

# アセンブラ

TINYCPU をターゲットとするアセンブラ TINYASM

## ソースコード

```
#!/usr/bin/perl -W

%MCODE = (HALT=>0x0000,
          PUSH=>0x1000,
          PUSH=>0x2000,
          POP=>0x3000,
          JMP=>0x4000,
          JZ=>0x5000,
          JNZ=>0x6000,
          IN=>0xD000,
          OUT=>0xE000,
          ADD=>0xF000,
          SUB=>0xF001,
          MUL=>0xF002,
          SHL=>0xF003,
          SHR=>0xF004,
          BAND=>0xF005,
          BOR=>0xF006,
          BXOR=>0xF007,
          AND=>0xF008,
          OR=>0xF009,
          EQ=>0xF00A,
          NE=>0xF00B,
          GE=>0xF00C,
          LE=>0xF00D,
          GT=>0xF00E,
          LT=>0xF00F,
          NEG=>0xF010,
          BNOT=>0xF011,
          NOT=>0xF012);

$addr=0;
while(<>){
    push(@source,$_);
    if(/({w+}):/){
        $label{$1}=$addr;
        s/{w+}://;
    }
    if(/(-?#d+|[A-Z]+)/){
        $addr++;
    }
}

print "*** LABEL LIST ***\n";
foreach $l (sort(keys(%label))){
    printf "%-8s%03X\n",$l,$label{$l};
}

$addr=0;
print "\n*** MACHINE PROGRAM ***\n";
foreach (@source){
    $line = $_;
    s/{w+}://;
    if(/PUSH#{s+}(-?#d+)/){
        printf "%03X:%04X\t$line",$addr++,$MCODE{PUSH}+(($1&0xfff));
    } elsif(/(PUSH|POP|JMP|JZ|JNZ)#{s+}({w+})/){
        printf "%03X:%04X\t$line",$addr++,$MCODE{$1}+$label{$2};
    } elsif(/(-?#d+)/){
        printf "%03X:%04X\t$line",$addr++,$1&0xffff;
    } elsif(/([A-Z]+)/){
        printf "%03X:%04X\t$line",$addr++,$MCODE{$1};
    } else {
        print "\t\t\t$line";
    }
}
```

## 変数など

変数名	タイプ	
-----	-----	--

%MCODE	連想配列	ニーモニックを機械語コードに変換
\$addr	スカラー変数	現在処理している行の対応するアドレス
@source	リスト	入力されるソースコードを2回目のパスのために保存
%label	連想配列	ラベルリスト．各ラベルのアドレスを保存