

Chapter 42 - Coprocessor Positive Cookbook

Chapter 32 explains the coprocessor registers and the failure path. This chapter shows the positive path: put a tiny worker image in guest RAM, start it, enqueue a request, wait for completion, inspect the shared mailbox, then stop the worker.

The worker in this chapter is a 6502 service. It does not use a hidden file. BASIC enters the 6502 bytes into memory and starts them with `COPROC_CMD_START_MEM`.

42.1 The Mailbox Shape

Each worker has a ring inside the shared mailbox at `$790000`. The 6502 worker is CPU type 3, and its ring index is 1, so its ring base is:

$$\$790000 + 1 * \$300 = \$790300$$

Inside the 6502 worker address space, that same ring appears at `$2300`, because the worker maps the mailbox at CPU address `$2000`.

Field	Bus address	6502 address
Ring head	<code>\$790300</code>	<code>\$2300</code>
Ring tail	<code>\$790301</code>	<code>\$2301</code>
Ring capacity	<code>\$790302</code>	<code>\$2302</code>
Request slot 0	<code>\$790308</code>	<code>\$2308</code>
Response slot 0	<code>\$790508</code>	<code>\$2508</code>
Response result code	<code>\$790510</code>	<code>\$2510</code>

The example assumes a fresh worker and sends one request, so slot 0 is enough.

42.2 The 6502 Service Bytes

This service waits until head and tail differ. It reads the request operation byte from request slot 0, adds 1, writes that value into the response descriptor result-code field, sets response status to `COPROC_TICKET_OK`, advances the tail to 1, and waits for more work.

```
AD 00 23 CD 01 23 F0 F8 AD 10 23 18 69 01 8D 10 25
A9 00      8D 11 25 8D 12 25 8D 13 25
8D 14 25 8D 15 25 8D 16 25 8D 17 25
A9 02      8D 0C 25
A9 00      8D 0D 25 8D 0E 25 8D 0F 25
A9 01      8D 01 23 4C 00 00
```

The same bytes as 6502 instructions:

```

$0000 LDA $2300      ; head
$0003 CMP $2301     ; tail
$0006 BEQ $0000
$0008 LDA $2310     ; request op, low byte
$000B CLC
$000C ADC #$01
$000E STA $2510     ; response result code low byte
$0011 LDA #$00
$0013 STA $2511
$0016 STA $2512
$0019 STA $2513
$001C STA $2514     ; response length = 0
$001F STA $2515
$0022 STA $2516
$0025 STA $2517
$0028 LDA #$02
$002A STA $250C     ; response status = OK
$002D LDA #$00
$002F STA $250D
$0032 STA $250E
$0035 STA $250F
$0038 LDA #$01
$003A STA $2301     ; tail = 1
$003D JMP $0000

```

42.3 Type The Positive Example

This BASIC listing enters the service bytes, starts the worker from guest RAM, enqueues operation 7, waits for completion, and inspects the response descriptor.

```

10 REM 6502 COPROCESSOR POSITIVE PATH
20 S=MEMALLOC(64,16)
30 FOR I=0 TO 63:READ B:POKE8 S+I,B:NEXT
40 POKE32 &H000F2344,3
50 POKE32 &H000F235C,S
60 POKE32 &H000F2360,64
70 POKE32 &H000F2340,6
80 PRINT "START ";PEEK32(&H000F2348),PEEK32(&H000F234C)
90 REQ=MEMALLOC(4,4):RESP=MEMALLOC(4,4)
100 POKE32 REQ,&H12345678
110 T=COCALL(3,7,REQ,4,RESP,4)
120 PRINT "TICKET ";T
130 COWAIT T,1000
140 PRINT "STATUS ";COSTATUS(T)
150 PRINT "RESULT ";PEEK32(&H00790510)
160 COSTOP 3
170 DATA &HAD,&H00,&H23,&HCD,&H01,&H23,&HF0,&HF8
180 DATA &HAD,&H10,&H23,&H18,&H69,&H01,&H8D,&H10
190 DATA &H25,&HA9,&H00,&H8D,&H11,&H25,&H8D,&H12
200 DATA &H25,&H8D,&H13,&H25,&H8D,&H14,&H25,&H8D
210 DATA &H15,&H25,&H8D,&H16,&H25,&H8D,&H17,&H25
220 DATA &HA9,&H02,&H8D,&H0C,&H25,&HA9,&H00,&H8D
230 DATA &H0D,&H25,&H8D,&H0E,&H25,&H8D,&H0F,&H25
240 DATA &HA9,&H01,&H8D,&H01,&H23,&H4C,&H00,&H00

```

Expected result: START is followed by status 0 and command error 0, TICKET is non-zero, STATUS is 2, and RESULT is 8. The exact ticket number is not fixed. It only has to be non-zero. The result is 8 because the request operation was 7 and the worker adds 1.

42.4 What The Lines Do

Lines 20 to 30 allocate public low memory and enter the 6502 image. Lines 40 to 70 start a CPU type 3 worker from that memory by writing the raw coprocessor MMIO command 6.

Line 110 enqueues a normal BASIC COCALL. BASIC checks that REQ and RESP are public buffers created by MEMALLOC.

Line 130 waits for the worker. Line 140 prints the ticket status: 2 means COPROC_TICKET_OK. Line 150 reads the response descriptor inside the mailbox. This example uses the descriptor result code rather than a response byte buffer, because that keeps the 6502 service short and makes the mailbox visible to the reader.

42.5 Side Effects and Limits

- A running worker remains online until COSTOP or until the worker exits.
- COCALL returns ticket 0 if enqueue fails. Read COPROC_CMD_ERROR at \$F234C for the command error.
- COWAIT uses the timeout in milliseconds. If it times out, COSTATUS(ticket) reports 4.
- This first service handles the first request slot only. A full service must use the current tail value to choose a request and response slot.
- The 6502 worker sees its own 64 KB service memory plus the mailbox window. It does not automatically see every BASIC buffer by 16-bit address.

Chapter 43 shows how to debug this kind of worker with IE Mon and IE Script.