

november 2020

# Commacode for MSP430

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/
for @+ u. ." , "
      dup 7 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! ( du a -- )
  sp )+ tos ) mov   sp )+ 2 tos x) mov
  sp )+ tos mov   next end-code
commacode
```

The output will be:

```
code 2!   44B7 , 0 , 44B7 ,
2 , 4437 , 4F00 , 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. Examples:

### An interrupt routine that uses a variable

```
variable SUM
routine PLUS ( -- )    \ does: 1 sum +!
  #1 sum & add reti    \ return
end-code
commacode
```

```
code PLUS 5392 , 250 , 1300 , end-code
```

- ◆ Replace "code" with "routine".
- ◆ Replace the address of SUM with "sum".

```
sum u. ← 250 OK
```

```
variable SUM
routine PLUS 5392 , sum , 1300 , end-code
```

### A subroutine call that uses a value

```
value SCORE
routine MORE ( -- )    \ does: 1 +to score
  #1 adr score & add sp )+ pc mov \ return
end-code
commacode
```

```
code MORE 5392 , 252 , 4430 , end-code
```

```
adr score u. ← 252 OK
```

```
value SCORE
routine MORE ( -- ) 5392 , adr score ,
  4430 , end-code
```

## A code word that does a subroutine call

```
code TEST ( -- ) more # call next end-code  
commacode
```

```
code TEST 12B0 , EE98 , 4F00 , end-code
```

```
more u. ← EE98 OK
```

```
code TEST 1280 , more , 4F00 , end-code
```