

AMD HSMP interface

Newer Fam19h EPYC server line of processors from AMD support system management functionality via HSMP (Host System Management Port).

The Host System Management Port (HSMP) is an interface to provide OS-level software with access to system management functions via a set of mailbox registers.

More details on the interface can be found in chapter "7 Host System Management Port (HSMP)" of the family/model PPR Eg: https://www.amd.com/system/files/TechDocs/55898_B1_pub_0.50.zip

HSMP interface is supported on EPYC server CPU models only.

HSMP device

amd_hsmp driver under the drivers/platforms/x86/ creates miscdevice /dev/hsmp to let user space programs run hsmp mailbox commands.

```
$ ls -al /dev/hsmp crw-r--r-- 1 root root 10, 123 Jan 21 21:41 /dev/hsmp
```

Characteristics of the dev node:

- Write mode is used for running set/configure commands
- Read mode is used for running get/status monitor commands

Access restrictions:

- Only root user is allowed to open the file in write mode.
- The file can be opened in read mode by all the users.

In-kernel integration:

- Other subsystems in the kernel can use the exported transport function `hsmp_send_message()`.
- Locking across callers is taken care by the driver.

An example

To access hsmp device from a C program. First, you need to include the headers:

```
#include <linux/amd_hsmp.h>
```

Which defines the supported messages/message IDs.

Next thing, open the device file, as follows:

```
int file;

file = open("/dev/hsmp", O_RDWR);
if (file < 0) {
    /* ERROR HANDLING; you can check errno to see what went wrong */
    exit(1);
}
```

The following IOCTL is defined:

```
ioctl(file, HSMP_IOCTL_CMD, struct hsmp_message *msg)
```

The argument is a pointer to a:

```
struct hsmp_message {
    __u32  msg_id;                /* Message ID */
    __u16  num_args;              /* Number of input argument words in message */
    __u16  response_sz;          /* Number of expected output/response words */
    __u32  args[HSMP_MAX_MSG_LEN]; /* argument/response buffer */
    __u16  sock_ind;             /* socket number */
};
```

The `ioctl` would return a non-zero on failure; you can read `errno` to see what happened. The transaction returns 0 on success.

More details on the interface and message definitions can be found in chapter "7 Host System Management Port (HSMP)" of the respective family/model PPR eg: https://www.amd.com/system/files/TechDocs/55898_B1_pub_0.50.zip

User space C-APIs are made available by linking against the `esmi` library, which is provided by the E-SMS project <https://developer.amd.com/e-smi/>. See: https://github.com/amd/esmi_ib_library