RAMDISK/COM200.SYS

Análisis del Archivo COM200.SYS - Comunicación RPC Multitarea

□□ Descripción General

Sistema de comunicación RPC (Remote Procedure Call) para intercambio de datos entre el robot ABB IRB 6400 y un sistema externo (posiblemente un PLC o supervisor). Implementado como tarea secundaria (Task 1) en un entorno multitarea.

□□ Contexto de la Aplicación

```yaml
Estación: FFS Schaufelbearbeitung (procesamiento de palas)
Ubicación: ABB Kraftwerke Birr/Bexbach (central eléctrica)
Función: Comunicación robot ↔ sistema de control externo (RPC)
Autor: Bernd Luckas - ABB Flexible Automation GmbH
Fecha: 22.08.1995 - 24.09.1996
Versión: 2.00
```

□□ Especificaciones Técnicas

```yaml
```

Robot: IRB 6400 (modelo específico XXXX)

Hardware: M94A

Software: Basic CommWare 2.0 + Advanced Functions + Multitasking

Programa: 3HAB2317-1/05

Comunicación: RS232 por puerto sio1:

Protocolo: Telegramas personalizados (STX/ETX/ACK/NAK)

```

## □□ Estructura del Protocolo de Comunicación

### 1. □□ Telegrama de ESTADO (Robot → RPC)

```rapid
Formato: [STX] [Z] [Datos(12)] [ETX]

Posiciones:

  1: STX (2) - Start of Text

  2: Z (90) - Identificador "Zustand" (Estado)

  3: nAlr - Número de alarma

  4-6: nAnrC1-3- Número de alarma convertido (3 chars)

  7: nBtrz - Estado operativo (A=Auto, M=Manual, S=Stop)

  8: nQpoa - ???

  9: nEpoa - ???

  10: nSpb - ???

  11: nQtta - ???
```
```

12: nEtta - ???

13: nStb - ???

14: ETX (3) - End of Text

```

2. □□ Telegrama de ORDEN (RPC → Robot)

```rapid

Formato complejo con múltiples campos:

- Tipo de orden (AuftragsArt)

- Estación (StationsNr)

- Posiciones (PlatzNrStation, PlatzNrIRB)

- Tipo de transferencia (TransArt)

- Peso (Gewicht)

- Desplazamientos X,Y,Z (VerschiebX/Y/Z)

- Rotación Z (DrehungZ)

- Tipo de adaptador

- Dirección de aproximación

- Patrón de colocación

- Tipo de pala (SchaufelTyp)

```

□□ Diagrama de Flujo del Sistema
```

```
┌──────────────────┐ RS232 ┌──────────────────┐
│ Robot ABB │◄──────────►│ Sistema RPC │
│ IRB 6400 │ sio1: │ (Supervisor) │
│ │ │ │
│ TASK 0 (Main) │ I/O │ │
│ │ │ Digital │ │
│ ▼ │◄──────────►│ │
│ TASK 1 (COM) │ di/do │ │
└──────────────────┘ └──────────────────┘

▲▲

└──────── Telegramas ────────┘
```

⚙ Variables de Comunicación entre Tareas

Señales Digitales Task 0 ↔ Task 1:

```rapid
diCTENQ → Enable Request (Task 0 solicita comunicación)
doCTACK → Acknowledge (Task 1 confirma recepción)
```

Variables Compartidas (posiblemente en DAT_MT16.SYS):

```rapid
nZ_tgm → Flag telegrama estado procesado
nA_tgm → Flag telegrama orden procesado
nAnr → Número de alarma actual
nAlr → ???
nQpoa, nEpoa, nSpb, nQtta, nEtta, nStb → Parámetros de estado
```

□□ Funciones Principales

main() - Ciclo Principal

```rapid
IF bInit=FALSE initialisiere; → Inicializa comunicación
ctrtask0; → Escanea solicitudes Task 0
ctrbetrz; → Controla estado operativo
IF bSend=TRUE send; → Envía telegrama estado
IF bSend=FALSE receive; → Recibe telegrama orden
IF bProcess_Atgm=TRUE process_Atgm; → Procesa orden recibida
stop_task; → Controla estado de tarea
```

send() - Envío de Telegramas

```rapid

1. Prepara buffer de salida (14 bytes)

2. Envía por RS232 (WriteBin)

3. Espera ACK (6) del RPC

4. Reintenta hasta 3 veces si falla

```

receive() - Recepción de Telegramas

```rapid

1. Lee bytes del puerto serie

2. Detecta STX (inicio) y ETX (fin)

3. Responde ACK al telegrama completo

4. Activa procesamiento si es válido

```

process_Atgm() - Procesamiento de Órdenes

```rapid

Convierte datos ASCII a valores numéricos:

- Campos de 1-5 dígitos

- Signo negativo detectado por carácter 45 (-)

- Escalas aplicadas (×0.1 para desplazamientos)
```

```

□□ Sistema de Gestión de Errores

Códigos de Error (stFMTCOM01-07):

```yaml
01: "Kein ACK in RPC Antwort TGM" - Sin confirmación
02: "ERR_DEV_MAXTIME" - Timeout en dispositivo
03: "Kein ACK" (reintento)
04: "Keine Antwort von RPC" - Sin respuesta tras 3 intentos
05: "TGM-Kennung undefiniert" - Identificador desconocido
06: "ERR_FILEACC" - Error acceso a archivo
07: "ERR_DEV_MAXTIME inerhalb RPC-TGM" - Timeout interno
```

Prioridades:

· 0: Solo mensaje en Teach Pendant
· 1: Mensaje + detiene tarea (requiere OK)

□□ Ejemplo de Configuración para Repositorio

Archivo: □□ Configuraciones/Comunicación/COM200_SYS.md
```

```markdown

## Configuración RS232 Implícita

(Puerto sio1: configurado en SIO.CFG)

- Baudrate: (desconocido, verificar en SIO.CFG)

- Paridad: (desconocido)

- Bits de datos: (desconocido)

- Bits de parada: (desconocido)

## Mapeo de Estados Operativos

nBtrz (Estado del robot):

  65 'A' → AUTO (diCProg_run=1 AND diCAuto_Ein=1)

  77 'M' → MANUAL (sin error, no auto)

  83 'S' → STOP (diCFehler=1)

## Protocolo de Handshake

1. Task 0 activa diCTENQ

2. Task 1 envía telegrama estado

3. Task 1 activa doCTACK

4. Task 0 desactiva diCTENQ

5. Task 1 desactiva doCTACK

```

□□ Dependencias con Otros Archivos

```

COM200.SYS depende de:

├── BASE_MT.SYS (definiciones base)

├── DAT_MT16.SYS (variables compartidas)

└── SIO.CFG (configuración puerto serie)

COM200.SYS comunica con:

└── PROG201.PRG (Task 0 principal)

```

□□ Notas de Implementación

1. Multitarea Real → Task 1 independiente de movimiento

2. Comunicación Asíncrona → No bloquea movimiento robot

3. Robustez → Reintentos y manejo de errores

4. Industrial → Diseñado para 24/7 en planta

¿Necesitas que explique cómo configurar el puerto serie SIO.CFG o cómo integrar este sistema en una aplicación nueva?

Revision #1
Created 30 December 2025 15:47:21 by Rafael Carbonell Lázaro
Updated 2 January 2026 14:33:37 by Rafael Carbonell Lázaro