november 2020

Commacode for RISC-V

Loading the assembler in order to define one or two words of your program in assembler takes relatively a lot of space.
COMMACODE produces 'comma code' for the most recently defined word and makes it easy to compile a low level forth word without the use of an active assembler.

```
: COMMACODE ( -- )
cr ." code "
created lfa>n count hx 1F and type 3 spaces
created lfa> >body   chere over - 2 + 4 /
for dup chere 2 - =
    if   h@+ . ." h, "
    else @+ u. ." , "
    then dup hx 0F and 0= if cr else space then
next drop ." end-code " cr ;
```

How to use it

1. include the assembler in noForth
2. include the COMMACODE definition (select and copy the code and paste it into your noForth terminal)
3. define your word in assembler
4. execute the word COMMACODE

Example:

```
code 2@ ( a -- lo hi )
day 4 tos x) .mov   tos tos ) .mov
sp -) day .mov      next end-code   commacode
```

The output will be:

```
code 2@   40004054 ,  C09414F1 ,  8382 h,  end-code
```

✦ Paste the output into your program.

Forth addresses in COMMACODE

It is not a good idea to have absolute addresses of variables, values, subroutines, etc. in commacode. Replace them with their forth names.
But, when you load a number into a register in RISC-V code, you will see that the number does not appear in the hex dump of the code.
Here is a method to get around that problem:
Start with `AHEAD, 0 H, ..data field.. THEN,`
The HOP register contains the address where the code starts, so the data field begins at HOP+4.

Example : An interrupt routine that uses a value

```
value SUM
code PLUS ( -- )     \ xyz @ sum +!
ahead, 0 h,
adr sum ,
then,
day 4 hop x) .mov                \ adr sum
moon day ) .mov                  \ sum
moon 1 .addi    day ) moon .mov
mret  end-code  commacode
```

```
code PLUS   A021 ,  20000D44 ,
429C4254 ,  C29C0785 ,  30200073 ,  end-code
```

✦ Replace code with routine.
✦ Replace the address of the value with "ADR SUM".

```
adr sum . ↵ 20000D44  OK.0
```

```
routine PLUS   A021 ,  adr sum ,
429C4254 ,  C29C0785 ,  30200073 ,  end-code
```