

[illegible]

--	--	--	--	--	--	--

[illegible]

Diagram illustrating the structure of the data:

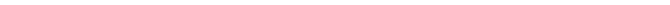
- Top row: 30 boxes.
- Middle row: 10 boxes, followed by the number 11, followed by 19 boxes.
- Bottom row: 30 boxes.

[illegible]



 “

 ”







[illegible]

" "

[illegible]

“ ”

HR

--	--	--	--

[illegible]

“ [25 boxes] ”

[50 boxes]

[25 boxes] “ [25 boxes] ”

[50 boxes]

[25 boxes]

[illegible]

1

Diagram illustrating a string in memory. The string "zxgk.court.gov.cn" is shown, with the first character 'z' at address 0x00000000 and the last character 'n' at address 0x00000010. The string is null-terminated with a '\0' character at the end.

--	--	--	--	--

API ☐ ERP

--	--	--	--	--	--	--	--

Diagram illustrating a memory buffer structure. The buffer is represented as a horizontal array of 1000 slots. The first 100 slots are filled with 'A's, and the next 100 slots are filled with 'B's. The remaining 800 slots are empty. An 'ERP' label is positioned above the 800-slot empty region.

API

API

--	--

The diagram shows three horizontal bars, each composed of small squares. The top bar is the longest, followed by the middle bar, and then the bottom bar. These bars represent the relative lengths of the three longest rivers in the world.

API ERP

[illegible][illegible]

--	--	--	--	--	--	--	--	--

The diagram shows a 32-bit instruction format. It is divided into three main sections: a 5-bit opcode field, a 27-bit operand field, and a 1-bit API field. The 27-bit operand field is further divided into a 16-bit register field and an 11-bit constant field. The 1-bit API field is located at the end of the instruction.

ERP

--	--	--	--	--	--

- 
- 

[illegible][illegible][illegible][illegible]



- 
- 
- 
- 



Revision #72

Created 2026-08-22 01:38:52 UTC by CCSST

Updated 2026-08-22 13:28:56 UTC by CCSST