応用計算機設計　課題(1)(2)(3)

4J04　岩崎悠紀

課題について

　各課題のフローチャートとコーディングシートについては，末尾へまとめることとする。フローチャートには図の題名として，コーディングシートには欄外に対応課題について記述している。

課題(1)

[課題 A]
<目的>
　指定レジスタに，1Byte 定数を入力する。
<動作確認>
　レジスタの内容を表示し，正しいことを確認する。
<結果>
　レジスタの中身に，プログラムで指定した内容が入っていたので，正しく動作したと言える。
<考察>
　LD という命令が，レジスタに値を入れるものだということが体感的に理解できた。
<感想>
　最初の課題だったので，課題をやるまでの操作に時間がかかってしまった。

[課題 B]
<目的>
　指定レジスタに，2Byte の定数を入力する。
<動作確認>
　レジスタの内容(B と C それぞれ)を表示し，正しいことを確認し記録する。
<結果>
　レジスタの内容が，プログラムで指定した 2Byte の定数になっていたので，正しく動作していると言える。
<考察>
　全課題では 1Byte の定数をレジスタに入れたが，二つのレジスタに 2Byte の定数を入力できることが理解できた。
<感想>
　定数をオペコードにするときに，1Byte ずつで逆の順番になるのが興味深かった。

[課題 C]
<目的>
　レジスタ間でデータを転送する。
<動作確認>
　レジスタの内容（それぞれ）を表示し，正しいことを確認し記録する。
<結果>
　はじめにレジスタ A に入れていた CDH という定数がレジスタ B，H にも入っていたので，正しく動作していると言える。
<考察>
　これまでは，レジスタにそのまま定数を入れるだけだったが，今回の課題によってレジスタどうして値を転送できることが理解できた。
<感想>
　レジスタ同士でのデータ転送の方法を覚えたので，できることの幅が広がったと思った。

[課題 E]
<目的>
　指定メモリアドレスとレジスタの間でデータを転送する。
<事前設定>
　8100H に，例として 45H を設定しておくこと。
<動作確認>
　レジスタの内容を表示し，正しいことを確認し記録する。
<結果>
　レジスタ A に 8100H にあった 45H という値が入っていたので，正しく動作してと言える。
<考察>
　今回の課題では，レジスタだけではなくメモリからもレジスタに値を入れることができるということを学んだ。
<感想>
　アドレスの表現で()をつけるということを学んだ。

[課題 F]
<目的>
　指定メモリアドレスとレジスタの間でデータを転送する。間接アドレスはレジスタを使う。

<事前設定>
8200H に，例として 43H を設定しておくこと。
<動作確認>
１ステップ毎にレジスタ内容を表示し，正しいことを確認し記録する。
<結果>
レジスタ HL に 8200H という定数が入っており，レジスタ B には 8200H の番地のデータである 43H が入っていたので，正しく動作していると言える。
<考察>
今回の課題で，間接アドレスは定数として決め打ちで書かなければならないものではなく，レジスタに入っていても良いということがわかった。
<感想>
アドレスの中身の値を持ってくることを間接アドレスということを知れた。

[課題 G]
<目的>
指定メモリアドレスとレジスタの間でデータを転送する。間接アドレスはレジスタを使う。
<動作確認>
１ステップ実行して，8200H の内容を表示し，正しいことを確認し記録する。
<結果>
8200H の番地に 67H という指定した値が入っていたため，正しく動作していると言える。
<考察>
今回の課題で，メモリの番地にレジスタから値を書き込む方法を知ることができた。これでメモリとレジスタ間の双方向転送ができるようになった。
<感想>
今までメモリからレジスタに転送はしていたが逆はしていなかったので，これでできることの幅が広がったと思う。

[課題 H]
<目的>
指定メモリアドレスとレジスタの間でデータを転送する。間接アドレスを使う。
<動作確認>
１ステップ実行して，8300H の内容を表示し，正しいことを確認し，記録する。
<結果>
8300H の番地に 94H という値が入っていたので，正しく動作してと言える。

<考察>
今回の課題では，間接アドレスのオペランド二つどちらもレジスタの値を使用していたので，レジスタのみを用いてもできるということが理解できた。
<感想>
HALT に入ってしまうので，ソフトフェアリセットをかけなくてはいけないのが少し面倒であった。

[課題 I]
<目的>
2Byte 定数を一度に入力する。(汎用レジスタを 2 個使う)
<事前設定>
8100H に例えば ABH，8101H に例えば CDH を設定しておく。
<動作確認>
1 ステップ毎に 2 個のレジスタの内容を表示し，正しいことを確認し記録する。特に順序に注意する。
<結果>
レジスタ BC に 8100H 番地から 2 Byte のデータが入っていたので，正しく動作していると言える。
<考察>
今回の課題では，2Byte のデータを間接アドレスによって転送する方法を学ぶことができた。
<感想>
2Byte のデータを間接アドレスによって転送するときは，C 言語などだと気にしていなかったが，逆の順番になっていることに驚いた。

[課題 J]
<目的>
2 個の汎用レジスタ内のデータを，一度にメモリの指定アドレスに書き込む。
<動作確認>
レジスタ内容（B と C それぞれ）を表示し，正しいことを確認し記録する。順序に注意する。
<結果>
レジスタから指定したアドレスの番地に 2Byte の正しいデータか入っていたので，正しく動作していると言える。
<考察>
前課題では，今回の課題と逆の内容となっていたが，実際は同じようなことなので，理

解がしやすかった。
<感想>
　前課題と同じ感じであったが，やっていることは違ったので，新しいことを覚えられて面白かった。

[課題 K]
<目的>
　プログラムの流れを強制的に変える。ラベルの使用法を把握する（ラベルは，アセンブラではラベル欄はなく，オペランド欄のラベルはアドレスになる)。ループの確認をする。
<動作確認>
　１ステップ毎にアドレスを確認する。無限ループになるはず。
<結果>
　アドレスを確認しつつステップ実行していると，アドレスが 8000H 番地に戻っていて，無限ループになっていたので正しく動作していると言える。
<考察>
　今回の課題によって，C 言語でいう，goto 命令を実装できるようになった。これで，条件も追加できれば，if 分などの条件分岐もできるようになると考えた。
<感想>
　今回の課題ではじめてプログラミングのような動作ができたので，とても面白かった。

[課題 L]
<目的>
　条件を指定してジャンプするか判断させる。
<動作確認>
　１ステップ毎に実行して，３回でループを抜けることを確認する。各ステップでフラグレジスタの内容を確認し，変化した場合の意味を確認する。
<結果>
　３回のループが終わるとジャンプせずにプログラムが進んだため，正しく動作しているといえる。
<考察>
　今回の課題ではただ無条件でジャンプするだけの方法ではなく，条件のついたジャンプの方法を理解することができた。
<感想>
　これで，for 文や if 文のような動きができるようになった。

[課題 M]
<目的>
　各種の加算の方法をを確認する。
<動作確認>
　１ステップ毎に実行して，A レジスタ内容を確認する。
<結果>
　レジスタ A に 02H と 03H を加算した，05H という値が入っていたので，正しく動作していると言える。
<考察>
　今まではデータの転送だけだったが，今回の課題で，データの演算かできるようになった。
<感想>
　演算ができればプログラム自体の幅がかなり広がると考えた。

[課題 N]
<目的>
　指定メモリアドレスとレジスタの間でデータを転送する。間接アドレスの指定方法は(数値)ではなく(HL)の記法等によってレジスタを使う。
<動作確認>
　１ステップ毎に使用した全てのレジスタ内容を表示し，また，8200H 番地を表示して，結果が正しいことを確認する。さらにフラグレジスタの値を確認し，その意味を調べる。
<結果>
　8200H 番地のメモリに 03H から 02H を引いた値である 01H が入っていたため，プログラムは正しく動作していると言える。
<考察>
　今回の課題では加算だけではなく，減算を使用したため，減算の方法について理解することができた。
<感想>
　加算の時はレジスタを指定できたのに，減算の時はレジスタを指定しないプログラムだったので，そこが不思議だった。

[課題 P]
<目的>
　演算の繰り返しのプログラミングを行う場合の準備や注意を確かめる。
<動作確認>
　結果が正しいことを確認する。

<結果>
　今回のプログラムは値がオーバーフローするまでレジスタ A にカウントを足していくというものであったが，１ステップ毎に実行していくとオーバーフローしたところでフラグが立ち，ループを抜けていたので，正しく動作していたと言える。
<考察>
　今回のプログラムは，while 文に近いものであったので，C 言語のようなプログラムを作成することについての理解が深まった。
<感想>
　やっとプログラムらしいプログラムを書くことができた。このようなものを組み合わせていけば様々なものができると思った。