



Struct sigaction

La estructura sigaction especifica como tratar un signal. En esta estructura, tenemos distintos miembros.

sa_handler

Puede tomar tres valores distintos:

- **SIG_IGN**: Ignorará el signal recibido y no realizará ninguna acción.
- **SIG_DFL**: Usará el tratamiento por defecto del signal.
- **Cabecera_funcion**: Toma por tratamiento una funcion definida por el propio usuario. Habrá que definir como valor el nombre de la función.

sa_mask

Se añaden signals a la mascara, con fin de añadirlas al grupo de signals bloqueadas.

- Si la máscara está vacía, solo se añade el signal que se quiere capturar.
- Una vez se ha salido del tratamiento, se debe restaurar la máscara inicial.

sa_flags

Se usan para configurar el comportamiento de los signals. El valor 0 define la configuración por defecto. Tratamos los siguientes flags:

- **SA_RESETHAND**: Se ejecuta y posteriormente se reestablece el tratamiento por defecto del signal.
- **SA_RESTART**: Si un proceso esta bloqueado y recibe el signal, se reinicia la llamada que lo ha bloqueado.

```
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
```

```

#include <signal.h>
#include <string.h>
void error_y_exit(char *msg,int exit_status)
{
    perror(msg);
    exit(exit_status);
}
int segundos=0;
void funcion_alarma(int signal)
{
    char buff[256];
    segundos=segundos+10;
    sprintf(buff,"ALARMA pid=%d: %d segundos\n",getpid(),segundos);
    write(1, buff, strlen(buff) );
}
int main (int argc,char * argv[])
{
    struct sigaction sa;
    sigset_t mask;
    /* EVITAMOS QUE NOS LLEGUE EL SIGALRM FUERA DEL SIGSUSPEND */
    sigemptyset(&mask);
    sigaddset(&mask, SIGALRM);
    sigprocmask(SIG_BLOCK,&mask, NULL);
    while (segundos<100) {
        alarm(10);
        switch (fork()) {
            case 0:
                /* REPROGRAMAMOS EL SIGNAL SIGALRM */
                sa.sa_handler = &funcion_alarma;
                sa.sa_flags = SA_RESTART;
                sigfillset(&sa.sa_mask);
                if (sigaction(SIGALRM, &sa, NULL) < 0) error_y_exit("sigaction", 1);
                execlp("./loop", "./loop", 1);
                break;
            case -1: error_y_exit("fork", 1);
        }
        sigfillset(&mask);
        sigdelset(&mask, SIGALRM);
        sigdelset(&mask, SIGINT);
        sigsuspend(&mask);
    }
    exit(1);
}

```



Como podemos ver en el código superior (resaltado en morado), la función de tratamiento de la señal esta redefinida a `funcion_alarma(int s)`. Siempre que se active el conjunto de signals con esa máscara en concreto, se llamará a la función y no al tratamiento por defecto.