

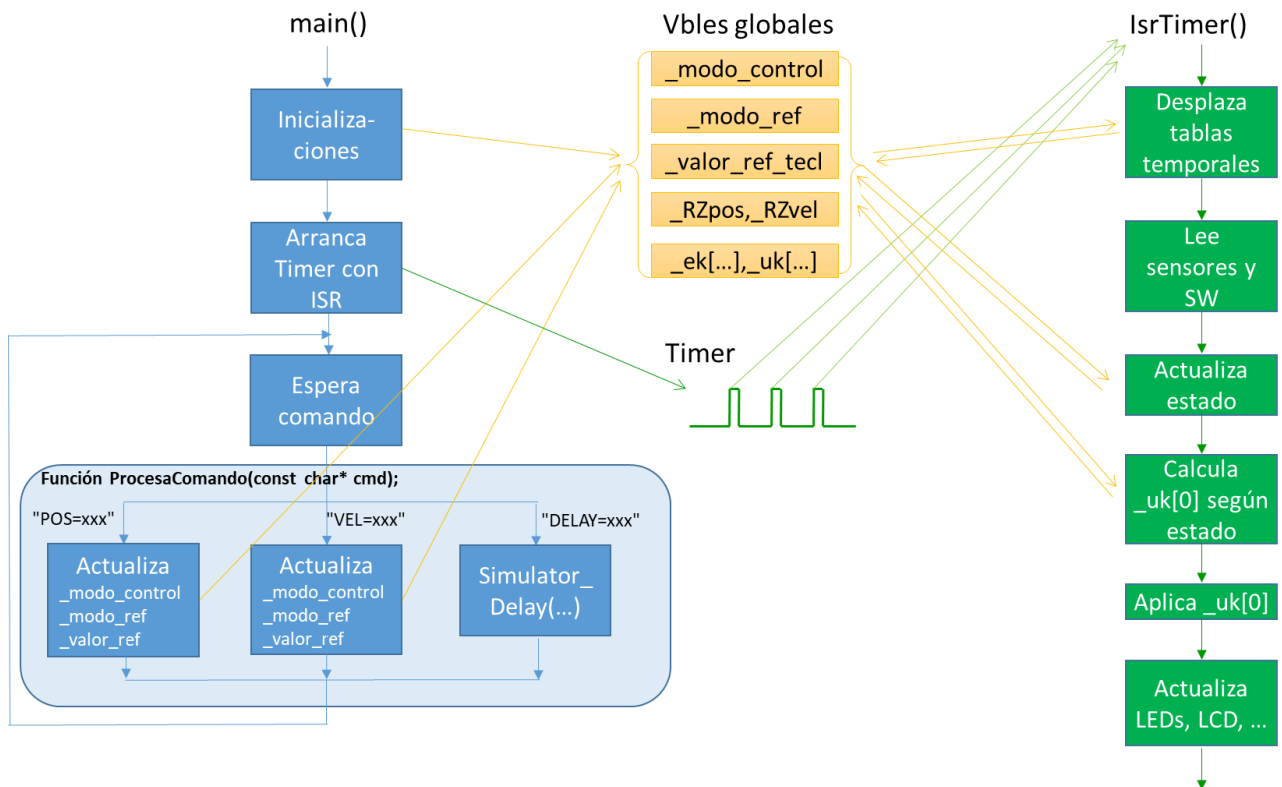


## Guía de Prácticas

ASIGNATURA: Informática Industrial y Comunicaciones  
CENTRO: Escuela Politécnica de Ingeniería de Gijón  
ESTUDIOS: Grado en Ingeniería Electrónica y Automática  
CURSO: 3º CUATRIMESTRE: 1  
CARÁCTER: Obligatoria CRÉDITOS ECTS: 6  
PROFESORADO: Ignacio Alvarez, José Mª Enguita, Angel Navarro, Mariam Saeed

PRACTICA 10: Gestión de interrupciones

1. Reorganizar el programa realizado en prácticas anteriores para que el lazo de control se ejecute bajo interrupción del temporizador 0, de forma que el programa principal se pueda dedicar a la espera de comandos por teclado y el servicio de los mismos.



Aplicar todo o parte de la lógica de control, en función del estado de avance de PLs anteriores.

## Esquema general del programa

```

#include...

Declaración variables globales:

- Todas aquellas que necesitan compartir main() y MiFnControl()
- Todas aquellas que sólo usa MiFnControl() pero necesitan mantener su valor


Algunas a modo de ejemplo:
struct RZ  rz pos, rz vel;
float *_ek_pos,*_ek_vel,*_uk...;
int _modo_control,_modo_ref;

void ProcesarComando(const char* cmd)
{
    Actualiza vbles globales en función de contenido de cmd
}

void MiFnControl()
{
    Declaración variables locales a MiFnControl() - sólo usadas aquí y que
    no necesitan mantener su valor anterior

    Actualizacion tablas temporales: Desplazar tablas uk, ek, ...
    Leer nuevos valores posk,velk
    Lógica de control:
    if (Switch 7 activo)
        _uk[0]=0;
    else
    {
        if (modo control es POS)
        {
            Calcular ek[0]
            _uk[0]=valor control POS según _modo_ref y switches
        }
        if (modo control es VEL)
        {
            Calcular _ek[0]
            uk[0]=valor control VEL según modo ref y switches
        }
    }
    AplicarTensionMotor(_uk[0]);
    Actualizar LCD y bits de indicación de sentido de movimiento
}

int main()
{
    char cmd[80];
    Otras vbles locales
    Dar valores iniciales a vbles globales: reguladores, ek, uk, modo control, etc.
    Simulator_ConnectWss(...);
    Simulator_SetTimerInterrupt(0,100,MiFnControl);

    while (1)
    {
        printf(">> ");
        gets(cmd);
        ProcesarComando(cmd); // → Actualiza vbles globales según contenido de cmd
    }
}

```

En caso de usar varios archivos .c :

- Declaración de variables globales "tal cual": en uno solo de los .c
- Declaración de variables globales "extern": en el resto que las necesitan

| main.c  | other.c   |
|---|---|
| <pre> #include... ... <b>int _modo_control,_modo_ref;</b> ...  void main() {     ...     <b>_modo_control=valor inicial;</b>     ...     ... } </pre> | <pre> #include... ... <b>extern int _modo_control,_modo_ref;</b> ... void MiFnControl() {     ...     <b>usar _modo_control, _modo_ref</b>     ... } </pre> |