

## 第9回 演習

9kai.tar 中に必要なファイルはあるので利用されたい。

演習 1: 電子サイコロ `module dice(clk, rst_n, stop, count);` を設計せよ。このサイコロは、`rst_n = 0` で `count=1` に初期化される。`stop=0` ならば、`clk` の立ち上がり同期して `count=6` まで数えて 1 に戻る。`stop=1` では停止する。テストには `test_dice` を利用せよ。

演習 2: 0 番地から 7 番地まで並んでいる 8 つの数の平均を計算する POCO のプログラム `ave.asm` を書け。

ヒント: 8 で割るのは、3 ビット右にシフトすれば良い。ただし 8 回加算はダメ。

演習 3: `r0` で示した番地から `r1` で示す番地まで `r2` 個分にデータを転送するサブルーチン `block.asm` を書け。ただし、転送する番地は互いに重ならず、`r2` は 0 にしないとする。このサブルーチンを用いて 0 番地から 4 番地に 4 個分のデータを転送することで、その動作を確認せよ。(つまりはメインルーチンも `block.asm` に付ける、ということ)

演習 4: POCO には `LDI`, `LDIU` があるので、レジスタの下位 8 ビットに数を入れるのは便利だが、これだと上位 8 ビットは全て 1 か全て 0 になってしまう。そこで、上位 8bit に指定した数を入れて下位 8 ビットは 0 にクリアするのが `LDHI`(Load High Immediate) 命令を以下のフォーマットで設ける。

```
LDHI rd, #X      01010 ddd XXXXXXXX
```

例えば、

```
LDHI r1, #0xaa
```

を実行すると `r1` は 1010101000000000 になる。この `LDHI` を `poco1.v` に実装せよ。テストは、`ldhitst.asm` を使えば良い。