

El software de depuración DTC_DS_1

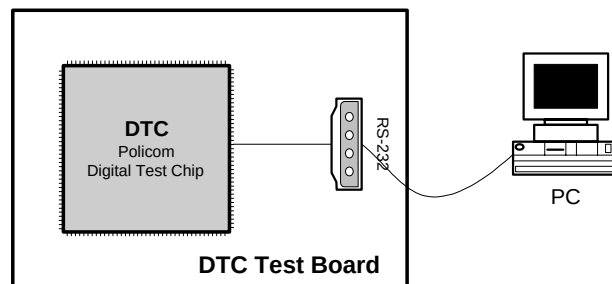
El software de depuración DTC_DS_1 está formado por un conjunto de rutinas repartidas entre el Test Chip Digital y un PC, empleando como interfaz de comunicación un canal de comunicaciones serie UART.

El objetivo fundamental de este conjunto de rutinas es proporcionar una vía de comunicación para enviar toda la información que se necesite del interior del test chip hacia un observador externo, en especial la evolución de los registros internos del chip.

Esta sección describirá en detalle cómo se ha diseñado el DTC_DS_1. Esta versión está orientada a chequear el funcionamiento del SBM. Sólo es necesario realizar pequeñas modificaciones en el código para adaptarlo a otras necesidades.

Los recursos hardware empleados

Distinguimos dos categorías de recursos hardware empleados en la configuración del software de depuración. La figura muestra el esquema hardware de trabajo.



Por un lado tendremos un PC que será el encargado de mostrar la información que le llega del DTC.

En el otro lado tendremos el hardware relacionado con el DTC. Las celdas del DTC involucradas son las siguientes:

- UART → encargada de llevar a cabo la comunicación.
- TIMER → que servirá para temporizar los envíos de información al PC.

Las comunicaciones PC-DTC

El software de depuración DTC_DS define un protocolo de comunicación que se llevará a cabo durante la ejecución de dicho software. Asimismo se define los formatos de trama que se emplearán en las transacciones de datos.

Especificación del protocolo de comunicación PC-DTC

Se definen los siguientes servicios del protocolo:

- CONFIG: utilizada para iniciar la comunicación DTC-PC y enviar la configuración inicial.
- DATA: empleado para intercambio de datos. Los datos contienen información sobre el interior del DTC.
- EOS: final de simulación. Cierra la comunicación entre ambos dispositivos.

En el diagrama de la página siguiente se describe el protocolo. Inicialmente es el DTC el que comienza el diálogo con el PC, pidiendo que le sea enviada la configuración. El PC responderá al DTC con tres tramas que especifican los siguientes parámetros:

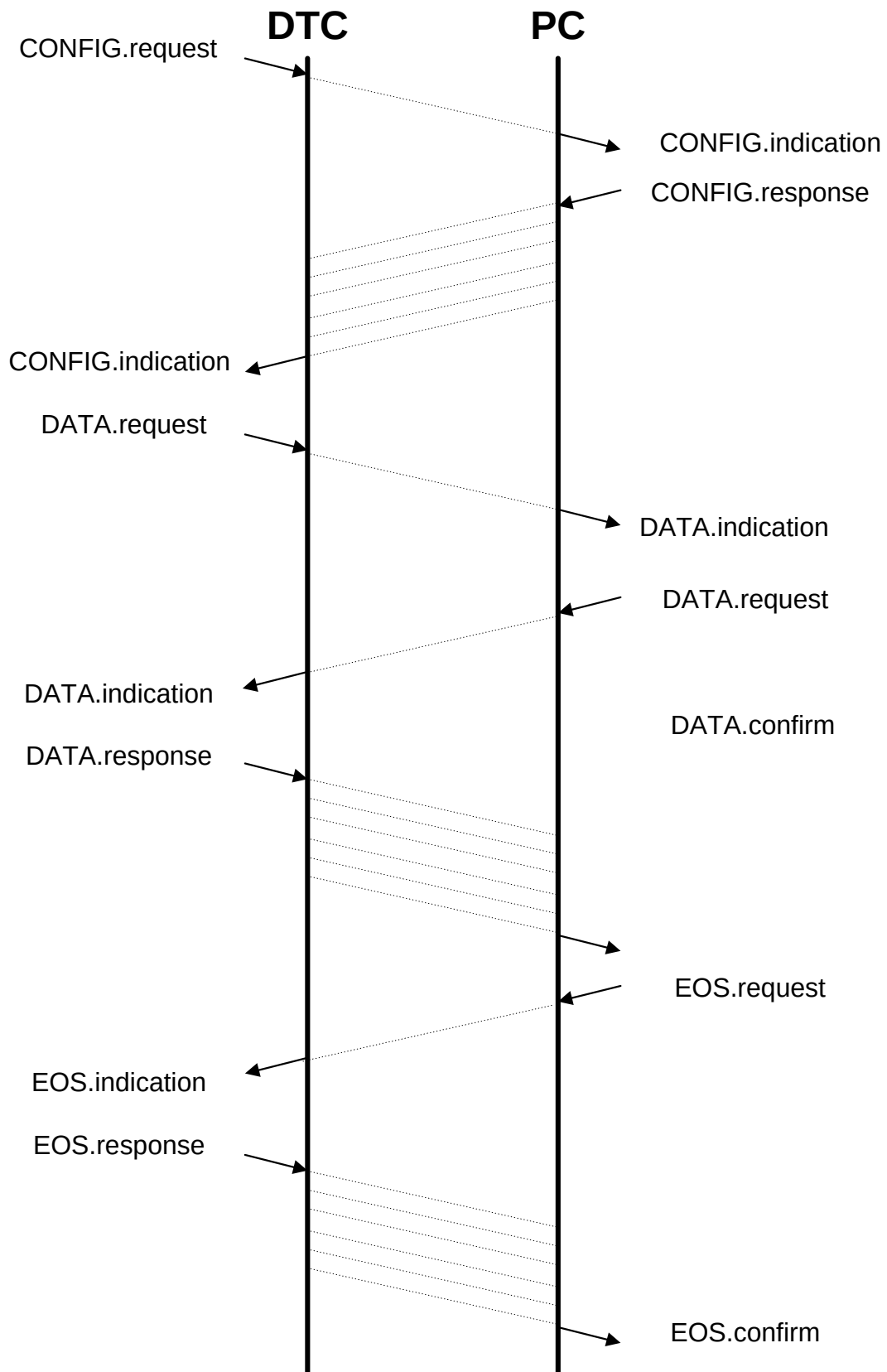
La información requerida para configurar el dispositivo bajo estudio, en este caso el SBM.

La información de configuración del Timer.

La dirección de los registros del DTC que el PC quiere monitorizar.

Una vez configurado todos los parámetros del DTC, este mandará información de cómo ha ido su proceso de configuración. A continuación el PC pedirá que comience el envío de las variables que el usuario quiere monitorizar.

El envío de las variables internas del DTC se realizará periódicamente según el período que esté configurado en el Timer. Este procedimiento periódico puede ser abortado por el PC cuando solicite un servicio EOS (final de simulación). En ese momento el DTC enviará alguna información interna de utilidad para el usuario y dará por finalizado el proceso de depuración.



Formatos de tramas

Se definen las siguientes tramas:

- **CONTROL:** empleada para enviar información de control.
- **DATOS:** envía información datos entre DTC y PC. Distinguimos los siguientes tipos:
 - **DF_0:** datos sin formato DTC→PC ó PC→DTC.
 - **DF_1:** empleada para enviar configuración del SBM en dirección PC→DTC
 - **DF_2:** PC→DTC con la información del timer.
 - **DF_3:** PC→DTC con la información que el PC pide que le mande periódicamente.
 - **DF_4** formato de la trama que contiene los datos DTC→PC con los las variables que el usuario quiere monitorizar.

La siguiente tabla informa de qué tipo de trama emplea cada primitiva:

Servicio	Primitiva	Trama	Subtrama
CONFIG	request	control	-
	response	datos	CF_1 CF_2 CF_3
DATA	request	control	
	response	datos	CF_4
EOS	request	control	-
	response	datos	CF_0

La trama de control

La trama de control tiene la siguiente estructura:

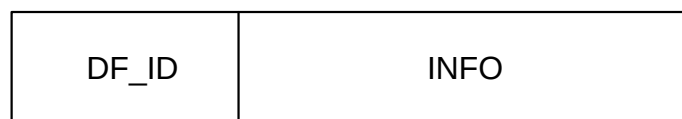
CF_ID	CB	SB
-------	----	----

- **Byte CF_ID:** es el identificador de trama de control. Sirve para saber que se trata de una trama de control.

- Byte CB: es el octeto de control. Se emplea para enviar información de control. Puede incluir los siguientes mensajes:
 - IMALIVE → el chip comunica que se encuentra activo.
 - EOC → el chip avisa del final de la comunicación.
 - EOS → orden de final de simulación
 - EOT → final de transmisión
 - STARTING → comienza la simulación.
- Byte SB_SB → información sobre estado requerido. Su contenido puede ser:
 - Estado tras configurar el DTC
 - Otros no implementados hasta ahora.

Las tramas de datos

Existen un total de cinco tipos de tramas datos definidas. Su estructura general es la siguiente:



- DF_ID → identifica el tipo de subtrama de datos.
- INFO → campos de información, que dependerá del tipo de subtrama.

SUBTRAMA DE DATOS DF_0

Empleada para enviar un byte de uso general en cualquier sentido. Su identificador es DF_0 y el campo de información está compuesto por un solo byte de información.

SUBTRAMA DE DATOS DF_1

Sirve para enviar información de configuración del PC → DTC. La información contenida en su campo de información va encaminada a configurar el dispositivo que se quiere monitorizar, en nuestro caso el SBM.

Adicionalmente podrán definirse otras subtramas de configuración de otros dispositivos.

La estructura es la siguiente:

DF_1	IER	CCR[0]	CCR[1]	BDT[0]	BDT[1]	BPT[0]	BPT[1]	FRQ[0]	FRQ[1]
------	-----	--------	--------	--------	--------	--------	--------	--------	--------

- DF_1 → identificador de este tipo de subtrama.
- IER → registro de habilitación de interrupciones.
- CCR → registro de control de comunicación.
- BDT → umbral de decisión.
- BPT → umbral de detección.
- FRQ → frecuencias portadoras para cada canal.

SUBTRAMA DE DATOS DF_2

Se emplea para enviar los datos configuración del Timer 0 empleado en la monitorización.

Su estructura es la siguiente:

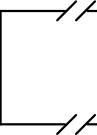
DF_1	LR[0]	LR[1]	CR
------	-------	-------	----

- DF_1 → Identificador de trama de configuración del Timer 0.
- LR → Registro de carga del Timer 0.
- CR → Registro de Configuración del Timer 0.

SUBTRAMA DE DATOS DF_3

Utilizada para enviar del PC→DTC cuáles son los registros que va a monitorizar durante el proceso.

Su estructura es la siguiente:

DF_3	RN	Addr_0	Size_0		Addr_n	Size_n
------	----	--------	--------	--	--------	--------

- DF_3 → es el identificador de subtrama del tipo 3
- RN → número de registros a monitorizar.
- Add_x → dirección del registro x a monitorizar. Ancho de 32 bits
- Size_x → tamaño real del registro x en número de bits.

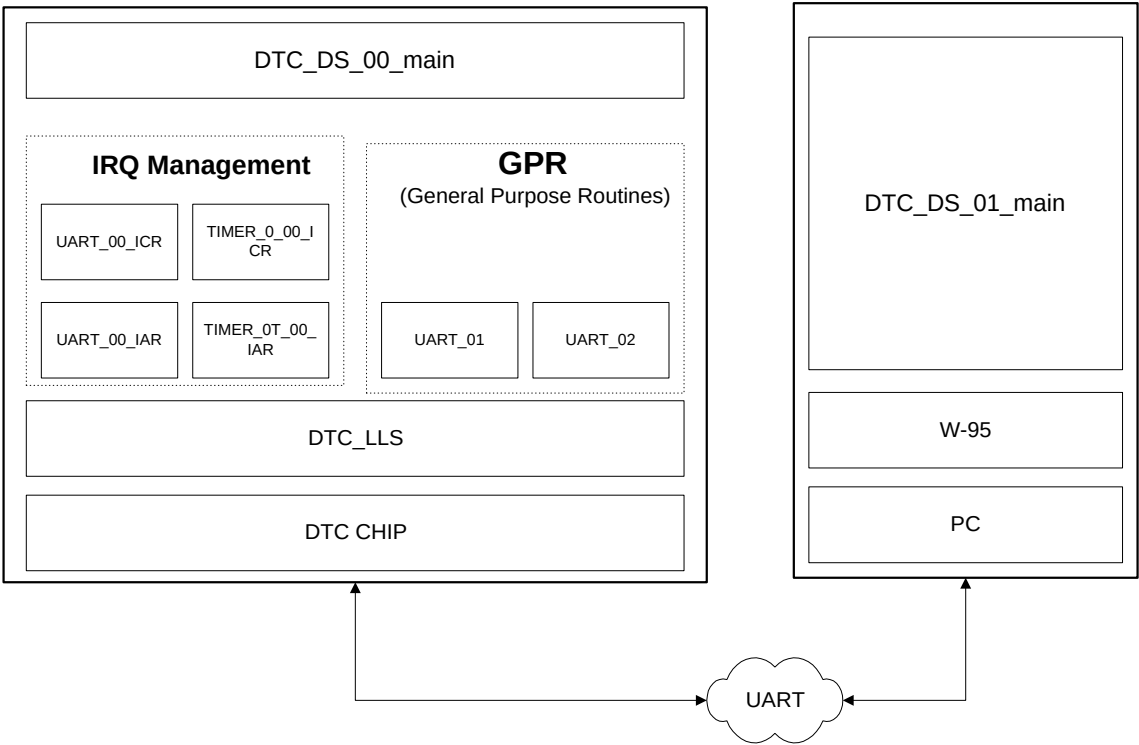
La versión DTC_DS_1 sólo permite monitorizar un solo registro.

Los recursos software empleados

Igualmente aquí dividimos el software en dos grupos: el que irá sobre el PC y el que se ejecutará sobre el DTC.

Nos centraremos en el software que va a ejecutarse en el DTC debido a su complejidad.

El esquema siguiente muestra un diagrama con todas las rutinas involucradas en el proceso.



Las rutinas

A continuación se describen en detalle la dos principales rutinas que forman el DTC_DS_00. Es resto de rutinas, y debido a su simplicidad, se puntualizan en el cuadro resumen al final de la sección.

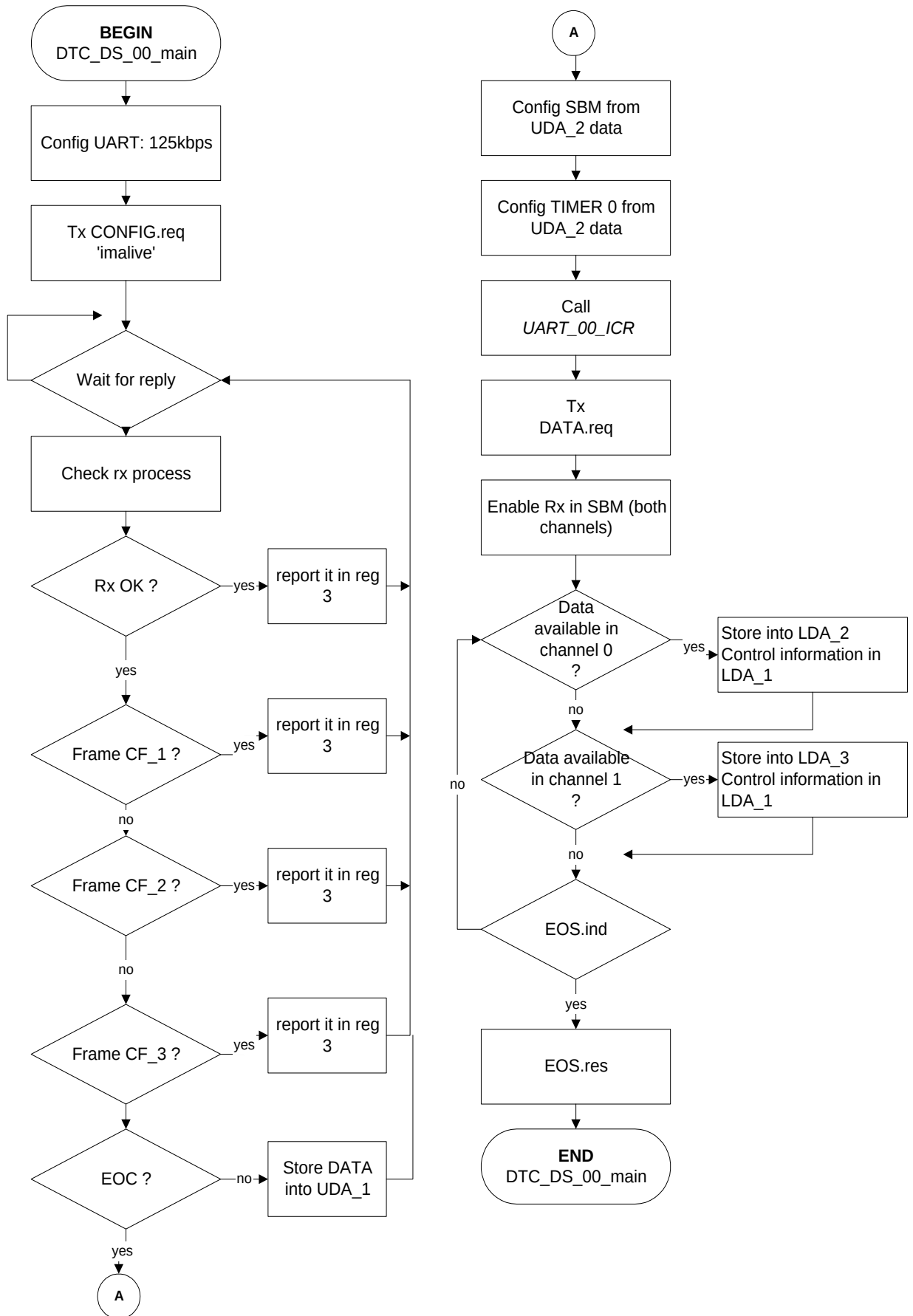
DTC_DS_00_main

La siguiente tabla muestra la información más relevante de la rutina.

El código de la rutina se encuentra disponible en el Anexo VI.

Funcionalidad	Programa principal de ejecución en el DTC
Parámetros de la llamada	Ninguno. Es invocado desde el Reset_EH desde memoria externa.
AMM	UDA_1 → mensaje inicial del DTC al PC UDA_2 → con la configuración del PC al DTC LDA_1, 2 y 3 → datos locales
Llamadas a subrutinas	UART_01 UART_02
Llamadas a ICR's	UART_00_ICR TIMER_0_99_ICR
IAR's presentes en su ejecución	UART_00_IAR TIMER_0_00_IAR

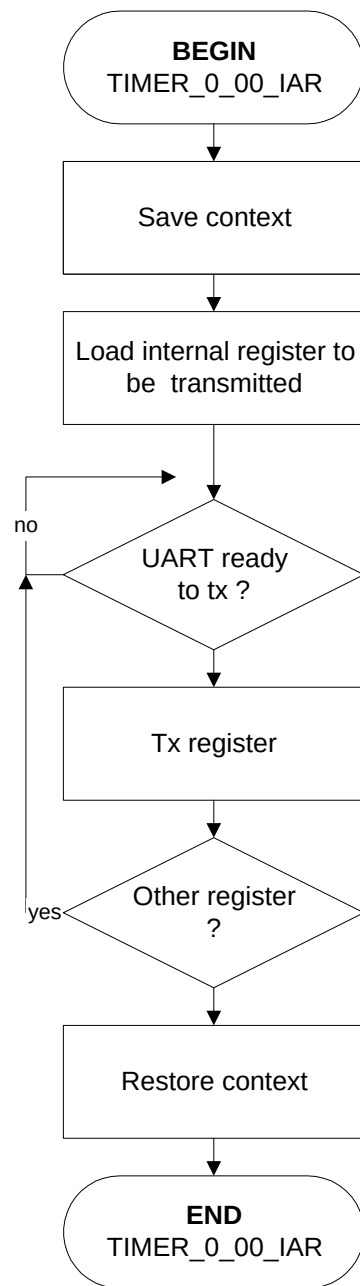
DIAGRAMA DE FLUJO



Timer_0_00_IAR

Funcionalidad	Se encarga de transmitir periódicamente al exterior del DTC la información requerida por el usuario.
Parámetros de la llamada	Ninguno. Se trata de una retina de atención a interrupción del Timer.
AMM	TDA_1 → datos del timer
Llamadas a subrutinas	Ninguna
Llamadas a ICR's	Ninguna
IAR's presentes en su ejecución	UART_00_IAR, en recepción

DIAGRAMA DE FLUJO



Condiciones de ejecución

El software de depuración que se ejecute en el DTC lo hará residente en memoria externa y su codificación será, en primera aproximación, en modo ARM.

El software diseñado está orientado a depurar el hardware SBM. Por ello en el programa principal se encuentra una parte de código que realiza ciertas tareas con el SBM. Esa parte es independiente del resto del DST_DS, por lo que puede ser sustituida por otro código y simplemente borrada.

El AMM

El Mapa de Memoria de Aplicación (AMM) es la organización de los recursos de memoria que emplea el software que se encuentra ejecutando en el DTC.

Para el software de depuración distinguimos dentro del AMM las siguientes áreas:

- UDA_x: áreas de datos de la uart.
- TDA_x: áreas de datos del timer
- LDA_x: áreas de datos locales.

Las UDA son áreas de datos en las que se encuentra datos que han sido recibidos y van a ser transmitidos por la UART.

Las TDA's son datos que se utilizan en una IAR del timer 0.

Las LDA's son las áreas locales de datos en las cuales se guardan datos temporales que sólo utiliza la misma rutina. Las LDA's no pueden dar problema de inconsistencia de datos, ya que no son utilizadas por las IAR's.

El siguiente cuadro caracteriza el AMM que emplea el software de depuración:

xDA	Utilización	Leída por:	Escrita por:	Tamaño (bytes)	Localización	Dirección
UDA_1	datos para tx por la UART	UART_01	main	32	i-RAM	INT_RAM_USER_DATA_Base
UDA_2	Datos recibidos por la UART	main	main	32	i-RAM	UDA_1 + 0x40
UDA_3	Datos recibidos por UART	main	UART_00_IAR	8	i-RAM	UDA_2 + 0x20
LDA_1	Variables locales de main	main	mian	32	i-RAM	UDA_3 + 0x10
LDA_2	variables locales main. Datos recibidos por el SBM	main	main	8 K	e-RAM	EXT_RAM_Base
LDA_3	variables locales que contienen los datos del canal 1 del SBM	main	main	8 K	e-RAM	LDA_1 + 0x2000
TDA	Variable empleadas por el timer	TIMER_0_00_IAR	main	32	i-RAM	LDA_1 + 0x20

Cuadro resumen

Nombre	Función	Parámetros	Llamadas a otras rutinas	IAR's presentes	AMM Empleado
DTC_LLS	Soporte de todas las llamadas a ejecución que se provoquen en el sistema.	Ninguno	Reset_EH → DTC_DS_00_main	-	-
UART_00_ICR	Rutina de configuración de interrupción de la UART con o sin interrupción	R0→ Vector de interrupción. R1→ prioridad R2→ tasa de bits R3→ registro de control de línea R4→ registro de habilitación de irq R5→ control de modem.	-	-	-
UART_00_IAR	Rutina de atención de interrupción de la uart	-	-	TIMER_0 con mayor prioridad	UDA_3 (escritura)
TIMER_0_00_ICR	Rutina de configuración de interrupción.	R0→ Vector de interrupción. R1→ prioridad R2→ registro de carga del timer R3→ registro de control del timer			-
TIMER_0_00_IAR	Rutina de atención de interrupción del timer	-	-	UART con menor prioridad	TDA_1
UART_01	Transmite por la UART un buffer completo en espera activa.	R0→ nº de bytes a transmitir R1→ dirección del buffer a transmitir	-	-	UDA_1 (lectura)
UART_02	Transmite por la UART un registro de 32 bits.	R0→ registro a transmitir.	-	-	-
DTC_DS_00_main	Programa principal de depuración.	TIMER_0_00_IAR UART_00_IAR	UART_00_ICR TIMER_0_00_ICR UART_01 UART_02		UDA_1 (escritura) UDA_2 (lectura y escritura) UDA_3 (lectura) LDA_1(l y e) LDA_2 (l y e) LDA_3 (l y e)

