押す。結果は (d) と同じである。

(f) PC に 0x10 をセットする方法 (手順)

SEL を SEL=2 に設定し、(d) と同様に DATA=0x10 を入力し、SET スイッチを押す。結果は (d) と同じである。

(g) 0x80 番地の内容を見る方法 (手順)

ADDRESS 8~0(9 ビットディップ SW) に 2 進数で $128_{(10)}$ となるように入力し、IMC スイッチを 'CHECK' にする。結果、ADDRESS の 7seg LED と LED に番地が表示される。DATA の 7seg LED と LED にはまだ何も入力してないので FF と表示された。

(h) 0x80 番地に 0x34 をセットする方法 (手順)

(g) の操作をした後に DATA に 0x34 に相当する 2 進数を入力し、SET スイッチを押す。結果、DATA の 7seg LED、LED にそれぞれ 16 進数、2 進数で 128_{10} が表示された。

(i) 0x00 番地から 0x0F 番地までの内容を連続的に見る方法 (手順)

ADRINC スイッチを押す。その結果、1 回押すごとに 0x00 番地、0x01 番地 … と連続で見ることが出来た。

(j) 0x90 番地から 0x9F 番地までの内容を連続的に見る方法 (手順)

ADDRESS 8~0(9 ビットディップ SW) に 0x90 番地を 2 進数で表し、IMC スイッチを CHECK にする。ADDRESS 8~0 を 0x91 番地、0x92 番地と設定を変えていけば連続に内容を見ることが出来た。

(k) 0x00 番地から 0x0F 番地までの内容を全て 0x11 にセットする方法 (手順)

0x00 番地に行き、DATA 7~0 を '00010001' に設定し SET スイッチを押したあと ADRINC スイッチを押す。すると 0x01 番地に移るの

た SET スイッチを押して ADRINC スイッチを押す。これを 0x0F 番地まで繰り返す。

- (l) 0x90 番地から 0x9F 番地までの内容を全て 0x11 にセットする方法 (手順)
(j) の操作の間に、(k) と同じように DATA 7~0 を '00010001' に設定し SET スイッチを押し、その後に ADDRESS 8~0 の値を変えていく。

- (m) プログラムを 0x00 番地から実行する方法 (手順)
0x00 番地に移動し、ACC や IX に必要な入力をした後 SS スイッチを押す。その結果 OP(LED) が光り、P0~P4 がクロック毎で動いて光った。

- (n) プログラムを 0x20 番地から実行する方法 (手順)
(j) の要領で 0x20 番地に移動し、(m) と同じ操作をする。

- (o) プログラムを 1 命令ずつ実行する方法 (手順)
SI(プッシュSW) スイッチを押す。結果、ランプが P0 から P1 に移動した。このことからプログラムが 1 命令だけ実行されたことがわかる。

- (p) プログラムを 1 フェーズずつ実行する方法 (手順)
SP(プッシュSW) スイッチを押す。その結果 P0~P4 が一通り光って停止した。このことから 1 フェーズだけ実行し、停止したことがわかる。

3.5 学籍番号の 6 個の数を足してその合計を求めるプログラムを各自で作成し、実行しなさい。ただし、学籍番号の 6 個の数は、データ領域の 0x00 番地 (0x100 番地) から 0x05 番地 (0x105 番地) に予め格納しておくこと。また、このプログラムを解析し、C 言語によるプログラムに書き換えなさい。

実験 (3)、「KUE-CHIP2 教育用ボードの概要 (抜粋)」を参考に表 2 に示してあるアセンブラプログラムを作成した。結果は $0+4+5+7+1+8=25$ になった。

このプログラムは図 2 で示したフローチャートでわかるように、まず初めに学籍番号の 6 個の数をデータ領域の 0x00 番地から 0x05 番地まで格納し、IX に 5,d に 0 を入れておく。そして実行すると、

ADRS 00 で ACC を初期化
 01,02 で ACC 足す IX+d 番地の値を ACC に格納
 03,04 で IX-1 を IX に格納
 05,06 で IX が 0 でなければ 01 に戻り、0 ならばループから抜ける
 07 で停止
 という処理がなされる。

もうすでにデータ領域に値を格納しているの、そこから値を引っ張り出すために、ADRS 01 の命令を B7(ADD ACC,(IX+d)) に設定し、データ領域の値を足すようにした。また、ADRS 03 で AA(SUB IX,01H) とすることで、前回の 1 個前のアドレスに移ることができる。最初に IX に 5 の値を入れておくことがミソである。

表 2:アセンブラプログラム

ADRS	DATA	OPECODE
00	C0	EOR ACC,ACC
01	B7	ADD ACC,(IX+d)
02	00	
03	AA	SUB IX,01H
04	01	
05	31	BNZ,01H
06	01	
07	0F	HLT

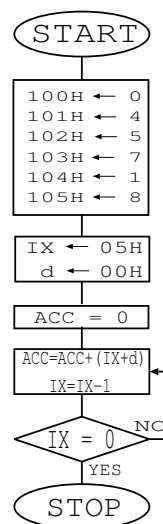


図 2: プログラムのフローチャート

このことを含めて、同じ動作をする C 言語プログラムを作った。そのソースを以下に示す。結果はもちろん 25 であった。

```

#include<stdio.h>
main()
{ int m,i,ix;
  int n[6] = {0,4,5,7,1,8};
  m=0;
  ix = 5;
  for(i = ix; i >= 0; i--){
    m = m + n[i];
  }
}

```

```
}  
printf("045718 を全て足すと%d\n",m);  
}
```

この C 言語のプログラムは、学籍番号のデータを格納するために配列を使い、変数 ix に 5 を入れアセンブラプログラムと同じ働きにするようにした。

4 考察

4.1 実験 (3),(5) について

アセンブラプログラムと C 言語プログラムの違いは柔軟性である。C 言語は変数や処理方法が豊富に宣言できるのに比べ、アセンブラプログラムは処理方法が限られている。そのため、アセンブラプログラムでも同じような処理を行うことはできるが、複雑な処理などは C 言語などの高水準な言語の方がより作りやすく、人間にとってわかりやすくなる。

4.2 実験 (4) について

実験 (4) の実験結果で書いた方法と別の操作方法があるか検討する。

(g) の 0x80 番地の内容を見る方法について考えてみる。

実験結果では、ADDRESS 8~0 を使ったが、0x80 になるまで ADRINC スイッチを押すという方法もある。しかし、これでは、ADRINC スイッチを 128 回押さないといけないことになり、効率的に考えると実験結果の手順法を用いた方がよいと思われる。

4.3 本実験を通しての考察

PC(プログラムカウンタ) : 次にどのアドレスの命令を読み込むかを表示するレジスタの 1 つ。メモリから命令を読み込むたび、値が 1 ずつ増えていく。また、ジャンプ命令が実行されたときは、そのジャンプ先のアドレス値のあたいが表示され、命令を読み込むとまたそこから値が 1 ずつ増えていく。

5 調査課題

下記の各項目について調査し、その結果を報告せよ。

(a) コンピュータの主要構成要素に関して、以下の設問に答えよ。

a-1 調査対象となるコンピュータを、自分のコンピュータあるいは自宅や大学にあるコンピュータの中から1台を選択し、そのメーカー名、商品名、型番などを示せ。

メーカー名：APPLE

商品名：iBook G4

型番：A1054

a-2 入力装置について、以下の設問に答えよ。

i. 入力装置の役割

コンピュータにデータを入力する。

ii. 入力装置の具体例を3つ以上挙げ、それぞれの特徴を説明せよ。

スキャナ：画像をコンピュータに入力する。

キーボード：文字や数字を入力する。

マイク：音声をコンピュータに入力する。

iii. 上記 a-1 で選択したコンピュータに内蔵あるいは外付けされている入力装置を全て列挙せよ。

キーボード、マウス、マイク

a-3 記憶装置について、以下の設問に答えよ。

i. 記憶装置の役割

データを記憶する装置。「主記憶装置」と「補助記憶装置」に分けられる。

ii. 補助記憶装置の例を5つ以上挙げ、それぞれの特徴や違いなどを説明せよ。

フロッピー・ディスク：磁気によってデータを記憶する。

CD-ROM：読み出し専用の記憶装置

MO：書き換え可能な記憶装置

フラッシュメモリ：RAMとROMの要素を兼ね備えたメモリ

DVD-RAM：容量が大きい記憶装置

iii. 上記 a-1 で選択したコンピュータに内蔵あるいは外付けされている記憶装置をすべて列挙し、それぞれの容量調べて報告せよ。

メモリ (DDR DSRAM) : 512MB

ハードディスク : 40GB

a-4 出力装置について、以下の設問に答えよ。

i. 出力装置の役割

コンピュータで処理したデータを出力する装置。

ii. 出力装置の例を3つ以上挙げ、それぞれの特徴を説明せよ。

ディスプレイ : パソコンの画面にデータを出力する。

スピーカー : 音声に出力する装置。

プリンタ : 用紙にデータを出力する装置。

iii. 上記 a-1 で選択したコンピュータに内蔵あるいは外付けされている出力装置をすべて列挙せよ。

ディスプレイ, スピーカー

a-5 プロセッサ (CPU, = 演算装置 + 制御装置) について以下の設問に答えよ。

i. 演算装置の役割を簡単に説明せよ。

四則計算や論理計算など、算術的な処理を行う装置。

ii. 制御装置の役割を簡単に説明せよ。

コンピュータの他の装置 (入力、出力、記憶、演算) の制御を行う装置。

iii. 上記 a-1 で選択したコンピュータに搭載されているプロセッサ (CPU) の性能や特徴について報告せよ。

powerPC G4 : 1.33GHz

特徴 : SIMD 型の並列演算機能である Velocity Engine 機能を搭載している。

(b) コンピュータには、プロセッサ (CPU) から近い順に、レジスタ、キャッシュ、メインメモリ、ハードディスクなどの記憶装置が配置されている。これらの各記憶装置の役割を説明せよ。また、このように多種多様な記

憶装置が用いられている理由について、コストや記憶容量などの観点から考察しなさい。

プロセッサ：ある命令が入力されたときに、その指示にしたがって処理を実行する装置

レジスタ：演算や実行状態を保持する

キャッシュ：使用頻度のたかいデータを保持する

メインメモリ：データやプログラムを記憶する

ハードディスク：データを記憶しておく装置

6 感想

設問が多くて疲れた。今回 KUE-CHIP2 を使って PC の意味を理解できたのが良かった。また、今回の実験は先生が細かく指導してくれ、TA の先輩方もいろいろ教えてくれたのでとてもわかりやすかった。

本実験での改善案は、しいていえば KUE-CHIP2 シミュレータの使い方をざっとして欲しかった。実験が終わった後、自分でやろうとして説明も呼んだけど、あまり意味がわからなかったので、ちょっと教えて欲しいなと思った。

参考文献

- [1] フローチャートの書き方
定平 誠 + 兼平 敦, 基本情報技術者 合格教本, 技術評論社
- [2] 主要構成要素に関して
<http://www.atmarkit.co.jp/icd/root/86/31958986.html>
<http://www.apple.com/jp/ibook/specs.html>
<http://www.asahi-net.or.jp/~ax2s-kmtn/hardware/storage.html>
<http://www.sfc.keio.ac.jp/mchtml/cns-guide/2001/2/1/2.html>
- [3] C 言語を作る際
<http://www9.plala.or.jp/sgwr-t/index.html>