Flujo de ejecución según ChatGPT

Perfecto, esa es la pregunta clave □□

En los ABB S4 el flujo de ejecución no es intuitivo porque NO empieza en el programa RAPID directamente, sino en el sistema. Vamos paso a paso, de forma lineal, como realmente ocurre dentro del controlador.

□□ Flujo real de ejecución en un ABB S4

Piensa en 3 capas:

1□ Arranque del sistema

2□ Carga de tareas RAPID

3□ Ejecución del programa de usuario

1□ ARRANQUE DEL CONTROLADOR

Cuando enciendes el S4:

➜ 1.1 Carga del sistema base

Se lee desde disquete (o disco):

BASE.SYS

Esto:

Inicializa CPU

Reserva memoria

Define qué otros .SYS se deben cargar

□□ Todavía NO hay RAPID ejecutándose

➜ 1.2 Carga de parámetros

Desde SYSPAR/:

SYS.CFG

MOC.CFG

PROC.CFG

EIO.CFG

...

Aquí el controlador aprende:

Qué robot es

Cómo se mueve

Qué entradas/salidas existen

Cuántas tareas habrá

□□ Aún no hay lógica de usuario

➜ 1.3 Carga de datos del robot

DAT_MT16.SYS

Define:

Longitudes de brazos

Ceros de ejes

Límites

⚠ Sin esto, el robot no puede moverse

➜ 1.4 Creación del RAMDISK

Todo lo anterior se copia a:

RAMDISK/

□□ Desde este punto el sistema ya corre desde RAM, no desde disquete.

2□ CARGA DE RAPID (NO ejecución todavía)

Ahora entra en juego USER.SYS

➜ 2.1 USER.SYS = “Índice de programas”

USER.SYS NO es un programa, es un loader.

Dentro dice cosas como:

Qué módulos RAPID cargar

En qué tareas

Qué módulo es principal

Ejemplo conceptual:

TASK PERS

LOAD PROG201.PRG

□□ Hasta ahora:

El programa existe en memoria

Pero no se está ejecutando

➜ 2.2 Compilación de RAPID

El controlador:

Lee PROG201.PRG

Lo compila

Verifica sintaxis

Si hay error → NO arranca

---

3 EJECUCIÓN DEL PROGRAMA RAPID

Aquí está la parte que suele confundir.

➜ 3.1 ¿Dónde empieza el programa?

En ABB S4 NO empieza en la primera línea.

Empieza en:

PROC main()

□□ SIEMPRE

No importa el nombre del archivo.

➜ 3.2 ¿Quién llama a main()?

El sistema operativo del S4:

BASE.SYS → USER.SYS → RAPID runtime → main()

Tú nunca llamas a main() manualmente.

➜ 3.3 Flujo típico dentro del programa

Ejemplo simplificado:

MODULE PROG201

```
PROC main()
    Init;
    WHILE TRUE DO
        Ciclo;
    ENDWHILE
ENDPROC

PROC Init()
    ! Inicialización
ENDPROC

PROC Ciclo()
    MoveJ p1, v100, z10, tool0;
ENDPROC

ENDMODULE
```

□□ El robot:

Entra a main()

Ejecuta Init

Se queda en el bucle infinito

□□ Resumen mental (MUY IMPORTANTE)

El flujo NO es:

Disquete → PROG201.PRG → primera línea

El flujo REAL es:

POWER ON

↓

BASE.SYS

↓

SYSPAR/*.CFG

↓

DAT_MT16.SYS

↓

RAMDISK

↓

USER.SYS

↓

Carga módulos RAPID

↓

Busca main()

↓

Ejecuta main()

⚠ Errores comunes

 “No hace nada al arrancar”

→ main() está vacío o no existe

 “Carga el programa pero no se mueve”

→ No hay bucle o está esperando una señal

 “No arranca el sistema”

→ Error en MOC.CFG o DAT_MT16.SYS

□□ Ejercicio mental rápido

Si tú borras:

PROG201.PRG → el sistema arranca, pero sin programa

USER.SYS → no se carga ningún RAPID

BASE.SYS → el S4 no arranca

main() → el programa carga pero no corre

Si quieres, en el siguiente mensaje puedo:

Dibujarte el flujo como diagrama

Explicarte qué pasa cuando pulsas START en el FlexPendant

Revisar contigo un USER.SYS real línea por línea

Revision #1
Created 30 December 2025 18:12:35 by Rafael Carbonell Lázaro
Updated 2 January 2026 14:33:37 by Rafael Carbonell Lázaro