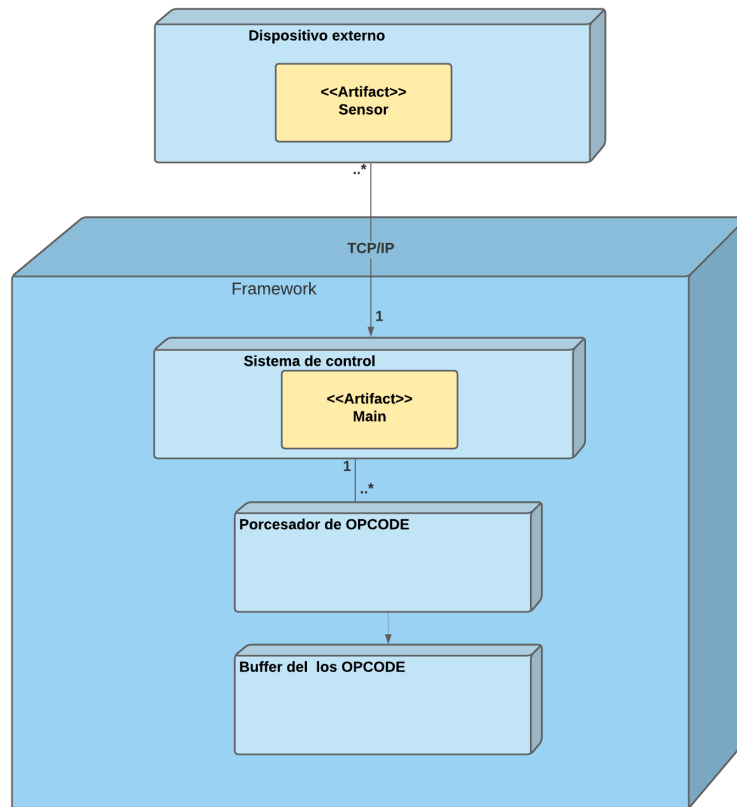


ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN
SISTEMAS OPERATIVOS
Proyecto-Rover

1. Diagrama de despliegue



2. Explicación

Mi proyecto se encarga de implementar un framework para la NASA, el cual por medio del protocolo TCP/IP recibe información de los sensores, al mismo tiempo crea hilos por cada sensor que es atendido y así se puede mantener la recepción de información de manera concurrente. A su vez cada hilo que atiende a un sensor evalúa las operaciones en donde se tienen que almacenar el dato recibido, para que así genere dos hilos por cada tipo de operación, uno para realizar la operación luego de un periodo de tiempo, y el otro para la evaluación del tiempo que ha esperado el framework a que le llegue información del sensor y que el mismo elimine a la operación en caso de que no haya recibido algo en X unidades de tiempo.

En el proyecto se manejan dos estructuras principales y 3 estructuras de apoyo, entre las principales contamos con la del OpProcesor, el cual se encarga de mantener almacenada la información por operación que envían cada sensor y el OpCodeBuffer, la cual mantiene la información en Colas. Adicional existen dos métodos esenciales, el verificador de

frecuencia, el cual realizará la operación sobre el buffer cada cierto tiempo, y el método verificar tiempo, el cual verificará si el sensor ha dejado de enviar información, estos métodos se encuentran en el archivo framework.c

Para iniciar el programa se lo debe ejecutar de la siguiente forma:

Para el framework:

```
./framework <puerto> -f frecuencia -x tiempo-espera [-d]
```

-d para entrar en modo debug, caso contrario real time

Para el sensor:

```
./sensor <ip:localhost> <puerto> "[op_code1,op_code2,op_code3]" -t  
intervalo-tiempo-envio-información
```

Adicional en el proyecto se agregaron dos archivos para testing, los cuales ejecutan el test para el simulador y el otro para el sensor, puede cambiar los valores a su conveniencia, el modo de ejecutarlo es:

```
./test_framework
```

Para el sensor:

```
./test_sensor
```

Nota:

En framework:

Unidades de frecuencia: Segundos

Unidades de tiempo-espera: Minutos

En sensor:

Unidades de t: Milisegundos