

■# BSO2 Zero Page Reference

Monitor ZP range: `\\$30-\\$8F`

User ZP range: `\\$90-\\$FF`

Label	Size	Address	Use	Ref
CORE_WORKSPACE	72 bytes	`\\$30-\\$77`	Monitor workspace (pointers, parser, debug scratch)	
1				
GAMEASK_PENDING	1 byte	`\\$78`	One-shot game prompt latch	3
BRK_FLAG	1 byte	`\\$79`	Debug/BRK context-valid flag	5
TERM_COLS	1 byte	`\\$7A`	Terminal width preference (`28/50/84`)	7
TERM_WIDTH_TIMEOUT	1 byte	`\\$7B`	Width prompt timeout seconds (`00=forever`, `01-FF=seconds`; default `08`)	8
RESERVED_GAP_A	2 bytes	`\\$7C-\\$7D`	Reserved gap before prompt scratch	
TERM_WAIT_LED	1 byte	`\\$7E`	Width prompt LED blink-pattern scratch	
TERM_WAIT_SECS	1 byte	`\\$7F`	Width prompt countdown scratch	
RST_HOOK	3 bytes	`\\$80-\\$82`	Reset trampoline target	9
NMI_HOOK	3 bytes	`\\$83-\\$85`	NMI trampoline target	11
IRQ_HOOK	3 bytes	`\\$86-\\$88`	IRQ trampoline target	13
BRK_HOOK	3 bytes	`\\$89-\\$8B`	BRK sub-dispatch trampoline target	10
HW_HOOK	3 bytes	`\\$8C-\\$8E`	Hardware IRQ sub-dispatch trampoline target	12
RESERVED_GAP_B	1 byte	`\\$8F`	Reserved gap after trampolines	
USER_ZP	112 bytes	`\\$90-\\$FF`	User-owned ZP range	15

Command Map (BSO2)

Cmd	Symbol(s)	Purpose refs
`H`	`H P`	Fixed-address help lines
`V`		Vector chain display
`!M`		Manual byte poke for pinned bytes
`Q`		Halt/resume flow tied to NMI/Reset
`R`	`N`	Debug/step/execute flow using context + vectors

Reference Legend

1. Core monitor ZP workspace block (`\\$30-\\$77`)
3. Game prompt latch byte (`\\$78`)
5. Debug/BRK context flag (`\\$79`)
7. Terminal-width pin byte (`\\$7A`)
8. Terminal-width prompt timeout byte (`\\$7B`)
9. Reset trampoline (`\\$80-\\$82`)
10. BRK sub-dispatch trampoline (`\\$89-\\$8B`)
11. NMI trampoline (`\\$83-\\$85`)
12. Hardware IRQ sub-dispatch trampoline (`\\$8C-\\$8E`)
13. IRQ trampoline (`\\$86-\\$88`)
15. User-reserved ZP (`\\$90-\\$FF`)