

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

**CARRERA DE INGENIERÍA EN ELÉCTRICA,
AUTOMATIZACIÓN Y CONTROL**

**PROYECTO DE GRADO PARA LA OBTENCIÓN DEL
TÍTULO DE INGENIERÍA**

**ARQUITECTURA DE PROCESADORES ARM DE 32 BITS,
CIRCUITOS INTEGRADOS Y ENTORNOS DE
DESARROLLO**

BYRON RUBÉN MARTÍNEZ ARIAS

Sangolquí Ecuador

2007

CERTIFICACIÓN

Certificamos que el siguiente proyecto de grado “ARQUITECTURA DE PROCESADORES ARM DE 32 BITS, CIRCUITOS INTEGRADOS Y ENTORNOS DE DESARROLLO” fue desarrollado en su totalidad por el señor Byron Rubén Martínez Arias, bajo nuestra dirección.

Atentamente,

Ing. Elemer Guanoluiza
DIRECTOR

Ing. Flavio Pineda
CODIRECTOR

AGRADECIMIENTO

A mis padres, Bayron y Martha, por todo el cariño y comprensión que me han dado y ser el soporte fundamental de mi vida.

A mi hermano, Christian, por estar siempre conmigo en las buenas y en las malas dándome confianza y palabras de aliento para seguir adelante.

A mi abuelita, Laurita, por haberme criado como una madre.

A mis abuelitos Nelson y Blanquita, por todo el cariño que me han dado y por que siempre han estado preocupados por mi bienestar.

A mi tío, Edwin y su familia, por los consejos y el apoyo que me han brindado.

A todos mis tíos, Nelson, María, Rita, Olavo por su preocupación y el cariño que he recibido.

A mis primos por compartir tantos momentos plenos.

A mis amigos por todos los gratos momentos que hemos vivido.

Al Ing. Byron Navas, por que gracias a su investigación se pudo llevar a acabo el desarrollo de este tema de tesis.

Al director de este proyecto, Ing. EleMBER Guanoluiza y al codirector Ing. Flavio Pineda, por su apertura y disposición para encaminar el desarrollo de esta tesis.

Y a todas esas personas que de una u otra manera supieron apoyarme para la realización de esta meta.

DEDICATORIA

A mi hija, Samantha Carolina

A mis padres, Bayron y Martha

A mi hermano, Christian

A mis abuelitos:

Nelson, Blanca y Laura

PRÓLOGO

I.- ANTECEDENTES

La arquitectura ARM (Advanced RISC Machine) fue el primer microprocesador el cual tuvo un gran impacto en el mundo. El fundamento de estos microprocesadores es realizar cualquier proceso de una manera rápida y sencilla.

La arquitectura ARM es una alternativa muy importante en el diseño de procesadores, por que permite usar un número menor de instrucciones, alberga menos espacio de memoria y debido a su procesador de 32 bits puede desempeñar muchas mas aplicaciones que otras arquitecturas.

Gracias a su diseño sencillo, el ARM tiene relativamente pocos componentes en el chip, por lo que no alcanza altas temperaturas y tiene bajos requerimientos de energía. Estas características lo han hecho candidato perfecto para el mercado de aplicaciones integradas (embedded applications) que van desde un teléfono celular hasta una lavadora. Esta arquitectura tuvo su primera aparición en productos comerciales como los asistentes digitales personales Newton de Apple.

Basándose en la arquitectura ARM se ha logrado grandes avances tecnológicos; conforme ha ido avanzando su desarrollo el manejo de dispositivos se ha incrementado. Los procesadores basados en una arquitectura ARM han evolucionado tanto que han llegado a ser parte fundamental en el desarrollo de aplicaciones tanto en el área de las comunicaciones, en automatización industrial y en la electrónica de consumo.

Tanta es la importancia de estos procesadores que el diseño básico del ARM fue licenciado por diferentes compañías como: Freescale (una empresa que derivó de Motorola en el año 2004), IBM, Nintendo, Philips, Infineon Technologies, Texas Instruments, VLSI, Atmel, Sharp y Samsung.

II.- ALCANCE DEL PROYECTO

El presente proyecto de tesis consiste en realizar un estudio de la arquitectura ARM, sus circuitos integrados y herramientas de desarrollo, permitiendo de esta manera que los estudiantes cuenten la información necesaria para el desarrollo de proyectos con esta nueva tecnología.

III.- ESTRUCTURA DEL PROYECTO

En el primer capítulo se trata una breve introducción de la arquitectura ARM, estudia las principales familias y tecnologías que la componen y que ayudan a la arquitectura ARM en el desarrollo de proyectos.

El segundo capítulo esta dedicado al estudio de la estructura y funcionamiento interno de la arquitectura ARM, los set de instrucciones con los que trabaja y también se verán ciertas aplicaciones en el desarrollo de tecnología.

El tercer capítulo muestra las principales casas que se dedican a la fabricación de microprocesadores y microcontroladores ARM, enumerando los principales chips y sus propiedades.

El cuarto capítulo describe el circuito integrado AT91SAM7S64, su estructura interna y los principales periféricos que lo constituyen.

El quinto capítulo expone las herramientas para desarrollo de proyectos como son entornos de desarrollo, emuladores, y tarjetas de desarrollo.

El sexto capítulo esta dedicado a la descripción del entorno de desarrollo Keil uVision3 para el integrado AT91SAM7S64, se realiza una descripción general del entorno de desarrollo y de los principales simuladores que lo constituyen como son el de Puertos, Timers y el Conversor Análogo Digital. También se realizó un tutorial básico para la edición, compilación y simulación de proyectos.

El séptimo capítulo contiene las conclusiones y recomendaciones referentes al presente trabajo.

Finalmente en la sección anexos se incluye tablas comparativas de chips, entornos de desarrollo, emuladores y tarjetas de desarrollo.

ÍNDICE DE CONTENIDO

| | |
|---|----|
| PRÓLOGO | I |
| <hr/> | |
| ÍNDICE DE CONTENIDO | IV |
| <hr/> | |
| CAPÍTULO I – INTRODUCCIÓN | 1 |
| <hr/> | |
| 1.1. RESEÑA HISTÓRICA DE LA ARQUITECTURA ARM | 3 |
| 1.1.1. ARCON | 3 |
| 1.1.2. El Primer ARM | 4 |
| 1.1.3. ARM 2 | 5 |
| 1.1.4. ARM 3 | 6 |
| 1.1.5. ARM 4 y ARM 5 | 7 |
| 1.1.6. ARM 6 | 7 |
| 1.1.7. Actualización de la Arquitectura ARM | 7 |
| 1.2. FAMILIAS DE LA ARQUITECTURA ARM | 8 |
| 1.2.1. Familia ARM 7 | 8 |
| 1.2.2. Familia ARM 9 | 9 |
| 1.2.3. Familia ARM 10E | 10 |
| 1.2.4. Familia ARM 11 | 11 |
| 1.3. TECNOLOGIAS ARM | 14 |
| 1.3.1. Thumb | 14 |
| 1.3.2. Thumb-2 | 15 |
| 1.3.3. Thumb-2EE | 16 |
| 1.3.4. NEON | 17 |
| 1.3.5. TRUSTZONE | 19 |

| | |
|---------------------------------------|-----------|
| CAPÍTULO II – ARQUITECTURA ARM | 21 |
|---------------------------------------|-----------|

| | |
|---|-----------|
| 2.1. ESTRUCTURA INTERNA | 21 |
| 2.1.1. Pipelining | 23 |
| 2.2. PROCESADOR RISC | 25 |
| 2.3. TAMAÑO DE INSTRUCCIONES | 27 |
| 2.4. TIPO DE DATOS | 27 |
| 2.5. MODOS DE PROCESADOR | 28 |
| 2.6. REGISTROS | 29 |
| 2.6.1. Set de Registros ARM | 29 |
| 2.6.2. Función de los registros | 30 |
| 2.6.3. Los Registros de Almacenamiento FIQ | 31 |
| 2.6.4. Registros de Estado del Programa | 31 |
| 2.6.5. Bandera de Código de Condiciones | 32 |
| 2.6.6. Bits de Control | 32 |
| 2.6.7. Excepciones | 33 |
| 2.7. SET DE INSTRUCCIONES | 34 |
| 2.7.1. Set de Instrucciones ARM | 34 |
| 2.7.1.1. Condiciones de Código de Campo ARM | 35 |
| 2.7.1.2. Instrucciones de Procesamiento de Datos | 35 |
| 2.7.1.3. Instrucciones de Procesamientos de Datos ARM | 36 |
| 2.7.1.4. Instrucciones de Transferencia de Datos | 36 |
| 2.7.1.5. Transferencia de Instrucciones en Bloque | 37 |
| 2.7.1.6. Instrucciones Swap | 38 |
| 2.7.1.7. Interrupción de Software | 38 |
| 2.7.1.8. Instrucción Branching | 39 |
| 2.7.1.9. Branching Link | 39 |
| 2.7.1.10. Branching Exchange | 40 |
| 2.7.2. Set de Instrucciones ARM | 40 |
| 2.8. IMPLEMENTACIÓN DEL CONJUNTO DE INSTRUCCIONES DE LA ARQUITECTURA ARM | 41 |
| 2.8.1. StrongARM | 41 |
| 2.8.2. nnARM | 42 |

| | |
|--|----|
| 2.8.2.1. Opencores | 42 |
| 2.8.2.2. Organización | 42 |
| 2.8.2.2.1. Cache | 42 |
| 2.8.2.2.2 Cache de Instrucciones | 43 |
| 2.8.2.2.3 Cache de datos | 43 |
| 2.8.2.2.4 Prefetch de instrucciones | 43 |
| 2.8.2.2.5 Modos de Memoria | 44 |
| 2.8.2.2.6 Ancho del Bus | 44 |
| 2.9. APLICACIONES DE LA ARQUITECTURA ARM EN EL | |
| DESARROLLO DE TECNOLOGÍA | 44 |
| 2.9.1 Home Solutions | 44 |
| 2.9.1.1 Gaming | 45 |
| 2.9.1.2 TV digital | 45 |
| 2.9.1.3 Cámaras Digitales | 46 |
| 2.9.1.4 Set-Top-Box Digitales | 46 |
| 2.9.2 Aplicaciones Móviles | 46 |
| 2.9.2.1 Teléfonos de voz | 47 |
| 2.9.2.2 Asistentes personales digitales | 47 |
| 2.9.2.3 Teléfonos inteligentes | 47 |
| 2.9.2.4 Media players personales | 48 |
| 2.9.2.5 Sistemas de posicionamiento geográfico persona | 48 |
| 2.9.3 Sistemas integrados | 48 |
| 2.9.3.1 Estructura automotriz y sistemas de utilidad | 48 |
| 2.9.3.2 Sistema powertrain de vehículos | 48 |
| 2.9.3.2 Sistema powertrain de vehículos | 48 |
| 2.9.3.4 Carros telemáticos y multimedia | 49 |
| 2.9.3.5 Tarjetas inteligentes | 50 |
| 2.9.4 Aplicaciones en la Empresa | 50 |
| 2.9.4.1 Redes empresariales e infraestructura | 50 |
| 2.9.4.2 Redes de almacenamiento | 51 |
| 2.9.4.3 Redes para el hogar | 51 |
| 2.9.4.4 Impresoras | 51 |
| 2.9.4.5 VoIP | 52 |

CAPÍTULO III – CIRCUITOS INTEGRADOS CON ARQUITECTURA ARM

| | |
|--------------------------------|-----------|
| 3.1. MICROPROCESADORES | 52 |
| 3.1.1. Digi | 52 |
| 3.1.2. Zarlink Semiconductor | 54 |
| 3.1.3. Actel | 56 |
| 3.2. MICROCONTROLADORES | 57 |
| 3.2.1. Intel | 60 |
| 3.2.2. Atmel | 60 |
| 3.2.3. Philips | 69 |
| 3.2.4. Cirrus logia | 75 |
| 3.2.5 Analog Device | 80 |
| 3.2.6 Sanyo | 83 |

CAPÍTULO IV – DESCRIPCION DEL CHIP AT91SAM7S64

| | |
|--|-----------|
| 4.1. DESCRIPCIÓN | 85 |
| 4.2. DIAGRAMA DE BLOQUES GENERAL | 86 |
| 4.3 ARQUITECTURA DEL INTEGRADO | 89 |
| 4.3.1. Memorias | 89 |
| 4.3.2. Mapa de Memorias | 90 |
| 4.3.2.1. SRAM Interna | 90 |
| 4.3.2.2 ROM Interna | 90 |
| 4.3.2.3 Flash Interna | 90 |
| 4.3.3. Procesador y Arquitectura | 91 |
| 4.3.3.1. Procesador ARM7TDMI | 91 |
| 4.3.3.2. Depuración de Errores y Características de Prueba | 91 |
| 4.3.3.3. Controlador de Memoria | 91 |
| 4.3.3.4. Controlador de Periféricos DMA | 92 |
| 4.3.4. Periféricos | 92 |
| 4.3.4.1. Múltiplexión de Periféricos en las Líneas de PIO | 94 |
| 4.3.4.2 Múltiplexión del Controlador A de los PIO | 94 |

| | |
|--|-----|
| 4.3.5. Control del Sistema | 95 |
| 4.3.5.1. Mapa del controlador del sistema | 97 |
| 4.3.5.2. Controlador de Reset | 98 |
| 4.3.5.3. Generador de Clock | 98 |
| 4.3.5.4. Control del Manejo de Potencia | 98 |
| 4.3.5.5. Controlador de Interrupciones | 99 |
| 4.3.5.6. Unidad de Depuración | 99 |
| 4.3.5.7. Timer de Intervalos Periódicos | 99 |
| 4.3.5.8. Watchdog Timer | 99 |
| 4.3.5.9. Controlador de PIO | 99 |
| 4.3.5.10. Controlador Regulador de Voltaje | 100 |
| 4.3.6. Recepción y transmisión asincrónica y sincrónica (USAR) | 100 |
| 4.3.6.1. Controlador Serial Sincrónico | 101 |
| 4.3.7. Timers | 101 |
| 4.3.8. Conversor Análogo Digital | 102 |
| 4.3.9. Abastecimiento de Potencia | 103 |
| 4.3.9.1. Consumo de Potencia | 104 |
| 4.3.9.2. Características Eléctricas | 104 |

CAPÍTULO V – HERRAMIENTAS DE DESARROLLO DE PROYECTOS

| | |
|-------------------------------------|-----|
| 5.1. ENTORNOS DE DESARROLLO | 105 |
| 5.1.1. AdaMULTI | 105 |
| 5.1.2. Visual X-Tools | 106 |
| 5.1.3. Embest IDE for ARM | 107 |
| 5.2. EMULADORES | 108 |
| 5.2.1. Keil ULINK™ USB-JTAG Adapter | 110 |
| 5.2.2. Embest PowerICE for ARM | 110 |
| 5.2.3. Embest Emulator for ARM | 112 |
| 5.2.4. Embest UNetICE for ARM | 113 |
| 5.2.5. Nohau Emulator | 114 |
| 5.2.6. TRACE32-FIRE | 115 |
| 5.3. TARJETAS DE DESARROLLO | 116 |

| | |
|---|-----|
| 5.3.1. Embest ATEBSAM7S | 116 |
| 5.3.2. Embest ATEB9200 | 118 |
| 5.3.3. CSB637 (Atmel AT91RM9200 based uSBC) | 119 |
| 5.3.4. Embest ATEB40x | 120 |
| 5.3.5. Embest AX4510 | 121 |
| 5.3.6. Embest S3CEV40 | 122 |
| 5.3.7. Keil MCB2100 | 125 |
| 5.3.8. KB9202 | 126 |
| 5.3.9. Embest LPCEB2000 | 127 |
| 5.3.10. Embest S3CEB2410 | 128 |
| 5.3.11. Embest STDV710A | 129 |

CAPÍTULO VI – ENTORNO DE DESARROLLO (IDE), KEIL UVISON3 PARA EL INTEGRADO AT91SAM7S64

| | |
|--|------------|
| 6.1. DESCRIPCIÓN GENERAL DEL KEIL UVISON3 | 132 |
| 6.1.1 Manejo de Proyectos | 133 |
| 6.1.2 Base de Datos de los Dispositivos | 133 |
| 6.1.3 Componentes de un Proyecto | 135 |
| 6.1.4 Manuales de Usuarios y Libros Adicionales | 135 |
| 6.1.5 Creación y Configuración de Herramientas | 136 |
| 6.1.6 Desarrollo de Opciones de Herramientas | 136 |
| 6.1.7 Proceso Build | 137 |
| 6.1.8 Descarga de la Flash | 137 |
| 6.1.9 Fuentes de Navegación | 137 |
| 6.1.10 Depurador | 137 |
| 6.1.11 Ventana de Memoria y Visualización | 139 |
| 6.2. SIMULADOR | 139 |
| 6.2.1. Puertos | 139 |
| 6.2.2. Timers | 141 |
| 6.2.3. Conversor Análogo Digital | 144 |

| | |
|--|------------|
| 6.3. TUTORIAL BÁSICO PARA DESARROLLAR PROYECTOS; EDICIÓN, COMPILACIÓN, SIMULACIÓN | 148 |
| 6.3.1. Instalación | 148 |
| 6.3.2. Edición | 151 |
| 6.3.2. Compilación | 155 |
| 6.3.3. Simulación | 157 |

CAPÍTULO VII – CONCLUSIONES Y RECOMENDACIONES

| | |
|-----------------------------|------------|
| 7.1. CONCLUSIONES | 165 |
| 7.2. RECOMENDACIONES | 168 |

REFERENCIAS BIBLIOGRÁFICAS

ANEXOS

- ANEXO1: Tabla comparativa de Circuitos Integrados**
- ANEXO2: Tabla Comparativa de Entornos de Desarrollo**
- ANEXO3: Tabla Comparativa de Emuladores**
- ANEXO4: Tabla Comparativa de tarjetas de Desarrollo**

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

GLOSARIO

CAPÍTULO I

INTRODUCCIÓN

La arquitectura ARM (Advanced RISC Machine) fue creada por Acorn Computer Group, como el primer procesador RISC con gran impacto comercial en el mundo. La filosofía RISC (Reduced Instruction Set Computer) busca realizar un proceso de la manera menos complicado posible.

La arquitectura ARM es una alternativa muy importante en el diseño de procesadores, por que permite usar un número menor de instrucciones, alberga menos espacio de memoria y debido a su procesador de 32 bits puede desempeñar muchas más aplicaciones que otras arquitecturas.

Gracias a su diseño sencillo, el ARM tiene relativamente pocos componentes en el chip, por lo que no alcanza altas temperaturas y tiene bajos requerimientos de energía. Estas características lo han hecho candidato perfecto para el mercado de aplicaciones integradas (embedded applications) que van desde un teléfono celular hasta una lavadora. Esta arquitectura tuvo su primera aparición en productos comerciales como los asistentes digitales personales Newton de Apple. Basándose en la arquitectura ARM se ha logrado grandes avances tecnológicos; conforme ha ido avanzando su desarrollo el manejo de dispositivos se ha incrementado.

Los procesadores basados en una arquitectura ARM han evolucionado tanto que han llegado a ser parte fundamental en el desarrollo de aplicaciones tanto en el área de las comunicaciones, en automatización industrial y en la electrónica de consumo. Tanta es la importancia de estos procesadores que el diseño básico del ARM fue licenciado por diferentes compañías como: Freescale (una empresa que

derivó de Motorola en el año 2004), IBM, Nintendo, Philips, Infineon Technologies, Texas Instruments, VLSI, Atmel, Sharp y Samsung.

La arquitectura ARM es un procesador RISC de 32 bits usado en un amplio número de aplicaciones, son los procesadores más populares en el mundo utilizados en sistemas integrados. Hoy en día, cerca del 75% de los procesadores de 32 bits poseen este chip en su núcleo. La tecnología ARM es ampliamente usada en el diseño de system-on-chip formando el núcleo del sistema, también la encontramos a la vanguardia en el desarrollo de nuevos sistemas móviles como son: teléfonos celulares, organizadores personales, cámaras digitales, beepers, PDAs, videophones, teléfonos inteligentes (smartphone); la tecnología ARM también la encontramos en productos de consumo digital como: set-top box, gateway para el hogar, Sony PSP PlayStation, MP3 audio, MPEG4 video y en los próximos IPODS; en la industria automotriz la encontramos en la telemática y en los sistemas infotainment (sistemas de entretenimiento e información como son: sistema de navegación GPS, MP3, automotive DVD players, sistemas de procesamiento digital in – cabin, y otros, los cuales son instalados en las principales marcas de autos como son BMW, Audi, Alfa Romeo, Ford, Mercedes, Volvo); también pueden ser encontrada formando parte de otros productos como: dispositivos de seguridad, dispositivos de almacenamiento como discos duros, etc. Debido a su popularidad, es una de las arquitecturas cuyo IP (propiedad intelectual) se lo puede encontrar gratis (en sus versiones básicas); esto es importante porque se lo puede sintetizar e implementar directamente en un dispositivo programable.

Las diferentes familias de arquitectura ARM están estandarizadas por un núcleo procesador y periféricos estándares que ayudan a manejar la complejidad y compatibilidad de diferentes sistemas. Según se han ido desarrollando las diferentes familias de la arquitectura ARM como son ARM7, ARM9, ARM10, ARM11, han ido incrementado los adelantos tecnológicos, brindado un incontable número de herramientas para desarrollar proyectos con esta arquitectura. La investigación de la arquitectura ARM concederá bases teóricas y herramientas para el desarrollo de proyectos, lo que ayudará a que las personas que se dedican al

diseño electrónico realicen proyectos de una manera eficiente y a la par con las tecnologías utilizadas en la actualidad.

La mayoría de las compañías electrónicas internacionales han tomado la arquitectura ARM como el principal instrumento para el adelanto tecnológico, debido a que ofrece un extenso número de productos para el desarrollo de proyectos como; microprocesadores RISC de 16/32 bits, procesador de datos, procesador 3D, librerías digitales, memorias integradas, periféricos, software, herramientas de desarrollo; muchas de las cuales son de libre acceso. Debido a esto es necesario hacer un estudio de la arquitectura ARM, sus diferentes familias, fabricantes y herramientas de desarrollo para sentar bases teóricas con la finalidad de conocer los beneficios que brinda y poder desarrollar diferentes tipos de aplicaciones de acuerdo con los avances electrónicos que se den en el mundo.

1.1 RESEÑA HISTÓRICA DE LA ARQUITECTURA ARM

1.1.1 ACORN

El primer circuito integrado ARM fue desarrollado luego de una avanzada investigación y con un equipo de desarrollo en las computadoras Acorn, uno de los pioneros en el desarrollo de microcomputadores. En esos tiempos Arcon era una de las principales marcas de computadores personales británicos. El éxito inicial de Arcon se estableció cuando la Corporación de Radiodifusión Británica (BBC) creó un nuevo modelo de computador para el hogar de la compañía para ser vendido como el microcomputador BBC. El lanzamiento de este micro BBC en 1982 tomó el primer lugar en lo que era computadores para el hogar principalmente en Gran Bretaña, y el nombre BBC le dio al diseño Acorn más importancia y credibilidad frente a otras máquinas que fueron desarrolladas. El micro BBC fue la base para el procesador 6502 de 8 bits de Rockwell, el mismo chip que impulsó el Apple II. Los modelos iniciales de Arcon ofrecían gráficos de color y un acceso a memoria random de 32 Kbytes.

1.1.2 EL PRIMER ARM.

Los trabajos de desarrollo del chip ARM comenzaron hacerse en 1983, para que luego sean continuados en 1985 por Steve Furber ahora profesor de Ingeniería en Computadoras en la universidad de Manchester, Roger Wilson, quien a trabajado en el diseño del micro BBC y Robert Heaston encargado de guiar el grupo de trabajo para el diseño, formaron el equipo de trabajo para crear un dispositivo el cual reúna las características del procesador 6502 pero en un entorno de RISC de 32 bits e implementarlo en un dispositivo pequeño el cual sería fácil diseñar, probar y fabricarlo con un costo mínimo. La decisión más importante de tomar fue el uso de una instrucción de tamaño fijo y un modelo para cargar los datos. Otra sugerencia fue la de hacer una instrucción solamente para instrucciones básicas.

El primer modelo del set de instrucciones ARM fue escrito en BASIC, por lo cual los siguientes modelos ARM también fueron hechos en el mismo lenguaje fuente. El diseño físico actual del chip fue realizado usando tecnología VLSI que se acostumbraba usar para el diseño de herramientas de programación, para realizar pruebas de los circuitos integrados se diseño un simulador llamado even – driven el cual fue hecho en BASIC, el controlador de video (VIDC), el controlador de memoria (MEMC) y el controlador de I/O (IOC), también fueron diseñados y probados. Posteriormente se diseñaría un nuevo simulador el cual se realizó en Modula – 2 y posteriormente en C y fue conocido como ASIM, el cual es usado hasta hoy para el diseño y prueba de dispositivos por la ACORN y ARM LTD.

El primer procesador comercial RISC en el mundo y el primer procesador ARM fue el ARM1, el cual fue fabricado por primera vez en silicio en Abril de 1985 con tecnología VLSI¹, este mejoró las metas de diseño que fueron planteadas ya que uso menos de 25.000 transistores, estos ejemplares fueron fabricados usando 3 microprocesadores.

¹Tecnología VLSI, compañía que se dedica al diseño y manufactura de circuitos integrados

1.1.3 ARM 2

Luego de haber trabajado con la arquitectura ARM 1 en la programación de chips de muestra, se encontró que existían algunas áreas donde el set de instrucciones podía mejorarse para aumentar el desempeño de los sistemas que se basaban en esta arquitectura. La investigación de los laboratorios Acorn albergó el primer ejemplar de una nueva familia de procesadores RISC. Sin embargo los procesadores ARM fueron creados con la intención de potenciar la siguiente generación de computadores personales Acorn.

En 1987, una casa de computadores, los Archimedes, lanzo el primer chip comercial usando la arquitectura ARM, ofreciendo una versión de 8 MHz de la ARM2, un controlador de entradas/salidas y un sistema operacional simple. Los Archimedes no tuvieron una buena acogida en su lanzamiento; por que los computadores personales aparecieron para ser consolidados después de los estándares de los PC IBM mientras Acorn introducía un computador con un nuevo procesador, un nuevo sistema operativo y no tenia una software base para que el usuario pueda disponer del software que el necesite.

Los chips ARM posee 27 registros de los cuales 16 son accesibles en cualquier momento, cuatro modos de procesador disponibles:

- USR: modo de usuario
- IRQ: modo de interrupción (con una copia privada de R13 y R14.)
- FIQ: modo de interrupción rápida (copias privadas de R8 a R14.)
- SVC: modo de supervisión (copias privadas de R13 y R14.)

Del modo de usuario no se puede cambiar al modo del procesador dándose un hardware de seguridad y la memoria física es solo accesible por medio de un código propio de la compañía. Debido a que los seis últimos bits del program counter son usados para las banderas de estatus del procesador, este chip tendrá restricciones para direccionar los 16 bits de memoria, o un espacio de dirección de 64 Megabyte. Actualmente hay 8 bits de retención del registro estatus en el PC.

Debido a que una instrucción tiene siempre cuatro bits de largo los dos últimos bits del PC han sido llenados con cero cuando el registro ha sido usado como un PC. Cuando el registro es usado para otras operaciones los últimos bits se los llena de acuerdo al modo de operación que vaya a desarrollar el procesador. (00 – USR, 01 – IRQ, 10 FIQ y 11 - SVC)

Un tercer estado de instrucciones pipeline permite que el chip ejecute instrucciones rápidamente con una cantidad de transistores bastante baja. Por otro lado el pipeline hace una rotación o un cambio libre en todas las instrucciones. Combinando la ejecución de condiciones de todas las instrucciones, luego de las corridas de código sin subcódigos para que no se detengan en los pipeline; pudo ser alcanzada una velocidad de ejecución de instrucciones necesaria para la frecuencia del reloj (Alrededor de 0.6 instrucciones por ciclo de reloj en promedio).

1.1.4 ARM 3

Esta constituido por una macro celda formada por el núcleo de la ARM2 y también se le puso una especializada interfase de coprocesador, el set de registro no es alterado y no hay nuevos modos de procesador. En el chip ARM3 a diferencia de los anteriores se le agregó un chip de cache (4Kbyte, random remplazadle, 4 líneas de palabra, escritura a través de datos mixtos e instrucciones) y se dio una velocidad de reloj mucha más rápida. También se le hicieron ajustes en la interfaz del coprocesador en el chip comprendiendo 15 definiciones del coprocesador para tener un control de la memoria cache y una identificación del chip.

Finalmente una nueva instrucción fue sumada, la instrucción SWP, un registro monótono para el cambio de comandos de memoria útil para arreglos de multiprocesadores.

1.1.5 ARM 4 Y ARM 5

Estos nunca fueron realizados, En el cambio del diseño de los procesadores de Arcon a Armltd, el número del diseño del chip fue cambiado saltándose de esta forma las versiones 4 y 5.

1.1.6 ARM 6

Este es el primer chip el cual estuvo disponible comercialmente de todos los ARMs, este tenía una capacidad de direccionamiento de 32 bits, adicionalmente el procesador tenía 31 registros con seis nuevos modos de procesador.

El interés en las familias ARM fue creciendo, los diseñadores se veían más interesados en RISC y los diseños de ARM eran vistos como una necesidad para obtener un alto desarrollo, baja potencia de consumo y procesadores RISC de bajo costo. Para tener un engrandecimiento silencioso se hizo un acuerdo entre las compañías Arcon, VLSI Technology Inc. y Apple que era una compañía que se había mostrada interesada.

La compañía Arcon RISC Machine se volvió Advance RISC Machine y nació la compañía Advances RISC Machines Ltd. ARM Ltd. fue fundada con la clara misión de seguir con el desarrollo de procesadores ARM y facilitar su uso para el desarrollo de sistemas.

1.1.7 Actualización de la Arquitectura ARM

ARM Ltd. se expandió globalmente por el desarrollo de relaciones con otras compañías alrededor del mundo, desde su lanzamiento ARM se ha relacionado con más fundaciones quienes han licenciado sus diseños y los venden en diferentes mercados.

Desde sus inicios con Acorn, ARM ha trabajado en conjunto con VLSI Technology el cual ha sido su principal socio y el primer fabricante de dispositivos

ARM. En 1993 Sharp Corporation de Japón firmo un tratado para producir y comercializar procesadores ARM y productos asociados.

1.2 FAMILIAS DE LA ARQUITECTURA ARM

1.2.1 Familia ARM7

La familia ARM7 esta formado por un microprocesador RISC de 32 bits, trabaja sobre los 130MIPs, esta familia incorpora el set de instrucciones de 16 bits de la tecnología Thumb entregando un rendimiento de 32 bits con un sistema de 8 o 16 bits.

La familia ARM7 esta compuesta por los siguientes modelos de núcleos procesadores: ARM7TDMI, ARM7TDMI-S, ARM7EJ-S y el ARM720T, cada uno de los cuales tienen que ser desarrollados en base a los diferentes requerimientos de mercado como son:

- Versión sintetizable de la versión del procesador ARM7TDMI
- Núcleo sintetizable con tecnología DSP y Jazelle encargados de mejorar el rendimiento en Java.
- Núcleo con unidad de memoria magnética (MMU) que sirve de apoyo para trabajar con sistemas como: Windows CE, Palm OS, Symbian OS y Linux.

Aplicaciones

- Dispositivos de Audio como: MP3, WMA, AAC players
- En los microteléfonos inalámbricos
- Beepers

Características

- Posee una arquitectura RISC de 32 bits
- Tiene un rendimiento sobre los 130 MIPs con un procesador típico de 0.13um
- Diseño pequeño y tiene un consumo muy bajo de potencia
- Programación de alto nivel, comparable con los microcontroladores de 16 bits
- Amplio sistema operativo compatible con Windows CE, Palm OS, Symbian OS y Linux
- Posee un amplio número de herramientas de desarrollo para seleccionar
- Simulación acorde para los principales entornos de desarrollo EDA
- Excelente depuración de errores para los diseños SoC
- Migración y soporte a través de nuevos procesos tecnológicos.
- Su código es compatible con los procesadores ARM9, ARM9E, ARM10

1.2.2 Familia ARM9

La familia de procesadores ARM9 esta construida basándose en el procesador ARM9TDMI e incorpora los 16 bits de set de instrucciones Thumb, el cual mejora la densidad de código en un 35%. Las familias de la ARM9 comprenden la característica de habilitar el desarrollo para implementar los sistemas de última tecnología, mientras se proporciona un notable ahorro en el área del chip, costos de desarrollo y consumo de potencia. La familia Thumb ARM9 incluye macro celdas del procesador ocultas, cada una de las cuales ha sido desarrollada para requerimiento de diferentes aplicaciones como por ejemplo:

- Doble cache de 16K para aplicaciones en el funcionamiento de Symbian OS, Palm OS, Linux y Windows CE.
- Doble cache de 8K para aplicaciones de funcionamiento Symbian OS, Palm OS, Aplicaciones de Linux y Windows CE.

Aplicaciones

- Es la generación de productos hand-held, como por ejemplo: videophones, comunicaciones portátiles, PDAs.
- Productos de consumo digital, como por ejemplo: set-top boxes, gateways para el hogar, consolas de juegos, MP3, MPEG4 video.
- En la fotografía por ejemplo: desktop printers, cámaras de fotos digitales y en cámaras de video digital.
- En la industria automotriz en el sistema telematic e infotainment².

Características

- Son procesador RISC de 32 bits que poseen set de instrucciones ARM y Thumb.
- Realiza 5 estados de pipeline a 1.1 MIPS/MHz
- Bus de interfaz de 32 bits
- Unidad de memoria magnética (MMU) de apoyo en Windows CE, Symbian OS, Linux, Palm OS
- Instrucciones integradas y cache de datos
- Excelente soporte de depuración para diseños de SoC
- 8 buffer de entrada de escritura

1.2.3 Familia ARM10E

Ofrece una excelente combinación de alto rendimiento y bajo consumo de potencia, la familia ARM10E incluye nuevas características en la arquitectura para entregar el más alto MIPS/MHz de algunos productos ARM. Las características de esta nueva familia son: nuevos modos de ahorro de potencia, tiene una micro arquitectura la cual permite cargar y guardar 64 bits, norma IEEE754 la cual le permite tener compatibilidad al coprocesador punto flotante con operaciones de vectores, fácil integración del sistema, y un completo set de herramientas de desarrollo tanto para hardware como para software.

²Infotainment, sistema de navegación GPS y entretenimiento implementado en los vehículos de última generación Telematic, Integración de las comunicaciones con el caculo automático o proceso de datos, produciendo nuevos servicios para el tratamiento y distribución de la información.

La familia ARM10E esta formada por los procesadores ARM1020E y ARM1022E los cuales tienen una memoria cache de 32K o 16K respectivamente para direcciones de memoria, tiene diferentes superficies de silicón y requerimientos de costo. Ofrece un set de instrucciones para trabajar con DSP, tiene una capacidad de depuramiento integrada en el chip, memoria cache dual de 32 KByte y unidad de memoria magnética (MMU) con soporte para Windows CE, Symbian OS, Linux y PalmOS.

El procesador ARM1020E tiene una memoria dual cache de 16 KByte, tiene un procesador el cual entrega un nuevo nivel de rendimiento, funcionalidad y flexibilidad para nuevas aplicaciones innovadoras con SoC.

Aplicaciones

- Próximas generación de productos portátiles como:
Videophones, comunicaciones portátiles, computadoras, aplicaciones de internet
- Productos digitales de consumo como.
Set-top boxes, gateways para el hogar, consolas de juegos
- Industria de la fotografía como:
Impresoras láser, cámaras fotográficas digitales, cámaras de video digital
- Industria automotriz como:
En el sistema de potencia, sistemas infotainment
- Sistemas de control industrial

1.2.4 Familia ARM11

Basada en la microarquitectura del ARM11 comprende un rango de alto rendimiento de los microprocesadores. Esta familia esta constituida principalmente por cuatro procesadores: ARM1136J(F)-S, ARM1156T2(F)-S, ARM1176JZ(F)-S, cada uno optimizado por los requerimientos específicos de los diferentes mercados y el microprocesador ARM11 MPCore.

Productos de Apoyo. La familia de procesadores ARM 11 esta complementada por la plataforma PrimeXsys y ETK11 las cuales permiten un funcionamiento eficiente y una implementación rápida del diseño ARM11. La plataforma ARM PrimeXsys provee un amplio set de periféricos pre-configurables con un flexible y alto rendimiento de interconexión.

Aplicaciones:

Tabla. 1.1. Aplicaciones familia ARM11

| | ARM 1136J(F)-S | ARM 1156T2(F)-S | ARM 1176JZ(F)-S | ARM 11MPCore |
|---------------------------------|---|---|--|--|
| Industria Automotriz | Infotainment, DVD, navegación | Etapa de potencia | | Infotainment, DVD, navegación y reconocimiento por voz |
| Computadoras | PDA | Impresión, almacenamiento de datos. | PDA | PDA, impresión, industria, servidores. |
| Consumo | TV digital, DVD, PVR, juegos | Cámaras digitales | Set-top box | DTV, IPSTB, DSC, consolas de juegos. |
| Industria | | Sistemas integrados | Terminales EPOS | |
| Networking | En la infraestruc- tura switch y routers | Modem | | Terminales CPE, swithes, routers. |
| Wireless | Teléfonos inteligentes | Estaciones bases. | Procesadores banda base para teléfonos inteligentes. | PDA |

Características

- Arquitectura de set de instrucciones ARMv6
- Extensiones ARM DSP
- SIMD (una instrucción para múltiples datos)
- Posee tecnología ARM TrustZone para fundamentos de seguridad en el chip.
- Núcleo con tecnología Thumb-2 para mejorar el rendimiento, energía eficiente y densidad de código.
- Bajo consumo de potencia:
 - 0.6 mW/MHz (0.13um, 1.2V) incluidos los controladores de la cache
 - Modos de ahorro de energía
 - Manejador de energía inteligente (IEM)
- Alto rendimiento del procesador
 - 8 fase de pipeline entregando un alto nivel de frecuencia
 - Posee retorno a pila
- Alto rendimiento del diseño del sistema de memoria
 - Tamaño de la memoria cache de 4 – 64K
 - Posee dos memoria opcionales con DMA para aplicaciones multimedia
 - Sistema de memoria de 64 bits de alto rendimiento el cual permite acceder de forma más rápida a los datos para procesos multimedia y aplicaciones de redes
 - La arquitectura del sistema de memoria es ARMv6 el cual acelera el OS
- Interfaz del vector interrupción y el modo de latencia de interrupción baja que apresuran la respuesta de interrupción y el rendimiento en tiempo real.
- Coprocesador con vector opcional punto flotante para aplicaciones en el control automotriz o industrial y acelerador de gráficos 3D.
- Los núcleos ARM 11 entregan una sinopsis referente de la metodología ARM lo cual ayuda reducir el tiempo en la generación de proyectos tecnológicos.

1.3 TECNOLOGÍAS ARM.

1.3.1 Thumb.

Es una extensión de la arquitectura ARM la cual la encontramos en aplicaciones de control integrado como: teléfonos celulares, discos duros, módems y beepers, en donde habido un desempeño exitoso de sus recientes generaciones de controladores CISC, mediante esta extensión los diseñadores están viendo la forma para alcanzar el rendimiento y el espacios de memoria de una arquitectura de 32 bits pero sin tener que emigrar a un sistema de 32 bits.

Características

- Excelente densidad de código para un tamaño de memoria y costo mínimo.
- Se obtiene 32 bits de rendimiento de una memoria de 8 o 16 bits con un bus de 8 o 16 bits para sistemas de bajo costo.
- Tiene un tamaño final pequeño para una fácil integración y un costo mínimo del chip.
- Primera industria MIPS/Watt la cual busca obtener una máxima vida de batería y un rendimiento RISC
- Múltiples formas para proveer seguridad.

Thumb es una extensión para la arquitectura ARM de 32 bits, el set de instrucciones Thumb ofrece un subset que es el más usado en las instrucciones ARM de 32 bits, este ha comprendido un ancho de 16 bits de opcode. Durante la ejecución de las instrucciones de 16 bits estas se descomprimen para actuar como instrucciones de 32 bits en tiempo real sin perder el rendimiento.

Los diseñadores pueden usar ambos set de instrucciones el Thum de 16 bits o el ARM de 32 bits por esta razón tienen la facilidad para enfocarse más en una cosa como puede ser el rendimiento o el tamaño del código en un nivel de subrutina como sus aplicaciones requieran.

El núcleo de la extensión Thumb es un procesador ARM estándar con un descompresor Thumb para las instrucciones pipeline. Por consiguiente el diseño alcanza la potencia fundamental de una arquitectura ARM de 32 bits con una excelente densidad de código y trabajando con un sistema de 8 bits. La extensión de arquitectura Thumb tiene mejor densidad de código que la CISC o RISC de 8 y 16 bits.

1.3.2 Thumb-2.

Nos presenta una densidad de código mejorada con un eficiente rendimiento y potencia. Esta tecnología provee un nuevo set de instrucciones de la arquitectura ARM el cual genera un mejor nivel de rendimiento, consumo de energía y ofrece una densidad de código útil para un amplio rango de aplicaciones integradas.

La tecnología thumb-2 se construyó gracias al éxito de la Thumb, el innovador set de instrucciones para núcleos de microprocesadores ARM incrementa la potencia de estos haciéndolos útiles para el desarrollo de sistemas de alto rendimiento.

Esta tecnología es compatible con las existentes ARM y Thumb mientras se extiende las características disponibles del set de instrucciones Thumb-2, esto permite realizar más aplicaciones para beneficiarse de las mejores clases de código de la tecnología Thumb.

Para mejorar el rendimiento de código en la tecnología Thumb-2 se usa 31 por ciento menos de memoria para reducir el consumo del sistema, mientras se suministra sobre el 38 por ciento de rendimiento para mejorar más la densidad de código existente, el cual puede ser utilizado para prolongar la vida de la batería o para enriquecer las características del producto.

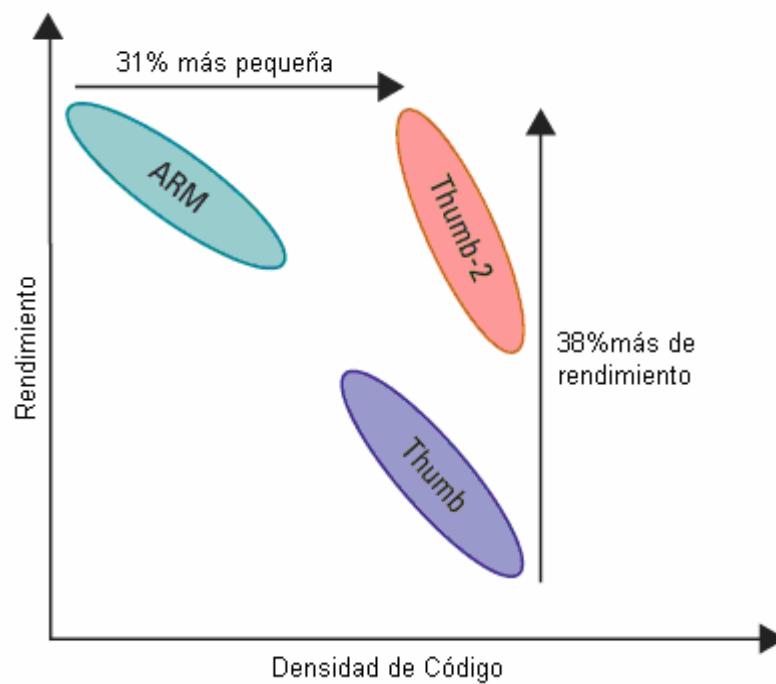


Figura. 1.1. Gráfica comparativa de las tecnologías ARM, Thumb, Thumb-2

1.3.3 Thumb-2EE

Conocida también como Jazelle RCT, fue dada a conocer en el 2005, primero aparece en los procesadores Cortex-A8. La tecnología Thumb-2EE proporciona una pequeña extensión a la Thumb 2, haciendo que el set de instrucciones particularmente se ajuste al código generado durante el tiempo de ejecución.

Las nuevas características provistas por Thumb-2EE incluyen: control de los punteros nulos durante toda la carga y almacenamiento de instrucciones, una instrucción para controlar la finalización de los arreglos y la capacidad para poder incrementar secciones pequeñas de código en el programa comúnmente usado para implementar características de lenguaje de alto nivel como por ejemplo la asignación de espacio de memoria para un nuevo objeto.

1.3.4 Neon

Esta tecnología presenta un extenso set de nuevas instrucciones para futuros procesadores ARM las cuales se enfocan para trabajar con medios de comunicación y aplicaciones con DSP; presenta nuevas herramientas las que pueden ser explotadas a través de un código de alto nivel sin la necesidad de utilizar funciones intrínsecas³ o lenguaje assembler y un hardware de ejecución independiente.

La tecnología NEON es una combinación de un set de instrucciones SIMD (Single Instruction Multiple Data) de 64 y 128 bits que proveen una aceleración estándar para sus diferentes aplicaciones tales como: codificación o decodificación de video, gráficos 3D, decodificación de audio comprimido, procesamiento de imagen, telefonía y síntesis de sonido.

La tecnología NEON es una arquitectura que trabaja con su propio pipeline y registro. Esta tecnología será implementada en la siguiente generación de procesadores ARM y presentará importantes características como control en el acceso de datos, soporte para tipos de datos enteros y punto flotante, un firme acoplamiento para el núcleo ARM y un extenso archivo de registros. La capacidad de poder operar en ambos tipos de datos entero y punto flotante permite realizar un amplio número de aplicaciones.

El firme acoplamiento del núcleo ARM provee un flujo de instrucción sencillo en una memoria unificada formando una plataforma de tarjeta de desarrollo única con una simple herramienta de flujo.

El archivo de registro de NEON con sus distintos parámetros habilita eficientemente la entrega de datos y minimiza el acceso a la memoria mejorando el rendimiento del flujo de los datos.

³ Intrínsecas, son funciones ajenas al programa

Tabla. 1.2. Características y beneficios de la familia NEON

| Características | Beneficios |
|---|---|
| Arquitectura SIMD híbrida de 64 y 128 bits. | Las instrucciones SIMD ⁴ de NEON permiten a los 16 elementos ser procesados paralelamente acelerando de esta manera los medios de comunicación y aplicaciones de procesamiento de señales. |
| Acoplamiento para el núcleo | La integración da una unificación de los parámetros de la memoria, la cual es compartida con el núcleo ARM. |
| Soporte para el acceso de datos | Habilita la carga eficiente del vector de datos en el código compilado. |
| Carga y almacenamiento de la estructura SIMD | Elimina los arreglos de datos que están sobrecargados y optimiza el acceso a la memoria de datos. |
| Archivo de registro. | Instrucciones definidas a través de un buen manejo de datos para una eficiencia máxima del código compilado, permitiendo hacer intercambios entre la longitud del vector de datos y el número de registros disponibles. |
| Soporte para tipos de datos entero, punto fijo y punto flotante | Da la facilidad para que se pueda realizar un amplio número de aplicaciones de voz o compresión de audio. |
| Tamaño de datos de 8, 16, 32 y 64 bits | Garantizan eficientemente el empaquetamiento del vector de datos para obtener un máximo ciclo procesamiento de datos. |
| Archivo de registro independiente | Un archivo de registros permite que algunos resultados intermedios sean guardados internamente decrementando el número de accesos |

⁴ Instrucción SIMD (Single Instruccion Multiple Data), procesa múltiples datos con una sola instrucción
Arquitectura híbrida, compuesta por dos tamaños de instrucciones

| | |
|-------------------------------|---|
| | de datos a la memoria e incrementando el desempeño del procesamiento. |
| Codificación en ARM y Thumb-2 | Asegura un alto rendimiento con una óptima densidad de código. |

1.3.5 TrustZone

Es una tecnología la cual ofrece parámetros de seguridad para la arquitectura ARM especialmente en productos de consumo de seguridad tales como: teléfonos portátiles, PDAs, juegos de video y en el funcionamiento de sistemas operativos tales como: Symbian OS, Linux y Windows CE. La tecnología TRUSTZONE garantiza la implementación fiable de aplicaciones de seguridad y servicios tales como: antivirus, ingreso promedios de usuarios incrementados a través de cuentas móviles, transacciones comerciales y la protecciones de secretos de los usuarios con claves personales.

La tecnología TrustZone esta implementada dentro del mismo núcleo del procesador permitiendo la protección de la memoria del chip y sus periféricos de ataques de otros software. Desde que los elementos de seguridad del sistema son diseñados dentro del núcleo del hardware cualquier propuesta de seguridad que esté fuera del núcleo es negada. De esta forma la seguridad es mantenida como una característica intrínseca en el corazón del dispositivo con un impacto mínimo en el área del núcleo o en el rendimiento, habilitando concesiones para construir cualquier característica adicional de seguridad deseada como: criptografía con el propósito de tener un hardware seguro.

“Monitor de Seguridad”, es un nuevo sistema el cual actúa dentro del núcleo poniendo una barrera por medio de interruptores entre los estados seguros y no seguros. Cuando el monitor cambia el sistema a un estado de seguridad el procesador da niveles de privilegio para ejecutar solo código que sea seguro. Lo cual permitirá posteriormente manejar tareas como: autenticación, manipulación de firmas y procesamiento de transacciones seguras.

La división del código de seguridad y datos dentro del sistema y la separación entre información segura y no segura, han habilitado un código de seguridad y datos para ejecutarse junto a un eficiente sistema operativo. El nivel de seguridad dentro de los sistemas con tecnología TrustZone puede variar desde medidas de seguridad implementadas dentro del núcleo a un primer nivel de sistema de memoria con el propósito de seguir implementando más seguridad que cubrirían las memorias y periféricos.

CAPÍTULO II

ARQUITECTURA ARM

2.1 ESTRUCTURA INTERNA

En la figura. 2. 1 se muestra un diseño genérico de la estructura interna de un procesador ARM, el cual es muy semejante al resto de microcontroladores que existen en el mercado. El sistema On-chip esta constituido por el núcleo ARM, los periféricos y el controlador de interrupciones. Cuando se realiza una interrupción el controlador de interrupciones se encarga de diferenciar de qué tipo es y procede a enviar la correspondiente señales IRQ o FIQ la cual va al núcleo ARM. Estas interrupciones pueden ser hechas por hardware para ser priorizadas.

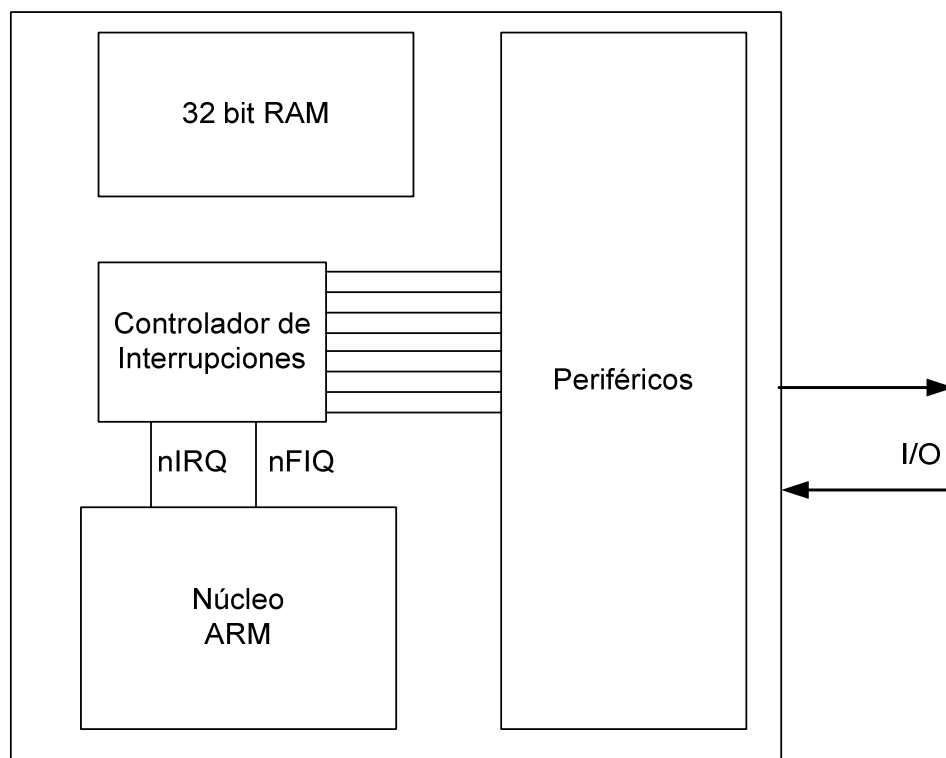


Figura. 2.1. Diagrama de bloques de estructura interna ARM

La organización interna del ARM esta constituida de la siguiente forma:

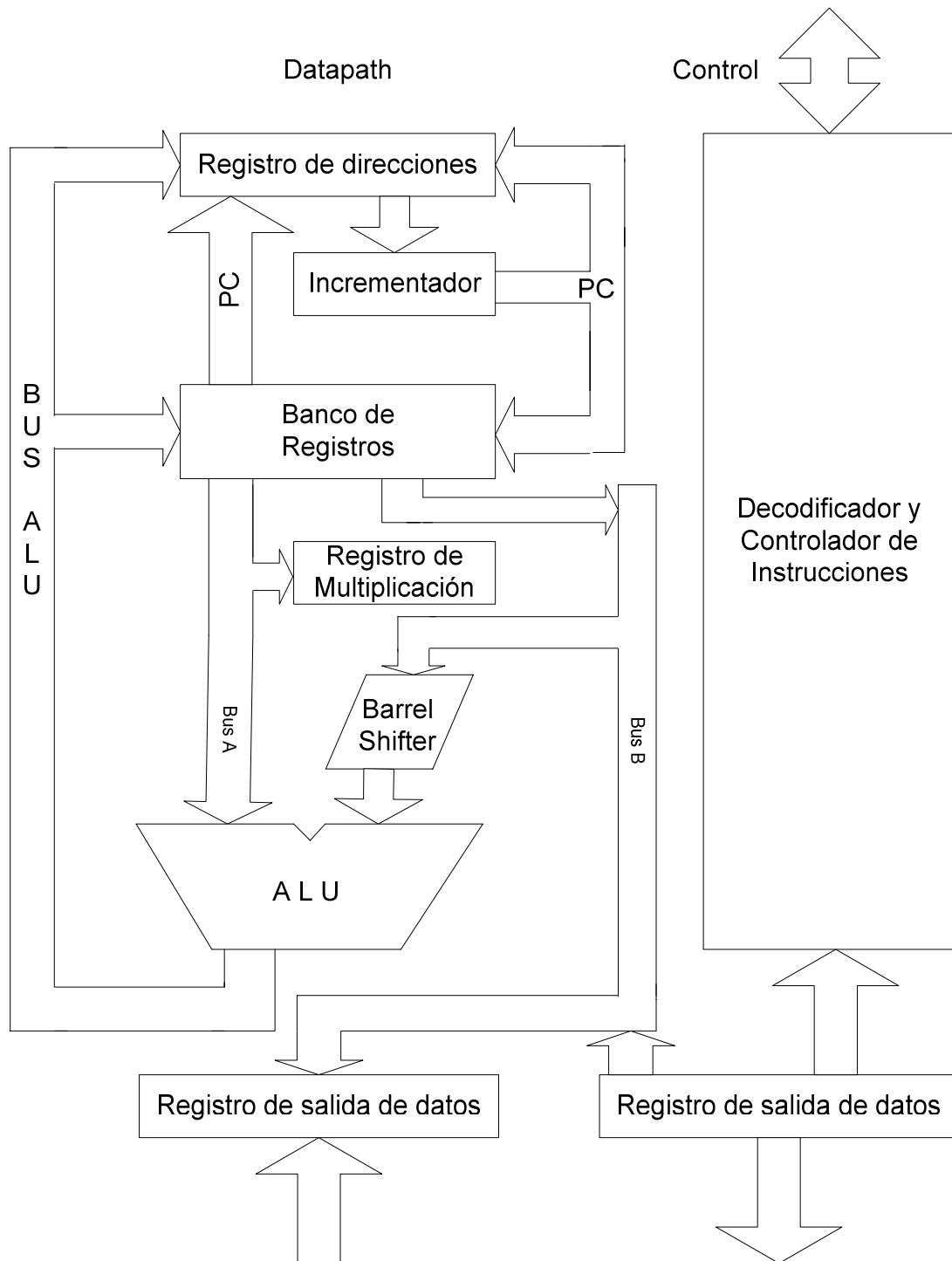


Figura. 2.2. Organización interna del procesador ARM

Esta formado por dos bloques principales el datapath y el decoder (bloque de descodificación).

- Posee un banco de registros desde r0 a r15.
- Dos puertos de lectura que están conectados al bus de datos A o al B.
- Un puerto de escritura.
- Puertos adicionales de lectura o escritura para el contador de programa (program counter) r15.
- *Barrel shifter* el cual permite rotar o mover el segundo operando en cualquier número de bits.
- Unidad aritmética lógica.
- Consta de registros de direcciones, también llamadas direcciones de PC (program counter) o direcciones de operándos.

Funcionamiento

- El registro de datos retiene los datos que van a ser leídos de la memoria o escritos en la memoria.
- El decodificador de instrucciones decodifica el código de instrucciones de la máquina para producir señales de control al datapath.
- En un ciclo simple de proceso de datos, se puede notar que los valores de los datos son leídos en los buses A y B y los resultados de la unidad aritmética lógica son escritos nuevamente dentro del banco de registros.
- El valor del program counter (contador de programa) es incrementado y copiado nuevamente al registro r15, esto permite obtener un adelanto en el tiempo para que se pueda ejecutar otra instrucción.

2.1.1 Pipelining

El pipeline es la parte fundamental en esta arquitectura debido a que incrementa la velocidad de ejecución de las instrucciones, es decir ejecuta más instrucciones en un simple ciclo de reloj.

La arquitectura ARM utiliza 3 etapas para realizar el pipeline.

- *Fetch*: Extrae el código de instrucción desde la memoria dentro de la instrucción pipeline.
- *Decode*: Decodifica la instrucción para obtener las señales de control para el datapath quedando listo para realizar el último paso.
- *Execute*: En esta etapa se trabaja con instrucciones propias del datapath, se produce la lectura de los registros, la variación de estos y los resultados generados por el ALU son escritos nuevamente en el registro.

Los resultados de cada etapa del pipeline son guardados en los registros. La ventaja del pipeline es que el periodo de reloj es mucho más pequeño que trabaja sin pipeline.

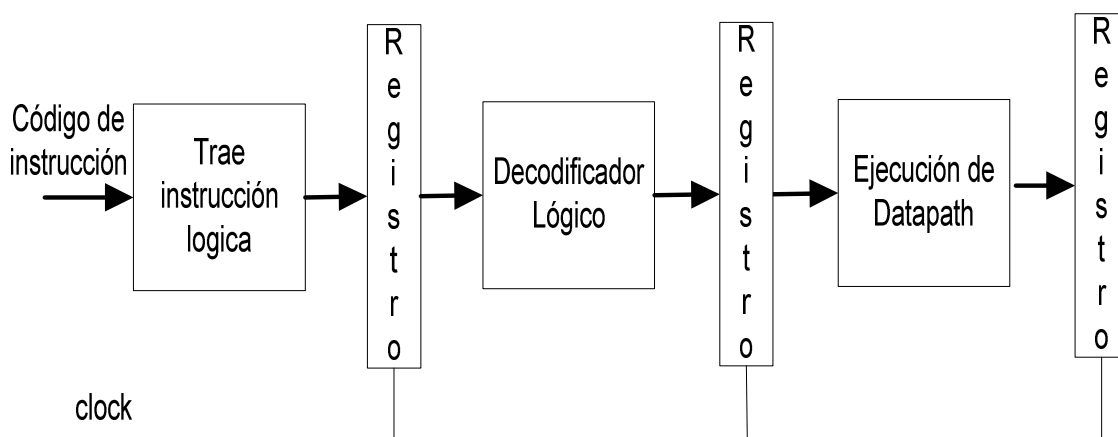


Figura. 2.3. Almacenamiento de cada etapa del pipeline en los registros

En cualquier instante de tiempo, tres instrucciones diferentes pueden ocupar cada uno de las tres etapas del pipeline, esto puede tomar 3 ciclos para completar una instrucción de ciclo simple es decir tiene tres ciclos latentes.

Una vez que el pipeline es realizado, el procesador completa una instrucción de ciclo simple en cada ciclo de reloj. Sin embargo el rendimiento es el de una instrucción por ciclo.

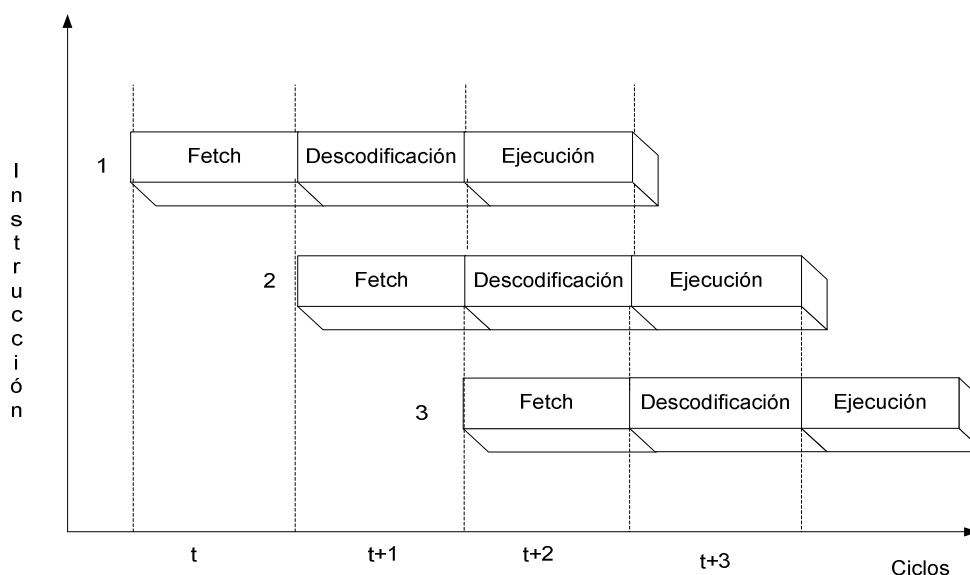


Figura. 2.4. Funcionamiento del Pipeline

Una de las características importantes del pipeline es el de eliminar los saltos a subrutinas logrando así obtener un mejor flujo del código.

2.2 PROCESADOR RISC

Buscando aumentar la velocidad del procesamiento se descubrió en base a experimentos que, con una determinada arquitectura de base, la ejecución de programas compilados directamente con microinstrucciones y residentes en memoria externa al circuito integrado resultaban ser más eficientes, gracias a que el tiempo de acceso de las memorias se fue decrementando conforme se mejoraba su tecnología de encapsulado.

Debido a que se tiene un conjunto de instrucciones simplificado, éstas se pueden implantar por hardware directamente en la CPU, lo cual elimina el microcódigo y la necesidad de decodificar instrucciones complejas.

En investigaciones hechas a mediados de la década de los setentas, con respecto a la frecuencia de utilización de una instrucción en un CISC y al tiempo para su ejecución, se observó lo siguiente:

- Alrededor del 20% de las instrucciones ocupa el 80% del tiempo total de ejecución de un programa.
- Existen secuencias de instrucciones simples que obtienen el mismo resultado que secuencias complejas predeterminadas, pero requieren tiempos de ejecución más cortos.

Las características esenciales de una arquitectura RISC pueden resumirse como sigue:

- Estos microprocesadores siguen tomando como base el esquema moderno de Von Neumann.
- Las instrucciones, aunque con otras características, siguen divididas en tres grupos: transferencia, operaciones y control de flujo.
- Reducción del conjunto de instrucciones a instrucciones básicas simples, con la que puede implantarse todas las operaciones complejas.
- Es una arquitectura del tipo load-store (carga y almacena). Las únicas instrucciones que tienen acceso a la memoria son 'load' y 'store'; registró a registro, con un menor número de acceso a memoria.
- Casi todas las instrucciones pueden ejecutarse dentro de un ciclo de reloj. Con un control implantado por hardware (con un diseño del tipo load-store), casi todas las instrucciones se pueden ejecutar cada ciclo de reloj, base importante para la reorganización de la ejecución de instrucciones es por medio de un compilador.
- El pipeline (ejecución simultánea de varias instrucciones) es la posibilidad de reducir el número de ciclos de máquina necesarios para la ejecución de la instrucción, ya que esta técnica permite que una instrucción pueda empezar a ejecutarse antes de que haya terminado la anterior.

El hecho de que la estructura simple de un procesador RISC conduzca a una notable reducción de la superficie del circuito integrado, se aprovecha con frecuencia para ubicar en el mismo, funciones adicionales como:

- Unidad para el procesamiento aritmético de punto flotante.
- Unidad de administración de memoria.
- Funciones de control de memoria cache.
- Implantación de un conjunto de registros múltiples.

La relativa sencillez de la arquitectura de los procesadores RISC conduce a ciclos de diseño más cortos cuando se desarrollan nuevas versiones, lo que posibilita siempre la aplicación de las más recientes tecnologías de semiconductores. Por ello, los procesadores RISC no solo tienden a ofrecer una capacidad de procesamiento del sistema de 2 a 4 veces mayor, sino que los saltos de capacidad que se producen de generación en generación son mucho mayores que en los CISC.

2.3 TAMAÑO DE INSTRUCCIONES

Las instrucciones ARM son exactamente del tamaño de una palabra (32 bits), y son agrupadas en un máximo de 4 byte. Las instrucciones Thumb son exactamente la mitad de una palabra (16 bits) y son agrupadas en un máximo de 2 byte.

2.4 TIPOS DE DATOS

ARM soporta los siguientes tipos de datos:

| | |
|----------|--|
| Byte | 8 bits |
| Halfword | 16 bits |
| | Los datos halfword deben ser agrupados en grupos de 2 bytes. |
| Word | 32 bits |
| | Los datos halfword deben ser agrupados en grupos de 4 bytes. |

Las operaciones de carga y almacenamiento pueden transferir bytes, halwords y words desde y hacia la memoria.

2.5 MODOS DE PROCESADOR

Existen varios tipos de modos de procesador, los cuales son mostrados en la siguiente tabla 2. 1:

Tabla. 2.1. Modos de procesador ARM

| Modo de Procesador | | | Descripción |
|--------------------|------------|-------|--|
| 1 | User | (usr) | Es el modo de ejecución del programa normal, en este modo se ejecutan la mayoría de las tareas |
| 2 | FIQ | (fiq) | Diseñado para soportar una alta velocidad de transferencia de datos, se activa cuando una interrupción de alta prioridad es ejecutada. |
| 3 | IRQ | (irq) | Se ejecuta cuando una interrupción de baja prioridad (normal) es activada. |
| 4 | Supervisor | (svc) | Modo de protección para el sistema cuando esta operando, produce un reset cuando una instrucción de interrupción de software es ejecutada. |
| 5 | Abort | (abt) | Usado para implementar memoria virtual y/o protección de memoria, cuando se ha producido una violaciones de acceso a memoria. |
| 6 | Indefine | (und) | Usado para manejar instrucciones indefinidas. |
| 7 | System | (sys) | Usado para ejecutar tareas privilegiadas del sistema operativo |

Los cambios de modos de procesador pueden ser hechos por software, causados por una interrupción externa o por ciertas excepciones. La mayoría de los programas se realiza en el modo usuario, los otros modos son conocidos como modos privilegiados y son habilitados por medio de interrupciones, excepciones o por recursos de protección de acceso.

2.6 REGISTROS

2.6.1 Set de registros ARM

El procesador ARM tiene un total de 37 registros, comprendidos de la siguiente forma:

- 30 registros de propósito general
- 6 registros de estado
- 1 que es el program counter

Sin embargo no todos estos registros pueden ser vistos ya que depende del modo del procesador teniendo así quince registros de propósito general (R0-R14), uno o dos registros de estado y el contador de programa. Los registros son ordenados en bancos parcialmente superpuestos, hay un banco de registro diferente para cada modo de procesador.

Tabla. 2.2. Set de registros ARM

| Usuario o sistema | Supervisor | Abort | Indefinidos | Interrupciones | Interrup. Rápidas |
|----------------------------------|-------------------|--------------|--------------------|-----------------------|------------------------------|
| R0 | R0 | R0 | R0 | R0 | R0 |
| R1 | R1 | R1 | R1 | R1 | R1 |
| R2 | R2 | R2 | R2 | R2 | R2 |
| R3 | R3 | R3 | R3 | R3 | R3 |
| R4 | R4 | R4 | R4 | R4 | R4 |
| R5 | R5 | R5 | R5 | R5 | R5 |
| R6 | R6 | R6 | R6 | R6 | R6 |
| R7 | R7 | R7 | R7 | R7 | R7 |
| R8 | R8 | R8 | R8 | R8 | R8_FIQ |
| R9 | R9 | R9 | R9 | R9 | R9_FIQ |
| R10 | R10 | R10 | R10 | R10 | R10_FIQ |
| R11 | R11 | R11 | R11 | R11 | R11_FIQ |

Tabla. 2.2. Set de registros ARM (Continuación)

| | | | | | |
|-----|---------|-----------|-----------|---------|---------|
| R12 | R12 | R12 | R12 | R12 | R12_FIQ |
| R13 | R13_SVC | R13_ABORT | R13_UNDEF | R13_IRQ | R13_FIQ |
| R14 | R14_SVC | R14_ABORT | R14_UNDEF | R14_IRQ | R14_FIQ |
| PC | PC | PC | PC | PC | PC |

| | | | | | |
|------|----------|------------|------------|----------|----------|
| CPSR | CPSR | CPSR | CPSR | CPSR | CPSR |
| | SPSR_SVC | SPSR_ABORT | SPSR_UNDEF | SPSR_IRQ | SPSR_FIQ |

2.6.2 Función de los registros

Los registros del 0-12 van siempre libres para propósitos generales del usuario. Los registros 13 y 14, a pesar que son tomados para propósitos generales también tiene específicos roles:

Registro 13 (también conocido como el stack pointer SP (puntero de pila) es acumulado por cada modo para proveer un puntero de pila privado para cada uno.

Registro 14 (también conocido como registro de vinculo LR (Link Register) es usado para salvar la dirección de la siguiente instrucción después del retorno de una subrutina. El R14 es almacenado por cada modo, también puede ser utilizado como un registro de propósitos generales.

Registro 15 es utilizado específicamente para guardar el program counter (contador de programa). Cuando el R15 es leído, los bits [1:0] son cero y los bits [31:2] contiene el PC. Cuando el R15 es escrito, los bits [1:0] son ignorados y los bits [31:2] son escritos en el PC. Dependiendo de cómo este es usado, el valor del PC puede ser cualquier dirección de la instrucción más n (donde n es 8 para el estado ARM y cuatro para el estado Thumb) o es impredecible.

CPSR es el registro de estado del programa. Este es accesible en todos los modos de procesador y contiene los códigos de condición o banderas, habilita las banderas de interrupciones, y los modos de procesador.

2.6.3 Los registros de almacenamiento FIQ

El modo FIQ esta constituido por los registros que van desde el R8 al R12. Los registros R8_FIQ, R9_FIQ, R10_FIQ, R11_FIQ, R12_FIQ son provistos para permitir un procesamiento de interrupciones rápidas y conservar los valores a través del llamado de interrupciones (es decir que estos registros no necesitan ser restaurados de la memoria).

2.6.4 Registros de estado del programa

La arquitectura ARM consta de un registro de estado del programa actual (CPSR) (Current Program Status Register), más cinco registros de estado del programa guardado (SPSR) (Saved Program Status Registers) para ser usados con excepciones. Los CPSR realizan:

- Guardan la información de las ultima operación realizada por la ALU
- Controla la habilitación y suspensión de las interrupciones
- Pone los modos de operación del procesador.

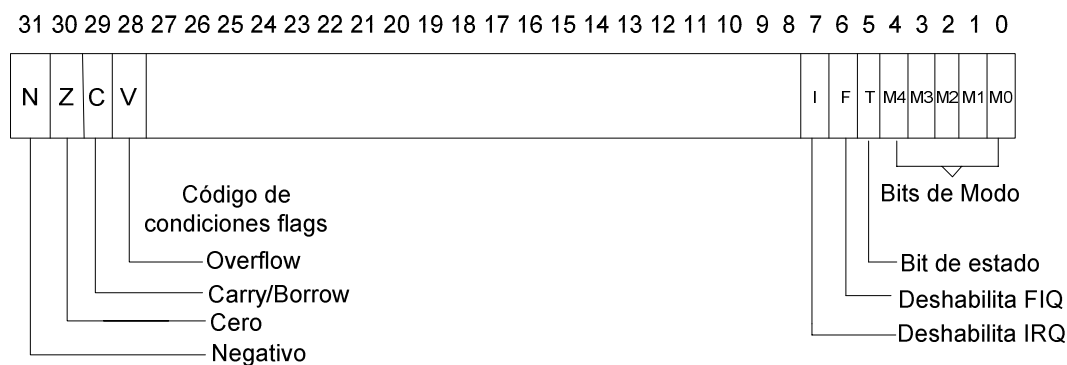


Figura. 2.5. Formato del registro de estado

2.6.5 Banderas de código de condiciones

Los bits N, Z, C y V (negativo, cero, carry y overflow) son colectivamente conocidos como condiciones de banderas. Estas condiciones de banderas en el CPSR pueden ser cambiadas como un resultado de una operación aritmética o lógica en el procesador, y pueden ser usadas por todas las instrucciones ARM para determinar si la instrucción tiene que ser ejecutada, de esta manera todas las instrucciones ARM pueden ser ejecutadas condicionalmente.

2.6.6 Bits de control

Los últimos bits 8 de un PSR (Program status register) (integrando I, F, T y los M [4:0]) son conocidos como los bits de control. Estos cambian cuando una excepción se produce y pueden ser alterados por software solo cuando el procesador está en un modo privilegiado.

Bits que deshabilitan la interrupción son los bits I y F, cuando se fijan estos bits se deshabilitan las interrupciones IRQ y FIQ respectivamente.

Bit de estado: El bit T está constituyendo el bit de estado del procesador, cuando el bit de estado está puesto en 0 esto indica que el procesador está en estado ARM (trabaja con instrucciones de 32 bits), y cuando el bit de estado está en 1 indica que el procesador está en estado Thumb (trabaja con instrucciones de 16 bits).

Bits de modo: Son los bits M4, M3, M2, M1 y M0, estos determinan el modo en el cual el procesador operará. No todas las combinaciones de bits de modo son válidas en la tabla 2.3. se puede ver los únicos que pueden ser usados.

Tabla. 2.3. Bits de modo

| M [4:0] | Modo | Registros accesibles |
|----------------|-------------|---|
| 10000 | User | PC, R14 a R0, CPSR |
| 10001 | FIQ | PC, R14_fiq a R8_fiq, R7 a R0, CPSR, SPSR_fiq |
| 10010 | IRQ | PC, R14_irq a R13_irq, R12 a R0, CPSR, SPSR_fiq |

| Tabla. 2.3. Bits de modo (Continuación) | | |
|---|--------|---|
| 10011 | SVC | PC, R14_svc, R13_svc, R12 a R0, CPSR, PSR_svc |
| 10111 | Abort | PC, R14_abt, R13_abt, R12 a R0, CPSR, PSR_abt |
| 11011 | Undef | PC, R14_und, R13_und, R12 a R0, CPSR, PSR_und |
| 11111 | System | PC, R14 a R0, CPSR |

El modo de usuario y el modo de sistema no tienen un SPSR, ya que ellos no entran en ninguna excepción, por lo tanto no necesita un registro en el cual conserve el CPSR.

2.6.7 Excepciones

Son generadas por fuentes internas o externas para desviar el procesador para manejar un evento, por ejemplo interrupciones generadas externamente o un intento para ejecutar una instrucción indefinida. El estado interno del procesador un momento antes de manipular la excepción debe ser guardado de tal forma que el programa original pueda ser continuado una vez que la excepción haya sido completada.

El procesador ARM soporta siete tipos de eventos y tiene un modo de procesador privilegiado para cada tipo. Los modos de procesamiento de excepciones lista cada tipo de excepción y el modo de procesador usado para procesar esa excepción. Cuando una excepción ocurre, la ejecución es forzada de una dirección de memoria fija correspondiente al tipo de excepción. ARM colectivamente nombra a estas direcciones fijas como *hard vectors*.

Tabla. 2.4. Modo de procesamiento de excepciones

| Tipo de excepción | Modo de Excepción | Dirección de vector |
|-----------------------------------|-------------------|---------------------|
| Reset | Supervisor | 0x00000000 |
| Instrucciones indefinidas | Indefinido | 0x00000004 |
| Interrupciones por software (SWI) | Supervisor | 0x00000008 |

| Tabla. 2.4. Modo de procesamiento de excepciones (Continuación) | | |
|--|-----------------------------|------------|
| Prefetch Abort (instrucción fetch) | Suspender ejecución (Abort) | 0x0000000c |
| Data Abort (Suspender la ejecución de la memoria de acceso de datos) | Suspender ejecución (Abort) | 0x00000010 |
| IRQ (Interrupción) | IRQ | 0x00000018 |
| FIQ (Interrupción rápida) | FIQ | 0x0000001c |

2.7 SET DE INSTRUCCIONES

Existen dos tipos de set de instrucciones:

- El set de instrucciones ARM el cual trabaja como estándar con 32 bits
- El set de instrucciones THUMB el cual es una forma comprimida y trabaja con 16 bits, tiene una densidad de código mucho mejor que la arquitectura CISC, la descompresión de las instrucciones es en forma dinámica a través del pipeline.

2.7.1 Set de instrucciones ARM.

Para el set de instrucciones ARM (ISA) se debe considerar algunos aspectos en detalle, como considerar el formato de algunas instrucciones codificadas y su relación con la instrucción en assembler.

31 30 29 28 27 26 25 24 23

0

| | | | |
|-----------|-------|---|--------------------------------|
| Condición | 1 0 1 | L | 24 bits para la palabra (word) |
|-----------|-------|---|--------------------------------|

Figura. 2. 6. Formato de una instrucción

Como se nota en la figura. 2. 6. los cuatro primeros bits [31:28] son siempre usados para especificar la condición bajo la cual la instrucción es ejecutada. El bit L (bit 24) es activado si es una sección con una instrucción vinculada. Los restantes 24 bits desplazados especifican el destino de la sección. El rango de la sección es de +/- 32 Mbytes.

2.7.1.1 Condiciones de códigos de campo ARM

Tabla. 2.5. Códigos de campo ARM

| Opcode | Mnemónico | Interpretación | Condición de la bandera de estado para la ejecución |
|--------|-----------|--------------------------------|---|
| 0000 | EQ | Igual / Igual a cero | Z set |
| 0001 | NE | Desigual | Z clear |
| 0010 | CS/HS | Carry set | C set |
| 0011 | CC/LO | Carry clear | C clear |
| 0100 | MI | Negativo | N set |
| 0101 | PL | Mas/Positivo o cero | N clear |
| 0110 | VS | Overflow | V set |
| 0111 | VC | No overflow | V clear |
| 1000 | HI | Superior sin signo | C set y Z clear |
| 1001 | LS | Inferior o igual sin signo | C clear o Z set |
| 1010 | GE | Mayor que o igual con signo | N igual V |
| 1011 | LT | Inferior que con signo | N no es igual a V |
| 1100 | GT | Superior que con signo | Z clear y N son iguales a V |
| 1101 | LE | Inferior que o igual con signo | Z set o N no son iguales a V |
| 1110 | AL | Siempre | Cualquiera |
| 1111 | NV | Nunca | Ninguno |

2.7.1.2 Instrucciones de procesamiento de datos

Estas instrucciones son las encargadas de realizar las operaciones aritméticas o lógicas, para esto se usa tres formatos de direcciones: los dos primeros operando tienen un tamaño de 32 bits, el primer y el segundo operando son registros aunque el segundo también puede ser el valor de un resultado anterior.

La tercera dirección es el resultado de la operación de los dos anteriores registros, tiene un tamaño de 32 bits y es guardado en otro registro

2.7.1.3 Instrucciones de procesamiento de datos ARM.

Tabla. 2. 6. Instrucciones del procesamiento de datos

| Opcode [24:21] | Mnemónico | Significado | Efecto |
|-------------------|-----------|------------------------------------|--------------------------------|
| 0000 | AND | Bit lógico AND | $Rd = Rn \text{ AND } Op2$ |
| 0001 | EOR | Bit lógico exclusión OR | $Rd = Rn \text{ EOR } Op2$ |
| 0010 | SUB | Substracción | $Rd = Rn - Op2$ |
| 0011 | RSB | Substracción invertida | $Rd = Op2 - Rn$ |
| 0100 | ADD | Sumar | $Rd = Rn + Op2$ |
| 0101 | ADC | Sumar con acarreo | $Rd = Rn + Op2 + C$ |
| 0110 | SBC | Substracción con acarreo | $Rd = Rn - Op2 + C - 1$ |
| 0111 | RSC | Substracción invertida con acarreo | $Rd = Op2 - Rn + C - 1$ |
| 1000 | TST | Prueba | Scc on $Rn \text{ AND } Op2$ |
| 1001 | TEQ | Equivalencia del test | Scc on $Rn \text{ EOR } Op2$ |
| 1010 | CMP | Comparación | Scc on $Rn - Op2$ |
| 1011 | CMN | Comparación negativa | Scc on $Rn + Op2$ |
| 1100 | ORR | Bit lógico OR | $Rd = Rn \text{ OR } Op2$ |
| 1101 | MOV | Mover | $Rd = Op2$ |
| 1110 | BIC | Borrar bit | $Rd = Rn \text{ AND NOT } Op2$ |
| 1111 | MVN | Mover y negar | $Rd = \text{Not } Op2$ |

2.7.1.4 Instrucciones de transferencia de datos

Estas instrucciones son las encargadas de cargar y almacenar los datos en el correspondiente registro, también son usadas para mover palabras, medias palabras y bytes con signo y sin signo hacia un registro o de uno de ellos. Estas instrucciones también se las usa para cargar el program counter.

Tabla. 2. 7. Instrucciones de transferencia de datos

| Mnemónico | Significado |
|------------------|----------------------------------|
| LDR | Carga palabra |
| LDRH | Carga media palabra |
| LDRSH | Carga media palabra con signo |
| LDRB | Carga byte |
| LDRSB | Carga byte con signo |
| STR | Almacena palabra |
| STRH | Almacena media palabra |
| STRSH | Almacena media palabra con signo |
| STRB | Almacena byte |
| STRSB | Almacena byte con signo |

2.7.1.5 Transferencia de instrucciones en bloque.

Son instrucciones que permiten la carga (LMD) o el almacenamiento (STM) de un banco entero de registros o un subset copiado o restaurado de la memoria con una simple instrucción.

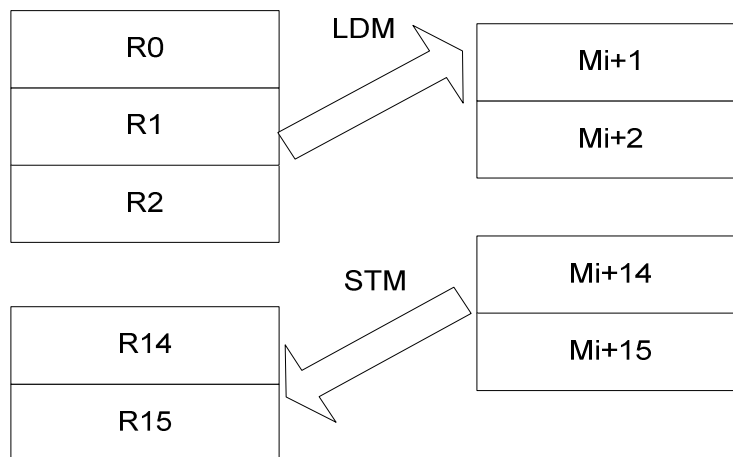


Figura. 2. 7. Instrucciones para transferencia en bloques

2.7.1.6 Instrucción Swap.

Permite realizar el cambio de una palabra la cual se encuentra en un registro a otro registro, esto lo realiza en dos ciclos simples pero con una sola acción.

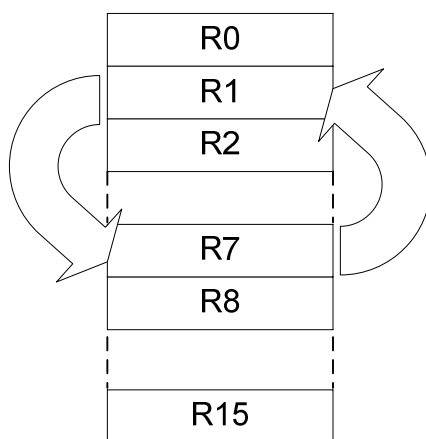


Figura. 2. 8. Instrucción Swap

2.7.1.7 Interrupción de software

Para el manejo interrupciones por software se tiene que trabajar con la instrucción SWI, la cual permite poner al CPU en un modo de inspección. Esta interrupción puede ser utilizada un máximo de 2^{24} veces y es usada principalmente para llamar al sistema operativo.

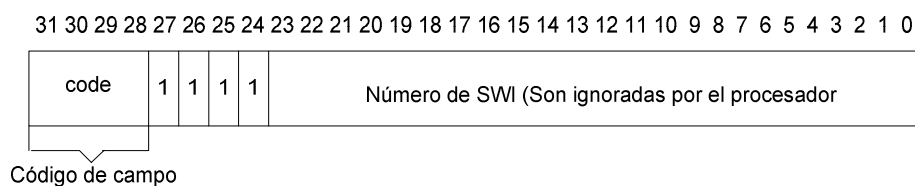


Figura. 2. 9. Interrupción por software

La sintaxis de la interrupción SWI es:

SWI {<cond>} <número SWI >

La interrupción SWI permite inspeccionar el número de SWI que se tiene, para de esta forma poder determinar el código de operación que ha sido requerido, por ejemplo si se tiene SWI1 se pediría que inicie una nueva tarea, SWI2 asignar un espacio de memoria, etc. De esta manera se puede obtener que el sistema operativo pueda realizar diferentes revisiones y pueda trabajar con la misma aplicación de código en cada revisión del sistema operativo.

2.7.1.8 Instrucción branching

Esta instrucción permite realizar saltos a diferentes etiquetas que estén formando el programa y retornar de ellas. Su sintaxis es la siguiente:

B <etiqueta>

2.7.1.9 Branching link (BL)

Guarda la dirección de retorno en el LR (Link Register) usando instrucciones STD/LDM cuando se invoca una subrutina dentro de otra subrutina, restaurando de esta manera el PC (program counter) desde el LR para poder continuar con la ejecución del programa. Su sintaxis es la siguiente.

BL <Subrutina>

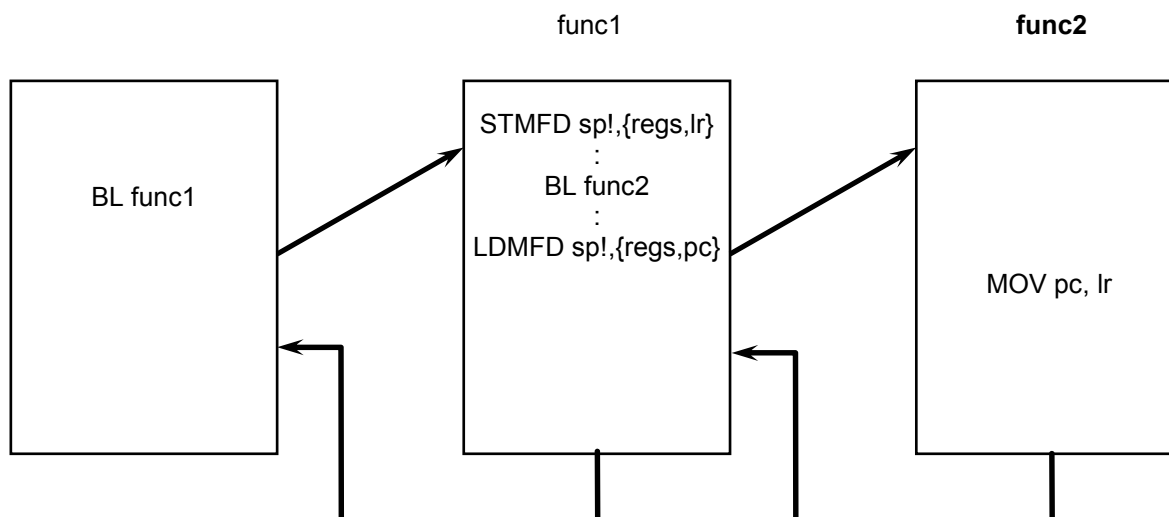


Figura. 2. 10. Instrucción Branching Link

2.7.1.10 Branching exchange (BX)

Esta instrucción permite realizar un intercambio entre el set de instrucciones ARM y el set de instrucciones Thumb, brindando más facilidades para el desempeño de tareas.

2.7.2 Set de instrucciones Thumb

El set de instrucciones Thumb fue diseñado para compiladores ARM en C, se hizo con el fin de observar cual set de instrucciones era usado más a menudo. Este set de instrucciones fue comprimido dentro de un código de 16 bits para tener una mejor densidad de código y un mejor rendimiento desde un espacio de memoria reducido.

Un procesador compatible Thumb es todavía un procesador de 32 bits, pero este tiene la habilidad de ejecutar ambos tipos de código, ya sea ARM o Thumb. Los dos set de instrucciones no pueden ser intercalados juntos en un mismo programa, una forma especial *branch* tiene que ser usada para cambiar el estado del set de instrucciones.

2.8 IMPLEMENTACIÓN DEL CONJUNTO DE INSTRUCCIONES DE LA ARQUITECTURA ARM

2.8.1 StrongARM

El microprocesador Intel StrongARM implementa el conjunto de instrucciones de la arquitectura ARM, con las siguientes características:

- Cache de Datos (Dcache) de 8 kbytes.
- Cache de Instrucciones (Icache) de 16 Kbytes.
- Minicache, para guardar los datos que sean desechados de los caches superiores (cache de datos, cache de instrucciones)
- Buffer de escritura de 8 entradas, entre 1 y 16 bytes cada una
- Buffer de lectura de 4 entradas, con 1, 4 o hasta 8 words

Aquí tenemos 32 entradas que se encuentran separadas, una para el flujo de instrucciones y otra para el flujo de datos. Cada uno de estas 32 entradas puede mapear segmentos, páginas grandes o páginas pequeñas de la memoria. El cache de datos y el cache de instrucciones están implementados como bloques de 32 bytes.

El minicache tiene 16 entradas y es asociado en 2 vías, implementando un algoritmo LRU (Last recent used). El buffer de lectura permite hacer un prefetch de datos importantes a través de software, previniendo esperas de pipeline durante las lecturas a memoria. El buffer de lectura tiene cuatro entradas, y cada entrada puede contener 1, 4 u 8 palabras (words). El buffer de escritura aumenta el desempeño global del sistema al fingir como buffer entre la frecuencia de reloj del CPU y el bus de la memoria cuando los datos comienzan a ser escritos del CPU a la memoria. El buffer de escritura es de 8 entradas y puede contener desde 1 hasta 16 bytes cada uno.

El controlador rutea todas las fuentes de interrupción al CPU, sin importar si son del tipo FIQ o IRQ. IRQ tiene una prioridad menor, puede ser interrumpido

por FIQ. FIQ es único para la arquitectura ARM y permite proveer de un servicio rápido para fuentes de interrupción determinadas por el usuario. Hay dos niveles en el servicio de interrupciones. El primer nivel alerta al usuario o al sistema operativo acerca de la interrupción y cual fue el modulo que lo produjo. El segundo nivel provee información acerca de cual es el evento específico en el módulo que causó una interrupción.

2.8.2 nnARM

2.8.2.1 Opencores

En los últimos años, el software producido en el esquema de código abierto (open source) ha demostrado poseer calidad, gran flexibilidad y tiempos de desarrollo cortos. Ejemplo de lo anterior, es el sistema operativo GNU/Linux, conformado por Linux como kernel y múltiples herramientas GNU para hacer un sistema operativo completo. El grupo opencores piensa que esto puede aplicarse al diseño e implementaciones de cores (núcleos). Uno de los proyectos de opencores es el liderado por ShengYu Shen, de NUDT. El proyecto nnARM (nnARM is not ARM) tiene como objetivo crear un procesador incrustado sintetizado de alta eficiencia que de soporte al conjunto de instrucciones de la arquitectura ARM.

2.8.2.2 Organización

El sistema esta organizado de la siguiente forma:

2.8.2.2.1 Cache

El procesador posee un cache de datos y uno de instrucciones. Están organizados de las siguientes maneras:

2.8.2.2.2 Cache de instrucciones

El cache de instrucciones tiene 256 bytes. Debido a que el cache es grande, el campo tag también será grande, por lo que para combinarlo con la dirección actual de la entrada es necesario utilizar lógica combinacional. El cache de instrucciones tiene 4 secciones, cada sección contiene 4 líneas y cada línea posee 4 palabras. La dirección [5:4] selecciona la sección, entonces el controlador del cache compara [31:6] con cada campo tag de las cuatro líneas en esta sección, si se encuentra la dirección deseada entonces usa la dirección [3:2] para seleccionar la palabra correspondiente a esta línea. Si ningún tag coincide con [31:6] entonces pone la señal de espera en uno para detener al que hizo la petición y va a la memoria para traer el bloque necesario.

2.8.2.2.3 Cache de datos

El cache de datos tiene 256 bytes. El cache tiene 4 secciones, cada sección tiene 4 líneas, cada línea contiene 4 palabras. La dirección [5:4] escoge la sección, entonces el controlador de cache compara [31:6] con cada campo tag de las 4 líneas en esta sección. Si encuentra la dirección deseada entonces usa [3:2] para seleccionar la palabra correspondiente en esta línea. Si ocurre un cache miss, el controlador determina si existe o no una línea en blanco en esta sección. Si la hay, va a la memoria para traer el bloque deseado y lo pone en la línea en blanco. En otro caso, ve si hay una línea que no este siendo ocupada en este momento, en caso afirmativo trae los datos necesarios de la memoria a este lugar. Si no la hay, traemos los datos necesarios de la memoria y lo ponemos en una línea seleccionada al azar.

2.8.2.2.4 Prefetch de instrucciones

Este es un buffer que contiene 8 entradas, cada una puede contener una instrucción de 32 bits. El buffer de prefetch es particionado en 2: Las primeras cuatro instrucciones y las cuatro últimas. Cuando uno esta utilizando la primera mitad, el circuito lógico del prefetch pedirá al cache de instrucciones las otras

cuatro y viceversa. Si la dirección solicitada no está en ninguna de las mitades, es lanzada una señal de espera al solicitante y se va al cache para traer el bloque de cache adecuado.

2.8.2.2.5 Modos de memoria

El byte menos significativo está en la dirección más baja y la dirección de una palabra (word) es la dirección de su byte menos significativo.

2.8.2.2.6 Ancho del bus

En la arquitectura ARM, antes de la versión 3 hay dos modos de ancho del bus de direcciones: 26 y 32 bits. Actualmente si se tiene un ARM con arquitectura versión 3 se tiene soporte para modos de 26 y 32 bits, por compatibilidad hacia atrás. A partir de la versión 4 se utilizan 32 bits y puede implementarse el soporte a 26 bits de manera opcional. nnARM únicamente da soporte a 32 bits.

2.9 APLICACIONES DE LA ARQUITECTURA ARM EN EL DESARROLLO DE TECNOLOGÍA

ARM provee en todos sus desarrollos tecnológicos la propiedad intelectual (IP) en la forma en la que diseña el núcleo del procesador, diseño de los sistemas SoC y la cache, estándares de productos de aplicaciones específicas (ASSPs), en todo lo que tenga que ver con el desarrollo de herramientas y software relacionados, ARM proporciona todo lo que una persona necesite para crear productos innovadores basados en componentes de la industria los cuales son compatibles con las siguientes generaciones

2.9.1 Home solutions

La arquitectura ARM brinda los fundamentos tecnológicos necesarios para que los clientes accedan y disfruten de diferentes tipos de contenidos mientras estén en el hogar como son: imágenes de video digital las cuales pueden ser

transmitidas a todas las partes del hogar a través de pequeñas redes de distribución.

ARM provee soluciones para las siguientes aplicaciones específicas en lo que tiene que ver en productos para el hogar: gaming, cámaras digitales, TV digital, digital Set-Top-Box.

2.9.1.1 Gaming

Los núcleos ARM son usados como la parte fundamental en el procesador y controladores en las consolas de juego las cuales pueden ser portátiles o hechas para el hogar. Las aplicaciones móviles, es decir los juegos portátiles requieren una operación extremadamente eficiente debido a que están limitadas por el tiempo que dure la batería, ARM se ha destacado en este mercado debido a que provee un excelente ahorro de energía por el tamaño de su núcleo de silicón y además por trabajar en miliwatts/megahertz. También en las aplicaciones para el hogar ha tenido un gran desarrollo como en los juegos de video los cuales tiene un sin número de periféricos inteligentes que requieren un alto consumo de energía.

2.9.1.2 TV digital

La TV digital es clasificada como una típica televisión pero con afinador digital integrada. Otra categoría que esta comenzando a ganar impulso es la iDTV, esta es una DTV integrada la cual incluirá alguna de las características encontradas hoy en día en STB. Con estos niveles de integraron es posible tener múltiples núcleos de CPU en un simple diseño, especialmente si son actualizados.

Con cuatro opciones de procesadores los cuales los encontramos solo en la familia ARM11, la arquitectura ARM ofrece la mejor opción para diseñadores que buscan el mejor rendimiento del sistema requeridos por el mercado de la televisión digital, incluyendo iDTV y HDTV.

2.9.1.3 Cámaras digitales

El mercado correspondiente a la imagen representa importantes requerimientos para los núcleos de los procesadores. Las cámaras digitales son las que más rápido crecen en el mercado año tras año, como el volumen incrementa nuevas características deben ser hechas para mejorar el producto a un bajo costo ya que la competencia se intensifica.

Los productos de imagen como: impresoras y cámaras digitales han incrementado su producción en todo el mundo en más del 20% anualmente. Las cámaras de fotografía son trasladadas desde películas de cinta a tecnología digital, y últimamente se ha incluido el almacenamiento de fotografías y audio, resultando esto una demanda para la nueva generación de productos. ARM es el que gobierna en el mercado del diseño de aplicaciones de imagen.

2.9.1.4 Set-Top-Box digitales

El mercado de los set-top box tiene un gran desarrollo en sistemas de entrega de información ya sea de manera terrestre, satelital, por cable o Internet. Los requerimientos de rendimiento son incrementados en orden para mejorar de acuerdo a las exigencias del consumidor. Hoy en día podemos encontrar audio, video y sistemas de transporte de datos integrados dentro de un solo chip, esta integración ha sido extendida para incluir más funciones como: redes, soportes de discos duros, demodulación y funciones de corrección de error en los sistemas.

ARM esta viendo crecientes éxitos como un microprocesador central en el mercado de los set-top box digitales, para logra alcanzar éxito en este mercado se debe tomar en cuenta el ofrecer un buen rendimiento mientras se soporta niveles muy altos de integración y uso de memoria.

2.9.2 Aplicaciones móviles

Los dispositivos móviles son aplicaciones en su gran mayoría diseñadas con tecnología ARM, los cuales funcionan gracias a una poderosa batería. El mercado

de móviles comprende una amplia variedad de comunicaciones móviles y dispositivos de cómputo portátiles, cada uno con sus propias características necesidades. Entre las principales aplicaciones móviles que provee ARM tenemos las siguientes: teléfonos de voz, asistentes digitales personales, teléfonos inteligentes, media players personales, sistemas de posicionamiento geográfico personal (GPS).

2.9.2.1 Teléfonos de voz

Los teléfonos de voz son teléfonos móviles que se caracterizan por ser agregados a nuevos subscriptores. Típicamente estos son de muy bajo costo y poseen características básicas, como la pantalla que todavía la tienen en escala de grises.

2.9.2.2 Asistentes personales digitales

Los asistentes personales digitales (PDA's) son computadoras móviles cuyo funcionamiento se basa en un sistema operativo abierto, estas son usadas de dos maneras, la primera es por medio de un dispositivo muy semejante a un esfero y la otra por medio de una entrada de teclado.

2.9.2.3 Teléfonos inteligentes

Los teléfonos inteligentes son teléfonos móviles que están designados para actualizar a los subscriptores. Este producto incluye un procesador con una aplicación y un sistema operativo en funcionamiento. Estos teléfonos son más costosos debido a que presenta características más avanzadas. ARM es la única arquitectura que tiene el apoyo de los más grandes vendedores de sistemas operativos como: Symbian, Microsoft, Linux, Palm, SavaJE.

2.9.2.4 Media players personales

Los reproductores de música digitales portátiles han roto el límite de ser un producto lujoso para pasar a convertirse en un dispositivo común con el cual cada persona puede escuchar su colección de música en cualquier momento y en cualquier lugar. ARM es el corazón de los reproductores de música digitales los cuales revolucionaron el lanzamiento de los núcleos RISC de 32 bits.

Después de los reproductores de música digital aparecieron los media players personales (PMP's) los que tenían muchas más aplicaciones como soporte para música, fotos y video. ARM también está en el núcleo de estos dispositivos logrando que estos tengan un consumo bajo de potencia y tenga el soporte de toda la comunidad ARM.

2.9.2.5 Sistemas de posicionamiento geográfico personal

Los sistemas GPS están conformados básicamente por un receptor GPS, un procesador y una pantalla. Estos dispositivos son usados típicamente para navegación personal razón por la cual han sido implementados como accesorios en los vehículos.

2.9.3 Sistemas integrados

2.9.3.1 Estructura automotriz y sistemas de utilidad

El confort del pasajero y los sistemas de instrumentación han sido lo más importante en esta categoría como por ejemplo: el controlador central del vehículo, tablero de instrumentos, el control de temperatura, manejadores de las ventanas automáticas, y entradas de control remoto. Más del 80% de la parte electrónica del vehículo está manejada por un microcontrolador hoy en día.

ARM provee un núcleo estándar a partir del cual se ha construido una base de software para futuros diseños de controladores. Para los módulos de la estructura y los sensores inteligentes del auto, el núcleo ARM7TDMI puede proveer similar

rendimiento que la tradicional arquitectura de 8 bits, pero con una frecuencia de reloj mucho más baja, reduciendo así la emisión de la disipación de potencia.

2.9.3.2 Sistema powertrain de vehículos

Powertrain es el principal campo donde el uso de microcontroladores de 32 bits es fundamental y utilizado al máximo en los vehículos hoy en día, debido a la integración de las funciones del motor y del control de transmisión que se ha estado llevando a cabo.

2.9.3.3 Sistema de seguridad automotriz

Los sistemas de seguridad abarcan un amplio campo de acción que va desde los sistemas ABS hasta los sistemas de seguridad anticollisiones. ARM provee una amplia variedad de núcleos de procesadores con código compatible que ofrecen soluciones óptimas para todos los sistemas. Basándose en los requerimientos de frenado y sistemas anticollisiones, los últimos núcleos ARM están compuestos por la más reciente tecnología para aplicaciones críticas de seguridad como son: soporte para detección o corrección de condiciones erróneas, protección de memoria, etc.

2.9.3.4 Carros telemáticos y multimedia

El éxito de la arquitectura ARM en aplicaciones inalámbricas ha sido la base necesaria para la construcción de productos telemáticos o productos de información y entretenimiento en la industria automotriz.

Los sistemas telemáticos suministran al conductor de servicios de información a través de la integración entre comunicaciones inalámbricas y sistemas de navegación. Los sistemas infotainment son la unión del sistema telemático con el sistema de entretenimiento del vehículo dando como resultado un componente multimedia como son audio y video.

2.9.3.5 Tarjetas inteligentes

La seguridad hoy en día juega un papel fundamental en lo correspondiente a redes de comunicación. Las tarjetas inteligentes están en el centro del sistema de diferentes aplicaciones como por ejemplo: teléfonos móviles, en televisión pagada, redes de comunicación, transferencia de datos y tarjetas de identificación.

Aparte de las aplicaciones de seguridad, confidencialidad e integración de los productos que son vendidos, estos requieren ser protegidos de falsificaciones y tener certificaciones de seguridad, razón por la cual ARM ha implementado los sistemas de seguridad en el centro del núcleo del dispositivo.

2.9.4 Aplicaciones en la empresa

Las aplicaciones empresariales son aquellas que nos permiten captar, guardar, transferir y alterar datos, son dispositivos los cuales usamos en nuestra vida diaria ya sea en nuestro trabajo o en nuestro hogar como por ejemplo: PC, los periféricos de un PC e impresoras. Estos dispositivos permiten que estemos comunicados y hacen nuestras actividades diarias más fáciles. Entre los principales productos tenemos: redes de comunicación, redes inalámbricas, redes de almacenamiento, redes para el hogar, impresoras, y VoIP.

2.9.4.1 Redes empresariales e infraestructura

El buen funcionamiento de las redes como: las de infraestructuras móviles, LAN de empresas, y redes de acceso depende significativamente de la energía que consuman, escalabilidad y fiabilidad. Estas aplicaciones han sido utilizadas principalmente por su flexibilidad, por ser fácilmente programables y su gran rendimiento.

Los núcleos RISC generados por ARM cumplen correctamente con las propiedades flexibilidad, escalabilidad, y rendimiento que son necesarias para las aplicaciones en la empresa y en la infraestructura.

2.9.4.2 Redes de almacenamiento

Diferentes topologías de redes como por ejemplo: las redes de área de almacenamiento (SAN) o la red de almacenamiento adjunto (NAS), son las que están siendo empleados para realizar la transferencia de datos entre redes LAN y dispositivos de almacenamiento masivo, logrando obtenerse una mejor velocidad de transferencia de datos y más facilidad de conectar dispositivos de almacenamiento al usuario final.

ARM provee dispositivos los cuales son muy utilizados para la realización de redes de almacenamiento, estos pueden ir desde los últimos dispositivos salidos al mercado para consumidores los cuales trabajan con redes SAN o NAS hasta la más reciente tecnología para transmitir datos por fibra.

2.9.4.3 Redes para el hogar

Hay un gran número de chips basados en ARM los cuales están formando parte de diferentes dispositivos como: routers, cable módems, módems DSL y gateways, para los cuales ARM ofrece software de aplicación en busca de mejorar su funcionamiento. Las principales ventajas de ARM en este mercado tan competitivo son: la rápida comercialización de sus productos, facilidad de integración, chips constituidos de silicón y software de desarrollo.

2.9.4.4 Impresoras

ARM se ha convertido en una de las principales opciones para aplicaciones de impresión, ha sabido afrontar los altos requerimientos del consumidor en la calidad de imagen y en el rendimiento del producto, ya sea para impresoras láser o impresoras de tinta.

2.9.4.5 VoIP

VoIP (Voice Internet Protocol) es la facultad para comunicarse utilizando la voz por medio del Internet, con más características y un costo más conveniente que la telefonía tradicional.

El soporte técnico y el uso de VoIP esta dentro de una gran variedad de productos los cuales van desde los dispositivos que conforman las tarjetas de teléfono hasta los teléfonos de escritorio. ARM tiene un amplio número de núcleos de procesadores que poseen las características necesarias para poder desarrollarse dentro de este campo con un buen rendimiento.

CAPÍTULO III

CIRCUITOS INTEGRADOS CON ARQUITECTURA ARM

2.10 ESTRUCTURA INTERNA

3.1.1 Digi

NetSilicon NS7520

El NetSilicon NS7520 es un microprocesador eficaz, tiene un alto rendimiento y es altamente integrable entre otras propiedades, esta conformado por 177 pines y posee un encapsulado de tipo PFBGA, se lo utiliza principalmente en dispositivos que actúan en diseño de redes o para dispositivos de monitoreo remoto. Este dispositivo forma parte de la familia NET+ARM de los sistemas on-chip para sistemas integrados. Esta constituido por un puerto Ethernet 10/100 y dos puertos seriales independiente cada uno de los cuales puede ser usado en los modos UART, HDLC o SPI. El CPU del NS7520 esta construido por un núcleo procesador ARM7TDMI RISC de 32 bits el cual trabaja a 36, 46, o 55MHz, aparte posee un suficiente número de periféricos y controladores de memoria para diferentes propósitos como:

- Conectar el dispositivo a varios tipos de memoria como: flash, SDRAM, EEPROM entre otras
- Timers programables
- Controlador DMA
- Módulos de expansión
- 16 pines de I/O para propósitos generales.

NetSilicon NET+50

Es un microprocesador rentable, esta constituido por un núcleo ARMTDMI de 32 bits y trabaja a 44 MHz, posee un encapsulado de tipo BGA (Ball – Gris Array) o PQFP (Plastic Quad Flat Pack), es usado principalmente en dispositivos que utilizan un gran ancho de banda especialmente cuando se trabaja con redes inteligentes. Este microprocesador esta basado en la arquitectura ARM, esta compuesto por un puerto MAC para Ethernet, controlador DMA y un controlador de memoria al cual puede ser acoplado a cualquier memoria utilizada en estos días.

NetSilicon NS9750

Esta constituido por el núcleo ARM 926EJ-S que es uno de los más poderos de la familia ARM9, posee un código de instrucciones para DSP y para Java. El NS9750 trabaja sobre los 200 MHz y esta constituido por una amplia variedad de periféricos para la industria como: USB, PCI, I²C, puertos seriales y un controlador de LCD de alto rendimiento. Se lo utiliza principalmente en equipos electrónicos integrado, o en dispositivos que están formando redes de comunicación.

NetSilicon NS9360

Este chip tiene las mismas características del NS9750, pero a diferencia del anterior este trabaja a 177 MHz, esta conformado por 388 pines y posee un encapsulado de tipo BGA, los periféricos que tiene a disposición son: USBd, USBh, I²C, puertos seriales y el controlador de LCD. Se lo puede utilizar en dispositivos de redes de comunicación y en equipo electrónico integrado.

NetSilicon NS9775

Este microprocesador trabaja con 32 bits a una velocidad de 200 MHz, posee un alto rendimiento y su núcleo esta formado por el ARM 926EJ-S. El NetSilicon NS9775 tiene cuatro canales de video independientes, cada uno tiene hardware de descompresión JBIG e interfaz de procesador láser. Además esta constituido

por los siguientes puertos de entrada salida como son: Ethernet 10/100, USB, PCI, puertos seriales y un controlador de LCD. Sus aplicaciones están dadas principalmente en las impresoras y en la automatización de oficinas.

NetSilicon NET+40

Es un microprocesador de 32 bits altamente integrable, fue diseñado para ser usado en dispositivos de conexiones de redes de computadoras y aplicaciones de Internet. Esta formado por el núcleo ARM7TDMI, consta de un puerto para Ethernet, controlador DMA y un sofisticado controlador de memoria que lo hace compatible con las más usadas

3.1.2Zarlink Semiconductor

ARM610

Es un microprocesador de 32 bits de propósitos generales con 4KByte de cache, buffer de escritura y una unidad de memoria magnética (MMU) que se encuentran dentro de un simple chip, este circuito integrado esta compuesto por 144 pines en un encapsulado de tipo TQFP (Thin Quad Flat Pack). Este microprocesador ofrece un alto rendimiento RISC y su diseño totalmente estático permite asegurar un consumo de potencia mínimo lo que lo hace ideal para sistemas portátiles. Las principales características que posee son:

- Procesador RISC de 32 BITS el cual trabaja a una velocidad de 25 MIPS a 33 MHz.
- Respuesta a las interrupciones en microsegundos, lo que lo hace muy útil para aplicaciones en tiempo real.
- Posee unidad de memoria virtual (MMU)
- Soporte para trabajar con lenguaje de alto nivel
- 4 Kbytes de instrucciones y de cache de datos

El ARM610 es usado idealmente para aplicaciones que requieran un funcionamiento RISC desde un procesador compacto y eficiente entre las que tenemos:

- Dispositivos de computadoras personales como por ejemplo PDAs
- Sistemas de control de alto rendimiento en tiempo real
- Telecomunicaciones portátiles
- Equipos de comunicación de datos
- Productos de consumo
- En la industria automotriz

ARM60

Es un microprocesador RISC de 32 bits para propósitos generales con un bajo consumo de potencia, es una implementación de la familia ARM6, este dispositivo es muy aconsejable para aplicaciones en las cuales se deba tener mucho cuidado con la potencia, esta compuesto por 100 pines en un encapsulado tipo QFP (Quad Flat Pack). Las principales características son:

- Procesador RISC de 32 bits, el cual puede trabajar a 21 MIPS a 30 MHz
- Bus de datos de 32 bits
- Bus de direcciones de 32 bits
- Un bajo consumo de potencia 1.5mA/MHz
- Respuesta a las interrupciones inmediata, para aplicaciones de tiempo real
- Soporte para sistema de memoria virtual
- Puede trabajar con lenguaje de alto nivel
- Posee un set de instrucciones simple pero muy fuerte

Entre las principales aplicaciones tenemos:

- Telecomms: control de terminales GPS
- Datacomms: conversión de protocolos
- Computación portátil: computadoras Palmtop

- Instrumentación portátil: unidades de adquisición de datos
- Industria automotriz: unidad magnética del motor

3.1.3 Actel

CoreMP7

Es un microprocesador el cual pertenece a la familia ARM7 de 32 bits y es utilizado para propósitos generales, ofrece un alto rendimiento y un bajo consumo de potencia. El CoreMP7 incluye periféricos como: controlador de memoria, controlador de interrupciones, timers, interfaz serial y buffer de entrada/salida.

Entre las características que presenta este procesador tenemos las siguientes:

- Arquitectura RISC (ARMv4T) de 16 o 32 bits
- Set de instrucciones ARM de 32 bits
- Set de instrucciones Thumb de 16 bits
- Bus de interfaz unificado de 32 bits
- 3 estados de pipeline
- 32 bits de unidad aritmética lógica
- 32 bits de rango de direccionamiento de memoria
- Todas las entradas y salidas son de libre acceso por el usuario

Entre sus principales aplicaciones tenemos.

- Dispositivos de audio personales: MP3, WMA y AAC players
- Asistentes digitales personales
- Teléfonos inalámbricos
- Beepers
- Cámaras digitales

- FBGA

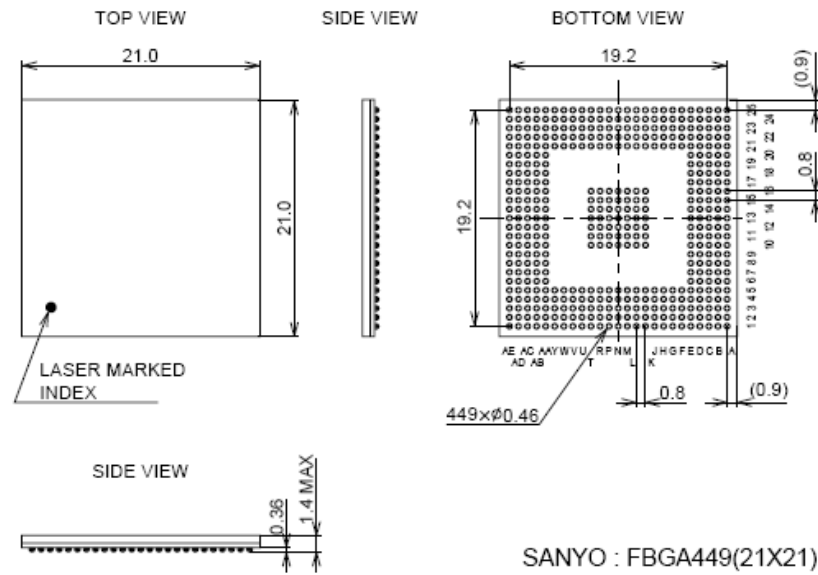


Figura 3.3. Encapsulado FBGA

- TFBGA

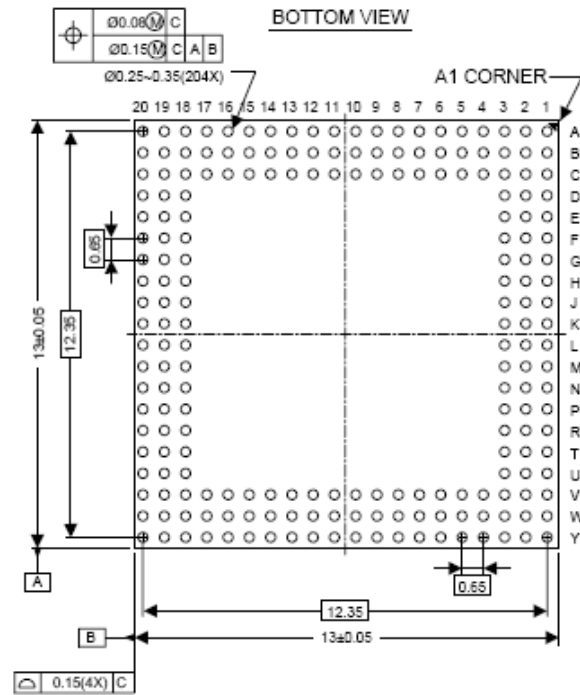


Figura 3.4. Encapsulado TFBGA

3.2.1 Intel

IXC1100

Este procesador combina el alto desempeño y el bajo consumo de potencia del núcleo Intel XScale con una serie de interconexiones y mejoras en el rendimiento del Intel IXC1100. Entre sus principales características tenemos:

- El código de la arquitectura ARMv.5TE es compatible con procesadores Intel StrongARM.
- Posee 492 pines en un encapsulado tipo PBGA de (35mm)
- Posee 32 KB de cache de instrucciones y 32 KB de cache de datos.
- Trabaja con tecnología RISC.
- Tiene dos puertos para Ethernet de 10/100
- Tiene un rendimiento de 32 BITS, con un controlador SDRAM integrado
- Posee controladores DMA para PCI y Ethernet
- 16 pines de I/O
- Puerto de depuración e interfaz estándar JTAG
- Soporte para trabajar con VxWorks, Linux, QNX y OSE

3.2.2 Atmel

AT91M42800A

- 8 Kbytes de SRAM on-chip
- Esta disponible en chips de 144 pines en encapsulado LQFP y en 144 pines en encapsulado BGA.
- Interfaz de bus externa
- Un canal para Timer/Counter
- 2 USART
- 2 interfaces SPI maestro esclavo

- 3 sistemas de Timers
- Un controlador magnético de potencia

AT91M55800A

- Presenta 8 Kbytes de SRAM on-chip
- Esta constituido por chips de 176 pines con encapsulado LQFP y BGA
- Un bus de interfaz externa
- Un canal para el Timer/Counter
- 3 UASART
- 2 interfaces SPI maestro esclavo
- 3 sistemas de Timers
- Un controlador magnético de potencia

AT91RM3400

- Presenta un procesador ARMTDMI que trabaja a 60 MIPS con 96 Kbytes de SRAM
- Esta disponible en chips de 100 pines con encapsulado LQFP
- 256 Kbytes de ROM
- Interfaz para dispositivo USB
- Controlador magnético de potencia
- Reloj de tiempo real
- Controlador serial sincrónico
- Un canal para el Timer/Counter
- Un canal USART
- Interfaz para periférico seria
- Interfaz de comunicación Two-wire
- Controlador de entradas salidas paralelas
- Este dispositivo no posee un bus de interfaz externa

AT91RM9200

Este microcontrolador ofrece una flexible configuración de memoria ON-CHIP y OFF-CHIP en conjunto con extenso set de controladores para comunicaciones y almacenamiento de datos. Su diseño esta basado en el núcleo ARM920T, trabaja a 200 MIPS, es un avanzado procesador RISC de 32 bits con una memoria de instrucciones de 16Kbyte y memoria cache de datos también de 16K byte, esta disponible en chips de 208 pines con encapsulado PQFP o en chips de 256 pines con encapsulado BGA. Ofrece 16 Kbytes de SRAM on-chip y 128 Kbytes de ROM.

Incluye un extenso set de periféricos como: puerto para ethernet, interfaz de tarjeta multimedia, tres controladores seriales sincrónicos (SSC), cuatro USARTs, interfaz de periférico serial maestro/esclavo (SPI), un timer counter (TC).

Este microcontrolador presenta una controlador de sistema que ayuda a tener un eficiente manejo del sistema, incluye también dos osciladores on-chip, un generador de reloj, un avanzado controlador de interrupciones, una unidad de depuración, un timer de intervalos periódicos, watchdog timer y un reloj de tiempo real con interrupción de alarma.

Sus aplicaciones están enfocadas principalmente en el campo de los sistemas telemáticos como receptor GPS, en comunicaciones GPRS, en la automatización industrial, instrumentación médica.

Series AT91SAM7A

AT91SAM7A1

Ofrece un procesador el cual trabaja a 36 MIPS con 4 Kbytes de SRAM, esta disponible en chips de 144 pines con encapsulado LQFP, un bus de interfaz externa completamente programable y un variado set de periféricos incluyendo un controlador CAN y un canal de 16 bits para PWM (Modulación por ancho de pulso). Este es una solución muy efectiva para aplicaciones de control integrado por computadora en la industria.

AT91SAM7A2

Esta constituido por un procesador ARM7TDMI que trabaja a 27 MIPS con 16 Kbytes de SRAM, esta disponible en chips de 176 pines con encapsulado LQFP, un bus de interfaz externa programable y un variado set de periféricos constituido por cuatro controladores CAN y un canal de 16 bits para PWM. Este ofrece una gran solución para aplicaciones de control integrado en el mundo industrial y automotriz.

AT91SAM7A3

Se basa en el procesado ARM7TDMI RISC de 32 bits y esta diseñado para aplicaciones de control integrado, esta disponible en chips de 100 pines con encapsulado LQFP

Esta formado por un extenso set de periféricos como son: dos controladores CAN, controlador PWM de 8 bits, nueve timers de propósitos generales de 16 bits, ADC de 10 bits, un puerto USB, interfaces seriales como: USARTs, SPI (Interfaz de periféricos seriales), SSC (Controlador Serial Sincrónico) y TWI (Interfaz two wire). El controlador del sistema es encargado de manejar las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando de esta manera el consumo de potencia.

Este es un microcontrolador de propósitos generales, utilizado particularmente para aplicaciones de control industrial en tiempo real las cuales requieren comunicarse por una red CAN.

Series AT91SAM7S

Esta serie de microcontroladores están basados en el procesador ARM7TDMI RISC de 32 bits, están constituidos por chips de 64 pines en encapsulado LQFP para los circuitos integrados AT91SAM7S256/128/64/321 y en chips de 48 pines en encapsulado LQFP para el circuito integrado AT91SAM7S32. Son

microcontroladores de propósitos generales se los puede utilizar en aplicaciones de 8 bits cuando se requiera un rendimiento adicional. En esta serie tenemos los siguientes dispositivos: AT91SAM7S321, AT91SAM7S32, AT91SAM7S64, AT91SAM7S128, AT91SAM7S256.

AT91SAM7S321

Presenta 32 Kbytes de flash de alta velocidad integrada y 8 Kbytes de SRAM. Este microcontrolador incluye los siguientes periféricos: puerto USB de 2.0, USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador de sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7S32

Presenta 32 Kbytes de flash de alta velocidad integrada y 8 Kbytes de SRAM. Consta de los siguientes periféricos: USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. El controlador del sistema es el encargado de manejar las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7S64

Presenta 64 Kbytes de flash de alta velocidad integrada y 16 Kbytes de SRAM. Este microcontrolador incluye los siguientes periféricos: puerto USB de 2.0, USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7S128

Presenta 128 Kbytes de flash de alta velocidad integrada y 32 Kbytes de SRAM. Este microcontrolador incluye los siguientes periféricos: puerto USB de 2.0, UARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7S256

Presenta 256 Kbytes de flash de alta velocidad integrada y 64 Kbytes de SRAM. Este microcontrolador incluye los siguientes periféricos: puerto USB de 2.0, UARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

Series AT91SAM7X

Estos microcontroladores están basados en el procesador ARM/TDMI RISC de 32 bits e integra interfaces para Ethernet, USB y CAN. Son microcontroladores de propósito general utilizados en aplicaciones de control integrado que requieran conectividad con redes Ethernet o CAN. Los microcontroladores que encontramos aquí son: AT91SAM7X128, AT91SAM7X256, AT91SAM7XC128, AT91SAM7XC256.

AT91SAM7X128

Posee 128 Kbytes de flash integrada y 32 Kbytes de SRAM, es disponible en chips de 100 pines con encapsulado LQFP. Este microcontrolador incluye los

siguientes periféricos: puerto USB de 2.0, puerto ethernet de 10/100, controlador CAN de 2.0A y 2.0B, USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7X256

Posee 256 Kbytes de flash integrada y 64 Kbytes de SRAM, es disponible en chips de 100 pines con encapsulado LQFP. Este microcontrolador incluye los siguientes periféricos: puerto USB de 2.0, puerto ethernet de 10/100, controlador CAN de 2.0A y 2.0B, USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7XC128

Posee 128 Kbytes de flash integrada y 32 Kbytes de SRAM, esta constituido por 100 pines en chips de encapsulado LQFP. Este microcontrolador incluye los siguientes periféricos: puerto USB de 2.0, puerto ethernet de 10/100, controlador CAN de 2.0A y 2.0B, un acelerador de encriptación AES128, USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. . El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

AT91SAM7XC256

Posee 256 Kbytes de flash integrada y 64 Kbytes de SRAM, esta constituido por 100 pines en chips de encapsulado LQFP. Este microcontrolador incluye los

siguientes periféricos: puerto USB de 2.0, puerto ethernet de 10/100, controlador CAN de 2.0A y 2.0B, un acelerador de encriptación AES128, USARTs, SPI, SSC, TWI un conversor análogo digital de 10 bits. El controlador DMA no permite que en el procesador se produzca un embotellamiento durante la transferencia de datos de los periféricos a la memoria. Su controlador del sistema maneja las interrupciones, clocks, potencia, tiempo, depuración y reset, minimizando el consumo de potencia.

Series AT91SAM9

AT91SAM9260

Es el primer microcontrolador de la familia ARM con pines compatibles que comparte el mismo modelo de programación con los microcontroladores de la familia ARM7, permitiendo directa migración entre microcontroladores que tienen diferentes núcleos ARM.

Este microcontrolador esta basado en el procesador ARM926EJ-S con 8Kbytes de memoria de instrucciones y 8 Kbytes de memoria cache de datos, esta constituido por 217 pines en chips de encapsulado LFBGA. Trabaja a 210 MIPS con un reloj de 190 MHz. Tiene un extenso set de periféricos entre los que podemos encontrar: puerto para ethernet a 10/100, interfase de sensor de imagen, interfase de tarjeta multimedia, controladores seriales sincrónicos, USARTs, interfaz de periféricos seriales maestro / esclavo, timer / counter y un converso análogo digital de 10 bits.

Posee un controlador de sistema el cual ayuda a tener un eficiente manejo de este, un controlador de reset, controlador de shutdown (apagado de sistema), un administrador de relojes, controlador de interrupciones, unidad de depuramiento, un timer de intervalos periódicos, watchdog timer y un reloj de tiempo real. Este microcontrolador se caracteriza por su alta conectividad con aplicaciones de procesamiento de imágenes como: cámaras y lectores de código de barras.

AT91SAM9261

Es un microcontrolador basado en el procesador ARM926EJ-S, con 16Kbyte de memoria de instrucciones y 16 Kbyte de memoria cache de datos, esta disponible en chips de 217 pines con encapsulado LFBGA trabaja a 210 MIPS con un reloj de 190 MHz. Este dispositivo ofrece 160 Kbyte de SRAM y 32 Kbyte de ROM.

Su extenso set de periféricos incluye interfaces de dispositivos, un controlador de LCD, tarjeta de interfaz multimedia, controladores seriales sincrónicos, USARTs, periféricos de interfaz serial maestro/esclavo, un canal de 16 bits de timer/counter e interfaz de comunicación por medio de dos cables.

Incluye un controlador del sistema el cual ayuda a tener un eficiente manejo de este, también consta de un controlador de reset, controlador de shutdown (apagado de sistema), un administrador de relojes, controlador de interrupciones, unidad de depuramiento, un timer de intervalos periódicos, watchdog timer y un reloj de tiempo real.

AT91FR40162S

Posee las siguientes características: 2 Mbits de SRAM on-chip y 2 Mbytes de memoria flash, es disponible en chips de 121 pines, su dimensión es de 10x10x1.2 mm con encapsulado BGA. Tiene un alto nivel de integración y su tamaño es pequeño ideal para aplicaciones reducidas. La memoria flash puede ser programada por medio de una interfaz JTAG/ICE o desde la fabrica.

AT91M40800

Ofrece las siguientes características: 8 Kbytes de SRAM on-chip, bus de interfaz externa, timer / counter, 2 puertos USART y un watchdog timer. El puerto JTAG/ICE es solo usado como puerto de depuración. Este circuito integrado esta disponible en chips de 100 pines con encapsulado LQFP.

AT91R40008

Este microcontrolador ofrece: 256 Kbytes de SRAM on-chip, bus de interfaz externa, timer / counter, 2 puertos USART y un watchdog timer. El puerto JTAG/ICE es solo usado como puerto de depuración. Este circuito integrado esta disponible en chips de 100 pines con encapsulado LQFP

3.2.3 Philips

LPC2101/2102/2103

- Están constituidos por el núcleo ARM7TDMI de 16/32 bits
- Presenta los siguientes tipos de encapsulado

Tabla 3.1. Tipos de encapsulado chips LPC2101/2102/2103

| Número de tipo | Encapsulado | |
|----------------|-------------|--|
| | Nombre | Descripción |
| LPC2101BBD48 | LQFP48 | Empaquetado cuadrado lizo, contorno delgado de plástico, dimensiones 7x7x1.4 m |
| LPC2102BBD48 | LQFP48 | |
| LPC2103FBD48 | LQFP48 | |

- Poseen memoria RAM estática de 2Kbyte/4Kbyte/2Kbyte respectivamente y una memoria flash de programación de 8Kbyte/16Kbyte/32Kbyte respectivamente.
- Conversor análogo digital de 10 bits generando 8 entradas análogas con tiempos de conversión muy bajos de 2.44 us por canal.
- Interfase serial múltiple que incluyen: dos UARTs, dos buses I²C, SPI y SSP.
- Reloj de tiempo real de baja potencia con una especial entrada de reloj de 32 KHz.

- Dos timers counters de 16 bits que combina tres canales de captura y 7 canales de comparación.
- Controlador del vector de interrupción
- Treinta y dos pines de entrada/ salida para propósitos generales, los cuales soportan 5V
- El reloj del CPU trabaja a una frecuencia máxima de 70MHz

LPC2104/2105/2106

- Están constituidos por el procesador ARM7TDMI de 16/32 bits
- Presenta los siguientes tipos de encapsulado

Tabla 3.2. Tipos de encapsulado chips LPC2104/2105/2106

| Número de tipo | Encapsulado | |
|----------------|-------------|---|
| | Nombre | Descripción |
| LPC2104BBD48 | LQFP48 | Empaquetado cuadrado lizo, contorno delgado de plástico, dimensiones 7x7x1.4 m |
| LPC2105BBD48 | LQFP48 | |
| LPC2106FBD48 | LQFP48 | |
| LPC2106FHN48 | HVQFN48 | Encapsulado cuadrado lizo muy delgado hecho en plástico térmico, dimensiones 7x7x0.85mm |

- Tiene una memoria RAM de 16/32/64 Kbytes respectivamente
- Memoria programable flash on-chip de 128Kbytes
- Controlador del vector de interrupción
- La interfaz serial incluye: dos puertos UARTs, I²C y SPI
- Posee dos timers de 32 bits (7 canales de captura / comparación), una unidad de PWM (6 salidas), reloj de tiempo real y Watchdog timer.
- 32 I/O de propósitos generales las cuales soportan hasta 5 V
- Cristal oscilador on-chip con un rango de operación de 1MHz a 30MHz
- Habilitación o deshabilitación individual de las funciones de los periféricos para optimizar el consumo de potencia.

LPC2114/LPC2124

- Esta basado en el procesador ARM/TDMI de 16 / 32 bits
- Presenta los siguientes tipos de encapsulado

Tabla 3.3. Tipos de encapsulado chips LPC2114/2124

| Número de tipo | Encapsulado | |
|----------------|-------------|--|
| | Nombre | Descripción |
| LPC2114FBD64 | LQFP64 | Empaquetado cuadrado lizo, contorno delgado de plástico, dimensiones 10x10x1.4 m |
| LPC2124FBD64 | LQFP64 | |

- Posee una memoria RAM on-chip de 16Kbytes
- Memoria programable flash on-chip de 128Kbytes / 256 Kbytes respectivamente
- Cuatro canales para el conversor análogo digital el cual trabaja con 10 bits y con un tiempo de conversión de 2.44 us
- 2 timers de 32 bits (a canales de captura y 4 canales de comparación), un PWM (6 salidas), reloj de tiempo real y posee un watchdog timer
- Múltiples interfaces seriales entre los que encontramos: dos UARTs, I²C y dos SPIs.
- Controlador del vector de interrupción
- 46 I/O de propósitos generales que soportan hasta 5V.
- Cristal oscilador on-chip con un rango de operación de 1MHz a 30MHz.
- Habilitación y deshabilitación individual de las funciones de los periféricos para obtener un mejor consumo de potencia
- 9 interrupciones disponibles

LPC2119/LPC2129

- Esta basado en el procesador ARM/TDMI de 16 / 32 bits
- Posee los siguientes tipos de encapsulado

Tabla 3.4. Tipos de encapsulado chips LPC2119/2129

| Número de tipo | Encapsulado | |
|----------------|-------------|--|
| | Nombre | Descripción |
| LPC2119FBD64 | LQFP64 | Empaquetado cuadrado lizo, contorno delgado de plástico, dimensiones 10x10x1.4 m |
| LPC2129FBD64 | LQFP64 | |

- Posee una memoria RAM on-chip de 16Kbytes
- Memoria programable flash on-chip de 128Kbytes / 256 Kbytes respectivamente
- 2 interfaces de interconexión CAN con modernos filtros de aceptación
- 4 canales para el conversor análogo digital el cual trabaja con 10 bits y con un tiempo de conversión de 2.44 us
- 2 timers de 32 bits (a canales de captura y 4 canales de comparación), un PWM (6 salidas), reloj de tiempo real y posee un watchdog timer
- Múltiples interfaces seriales entre los que encontramos: dos UARTs, I²C y dos SPIs.
- Controlador del vector de interrupción
- 46 I/O de propósitos generales que soportan hasta 5V.
- 9 interrupciones disponibles
- Cristal oscilador on-chip con un rango de operación de 1MHz a 30MHz.
- Habilitación y deshabilitación individual de las funciones de los periféricos para obtener un mejor consumo de potencia

LPC2131/2132/2134/2136/2138

- Esta basado en el procesador ARM/TDMI-S de 16 / 32 bits
- Posee los siguientes tipos de encapsulado

Tabla 3.5. Tipos de encapsulado chips LPC2131/2132/2134/2136/2138

| Número de tipo | Encapsulado | |
|----------------|-------------|---|
| | Nombre | Descripción |
| LPC2131FBD64 | LQFP64 | Empaquetado cuadrado lizo, contorno delgado |
| LPC2132FBD64 | LQFP64 | |

| | | |
|--------------|--------|--------------------------------------|
| LPC2134FBD64 | LQFP64 | de plástico, dimensiones 10x10x1.4 m |
| LPC2136FBD64 | LQFP64 | |
| LPC2138FBD64 | LQFP64 | |

- Posee una memoria RAM on-chip de 8 / 16 / 32 Kbytes
- Memoria programable flash on-chip de 32Kbytes / 64 Kbytes / 128 Kbytes / 256 Kbytes / 512 Kbytes respectivamente
- 2 canales para el conversor análogo digital el cual trabaja con 10 bits en un tiempo de conversión de 2.44 us
- 1 conversos digital análogo simple de 10 bits que provee salidas de las variables analógicas.
- 2 timers de 32 bits (2 canales de captura y 4 canales de comparación), un PWM (6 salidas), y un watchdog timer
- Múltiples interfaces seriales entre los que encontramos: dos UARTs, 2 buses de I²C y dos SPI y SSP.
- Controlador del vector de interrupción
- 47 I/O de propósitos generales que soportan hasta 5V.
- 9 interrupciones disponibles
- Cristal oscilador on-chip con un rango de operación de 1MHz a 30MHz y también puede trabajar con un oscilador externo hasta 50 MHz.
- Habilitación y deshabilitación individual de las funciones de los periféricos para obtener un mejor consumo de potencia
- BOD es una interrupción externa que permite al procesador volver a trabajar cuando este se encuentra en un estado de suspensión.

LPC2141/42/44/46/48

- Esta basado en el procesador ARM/TDMI-S de 16 / 32 bits
- Posee los siguientes tipos de encapsulado

Tabla 3.6. Tipos de encapsulado chips LPC2141/2142/2144/2146/2148

| Número de tipo | Encapsulado | |
|----------------|-------------|--|
| | Nombre | Descripción |
| LPC2141FBD64 | LQFP64 | Empaquetado cuadrado lizo, contorno delgado de plástico, dimensiones 10x10x1.4 m |
| LPC2142FBD64 | LQFP64 | |
| LPC2144FBD64 | LQFP64 | |
| LPC2146FBD64 | LQFP64 | |
| LPC2148FBD64 | LQFP64 | |

- Poseen una memoria RAM de 8Kbytes a 40Kbytes y una memoria flash de 32Kbytes a 512 Kbytes
- Tiene un controlador USB 2.0
- 2 conversores análogos digitales de 10bits que dan un total de 14 entradas análogas con un tiempo de conversión de 2.44 por canal.
- 1 DAC (conversor digital análogo) de 10 bits.
- 2 timers o contadores de eventos externos de 32 Bits, una PWM y un watchdog timer.
- Presenta múltiples interfaces seriales como son: dos UARTs, dos buses I²C, SPI y SSP
- Controlador del vector de interrupción (VIC) con configuración de prioridades y direccionamiento del vector
- 45 pines de I / O para propósitos generales los cuales trabajan con 5V
- 21 pines para interrupciones externas
- Habilitación y deshabilitación individual de las funciones de los periféricos para obtener un mejor consumo de potencia
- BOD es una interrupción externa que permite al procesador volver a trabajar cuando este se encuentra en un estado de suspensión.

3.2.4 Cirrus Logia

Las series EP7309, 7311, 7312 presentan los siguientes tipos de encapsulado: 208 pines LQFP (Low Profile Quad Flat Pack), 256 pines PBGA (Plastic Ball Grid Array), 204 pines TFBGA

EP7309

Presenta las siguientes características:

- Velocidades de reloj programadas dinámicamente de 18, 36, 49 y 79 MHz.
- 48 Kbytes de SRAM on-chip
- Controlador LCD
- 2 interfaces seriales sincrónicas (SSI1, SSI2)
- Interfaz de sonido CODEC
- Interfaz de audio digital
- 27 pines de I / O para propósitos generales
- 2 UARTs compatibles
- 2 módulos PWM
- un clock de tiempo real
- 2 timers de 16 bits para propósitos generales
- Controlador de interrupciones
- Entre sus aplicaciones tenemos: digital music players, aplicaciones de Internet, teléfonos celulares inteligentes y tiene la capacidad para descompresión de audio digital.

EP7311

Presenta las siguientes características:

- Este construido a partir del procesador ARM720T
- Controlador de LCD
- Interfaz SDRAM de 32 bits
- Puerto codec multimedia
- 2 interfaces seriales sincrónicas

- Interfaz de sonido CODEC
- 27 pines de I / O para propósitos generales
- 2 módulos UARTs
- 2 módulos PWM
- Reloj de tiempo real
- 2 timers de 16 bits de propósitos generales
- Controlador de interrupciones +
- Este microcontrolador es diseñado principalmente para aplicaciones como: PDAs, teléfonos celulares inteligentes y como instrumentos de mano para información industrial.

EP7312

Presenta las siguientes características:

- Este construido a partir del procesador ARM720T
- Una memoria SRAM de 48 Kbytes
- Trabaja con una velocidad de reloj de 74 y 90 MHz
- Controlador de LCD
- Interfaces SRAM/FLASH/ROM de 8/32/16 bits respectivamente
- 2 interfaces seriales sincrónicas
- Interfaz de sonido CODEC
- 27 pines de I / O para propósitos generales
- 2 módulos UARTs
- 2 módulos PWM
- Reloj de tiempo real
- 2 timers de 16 bits de propósitos generales
- Controlador de interrupciones
- Este microcontrolador es diseñado principalmente para aplicaciones como: dispositivos portátiles de entretenimiento, sistemas de audio tanto para vehículos como para el hogar y en aplicaciones de control industrial de propósito general.

Las series EP9301, EP9301, EP9301

EP9301

Presenta las siguientes características:

- Este construido por un procesador ARM920T que trabaja a 166 MHz
- Cache de instrucciones de 19 Kbytes
- Cache de datos de 16 Kbytes
- Soporte para trabajar con sistemas operativos como: Linux, Microsoft, Windows.
- Interfaz de 16 bits de SDRAM
- ROM /FLASH / SRAM de 16 bits
- Interfaz de EEPROM serial
- Puerto Ethernet de 1 / 10 / 100 Mbps
- 2 UARTs
- 2 puertos USB 2.0 (12 Mbits por segundo)
- Conversor análogo digital
- Puerto para comunicación serial (SPI)
- Interfaz de audio serial (I²S)
- 12 canales de acceso a memoria directamente
- Reloj de tiempo real con software
- Watchdog timer
- 2 timers de 16 bits para propósitos generales
- 1 timer de 32 bits para propósitos generales
- 1 timer de depuración de 40 bits
- Este microcontrolador es utilizado principalmente para: control industrial, servidores digitales, gateways, consolas de audio digital, set-top-box, sistemas de seguridad biométricos, Sistemas GPS, juguetes educativos, computadoras para la industria, máquinas de votación, equipo médico

EP9302

- Este construido por un procesador ARM920T que trabaja a 200 MHz
- El tipo de encapsulado que presenta es 208 pines LQFP
- Cache de instrucciones de 19 Kbytes
- Cache de datos de 16 Kbytes
- Soporte para trabajar con sistemas operativos como: Linux, Microsoft, Windows.
- Se utiliza principalmente para compresión de música digital y descompresión de algoritmos.
- Interfaz de 16 bits de SDRAM
- ROM /FLASH / SRAM de 16 bits
- Interfaz de EEPROM serial
- Puerto Ethernet de 1 / 10 / 100 Mbps
- 2 UARTs
- 2 puertos USB 2.0 (12 Mbits por segundo)
- Conversor análogo digital
- Puerto para comunicación serial (SPI)
- Interfaz de audio serial (I²S)
- 12 canales de acceso a memoria directamente
- Reloj de tiempo real
- Watchdog timer
- 2 timers de 16 bits para propósitos generales
- 1 timer de 32 bits para propósitos generales
- 1 timer de depuración de 40 bits
- Controlador de interrupciones
- Posee las mismas aplicaciones que el microcontrolador EP9301

EP9307 / EP9312 / EP9315

Presenta las siguientes características:

- Este construido por un procesador ARM920T que trabaja a 200 MHz

- El tipo de encapsulado que presenta es 272 pines TFBGA / 352 pines PGBA / 352 pines PBGA
- Cache de instrucciones de 16 Kbytes
- Cache de datos de 16 Kbytes
- Soporte para trabajar con sistemas operativos como: Linux, Microsoft, Windows.
- Se utiliza principalmente para compresión de música digital y descompresión de algoritmos.
- Interfaz de 32 bits de SDRAM
- ROM /FLASH / SRAM de 32 / 16 bits
- Interfaz de EEPROM serial
- Puerto Ethernet de 1 / 10 / 100 Mbps
- 3 UARTs
- 3 puertos USB 2.0 (12 Mbits por segundo)
- LCD e interfaz de trama con acelerador de gráficos
- Interfaz de pantalla táctil con ADC
- Puerto para comunicación serial (SPI)
- Interfaz de audio serial (I²S)
- 12 canales de acceso a memoria directamente
- Reloj de tiempo real
- Watchdog timer
- 2 timers de 16 bits para propósitos generales
- 1 timer de 32 bits para propósitos generales
- 1 timer de depuración de 40 bits
- Controlador de interrupciones
- Posee las siguientes aplicaciones: en computadoras para negocios y para el hogar, radio de Internet, dispositivos de acceso de Internet, computadoras industriales, terminales de punto de salida y en equipos de medición y pruebas.

3.2.5 Analog Devices

ADUC7019 / ADU7020

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Son chips de 40 pines y posee un encapsulado de tipo LFCSP
- Trabaja a una velocidad de 40 MIPS
- Memoria flash de 62 Kbytes
- Memoria RAM de 8192Bytes
- 14 pines de I / O para propósitos generales
- 5 canales para el conversor análogo digital
- Conversor digital análogo de 12 bits
- Rango de temperatura que soporta es de -40°C a 125 °C

ADUC7021

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Son chips de 40 pines y posee un encapsulado de tipo LFCSP
- Trabaja a una velocidad de 40 MIPS
- Memoria flash de 62 Kbytes
- Memoria RAM de 8192Bytes
- 13 pines de I/O para propósitos generales
- 8 canales para el conversor análogo digital
- Conversor digital análogo de 12 bits
- Rango de temperatura que soporta es de -40°C a 85 °C

ADUC7022

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Son chips de 40 pines y posee un encapsulado de tipo LFCSP
- Trabaja a una velocidad de 40 MIPS
- Memoria flash de 62 Kbytes
- Memoria RAM de 8192Bytes
- 13 pines de I/O
- 10 canales para el conversor análogo digital
- Rango de temperatura que soporta es de -40°C a 85 °C
- Interfaces de puertos seriales como: UART, SPI e I²C
- Comparador, power-on Reset (POR)

ADUC7024

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Son chips de 64 pines y posee un encapsulado de tipo LQFP
- Trabaja a una velocidad de 40 MIPS
- Memoria flash de 62 Kbytes
- Memoria RAM de 8192Bytes
- 30 pines de I / O para propósitos generales
- 10 canales para el conversor análogo digital
- Conversor digital análogo de 12 bits
- Rango de temperatura que soporta es de -40°C a 105 °C
- Interfaces de puertos seriales como: UART, SPI e I²C
- 3 fases de PWM
- Comparador, power-on Reset (POR)

ADUC7025

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Trabaja a una velocidad de 40 MIPS
- Son chips de 64 pines y posee un encapsulado de tipo LFCSP
- Memoria flash de 62 Kbytes
- Memoria RAM de 8192Bytes
- 30 pines de I / O para propósitos generales
- 12 canales para el conversor análogo digital
- Rango de temperatura que soporta es de -40°C a 105 °C
- Interfaces de puertos seriales como: UART, SPI e I²C
- 3 fases de PWM
- Power-on Reset (POR)

ADUC7026

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Son chips de 80 pines y posee un encapsulado de tipo LQFP
- Trabaja a una velocidad de 40 MIPS
- Memoria flash de 62 Kbytes
- Memoria RAM de 8192Bytes
- 40 pines de I / O para propósitos generales
- 12 canales para el conversor análogo digital
- Conversor análogo de 12 bits
- Presenta interfaz para memoria externa
- Rango de temperatura que soporta es de -40°C a 125 °C
- Interfaces de puertos seriales como: UART, SPI e I²C
- 3 fases de PWM
- Power-on Reset (POR)

ADUC7023

Presenta las siguientes características:

- Esta constituido por el núcleo ARM7 TDMI
- Son chips de 40 pines y posee un encapsulado de tipo LFCSP
- Trabaja a una velocidad de 40 MIPS
- Memoria flash de 126 Kbytes
- Memoria RAM de 8192Bytes
- 28 pines de I / O para propósitos generales
- 10 canales para el conversor análogo digital
- Interfaces de puertos seriales como: 2 UART, SPI e 2 I²C
- 5 timers de propósitos generales
- Generador de PWM de 16 bits

3.2.6 Sanyo

LC67F5006A / LC7F5104A

Presenta las siguientes características:

- El procesador esta formado por el ARM7TDMI
- Posee 100 pines en un encapsulado TQFP100 de 14x14 mm
- Posee 768/512 Kbytes de memoria ROM respectivamente
- 32 KBYtes de memoria RAM
- 68 pines de I / O
- Presenta 23 interrupciones internas y 7 interrupciones externas
- 5 timers de 16 bits
- Presenta un UART, DMAC
- Un conversor análogo digital de 8 bits

- Rango de voltaje de 2.25 a 2.75 en el CPU y de 3.0 a 3.6 en las I/O

LC690123A

Presenta las siguientes características:

- El procesador esta formado por el ARM926EJ_STM
- Posee 449 pines en un encapsulado de tipo FBGA
- Posee 768 Kbytes de memoria ROM
- 32 KBytes de memoria RAM
- 40 pines de I / O
- Presenta 27 interrupciones internas y 5 interrupciones externas
- 1 timer 32 bits
- Un conversor análogo digital de 8 bits
- Tarjeta de memoria SD
- Interfaces seriales SSP, UART
- Posee un dispositivo USB, Ethernet, interfaz de audio, interfaz de video
- Rango de voltaje de operación de 1.1 a 1.3 en el CPU y de 3.0 a 3.6 en las I/O

ANEXO 1. Tabla comparativa microcontroladores

CAPÍTULO IV

DESCRIPCIÓN DEL CHIP AT91SAM7S64

4.1 DESCRIPCIÓN

Es un microcontrolador basado en la arquitectura ARM RISC de 32 bits. Ofrece una flash de alta velocidad de 64 KB y una SRAM de 16 KB, un extenso set de periféricos incluyendo un dispositivo USB 2.0 y una gran variedad de funciones del sistema utilizadas para minimizar el número de componentes externos.

La memoria flash integrada puede ser programada de varias formas: por medio de la interfaz JTAG-ICE o por un programador el cual se debe encontrar puesto en paralelo. Integra bits de seguridad para proteger el firmware (programa del microcontrolador) de sobre escrituras y guarda su confidencialidad.

El AT91SAM7S64 abarca un controlador de reset capaz de manejar la secuencia de encendido del microcontrolador y el sistema completo. La correcta operación del dispositivo puede ser monitoreada por un incorporado detector de fallas y el watchdog fuera de funcionamiento.

Este microcontrolador es utilizado para propósitos generales. Gracias a su puerto USB lo hace un dispositivo ideal para aplicaciones que requieran ser conectadas a un PC o a un teléfono celular.

4.2 DIAGRAMA DE BLOQUES GENERAL

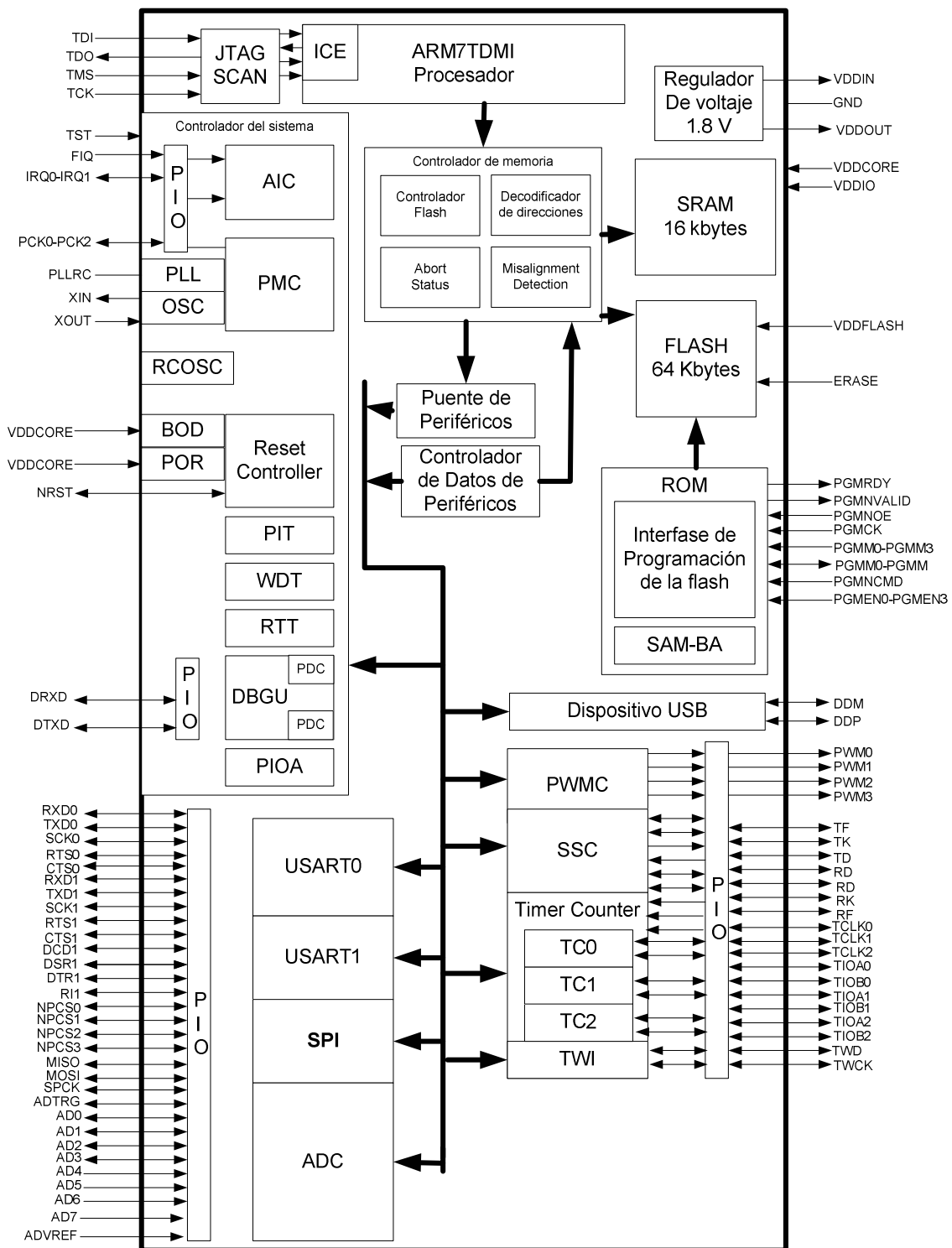


Figura 4.1. Diagrama de bloques AT91SAM7S64

En la tabla 4.1 podemos ver los diferentes pines del chip y la función que desempeña cada uno.

Tabla 4.1. Descripción de señales

| Nombre de la señal | Función | Tipo |
|------------------------------------|--|----------|
| Potencia | | |
| VDD IN | Entrada de voltaje y ADC | Potencia |
| VDDOUT | Salida de voltaje | Potencia |
| VDDFLASH | Fuente de potencia flash | Potencia |
| VDDIO | Subministra potencia a las líneas de I/O | Potencia |
| VDDCORE | Subministra potencia la núcleo | Potencia |
| VDDPLL | PLL | Potencia |
| GND | Tierra | Tierra |
| Relojes, Osciladores y PLLs | | |
| XIN | Entrada de oscilador principal | Entrada |
| XOUT | Salida de oscilador principal | Salida |
| PLLRC | Filtro PLL | Entrada |
| PCK0-PCK2 | Salida del reloj programable | Salida |
| ICE y JTAG | | |
| TCK | Clock de prueba | Entrada |
| TDI | Entrada de datos de prueba | Entrada |
| TDO | Salida de datos de prueba | Salida |
| TMS | Selección de modo de prueba | Entrada |
| JTAGSEL | Selección de JTAG | Entrada |
| Memoria Flash | | |
| ERASE | Configuración de la flash | Entrada |
| Reset / Prueba | | |
| NRST | Reset del microcontrolador | I/O |
| TST | Selección del modo de prueba | Entrada |
| Unidad de depuración | | |
| DRXD | Datos recibidos de la depuración | Entrada |
| DTXD | Datos transmitidos de la depuración | Salida |
| AIC | | |
| IRQ-IRQ1 | Entradas de interrupción externa | Entrada |
| FIQ | Entrada de interrupción rápida | Entrada |

| Tabla 4.1. Descripción de señales (Continuación) | | |
|--|---|---------|
| PIO | | |
| PA0 – PA31 | Controlador de entradas salidas paralelas | I/O |
| Dispositivo USB | | |
| DDM | Datos (-) del puerto USB | Análogo |
| DDP | Datos (+) del puerto USB | Análogo |
| USART | | |
| SCK0 – SCK1 | Reloj Serial | I/O |
| TXD0 – TXD1 | Datos transmitidos | I/O |
| RXD0 – RXD1 | Datos recibidos | Entrada |
| RTS0 – RTS1 | Respuesta al envió | Salida |
| CTS0 – CTS1 | Limpia el envió | Entrada |
| DCD1 | Detecta carrier | Entrada |
| DTR1 | Ready dato terminado | Salida |
| DSR1 | Ready dato puesto | Entrada |
| RI1 | Indicador | Entrada |
| Controlador serial sincrónico | | |
| TD | Datos de transmisión | Salida |
| RD | Datos de recepción | Entrada |
| TK | Reloj de transmisión | I/O |
| RK | Reloj de recepción | I/O |
| TF | Transmisión sincrónica | I/O |
| RF | Recepción sincrónica | I/O |
| Timer / Counter | | |
| TCLK0 – TCLK2 | Entradas de reloj externa | Entrada |
| TIOA0 – TIOA2 | Línea A I/O | I/O |
| TIOB0 – TIOB2 | Línea B I/O | I/O |
| Controlador de PWM | | |
| PWM0 – PWM3 | Canales de PWM | Salidas |
| SPI | | |
| MISO | Salida maestro/esclavo | I/O |
| MOSI | Entrada de esclavo, salida de maestro | I/O |
| SPCK | Reloj del serial SPI | I/O |

| Tabla 4.1. Descripción de señales (Continuación) | | |
|---|------------------------------------|---------|
| NPCS0 | Selección del periférico SPI [0] | I/O |
| NPCS1 – NPCS3 | Selección de periféricos SPI [1:3] | Salidas |
| Interfaz Two-Wire | | |
| TWD | Datos seriales Two-wire | I/O |
| TWCK | Clock Serial Two-wire | I/O |
| Conversor Análogo digital | | |
| AD0 – AD3 | Entradas análogas | Análogo |
| AD4 – AD7 | Entradas análogas | Análogo |
| ADTRG | Trigger ADC | Entrada |
| ADVREF | Referencia ADC | Análogo |
| Interfaces Programables | | |
| PGMEN0 – PGMEN2 | Habilita programación | Entrada |
| PGMM0 – PGMM3 | Modo de programación | Entrada |
| PGMD0 – PGMD15 | Datos de programación | I/O |
| PGMRDY | Ready | Salida |
| PGMNVALID | Dirección de datos | Salida |
| PGMNOE | Lectura de programación | Entrada |
| PGMNCK | Clock de programación | Entrada |
| PGMNCD | Comandos de programación | Entrada |

4.3 ARQUITECTURA DEL INTEGRADO

4.3.1 Memorias

- Posee 64 Kbytes de memoria flash
- 512 páginas de 128 bytes
- Tiempo de programación de una página: 6ms, incluyendo el auto borrado de la página
- Programación de la página sin auto borrado: 3ms
- Tiempo de borrado completo del chip: 15ms

- Posee 10000 ciclos de escritura, capacidad para retener los datos de 10 años
- 16 bits de bloqueo, protege 16 sectores de 32 páginas
- Modos de protección para asegurar el contenido de la Flash
- Posee 16 Kbytes de SRAM

4.3.2 Mapa de Memoria

4.3.2.1 SRAM interna

El microcontrolador AT91SAM7S64 tiene un banco de memoria SRAM de 16Kbytes. Es ejecutada después del reset y antes que el comando de reasignación sea ejecutado, la SRAM solo es accesible en la dirección 0x00200000, después de la reasignación la SRAM vuelve a ser útil en la dirección 0x0.

4.3.2.1 ROM interna

Esta memoria es integrada internamente y es asignada por defecto.

4.3.2.3 Flash interna

Tiene un banco de 64 Kbytes de flash, esta memoria en cualquier momento puede ser asignada a la dirección 0x00100000, es también accesible en la dirección 0x0 después del reset y antes del comando de reasignación.

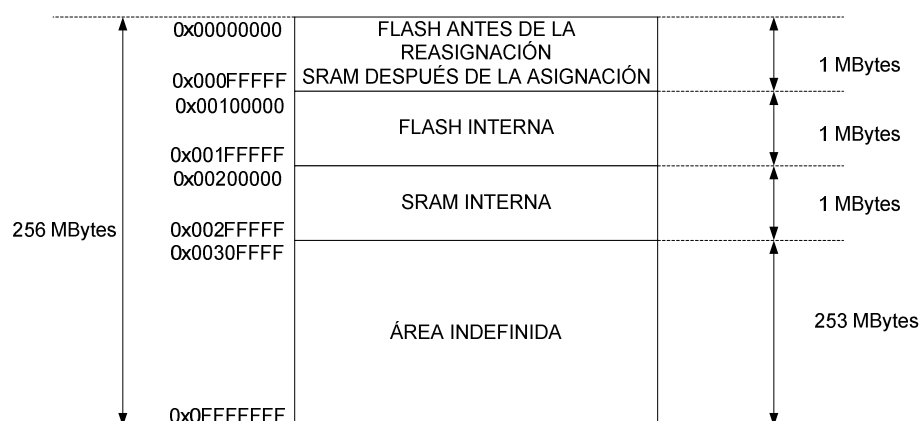


Figura 4.2. Mapa de Memorias

4.3.3 Procesador y arquitectura

4.3.3.1 Procesador ARM7TDMI

- Posee un procesador RISC basado en la arquitectura Von Neumann, trabaja sobre los 55 MHz y entrega 0.9 MIPS/MHz
- Puede trabajar con el set de instrucciones ARM de 32 bits y con el set de instrucciones Thumb de densidad de código de 16 bits
- Esta arquitectura posee tres estados de pipeline: la instrucción fetch, la instrucción decode, la ejecución de la instrucción

4.3.3.2 Depuración de errores y características de prueba

- Tiene integrado EmbeddedICE (Emulador integrado en el circuito)
 - Dos unidades de Prueba
 - Puerto de prueba accesible a través del protocolo JTAG
- Unidad de depuración
 - UART
 - Canal de comunicación de corrección de errores
 - Registro ID del chip

4.3.3.3 Controlador de memoria

- Bus
 - Realiza peticiones de manejo desde el núcleo ARMTDMI y el controlador de periféricos DMA
- El decodificador de direcciones proporciona señales para seleccionar:
 - Tres espacios de memoria internos de 1 Mbyte
 - Un espacio de periférico integrado de 256 Mbyte
- Detector de mal alineamiento
 - Chequea el alineamiento del acceso de los datos
 - No permite el acceso de los datos en caso de una mala alineación

- Controlador Flash integrado
 - Interfaz flash integrada, para más de tres estados de espera programables
 - Buffer de prefetch, sirve para reducir los estados de espera
 - Programa protegido por código, borrado y bloqueo/desbloqueo de la secuencia del programa
 - Un comando simple de borrado, programación y bloqueo de operaciones
 - Generación de interrupciones en caso de operaciones prohibidas

4.3.3.4 Controlador de periféricos DMA

- Maneja la transferencia de datos entre los periféricos y las memorias
- Posee once canales para:
 - Dos para cada USART
 - Dos para la unidad de depuración
 - Dos para los controladores sincrónicos seriales
 - Dos para las interfaces de los periféricos seriales
 - Uno para el conversor análogo digital
- Posee un ciclo del reloj necesario para una transferencia desde la memoria a los periféricos
- Posee dos ciclos de reloj maestro necesarios para transferir datos desde los periféricos a la memoria
- Puede controlarse el siguiente puntero para reducir los requerimientos de interrupciones

4.3.4 Periféricos

Cada periférico tiene asignado 16 Kbytes de espacio de direcciones.

| | | | |
|----------------------------|---------------------------|---|-----------|
| 0xF0000000 | Reservado | | |
| 0xFFFF9FFF 0xFFFFA0000 | TC0, TC1, TC ² | Timer/Counter 0,1 y 2 | 16 KBytes |
| 0xFFFFA3FFF 0xFFFFA4000 | Reservado | | |
| 0xFFFFAFFFF 0xFFFFB0000 | Reservado | | |
| 0xFFFFB3FFF 0xFFFFB4000 | Reservado | | |
| 0xFFFFB7FFF 0xFFFFB8000 | TWI | Interfaz Two-Wire | 16 KBytes |
| 0xFFFFBBFFF 0xFFFFBC000 | Reservado | | |
| 0xFFFFBFFFF 0xFFFFC0000 | USART | Transmisor, receptor sincrónico asincrónico universal | 16 KBytes |
| 0xFFFFC3FFF 0xFFFFC4000 | Reservado | | |
| 0xFFFFC7FFF 0xFFFFC8000 | Reservado | | |
| 0xFFFFCBFFF 0xFFFFCC000 | PWMC | Controlador PWM | 16 KBytes |
| 0xFFFFCFFFF 0xFFFFD0000 | Reservado | | |
| 0xFFFFD3FFF 0xFFFFD4000 | SSC | Controlador serial sincrónico | 16 KBytes |
| 0xFFFFD7FFF 0xFFFFD8000 | ADC | Conversor análogo digital | 16 KBytes |
| 0xFFFFDBFFF 0xFFFFD7FFF | Reservado | | |
| 0xFFFFDFFFF 0xFFFFE0000 | SPI | Interfaz de periféricos seriales | 16 KBytes |
| 0xFFFFE3FFF 0xFFFFE4000 | Reservado | | |
| 0xFFFFEFFFF | | | |

Figura 4.3. Mapa de periféricos

4.3.4.1 Múltiplexión de periféricos en las líneas de PIO

El AT91SAM7S64 presenta un controlador PIO (periféricos de entrada y salida), PIOA el cual multiplexa las líneas de entrada salida del set de periféricos.

El controlador de PIO A controla 32 líneas, cada línea puede ser asignada para una de las dos funciones del periférico, A o B. Algunas de estas entradas pueden ser multiplexadas con las entradas del controlador del conversor análogo digital.

En la tabla 4.2 se define como algunas de las líneas I/O de los periféricos A, B o las entradas analógicas son multiplexadas en el controlador de PIO.

4.3.4.2 Múltiplexión del controlador A de los PIO

Tabla 4.2. Múltiplexión en el controlador de PIO

| Controlador de PIO | | |
|--------------------|--------------|--------------|
| Líneas de I/O | Periférico A | Periférico B |
| PA0 | PWM0 | TIOA |
| PA1 | PWM1 | TIOB |
| PA2 | PWM2 | SCK0 |
| PA3 | TWD | NPCS3 |
| PA4 | TWCK | TCLK0 |
| PA5 | RXD0 | NPCS3 |
| PA6 | TXD0 | PCK0 |
| PA7 | RTS0 | PWM3 |
| PA8 | CTS0 | ADTRG |
| PA9 | DRXD | NPCS1 |
| PA10 | DTXD | NPCS2 |
| PA11 | NPCS0 | PWM0 |

| Tabla 4.2. Multiplexión en el controlador de PIO (Continuación) | | |
|--|-------|-------|
| PA12 | MISO | PWM1 |
| PA13 | MOSI | PWM2 |
| PA14 | SPCK | PWM3 |
| PA15 | TF | TIOA1 |
| PA16 | TK | TIOB1 |
| PA17 | TD | PCK1 |
| PA18 | RD | PCK2 |
| PA19 | RK | FIQ |
| PA20 | RF | IRQ0 |
| PA21 | RXD1 | PCK1 |
| PA22 | TXD1 | NPCS3 |
| PA23 | SCK1 | PWM0 |
| PA24 | RTS1 | PWM1 |
| PA25 | CTS1 | PWM2 |
| PA26 | DCD1 | TIOA2 |
| PA27 | DTR1 | TIOB2 |
| PA28 | DSR1 | TLCK1 |
| PA29 | RI1 | TCLK2 |
| PA30 | IRQ1 | NPCS2 |
| PA31 | NPCS1 | PCK2 |

4.3.5 Control del sistema

El controlador del sistema maneja los principales bloques del microcontrolador como son: interrupciones, clocks, potencia, timers, corrección de errores y reset.

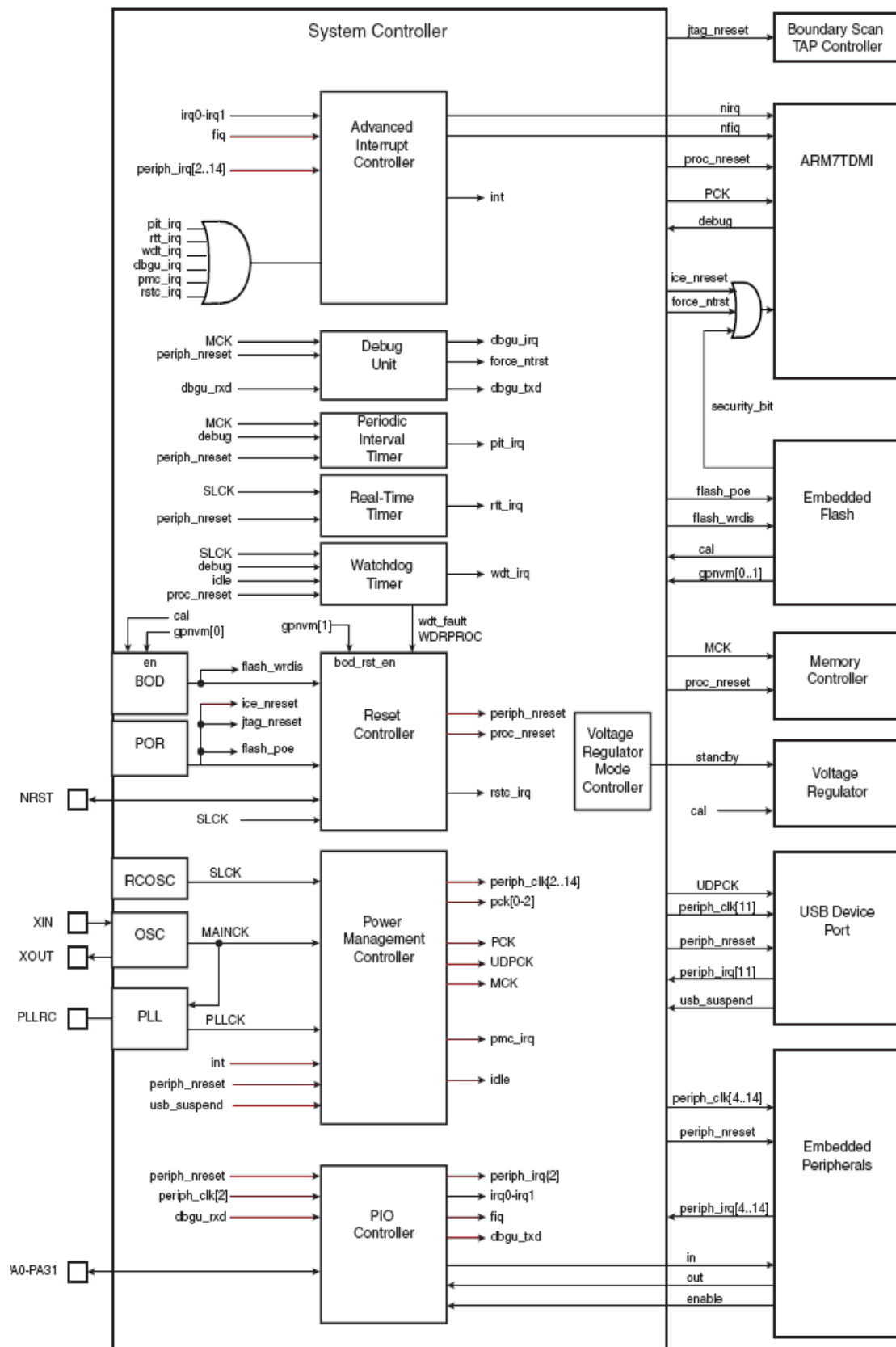


Figura 4.4. Diagrama de bloques del controlador del sistema

4.3.5.1 Mapa del controlador del sistema

Todos los periféricos del controlador del sistema son mapeados para así obtener 4Kbytes de espacio de dirección, entre las direcciones 0xFFFF F000 y 0xFFFF FFFF.

En la figura 4.5 se muestra el mapa del controlador del sistema. Hay que notar que la configuración del controlador de memoria usa interfaces y es también asignado dentro de este espacio de memoria.

| | | | |
|--|-----------|-----------------------------------|----------------------|
| 0xFFFFF000 | AIC | Unidad de depuración | |
| 0xFFFFF1FF 0xFFFFF200 | | | |
| 0xFFFFF3FF 0xFFFFF400 | PIO A | | Controlador A de PIO |
| 0xFFFFF5FF 0xFFFFF600 | | | |
| 0xFFFFFBFF 0xFFFFFC00 | PMC | Controlador de manejo de potencia | |
| 0xFFFFFCFF 0xFFFFFD00 | | | RSTC |
| 0xFFFFFD0F | Reservado | Controlador de reset | |
| 0xFFFFFD20 | RTT | Timer de tiempo real | |
| 0xFFFFFC2F 0xFFFFFD30 | | | PIT |
| 0xFFFFFC3F 0xFFFFFD40 | WDT | Watchdog timer | |
| 0xFFFFFD4F | | | Reservado |
| 0xFFFFFD60 0xFFFFFC6F 0xFFFFFD70 | VREG | Controlador regulador de voltaje | |
| 0xFFFFFEFF 0xFFFFF00 | | | Reservado |
| 0xFFFFFFF | MC | Controlador de Memoria | |

Figura 4.5. Diagrama de bloques del controlador del sistema

4.3.5.2 Controlador de reset

El controlador de reset esta basado en el power on reset y en un detector que entrega el estatus de reset, indicando si ha sido provocado por el power-up reset, un reset de software, provocado por el usuario o por el watchdog timer.

4.3.5.3 Generador de clock

El generador de clock integra un oscilador RC de baja potencia, un oscilador principal, y un PLL con las siguientes características:

- El oscilador RC trabaja entre 22Khz y 42 KHz
- El rango de frecuencia del oscilador principal está entre 3 y 20 MHz
- El oscilador principal puede ser desviado
- El rango de las salidas PLL está entre 80 y 220 MHz

4.3.5.4 Control del manejo de potencia

El controlador del manejo de potencia usa las salidas del generador de clock para proveer:

- El reloj del procesador PCK
- El reloj master MCK
- El reloj del USB UDPC
- El reloj de los periféricos, controlándolos independientemente
- 3 salidas de reloj programable

El reloj master es programable desde unos cientos de Hz hasta la frecuencia máxima de operación del dispositivo.

El reloj del procesador se apaga cuando este entra en modo inactivo, permitiendo reducir el consumo de potencia mientras espera por una interrupción.

4.3.5.5 Controlador de interrupciones

- Controla las líneas de interrupciones de un procesador ARM (nIRQ y nFIQ)
- Individualmente enmascara las fuentes de interrupción
- 8 niveles de control de prioridad
- Modos de protección, fácil depuración para prevenir operaciones automáticas
- Permite transformar cualquier fuente de interrupción en una interrupción rápida

4.3.5.6 Unidad de depuración

Comprende

- Un pin para transmisión y recepción serial asincrónica
- Una interfaz para soporte de canales de comunicación de depuración
- Un set de registros ID

4.3.5.7 Timer de intervalos periódicos

- 20 bit de contador programable más 12 bits de contador de intervalos

4.3.5.8 Watchdog timer

- Contador programable de 12 bits
- Genera un reset o señales de interrupciones para el sistema
- El contador puede ser parado mientras el procesador esta en estado de depuración o en modo inactivo

4.3.5.9 Controlador de PIO

- Un controlador de PIO, el cual controla 32 líneas de I/O

- Completamente programable a través de los registros set / clear
- Multiplexa dos funciones de periféricos por línea de I/O
- Para cada línea de I/O posee:
 - Cambio de entrada en la interrupción
 - Un medio o un período de reloj
 - Un adelanto automático en cada línea de I/O
- Salida sincrónica, provee un set y clear de varias líneas en una sola escritura

4.3.5.10 Controlador regulador de voltaje

El objetivo de este controlador es seleccionar el modo de potencia del regulador de voltaje entre modo normal y modo standby (de reserva).

4.3.6 Recepción y transmisión asincrónica y sincrónica (USART)

- Generador de tasa de baudios programable
- De 5 a 9 bits de comunicación full-duplex o serial asincrónica
 - 1, 1.5, o 2 stop bits en modo asincrónico
 - 1 o 2 stop bits en modo sincrónico
 - Generación de paridad y detección de error
 - Detección de error de framing (cuando no hubo buena comunicación y no llego todos los datos de la transmisión) y de overrun (excedente de datos)
 - Generación y detección de pausas opcionales
 - Manejo de señales del modem DTR-DSR-DCD-RI en el USART1
 - Modo Multi-drop para generación y detección de direcciones
- Señales de control para poder trabajar con interfaz RS485
- Protocolo para interfase con tarjetas inteligentes
- Modulación y desmodulación IrDA
- Modos de prueba

4.3.6.1 Controlador serial síncrono

- Provee comunicación serial usada en aplicaciones de audio y telecomunicaciones
- Posee un transmisor y receptor independiente y un divisor de reloj común
- La recepción y la transmisión pueden ser programada para iniciar automáticamente o desde un evento detectado
- La recepción y la transmisión incluyen una señal de datos, una señal de reloj y una señal síncrona

4.3.7 Timers

Cada canal del timer puede ser programado independientemente para ejecutar un amplio rango de funciones incluyendo mediciones de frecuencia, conteo de eventos, medición de intervalos, generación de pulsos, medición de retardos, y modulación de ancho de pulso.

Cada canal tiene tres entradas de reloj externas, cinco entradas de reloj internas y dos señales de entradas o salidas multipropósitos las cuales pueden ser configuradas por el usuario. Cada canal maneja una señal de interrupción interna la cual puede ser programada para generar interrupciones en el procesador.

El bloque del timer/counter tiene tres registros globales los cuales trabajan con los tres canales del timer counter.

Entre las características más importantes tenemos:

- Tiene 3 canales de Timer/Counter de 16 bits
- Ofrece un amplio rango de funciones como:
 - Dimensiones de frecuencia
 - Conteo de eventos

- Dimensionamiento de intervalos
- Generación de pulsos
- Delay
- Modulación por ancho de pulso (PWM)
- Cada canal es configurable y está compuesto por:
 - 3 entradas de reloj externas
 - 5 entradas de reloj internas, las cuales están definidas en la tabla 4.3

Tabla 4.3. Entradas de reloj

| TC Entrada de reloj | Reloj (MCK = frecuencia de reloj) |
|---------------------|-----------------------------------|
| Timer_Clock1 | MCK/2 |
| Timer_Clock2 | MCK/8 |
| Timer_Clock3 | MCK/32 |
| Timer_Clock4 | MCK/128 |
| Timer_Clock5 | MCK/1024 |

- 2 señales de entrada/salida multipropósitos
- 2 registros globales los cuales actúan en los 3 canales del timer/counter

4.3.8 Conversor análogo digital

El ADC esta basado en un registro de aproximación sucesiva (SAR) de 10 bits que convierte de análogo a digital. Este también integra un multiplexor de 8 a 1, haciendo posible la conversión análoga digital de las 8 líneas análogas. La conversión va en un rango de 0V a un $ADVREF$ (voltaje de referencia).

El ADC soporta un modo de resolución de 8 bits o 10 bits, y los resultados de la conversión son guardados en un registro común para todos los canales.

El ADC también integra un modo de Sleep, un secuenciador de conversión y un conector con el canal PDC. Estas características reducen el consumo de potencia y el trabajo del procesador.

Finalmente el usuario puede configurar los diferentes tiempos del ADC, como: el tiempo de inicio y el tiempo de muestreo y espera. Entra las características más importantes tenemos:

- Posee 8 canales para el conversor análogo digital
- Ofrece 8 entradas análogas independientes de 3.3 V
- Posee referencia de voltaje externa para mejorar la exactitud en las entradas de bajo voltaje
- Activación y desactivación individual de cada canal
- Fuente de trigger (disparo para activación del ADC) múltiple
 - Trigger por hardware o software
 - Pin de trigger externo
- El ADC continua trabando cuando el microcontrolador esta en modo Sleep
- 4 de 8 entradas análogas son compartidas con señales digitales

4.3.9 Abastecimiento de potencia

Este integrado tiene seis tipos de pines de suministro de potencia y también integra un regulador de voltaje el cual permite que el dispositivo sea abastecido con un solo voltaje. Los seis pines de suministro de potencia son los siguientes:

- VDDIN, este energiza al regulador de voltaje y el ADC; el rango de voltaje va desde 3.0 V hasta 3.6 V con un voltaje nominal de 3.3 V
- VDDOUT, es la salida del regulador de voltaje y entrega 1.8 V
- VDDIO, energiza las líneas de entrada salida y el dispositivo USB; su rango de voltaje esta entre 3.0 v y 3.6 V con un voltaje nominal de 3.3 V
- VDDFLASH, energiza una parte de la Flash, este voltaje es necesario para la operación correcta de la Flash; el rango de voltaje va desde 3.0 V a 3.6 V, con un voltaje nominal de 3.3 V
- VDDCORE, estos pines de voltaje energizan para que el dispositivo trabaje con lógica; el rango de voltaje va desde 1.65 V a 1.95 V. Este puede ser conectado al VDDOUT por medio de un capacitor.

4.3.9.1 Consumo de potencia

Este chip tiene un consumo de corriente continua menor a 60 μ A en el VDDCORE a 25°C. El consumo de potencia en el VDDCORE es menor a 50 mA a velocidad máxima cuando funciona fuera de la Flash. Bajo las mismas condiciones, el consumo de potencia en la VDDFLASH no excede los 10 mA.

4.3.9.2 Características eléctricas

- La temperatura de operación industrial es de -40°C a 85°C
- Temperatura de almacenamiento de -60°C a 150°C
- El voltaje en los pines de entrada con respecto a tierra es de 1.95V
- Voltaje de operación máximo (VDDIO, VDDIN y VDDFLASH) es de 3.6 V
- Corriente en todas las líneas de I/O es de 100 mA

CAPÍTULO V

HERRAMIENTAS DE DESARROLLO DE PROYECTOS

5.1 ENTORNOS DE DESARROLLO

4.3.4 AdaMULTI

El AdaMULTI es un completo ambiente de desarrollo utilizado para aplicaciones integradas usando Ada 95, C, C++ y la integración entre C++ y FORTRAN. El AdaMULTI puede funcionar con Windows y UNIX y soporta depuración remota de una variedad de tarjetas. Este entorno de desarrollo provee una interfaz grafica directa con todos los compiladores Green Hills y puede trabajar con múltiples lenguajes. Contiene las herramientas necesarias para desarrollar proyectos de programación entre las que podemos enumerar:

- El constructor de proyectos
- Depurador de Source-level
- Analizador de eventos
- Verificación de errores en tiempo real
- Análisis del código de respaldo
- Versiones del sistema de control
- Perfil de rendimiento
- Examinador gráfico

Si se toma por separado los componentes del AdaMULTI se tiene que cada uno de ellos es el más avanzado en su clase. Este IDE representa mucho más que una colección de software de alto nivel, es uno de los más destacados

ambientes de desarrollo. Sus componentes mantienen una constante comunicación entre ellos para saber como están trabajando y no se produzcan fallas de una manera muy rápida y eficiente.

Por ejemplo en el ciclo de edición-compilación-depuración, cuando un error es descubierto con el depurador del AdaMULTI, una señal de alerta es dada en la línea que contiene el error y dando un clic se abrirá un editor para hacer la respectiva corrección, tan pronto como las modificaciones son hechas en el archivo, el editor chequea las salidas del sistema de control, dando otro clic se guardará los cambios y chequeará el archivo de respaldo y por último construirá el programa y reiniciará la depuración.

4.3.5 Visual X-Tools

- Este IDE ofrece SmartPaste
- Código de navegación
- Argumentos completos
- Soporte para regrabado
- Ofrece un mejor lenguaje de programación
- Tiene soporte para seleccionar código por bloques
- Remplaza y lista los símbolos automáticamente
- Busca y remplaza directorios y archivos
- Integra FTP para operar transparentemente con archivos remotos
- Permite combinar la edición de un programa, tomando la versión original, luego separa las modificaciones que se la hagan y las guarda dentro de un archivo
- La configuración y organización del código ofrece soporte para espacios de trabajo y proyectos usando cuadros de diálogos amigables al usuario.
- Tiene una completa integración con GNU-Tools, Visual GDB y los comandos de línea GDB para que puedan ser llamados directamente desde “Build” que se encuentra en el menú o desde una mini barra de herramientas que se encuentra sobre la venta “Build” para de esta forma poder compilar el programa.

4.3.6 Embest IDE for ARM

Embest IDE es una solución completa para desarrollar aplicaciones integradas con chips ARM. Provee una simple y versátil interfaz gráfica para el usuario y herramientas para desarrollar proyectos que utilicen arquitectura ARM. Es un ambiente de desarrollo integrado que facilita manejar y construir proyectos, establece y dirige las comunicaciones entre el host y la tarjeta, corre y depura las aplicaciones realizadas. Embest IDE puede correr entre Windows 98 y Windows NT. Trabaja con los siguientes lenguajes de programación: ANSI C y ARM.

Entre sus principales características tenemos:

- Un integrado ambiente de desarrollo
- Facilidad para manejar proyectos
- Integra un editor de código fuente
- Compiladores GNU, assembler y linker
- Librerías GNU para ANSI C
- Soporte para set de herramientas ARM
- Corrección de errores
- Simulador para el set de instrucciones ARM
- Ejemplos de código fuente para Atmel / Samsung / Cirrus logia y procesadores con núcleo ARM
- Documentación online

Embest IDE ofrece soporte para microcontroladores basados en ARM7 o ARM9, como:

- ATMEL: AT91 series (AT91M40800, AT91F40816, AT91R40807, AT91FR4081, AT91M63200, AT91M55800, AT91RM9200)
- SAMSUNG: S3C3410, S3C44B0, S3C4510, S3C2410
- PHILIPS: LPC2114-2194, LPC22xx (LPC2104,2105,2106,2210)
- CIRRUS: EP7209, EP7212, EP7312, EP9312, EP9301
- OKI: ML671000, ML674000
- Sharp: LH75400, LH75410
- ST Microelectronic: STR710FZ2T6

5.2 EMULADORES

Los emuladores son utilizados para depurar, sugerir soluciones y probar sistemas integrados complejos. Los emuladores In-Circuit (integrados en el circuito) (ICEs) han presentado algunas fallas por lo cual se ha optado por remplazarlos por los emuladores JTAG. Estos emuladores no necesitan ninguna versión especial de un procesador para operar, también sus herramientas y actualizaciones son muy fáciles de encontrar. El tiempo ahorrado con los emuladores JTAG puede ser usado para mejorar sus características y elevar la experiencia del usuario.

Características del emulador JTAG

Tabla 5.1. Características emulador JTAG

| | |
|--|--|
| Reset del procesador | Envía un comando JTAG o un pulso a la línea de reset para resetear el procesador. |
| Lectura de Memoria | Realiza un ciclo de lectura de memoria usando JTAG, ejecuta el ciclo y retorna el resultado. |
| Escritura de la memoria | Realiza un ciclo de escritura de memoria usando JTAG, ejecuta el ciclo y escribe un valor en la memoria. |
| Lectura del registro | Lectura on-chip u off-chip del registro. |
| Escritura del registro | Escritura on-chip u off-chip del registro. |
| Funcionamiento del procesador | Comienza el procesador a funcionar. |
| Set Break Points | Hace que el procesador detenga la ejecución de una instrucción, lectura, escritura o acceso a una localización de datos. |
| Para del procesador | Para el procesador cuando esta funcionando. |
| Tabla 5.1. Características emulador JTAG (continuación) | |

| | |
|---|--|
| Ejecución del hardware de prueba JTAG | Prueba la memoria en la tarjeta del sistema. |
| Inicializa dispositivos | Envía comandos a los dispositivos para configurarlos apropiadamente para ciertas operaciones. |
| Conexión a una tarjeta en funcionamiento o detenida | Permite conectar el JTAG a una tarjeta sin distorsionar el estado de la tarjeta. La información de la tarjeta puede ser leída mientras el sistema este operando o este parado. |
| ID FLASH | Detecta el tipo de FLASH y como esta configurada en el sistema. |
| Dispositivos de programación de la FLASH | Utiliza dispositivos propios de programación. |
| Dispositivos de borrado de la FLASH | Utiliza dispositivos propios de programación. |
| Dispositivos de verificación de la FLASH | Verifica que el dispositivo este correctamente programado. |
| Comandos de ejecución, macros e inicialización de archivos. | Permite que el usuario defina los comandos más utilizados. |
| Depuración con GNU/GDB | Soporte para corregir errores basada en GDB y Guardian-SE JTAG. |
| Download | Descarga código a la RAM. |
| Registros personalizados | Crea registros personalizados para dar soporte a ciertos tipos de dispositivos. |
| BSDL Pin Scan | Usa un archivo BSDL para mostrar el estado de todos los pines del procesador que usan JTAG. |
| Depuración integrada por LINUX | Permite corregir errores por medio de LINUX. |
| Dissasemble | Despliega la memoria como instrucciones del procesador. |

5.2.1 Keil ULINK™ USB-JTAG Adapter

Permite conectar el puerto USB a la tarjeta de hardware de la computadora (por medio de JTAG o ACDS) para realizar la depuración de los mismos mientras estos se están ejecutando.

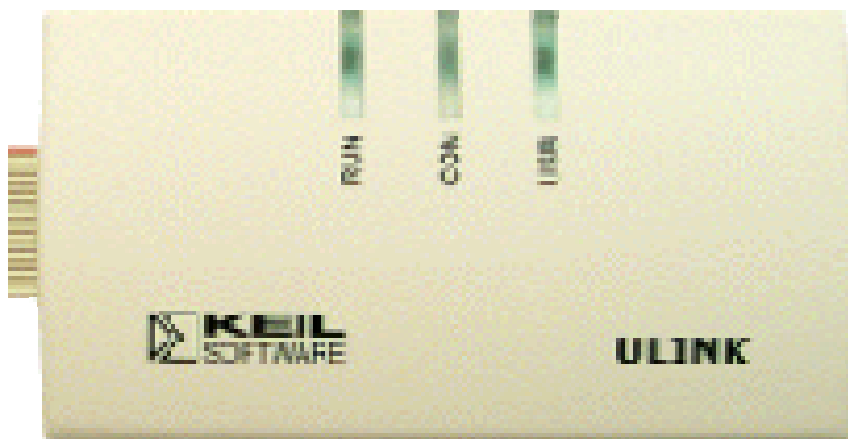


Figura 5.1. Adaptador USB-JTAG

Este puede ser usado para: corrección de errores on-chip y para programar la memoria flash. Usando el Keil uVision IDE/Debugger con el adaptador UNLINK, es mucho más fácil crear, descargar, y probar aplicaciones integradas en hardware de tarjetas actuales.

5.2.2 Embest PowerICE for ARM

Es un mejorado emulador JTAG en paralelo para procesadores ARM. Embest PowerICE provee para ARM un poderosa herramienta, rápida, fácil de usar y de bajo costo para mejorar el rendimiento.



Figura 5.2. Embest PowerICE

Embest PowerICE puede comunicarse con núcleos ARM por medio de JTAG usando los pines analizadores. Este emulador no usa tarjeta de memoria y no requiere puerto para conectar la tarjeta del sistema. El usuario puede controlar la tarjeta del procesador completamente con el Embest PowerICE, esto quiere decir que uno mismo puede iniciar, parar, ejecutar paso a paso, leer y escribir desde y hacia los registros, memoria y sistema de entradas salidas, también se puede descargar código a la RAM de la tarjeta, programar on-chip o en una FLASH. Embest PowerICE intensifica la velocidad de descarga de código para facilitar las pruebas y depuración de errores.

Entre las principales características tenemos:

- Da soporte para los núcleos ARM7TDMI, ARM720T, ARM9TDMI, ARM920T, ARM922T, ARM940T
- Da soporte a un reloj en tiempo real para poner interrupciones en el programa apropiadamente
- Puede escribir datos a la memoria flash de la tarjeta del sistema a través de la herramienta de programación del Embest Flash u otro software descargado
- Con el Embest IDE para ARM y Embest Power ICE, el usuario puede desarrollar completamente trabajos basados en procesadores ARM

Embest PowerICE tiene un puerto en paralelo para conectar al puerto paralelo de la computadora y una interfaz JTAG para conectar a la tarjeta del sistema.

Los niveles de voltaje de la tarjeta de interfaz del Embest PowerICE dependen de los niveles de entrada de voltaje, que por lo general están entre 3V y 5V. El PowerICE provee tres leds que muestran el estado de operación,

5.2.3 Embest Emulator for ARM

Es un emulador JTAG estándar para procesadores ARM, es fácil de usar, de bajo costo y presenta un alto rendimiento. Es un dispositivo accesible y económico para desarrollar cualquier trabajo.



Figura 5.3. Embest Emulator

Posee las mismas características que el Embest PowerICE pero se diferencian en lo siguiente:

- La primera diferencia esta en la velocidad de descarga y depuración, el PowerICE es más rápido que el Emulador
- El PowerICE puede ser energizado desde una tarjeta de desarrollo o desde una fuente de potencia externa, en cambio el Emulador solo puede ser energizado desde la tarjeta
- En PowerICE se pueden configurar diferentes velocidades de depuración, en el Emulador no se puede hacer esto.

5.2.4 Embest UNetICE for ARM

Es una nueva versión de emuladores JTAG de tiempo real de alta velocidad hecho por Embest. UNetICE conecta el puerto USB o el puerto Ethernet a la interfaz JTAG de la tarjeta de desarrollo ARM y permite la programación y depuración de la Flash. UNetICE da soporte para:

- Descargar programas de la tarjeta
- Examinar la memoria y registros
- Permite insertar múltiples interrupciones
- Ejecución de programas en tiempo real
- Programación de la Flash on-chip

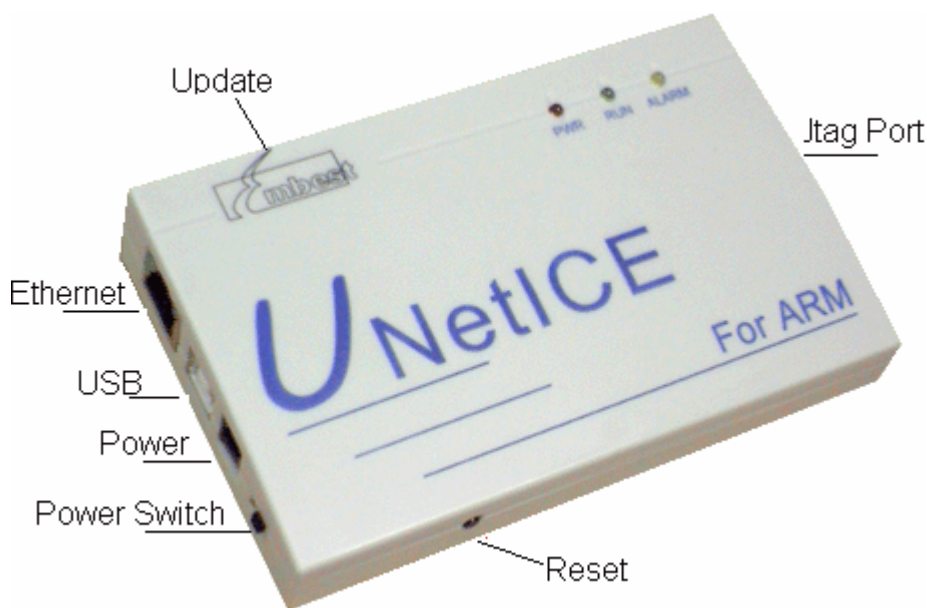


Figura 5.4. UNetICE Emulator

Este emulador incluye.

- Puerto JTAG (puerto de 20 pines) para conectar a la tarjeta de desarrollo
- Puerto USB o Ethernet para poder conectarlo al host del PC

- Botones de reset y actualización
- Fuente de potencia externa y switch de potencia
- Indicadores de estado del emulador de colores rojo, verde y amarillo

Características de UNetICE:

- Da soporte a las arquitecturas ARM7 y ARM9
- Puede trabajar con Windows 98/NT/2000/XP
- Posee un puerto JTAG de 20 pines para ser conectado a la tarjeta
- Soporta interrupciones de software y hardware
- La velocidad de descarga y depuración esta entre los 200 y 800 Kbytes/s
- Posee soporte de Internet para desarrollo y depuración remota
- Capacidad de actualización por firmware en no más de 30s

5.2.5 Nohau Emulator

- Depuración por medio de la interfaz JTAG la cual da soporte a todos los núcleos de las arquitecturas ARM7 y ARM9
- Incluye una interfaz de usuario, puede trabajar con Windows 98/ME/2000/XP
- Conexión USB de alta velocidad. El puerto USB energiza el hardware de depuración
- Apoyo completo para trabajar con los sets de instrucciones Thumb o ARM
- Posee uno de los más populares ambientes de desarrollo
- Soporte para programación de la Flash
- La comunicación con la tarjeta y el depurador permite que este lea y escriba la memoria mientras la tarjeta este funcionando.
- Incorpora en la línea de código de assembler el tipo de instrucción que es utilizada
- Soporte para múltiples dispositivos JTAG



Figura 5.5. UNetICE Emulator

5.2.6 TRACE32-FIRE

Desde que los procesadores RISC operan en buses con frecuencia mayores a los 100MHz, una redefinición de los emuladores tradicionales fue necesaria. El resultado de este nuevo concepto es el TRACE32-FIRE que fue desarrollado especialmente para aplicaciones con RISC. Este emulador garantiza una alta rentabilidad en calidad y soporte.



Figura 5.6. Emulador TRACE32-FIRE

Presenta las siguientes características:

- Emulador RISC de alta velocidad
- Soporte para 64 bits de datos y 32 bits de direcciones
- Memoria de emulación de puerto dual de alta velocidad de 16 Mbyte
- Trabaja en un amplio rango de voltaje tanto para microprocesadores y microcontroladores este rango esta entre 3.3V y 5V
- Trabaja con una gran cantidad de lenguajes de programación como: C, C++, JAVA, ADA, Perl, etc.
- EL TRACE32-FIRE puede ser unido a un PC en cualquier estación de trabajo o sistema informático.
- Los controladores de software son aconsejables para todos los sistemas operativos y para todas las interfaces gráficas de usuario
- Provee un depuramiento simbólico de las aplicaciones multitareas y da un análisis detallado del comportamiento del sistema en tiempo real
- EL TRACE32-FIRE ofrece un análisis de rendimiento en el programa, funciones o módulos bases en áreas definidas por el usuario
- Modo de emulación activo y pasivo
- Este emulador es controlado por un CPU RISC de alta velocidad
- Alta compatibilidad con sistemas de depuración ICE e ICD

5.3 TARJETAS DE DESARROLLO

5.3.1 Embest ATEBSAM7S

La tarjeta de desarrollo Embest ATEBSAM7S amplía todas las funciones del microcontrolador ATEBSAM7S, y da soporte para otros dispositivos de hardware entre los cuales se tiene: un led de 8 segmentos, interfaz para botones de pulsación y ADC. Con esta tarjeta de desarrollo el usuario puede evaluar fácilmente el circuito integrado AT91SAM7S64, empezar a desarrollar el código en la tarjeta de desarrollo empleándola como un diseño sólido de referencia.

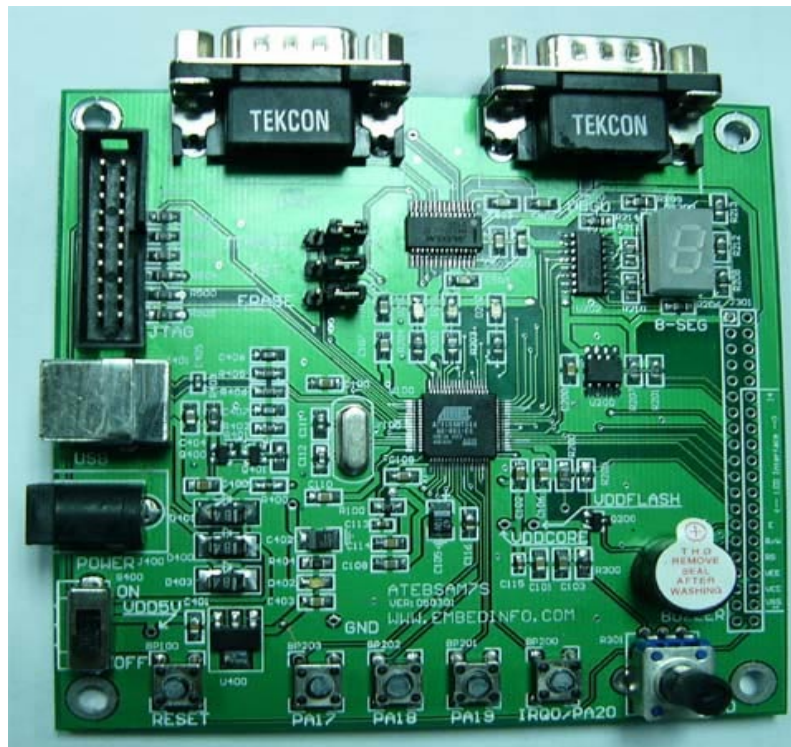


Figura 5.7. Tarjeta de desarrollo Embest ATEBSAM7S

Esta tarjeta presenta las siguientes especificaciones:

- Sus dimensiones son 100x100 mm
- La temperatura de trabajo esta entre 45 y 85 °C
- Entrada de voltaje de 5V
- Posee un puerto para dispositivo USB
- Un puerto serial para RS232
- Una interfaz SPI
- Interfaz PWM
- Un controlador ADC
- Varios puertos de I/O
- Un botón de reset y cuatro entradas de botones para propósitos generales
- Interfaz LCD
- Interfaz JTAG estándar de 20 pines

5.3.2 Embest ATEB9200

La tarjeta de desarrollo ATE9200 esta basada en el microcontrolador AT91RM9200 y trabaja a 180 MHz, esta tarjeta esta provista por varios ejemplos de programas. Su ambiente de desarrollo es completamente hecho en Linux lo cual cubre todas las necesidades de software y hardware para realizar diseños integrados con resultados rápidos. Entre sus aplicaciones más conocidas tenemos: sistemas médicos y automatización industrial que requieren ethernet y USB. Los usuarios pueden tener un completo uso de esta tarjeta para evaluación, desarrollo y aplicaciones.

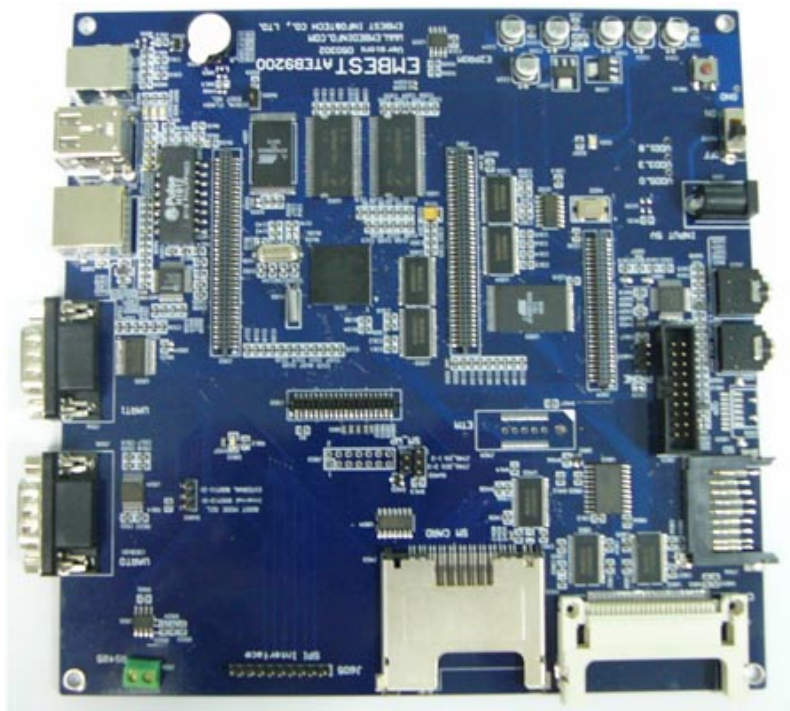


Figura 5.8. Tarjeta de desarrollo Embest ATEB9200

Esta tarjeta presenta las siguientes especificaciones:

- Sus dimensiones son 180x180 mm
- La temperatura de trabajo esta entre 45 y 85 °C
- Entrada de voltaje de 5V
- 2 M de memoria flash en paralelo
- 32 M de SDRAM

- 8 M de flash de datos
- Posee un puerto para dispositivo USB
- Una interfaz IrDA
- Una interfaz para trabajar con el protocolo RS485
- Una interfaz SPI que puede conectarse a un módulo CAN
- Un módulo de interfaz de ethernet 10/100
- Interfaz SD/MMC
- Interfaz SM
- Varios puertos de I/O
- Interfaz JTAG estándar de 20 pines

5.3.3 CSB637 (Atmel AT91RM9200 based uSBC)

Esta es una económica tarjeta de desarrollo que ofrece un núcleo ARM920T que trabaja a 184Mhz e integra puerto ethernet 10/100, dispositivo USB, Host USB, interfaz de audio I2S, entradas y salidas seriales y un módulo de rastreo integrado. A continuación se muestran las principales características:

Tabla 5.2. Características CSB637

| Características | CSB637 |
|------------------------|---|
| Núcleo | ARM920 a 184MHz, 16K de cache |
| SDRAM | 64Mbyte |
| FLASH | 8Mbyte |
| Controlador LCD | S1D13506 – 800x600 - LCD de 16 bits |
| Serial (UART) | RS-232 – Tres puertos adicionales UART |
| Ethernet | On-chip 10/100 |
| Reloj de tiempo real | RTC interno |
| Expansión del bus | Bus de dirección de 20 bits y bus de datos del CPU de 16 bits |
| Dispositivo USB | Puerto USB de 2.0 |
| Audio | Soporte I2S |
| JTAG | Estándar ARM JTAG |
| Voltaje | Entrada de 5V a 35V |

5.3.4 Embest ATEB40x

La tarjeta de desarrollo ATEB40X habilita la evaluación y desarrollo de código en tiempo real, da soporte a la series de microcontroladores AT91x40 con sus varias opciones de memorias. Esta tarjeta esta constituida por una tarjeta maestro y una tarjeta esclavo. La tarjeta esclavo es el núcleo del microcontrolador de la tarjeta.

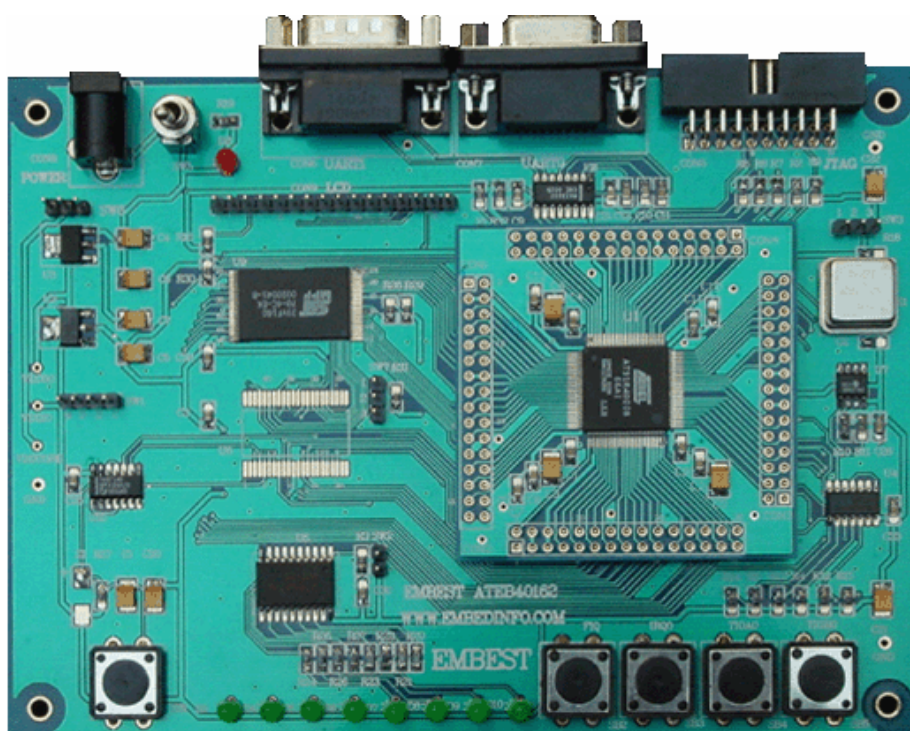


Figura 5.9. Tarjeta de desarrollo Embest ATEB40x

Esta tarjeta presenta las siguientes especificaciones:

- Tarjeta de desarrollo basado en las series AT91x40
- Trabaja solo con 5V
- Dos puertos seriales
- Botón de reset
- Cuatro botones para que el usuario pueda usarlos como el desee
- Ocho leds
- 1 MB de Flash

- 512 Bytes de EEPROM
- Conector de expansión de LCD de 16 pines
- Interfaz JTAG estándar de 20 pines

5.3.5 Embest AX4510

Embest Ax4510 es una plataforma aconsejable para el desarrollo de código del microcontrolador RISC Sansumg S3C4510B, especialmente en aplicaciones de redes y generales.

Embest Ax4510 esta constituido por una flash, SDARM, interfaz LCD, dos puertos de comunicación serial, interfaz ethernet, interfaz USB, interfaz PS2, puerto IDE exterior, interfaz JTAG y leds. Esta tarjeta proporciona un alto rendimiento y soluciones de bajo costo para diseños integrados.

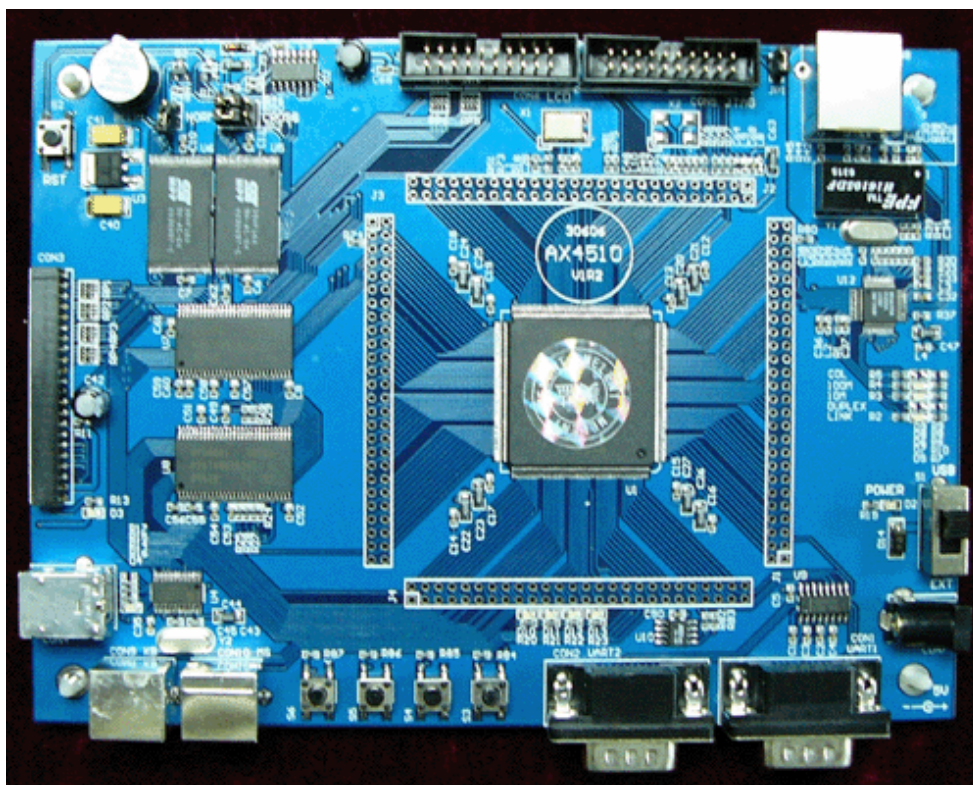


Figura 5.10. Tarjeta de desarrollo Embest AX4510

Esta tarjeta presenta las siguientes especificaciones:

- Sus dimensiones son 177x130
- Trabaja solo con 5V
- Posee 4 Mbytes de Flash
- 16 Mbytes de SDRAM
- 2 Kbyte de EEPROM
- 2 RS232
- Conector USB
- Conector para ethernet (RJ45) 10M/100Mbps
- Puerto LCD
- Dos puertos PS2
- Cuatro botones de propósitos generales
- Botón de reset
- Cuatro leds
- Conector para interfaz JTAG de 20 pines
- Todos los pines del CPU se pueden expandir

5.3.6 Embest S3CEV40

Embest S3CEV40 es aconsejable para desarrollo de código de los microcontroladores Samsung S3C44B0X de 16/32 bits RISC para dispositivos portátiles y aplicaciones en general.

El S3C44B0X consta de un núcleo en su CPU RISC de 16/32 bits, 8 KB de cache, SRAM interna opcional, controlador de LCD, dos canales UART, canal DMA, controlador del sistema, timers con PWM, puertos I/O de propósitos generales, RTC, ADC, interfaz SIO y PLL para el reloj.

El Embest S3CEV40 consta de S3C44B0X, Flash, SDRAM, puertos de comunicación serial, interfaz de Internet, interfaz USB, interfaz de teclado, puertos IDE exterior, RTC, interfaz JTAG. El Embest S3CEV40 EVB provee un alto rendimiento y soluciones de bajo costo para aplicaciones integradas.

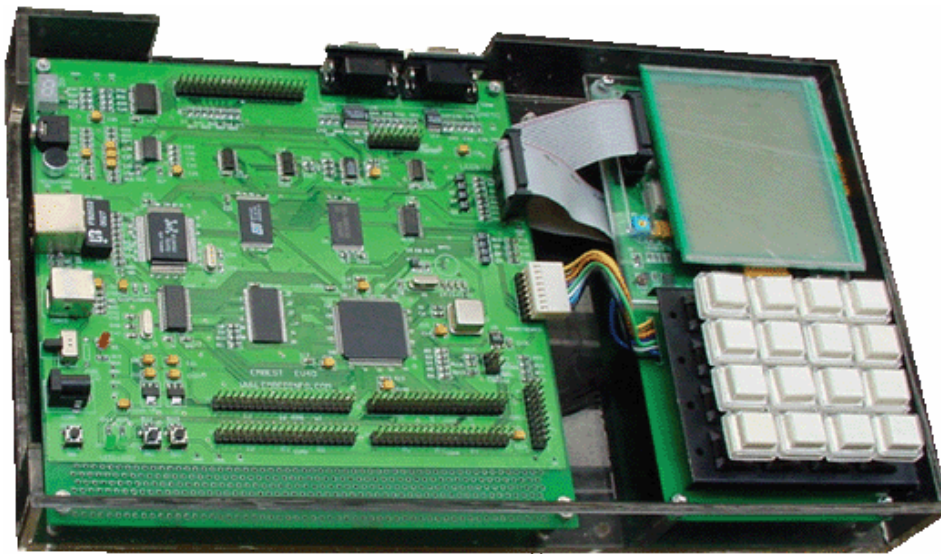


Figura 5.11. Tarjeta de desarrollo Embest S3CEV40

Presenta las siguientes especificaciones de hardware:

- Sus dimensiones son de 190x190mm
- Sea alimenta con 5V DC o de la fuente de potencia de puerto USB
- Flash de 2 Mbytes
- SDRAM de 8 Mbytes
- Protocolo de comunicación RS232
- Conector USB
- Conector de interfaz Ethernet
- LCD opcional de 320x240
- Botón de reset
- Presenta dos botones para generar interrupciones y dos leds
- Conector de interfaz JTAG de 20 pines

Especificaciones de software:

- Software de inicialización
- Ejemplos de código y software de prueba

- Ejemplos de aprendizaje para instrucciones en assembler ARM
- Ejemplos de aprendizaje para el funcionamiento de procesadores ARM en diferentes modos de operación
- Ejemplos de aprendizaje de instrucciones THUMB de ARM
- Ejemplos de aprendizaje para conocer como trabajar con interrupciones
- Programa de prueba para usar RTC
- Programa de prueba para acceso a memoria
- Programa de prueba para trabajar con leds y mostrar la forma de uso de las GPIO
- Programa de prueba para la lectura y escritura de la Flash
- Programa de prueba para timers
- Programa de prueba para el UART
- Programa de prueba para conocer como funciona el teclado
- Una aplicación de una red de comunicación para probar el puerto Ethernet

La tarjeta de desarrollo lista las siguientes herramientas:

- Tarjeta de desarrollo Embest S3CeV40
- Un cable serial DB9
- Fuente de voltaje de 5.0 V DC
- Cable USB
- El CD-ROM incluye:
 - Manual de usuario
 - Bosquejo esquemático del circuito
 - Software de inicialización

Esta tarjeta de desarrollo es capaz de soportar diferentes tipos de sistemas de depuración usando una interfaz ICE.

5.3.7 Keil MCB2100

La tarjeta de desarrollo Keil MCB2100 da soporte a la familia ARM LPC2100 de Philips y permite crear programas de prueba para trabajar con arquitecturas más avanzadas. El Keil MCB2100 se puede conectar al PC usando el puerto serial o la interfaz JTAG. Esta tarjeta consta de dos interfaces CAN y dos interfaces seriales que hacen de esta el punto de partida para futuros proyectos ARM.

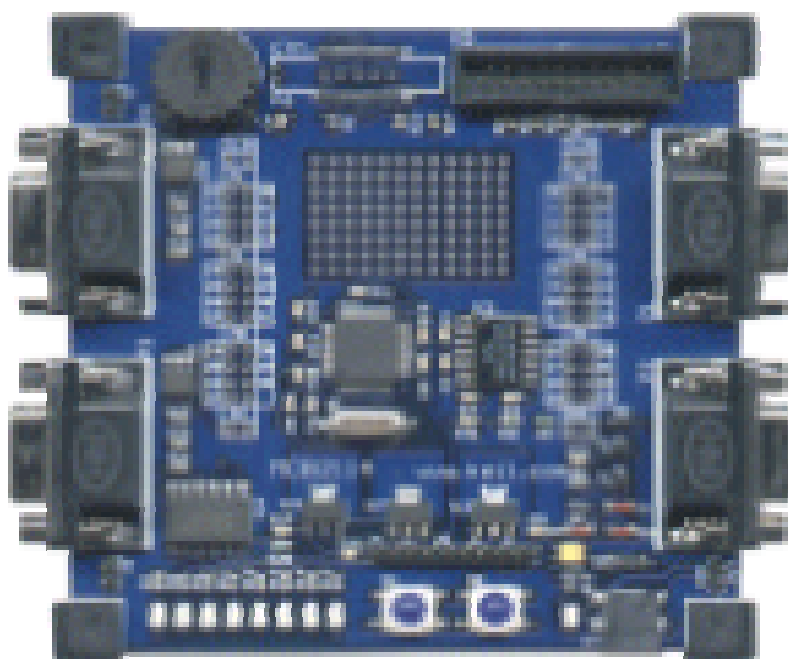


Figura 5.12. Tarjeta de desarrollo Keil MCB2100

Los requerimientos del sistema para poder trabajar con esta tarjeta son los siguientes:

- Un PC con un puerto serial disponible
- Windows 95/98/ME/NT/2000/XP
- CD-ROM

El MCB2100 presenta como herramienta de evaluación el RealView MDK-ARM. Esta herramienta ayuda iniciar la escritura de los programas, pruebas del microcontrolador y sus capacidades.

5.3.8 KB9202

Es una tarjeta de computadora basada en el núcleo ARM920T la cual es de bajo costo, trabaja con aplicaciones integradas, y desarrollo. El ARM920T esta capacitado para trabajar en LINUX y con el compilador GNU el cual es usado en sistemas que trabajan con WINDOWS.

Usando el cable serial se puede comenzar el desarrollo de aplicaciones integradas en un periodo muy corto de tiempo. Esta tarjeta incluye CD con archivos necesarios para la compilación de programas y el Linux Kernel, estos CD también incluyen ejemplos útiles y código fuente el cual se puede usar de referencia.



Figura 5.13. Tarjeta de desarrollo KB9202

Esta tarjeta de desarrollo presenta los siguientes aspectos técnicos:

- Un núcleo ARM altamente integrable el cual trabaja a una velocidad 200MIPS a 180 MHz
- Memoria independiente de instrucciones de 16KB y cache de datos de 16KB
- Su dimensión es de 100 x 160 mm

- SDRAM de 64 Mbps
- Ethernet 10/100
- Puerto USB de 2.0
- Interfaz para LCD en paralelo
- Protocolo de comunicación RS-232 y RS-485
- SRAM interna de 16KB
- ROM interna de KB
- Interfaz para memoria externa
- El reloj trabaja sobre los 180 MHz
- Reloj de tiempo real on-chip y DMA
- Interfaz JTAG
- Posee sensor de temperatura digital
- 3 leds indicadores

5.3.9 Embest LPCEB2000

La tarjeta de desarrollo LPCEB2000 esta conformada por un módulo de CPU y un panel de expansión que es llamado LPCEB2000-I cuyas características son:

- Su dimensión es de 182x139 mm
- Trabaja en los rangos de temperatura de -45 a 85 °C
- Voltaje de entrada es de 9V
- Dos puertos seriales
- Un puerto CAN
- Un botón de reset
- Teclado 4x4
- Ocho luces indicadoras
- Display de 8 dígitos de 8 segmentos
- Salida de señal análoga a parlante
- Puerto JTAG estándar de 20 pines

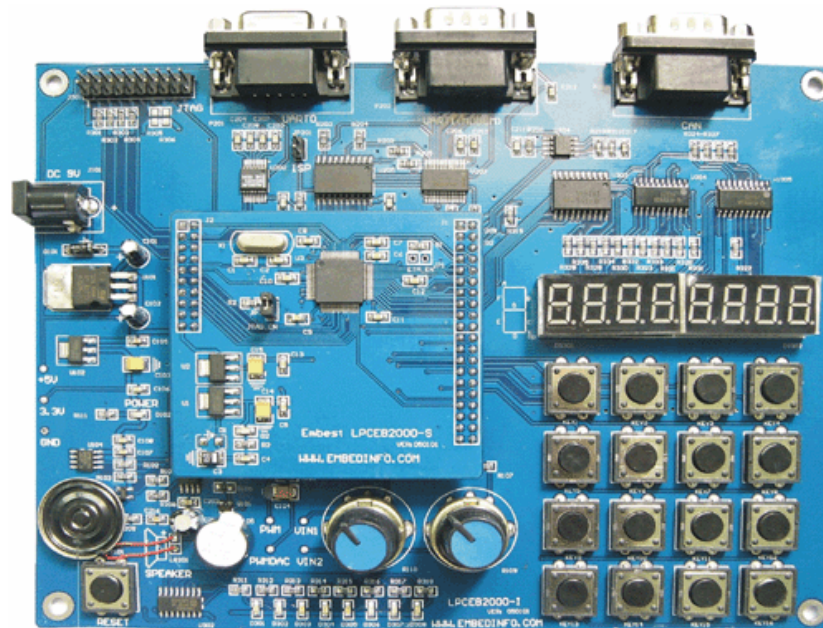


Figura 5.14. Tarjeta de desarrollo Embest LPCEB2000

5.3.10 Embest S3CEB2410

La tarjeta de desarrollo S3CEB2410 es una plataforma de desarrollo basada en el procesador Samsung S3C2410X. Esta tarjeta realiza todas las funciones del procesador permitiendo que el usuario pueda fácilmente desarrollar sus proyectos en esta plataforma, o use esta como referencia para desarrollar aplicaciones.

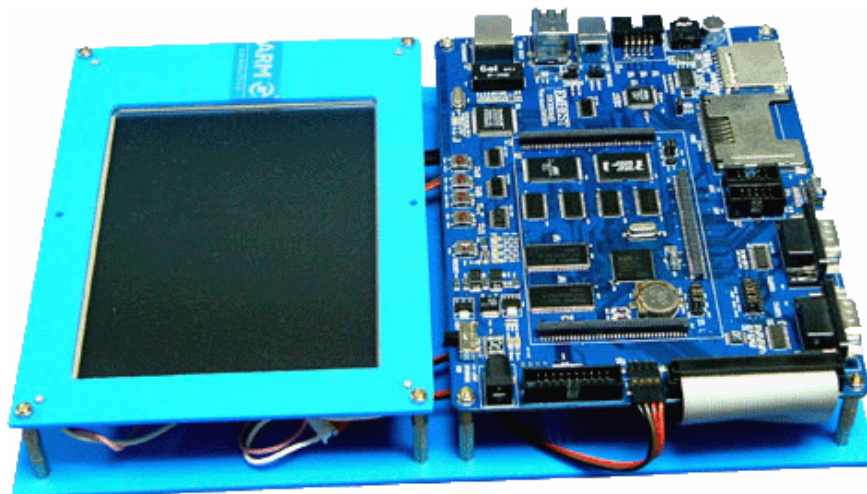


Figura 5.15. Tarjeta de desarrollo Embest S3CEB2410

Esta tarjeta presenta las siguientes especificaciones:

- Sus dimensiones son 190x140 mm
- Su sistema de potencia funciona de la siguiente forma: la tarjeta usa un transformador de 5V, 2A DC el cual regula el voltaje a 3.3 V para el CPU, I/O y periféricos
- Sistema de reloj: Puede funcionar con un cristal oscilador externo de 12 MHz
- SDRAM de 64 Mbytes
- RTC (reloj de tiempo real)
- Tres canales de UART
- 1 canal para dispositivo USB
- Interfaz SPI
- Interfaz de transferencia modular ADC de 10 bits
- Interfaz ethernet
- Cuatro botones para generar interrupciones externas o I/O
- Cuatro leds indicadores y dos indicadores de potencia
- Interfaz JTAG de 20 pines
- LCD a color 640*480 y Touch panel (optional)

5.3.11 Embest STDV710A

El STR710F-Z2T6 es un poderoso microcontrolador ARM con FLASH y RAM integradas. Esta basado en el núcleo ARM7TDMI de 32 bits presenta un extenso set de funciones de periféricos y I/O mejoradas. Esta microcontrolador posee una alta conectividad por lo cual utiliza varias interfaces como: CAN, USB, SPI, I2C, UART, EMI (External Memory Interface), etc. Este controlador es diseñado para tarjetas de aplicaciones de control industrial como: automatización de fábricas, máquinas de venta y equipos de prueba, también podemos encontrar en el campo de las comunicaciones como en conexiones y protocolos de transmisión.

La tarjeta de desarrollo Embest STDV710A se caracteriza por ser una plataforma de evaluación de bajo costo para dispositivos STR710F-Z2T6. La tarjeta despliega todas las características del microcontrolador STR710F-Z2T6 e integra varios periféricos adicionales incluyendo 512 Kbytes de SRAM y 2Mbytes

de Flash, leds, botones, LCD, ethernet, CAN, SPI, I2C, interfaz de dispositivo USB para crear plataformas de pruebas.

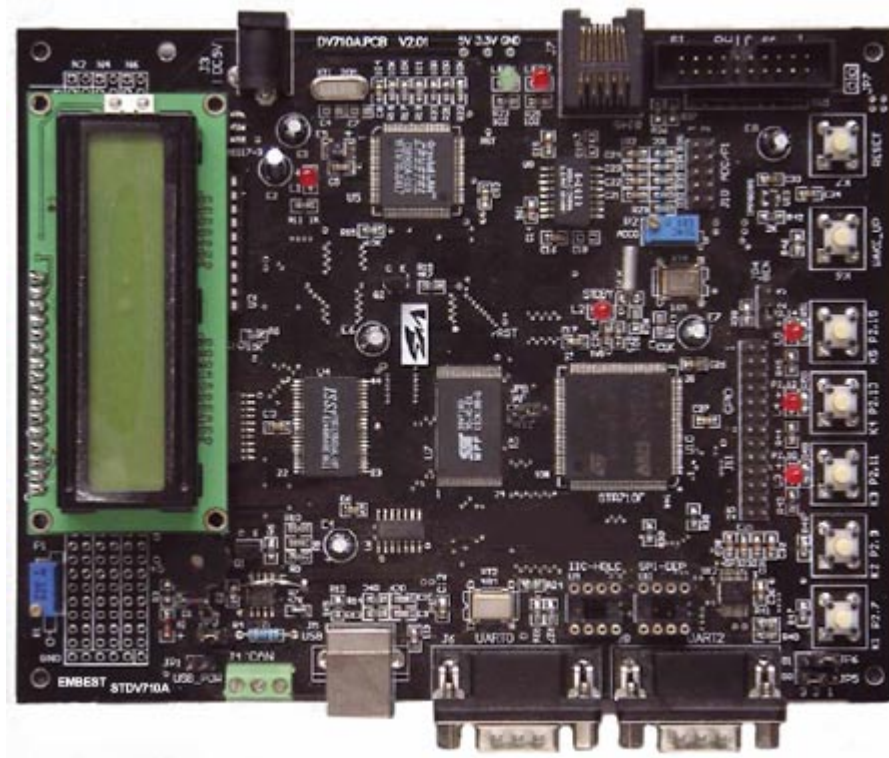


Figura 5.16. Tarjeta de desarrollo Embest STDV710A

Esta tarjeta presenta las siguientes características:

- Sus dimensiones son de 163x124 mm
- Rango de temperatura de trabajo entre -40 y 85 °C
- Potencia de entrada de 5.0V
- Interfaz de ethernet de 10Mbps
- Interfaz USB 2.0
- Interfaz de comunicación CAN 2.0
- Puerto UART0 y UART2 con 2 protocolos de comunicación RS-232
- Interfaz de periféricos seriales (SPI)
- Interfaz de LCD
- 5 leds indicadores
- 4 canales ADC
- 1 botón de reset

- 1 botón de wake up
- 5 teclas para uso general

5.4 Tabla comparativa de costos y características

ANEXO 2. Tabla comparativa de costos y características de ambientes de desarrollo

ANEXO 3. Tabla comparativa de costos y características de los emuladores

ANEXO 4. Tabla comparativa de costos y características de las tarjetas de desarrollo

CAPÍTULO VI

ENTORNO DE DESARROLLO (IDE), KEIL UVISION3 PARA EL INTEGRADO AT91SAM7S64

5.5 DESCRIPCIÓN GENERAL DEL KEIL UVISION3

El uVision3 es un popular ambiente de desarrollo (IDE) desarrollado por Keil, combina manejo de proyectos, edición de código fuente, depuración del programa y programación de la flash en un simple y poderoso ambiente.

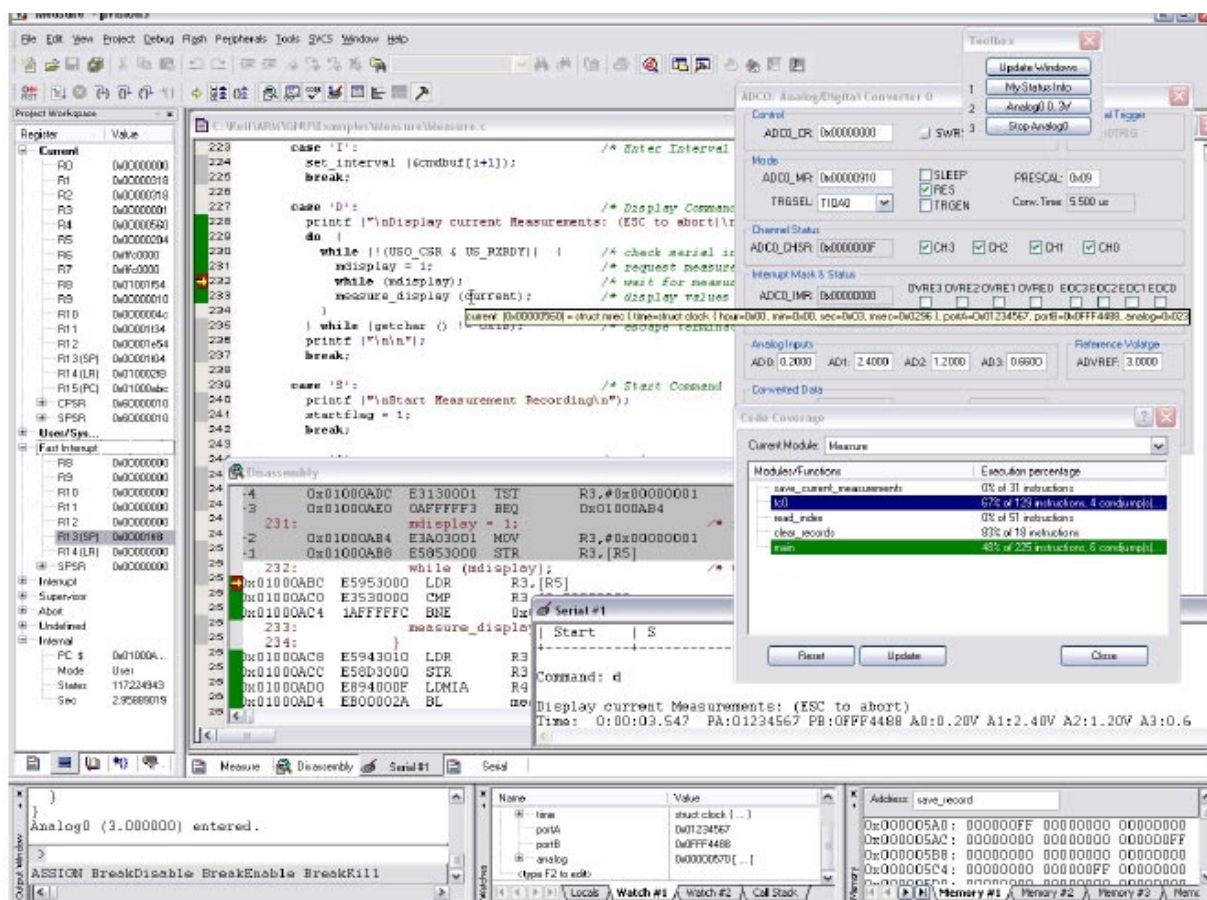


Figura. 6.1. Entorno de desarrollo Keil uVision3

Para el desarrollo de otros tipos de proyectos, el uVision3 incluye nuevas características como son: funciones de navegación, planillas de edición, buscadores con nuevas cosas, asistentes para configuración, analizadores lógicos, Simulación CAN e I²C, programación de la Flash y depuración JTAG.

El keil uVision3 IDE da soporte para más de 700 dispositivos incluyendo chips ARM, XC16x, C16x, STIO, 251 y microcontroladores basados en el 8051.

6.1.1 Manejo de proyectos

El uVision3 asegura un fácil y consistente manejo de proyectos. Un simple archivo de proyecto almacena el archivo fuente y guarda la información configurada para posteriormente compilarla, ensamblarla (assembler), enlazarla (linker), depurar (debugger), cargar en la Flash y otras utilidades.

El menú del proyecto da acceso a los archivos del proyecto y cuadros de diálogos para manejar el proyecto.

Open Project. Carga los archivos de proyectos existentes.

New Project. Permite al usuario crear sus propios proyectos.

6.1.2 Base de datos de los dispositivos

Cuando se inicia un nuevo proyecto en uVision3 se debe seleccionar una tarjeta para el dispositivo desde la base de datos y automáticamente el uVision3 colocará las opciones de herramientas requeridas y personalizará los cuadros de diálogos.

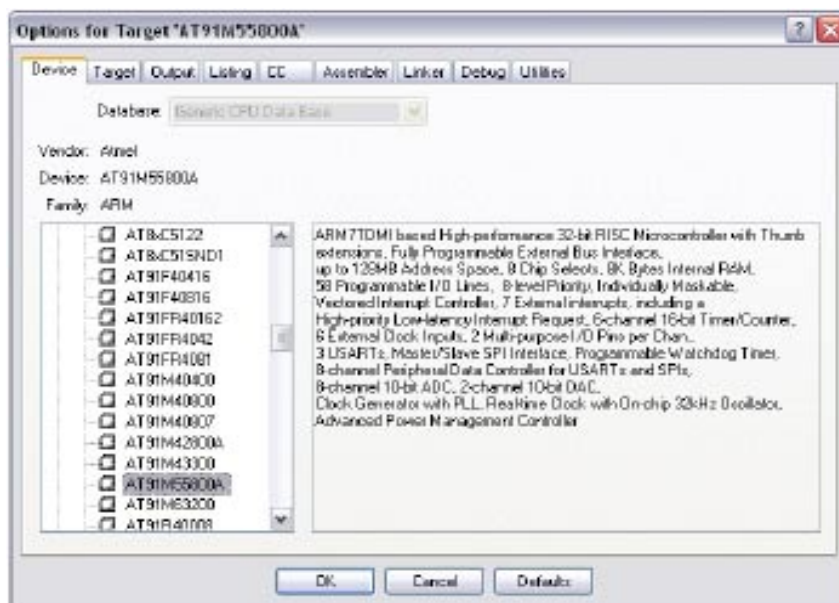


Figura. 6.2. Opciones de tarjeta

El uVision3 solo muestra las opciones que son pertenecientes al dispositivo seleccionado previniendo de esta manera la selección de directorios incompatibles.

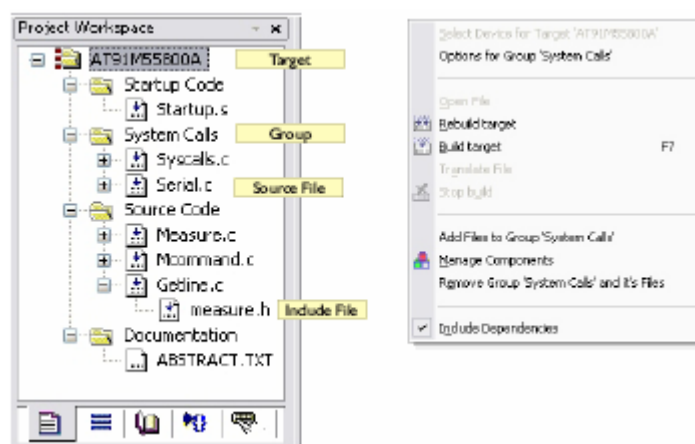


Figura. 6.3. Archivos del Workspace

Como podemos ver en la Figura. 6.3. los *Files Tab* en el espacio de trabajo del proyecto lista los archivos fuentes y cabeceras del programa.

6.1.3 Componentes de un proyecto

Grupos de archivos. Permite al usuario agrupar archivos. Los mismos que pueden ser usados para separar archivos en bloques funcionales o para identificar las partes dentro del programa.

Tarjetas de proyectos. Por medio de estas se pueden crear algunos programas desde un proyecto existente. El usuario puede requerir una tarjeta para realizar pruebas y otra para descargar la versión de la aplicación realizada. Cada tarjeta permite colocara herramientas independientemente para cada archivo de proyecto.

6.1.4 Manuales de usuarios y libros adicionales

La etiqueta de libros en el espacio de trabajo facilita una extensa lista de manuales de usuarios on-line para las herramientas y los microcontroladores seleccionados. El usuario pude adherir un libro adicional que el requiera usando el cuadro de dialogo *Manage Components*. Esto ofrece un rápido acceso al CPU y a la documentación del proyecto.

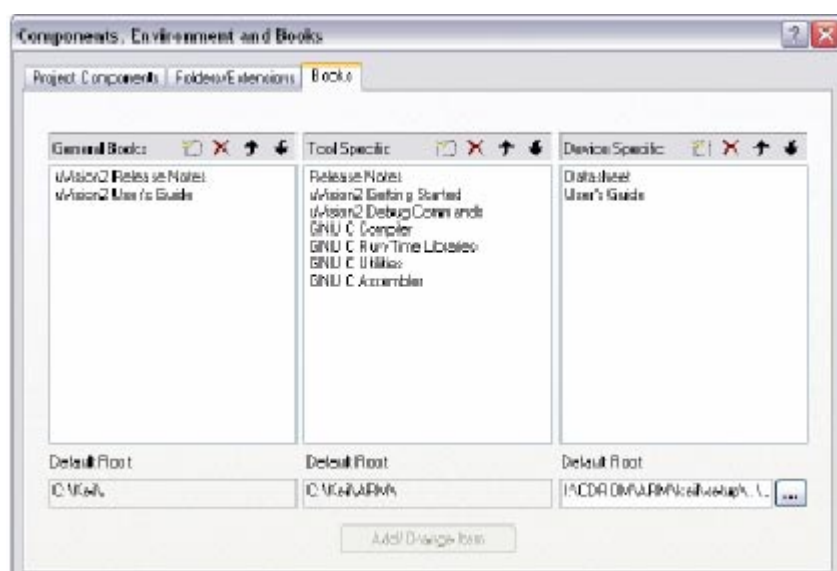


Figura. 6.4. Libros y hojas técnicas

6.1.5 Creación y configuración de herramientas

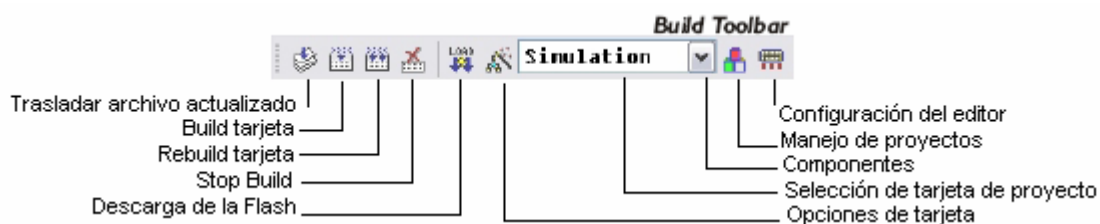


Figura. 6.5. Build Toolbar

6.1.6 Desarrollo de opciones de herramientas

uVision3 permite colocar opciones de herramientas para las tarjetas, grupos y archivos fuentes.

Las opciones de dialogo o cuadros de dialogo son abiertas por medio del contexto del menú del espacio de trabajo del proyecto y ofrece etiquetas para:

- **Dispositivos.** Permite al usuario ver y cambiar la tarjeta del CPU seleccionada.
- **Tarjeta.** Permite especificar el CPU y las opciones de memoria. Las tarjetas son usadas para configurar las herramientas básicas incluyendo las de linker, depuración, y simulación.
- **Output / Listing.** Permite configurar los objetos generados y enumerar los archivos.
- **Compiler / Assembler / Linker.** Permite al usuario ver y cambiar las opciones de específicas herramientas.
- **Debug.** Permite al usuario configurar el simulador y el depurador del uVision3.
- **Utilidades.** Permite al usuario configurar herramientas específicas del proyecto, como la programación de la Flash.

6.1.7 Proceso build

El comando de la tarjeta Build pone en funcionamiento el compilador y el ensamblador del programa. Adicionalmente se puede habilitar el Global Code de optimización el cual es ejecutado para incrementar la recopilación de los módulos que están en C.

6.1.8 Descarga de la flash

Las características de descarga de la Flash son incluidas en drivers de algunas tarjetas (como el emulador y programador EPM900 LPC y el adaptador JTAG de USB Keil). Adicionalmente se puede incluir una memoria flash externa al proyecto de Uvision3.

6.1.9 Fuentes de navegación

La integrada fuente de navegación mantiene una base de datos de los programas realizados por el usuario para tener una rápida navegación a través de los archivos fuentes. El usuario puede usar información detallada para optimizar los tipos de memoria de las variables.

6.1.10 Depurador

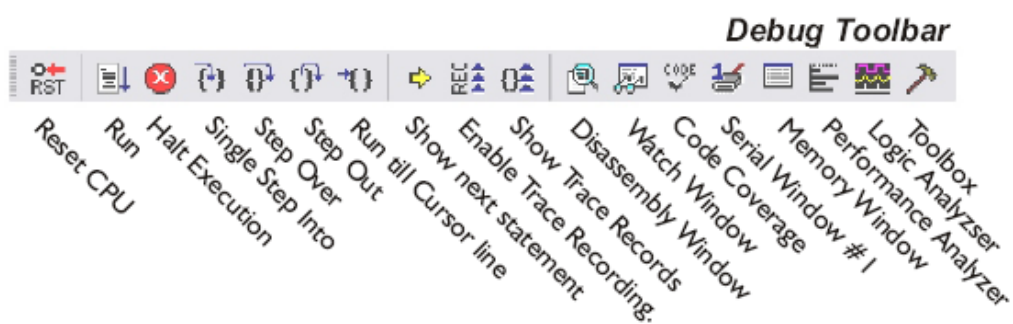


Figura. 6.6. Debug Toolbar

El uVision3 IDE / Simulador / Depurador ofrece un ambiente simple para la edición, simulación y hardware de tarjetas de prueba. La mayoría de las funciones

del editor y depurador pueden ser activadas rápidamente desde la barra de herramientas.

El usuario puede usar el menú de contexto o el editor de la barra de herramientas para insertar breakpoints. Los breakpoints son colocados mientras se realiza la edición cuando el usuario inicia el depuramiento.

El uVision3 marca el estado de cada línea de código en la columna de atributos de la ventana del editor. Esto ofrece una rápida visualización de la actualización de los breakpoints y el estado de la ejecución.

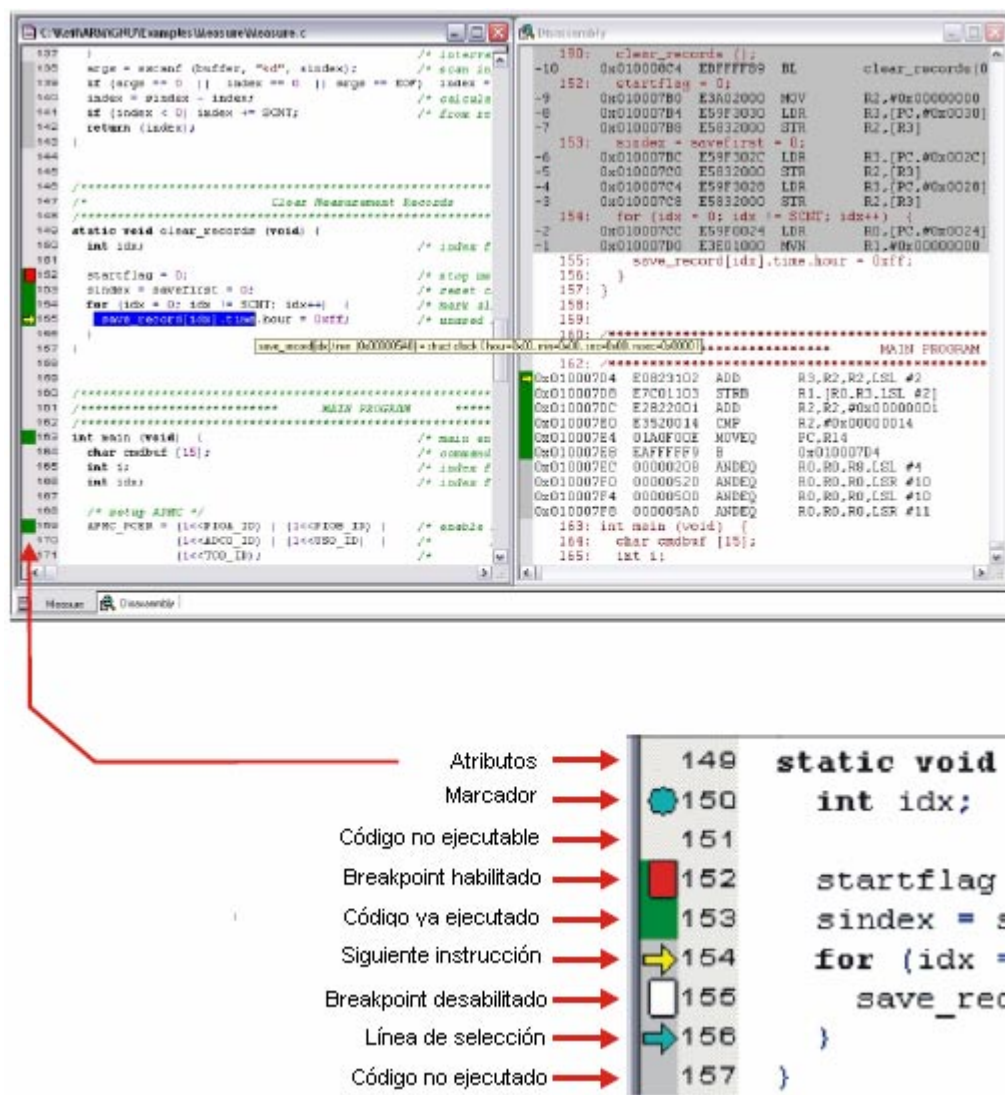


Figura. 6.7. Window disassembly

6.1.11 Ventana de memoria y visualización

La ventana de memoria muestra varias áreas de la memoria. El usuario puede seleccionar el formato de las salidas o cambiar los valores por medio del menú de contexto.

La venta de visualización muestra y permite que el usuario cambie los valores de las variables. Las variables de las funciones actuales son mostradas bajo la etiqueta *Locals*, considerando que el *Call Snack* lista las llamadas al programa.

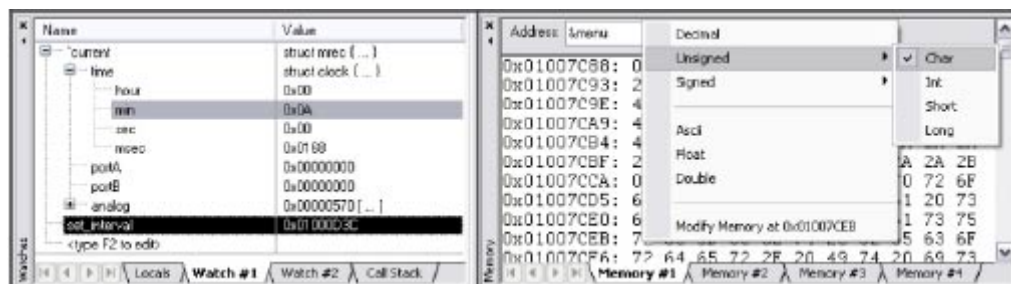


Figura. 6.8. Ventana de memorias

5.6 SIMULADOR

4.3.7 Puertos

Controlador de I/O paralelas

- El **PIOA_SR** (registro de estado de PIO) revisa los cuadros indicando cuales pines son controlados por periféricos asociados y cuales pines son controlados por el PIO.
- El **PIOA_OS** (registro de estado de salida) revisa los cuadros indicando el estado de salida (0 – no revisado para entrada, 1 – revisado para salida) del pin correspondiente.

- El **PIOA_IFSR** (registro de estado del filtro de entradas) revisa los cuadros indicando si habido un fallo técnico en la filtración para el correspondiente pin.
- El **PIOA_ODSR** (registro de datos de datos de salida) revisa los cuadros indicando el valor de salida de cada pin del puerto. Si se cambia las características de los cuadros de revisión escribirá en los registros de periféricos **PIOA_SODR** o **PIOA_CODR**
- **PIOA_MDSR** (registro de estado multifunción) revisa los cuadros indicando si cada pin del puerto esta abierto (open drain) o no.
- **PIOA_PUSR** (registro de estado de pull-up) habilita o deshabilita la colocación de una resistencia pull-up de 100K para cada bit.
- **PIOA_ABSR** (Registro de estado de selección AB) determina si los pines de entrada salida son controlados por el controlador de PIO o los correspondientes periféricos on-chip.
- **PIOA_OWSR** (Registro de estado de escritura de las salidas) muestra los bits de enmascaramiento de los datos de salida para escribirlos en el registro **PIO_OWER** o resetearlo para escribirlos al registro **PIO_OWDR**

Estado de los pines de datos

- **PIO_PDSR** (Registro de estado de los pines de datos) revisa los cuadrados indicando el estado del dato del pin de los pines del simulador MCU

Pines I/O

- Los pines que son revisados en los cuadrados indican el estado de los pines en el simulador MCU. Cuando son usados como salidas, estos tiene los mismos valores que los cuadrados de control **PIOA_ODSR**. Cuando los pines son usados como entradas el usuario puede setear los pines para que tengan un valor alto (1) o bajo (0).

Estado y enmascaramiento de interrupciones

- **PIOA_IMR** (Registro de enmascaramiento de la interrupción) revisa los cuadrados indicando las interrupciones habilitadas para cada pin correspondientemente.
- **PIOA_ISR** (registro de estado de la interrupción) revisa los cuadros de control permitiendo al usuario visualizar el estado de las interrupciones registrados para los pines PIOA.

PIOA: Parallel I/O Controller A

PIO / Output / Input Filter / Output Data / Multi Driver / Pull-up / AB Select / Output Write

PIOA_PSR: 0x00000000

PIOA_OSR: 0x00000000

PIOA_IFSR: 0x00000000

PIOA_ODSR: 0x00000000

PIOA_MDSR: 0x00000000

PIOA_PUSR: 0x00000000

PIOA_ABSR: 0x00000000

PIOA_OWSR: 0x00000000

Pin Data Status

PIOA_PDSR: 0x09000010

I/O Pins

Pins: 0x09000010

Interrupt Mask & Status

PIOA_IMR: 0x00000000

PIOA_ISR: 0x00000000

Figura. 6.9. Controlador de entradas y salidas paralelas

4.3.8 Timers

Para la simulación de este periférico se consta con cuadros de diálogos los cuales muestran y permiten cambiar la configuración de los periféricos.

TC: Timer Counter

| x | CV | RA | RB | RC | CLKSTA | COVFS | LOVRS | CPAS | CPBS | CPCS | LDRAS | LDRBS | ETRGS | MTIOA | MTIOB |
|---|-------|-------|-------|-------|--------|-------|-------|------|------|------|-------|-------|-------|-------|-------|
| 0 | 0000H | 0000H | 0000H | 0000H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0000H | 0000H | 0000H | 0000H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0000H | 0000H | 0000H | 0000H | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Block Control & Mode
☐ SYNC TC0XC0S: **TCLK0**

Selected Channel Control
 TC0_CCR: **0x00000000** ☐ SWTRG ☐ CLKDIS ☐ CLKEN

Selected Channel Mode
 TC0_CMR: **0x00000000** ☐ WAVE ETRGEDG: **None** LDRA: **None**
☐ CPCTRG LDRB: **None**
 TCCLKS: **MCK/2** ☐ CLKI ☐ ABETRG
☐ LDBSTOP
 BURST: **Not gated** ☐ LDBDIS

Selected Channel Counter Value & Compare/Capture Registers
 TC0_CV: **0x00000000** TC0_RA: **0x00000000** TC0_RB: **0x00000000** TC0_RC: **0x00000000**

Selected Channel Interrupt Mask & Status
 TC0_IMR: **0x00000000** MTIOB MTIOA CLKSTA ETRGS LDRBS LDRAS CPCS CPBS CPAS LOVRS COVFS
 TC0_SR: **0x00000000** ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

I/O Pins
☒ TCLK0 ☐ TCLK1 ☐ TCLK2 ☐ TIOA0 ☐ TIOA1 ☐ TIOA2 ☐ TIOB0 ☐ TIOB1 ☒ TIOB2

Figura. 6.10. Timer Counter

El cuadro de dialogo del Timer/Counter muestra y cambia las características asociadas con un bloque del Timer/Counter. Este realiza varios eventos como son el conteo y la medida de tareas.

Bloque y modo de control

- **SYNC** Crea una señal de entrada sincrónica
- **TC0XC0S** Define las entradas de señales de reloj externas para este timer counter

Control del canal de selección

- **TC0_CCR** Muestra el registro de control del canal del timer/counter
- **SWTRG** Ejecuta un trigger de software
- **CLKDIS** Deshabilita el reloj
- **CLKEN** Habilita el reloj

Modo del canal seleccionado

- **TC0CMR** Muestra el registro de modo de captura del timer/counter
- **WAVE** Habilita el modo de forma de onda
- **CPCTRG** Resetea el counter e inicia el reloj
- **TCCLKS** Selecciona la fuente y el tiempo base de la señal de reloj
- **CKLI** (Inversión del reloj) incrementa el ciclo negativo del reloj
- **BURST** Permite seleccionar una señal externa
- **ABETRG** (Trigger externos TIOA/TIOB) Se usa TIOA como un trigger externo cuando es chequeado, y se usa TIOB cuando no es chequeado
- **LDBSTOP** Detiene la cuenta del reloj cuando la carga del RB sucede
- **LDBDIS** Deshabilita cuenta del reloj cuando la carga del RB sucede
- **ETRGDGD** Selecciona el margen de crecimiento, el margen de caída o ambos para un trigger externo
- **LDRA** Selecciona la carga del registro A en margen del crecimiento del TIOA, margen de decrecimiento o ambos.
- **LDRB** Selecciona la carga del registro B en margen del crecimiento del TIOA, margen de decrecimiento o ambos.

Valor de cuenta del canal seleccionado y registros de comparación/captura

- **TC0_CV** Retiene el valor del Timer/Counter
- **TC0_RA** Retiene el valor del registro A del Timer/Counter
- **TC0_RB** Retiene el valor del registro B del Timer/Counter
- **TC0_RC** Retiene el valor del registro C del Timer/Counter

Enmascaramiento del canal de interrupción seleccionado y estado

- **TCO_IMR** Muestra el registro de enmascaramiento de la interrupción del timer/counter
- **TCO_SR** Muestra el registro de estado del timer/counter
- **MTIOB**

- **MTIOA**
- **CLKSTA** (estado del reloj) si es seteado el reloj está habilitado
- **ETRGS** es seteado si un trigger externo ocurre después de leer el registro de estado
- **LDRBS** es seteado si la carga del registro B ocurre después de la lectura del registro de estado
- **LDRAS** es seteado si la carga del registro A ocurre después de la lectura del registro de estado
- **CPCS** es seteado si la comparación del registro C ocurre después de leer el registro de estado
- **LOVRS** es seteado si los registros A o B fueron cargados 2 o más veces pero no leídos después de la lectura del registro de estado
- **COVFS** es seteado si ocurre un desborde en un contador después de la lectura del registro de estado

Pines de I/O

- **TCLK0** es la entrada de la señal de reloj del timer 0
- **TCLK1** es la entrada de la señal de reloj del timer 1
- **TCLK2** es la entrada de la señal de reloj del timer 2
- **TIOA0** es la entrada de la señal A del timer 0
- **TIOA1** es la entrada de la señal A del timer 1
- **TIOA2** es la entrada de la señal A del timer 2
- **TIOB0** es la entrada de la señal B del timer 0
- **TIOB1** es la entrada de la señal B del timer 1
- **TIOB2** es la entrada de la señal B del timer 2

4.3.9 Conversor análogo digital

Para la simulación de este periférico se cuenta con cuadros de diálogos los cuales muestran y permiten cambiar la configuración de los periféricos, a demás consta de un registro de tarjeta virtual (VTREGs) el cual da soporte a las entradas y salidas con el periférico.

ADC: Analog/Digital Converter

Control
 ADC_CR: 0x00000000 ☐ SWRST ☐ START

External Trigger
☐ ADTRG

Mode
 ADC_MR: 0x00000000 ☐ TRGEN ☐ LOWRES ☐ SLEEP
 TRGSEL: TIOA0 STARTUP: 0x00 PRESCAL: 0x00
 SHTIM: 0x00 Rate[Hz]: 2363636

Channel Status
 ADC_CHSR: 0x00000000 CH7 CH6 CH5 CH4 CH3 CH2 CH1 CH0
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Interrupt Mask & Status
 ADC_IMR: 0x00000000 EOC7 EOC6 EOC5 EOC4 EOC3 EOC2 EOC1 EOC0
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐
 ADC_SR: 0x0000C000 ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐
 RXBUFF ENDRX GOVRE DRDY OVRE7 OVRE6 OVRE5 OVRE4 OVRE3 OVRE2 OVRE1 OVRE0
☒ ☒ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Channel Data
 ADC_CDR0: 0x00000000 ADC_CDR4: 0x00000000
 ADC_CDR1: 0x00000000 ADC_CDR5: 0x00000000
 ADC_CDR2: 0x00000000 ADC_CDR6: 0x00000000
 ADC_CDR3: 0x00000000 ADC_CDR7: 0x00000000

Last Converted Data
 ADC_LCDR: 0x00000000
 LDATA: 0x0000

Analog Inputs
 AD0: 0.0000 AD1: 0.0000 AD2: 0.0000 AD3: 0.0000
 AD4: 0.0000 AD5: 0.0000 AD6: 0.0000 AD7: 0.0000

Reference Voltage
 ADVREF: 3.0000

Figura. 6.11. Conversor Análogo Digital

El simulador de conversor análogo digital permite mostrar y configurar este para diferentes propósitos. El control en este cuadro de diálogo está separado en algunos grupos lógicos como:

Control

- **ADC_CR** (Registro de control ADC) muestra y configura los registros de control del A/D
- **SWRST** (Reset de software) botón que resetea el simulador del ADC es un reset de hardware

- **START** (Inicia conversión) botón para comenzar la conversión análogo a digital

Trigger externo

- **ADTRG** (Pin de trigger externo del ADC) muestra el nivel de la entrada trigger externa del ADC

Modo

- **ADC_MR** (Registros de modo del ADC) muestra y configura los modos de operación del conversor análogo digital
- **TRGSEL** (Selección de Trigger) selecciona la fuente de trigger de hardware
- **TRGEN** (Habilitación de Trigger) habilita el trigger de hardware definido por la selección de trigger (TRGSEL), puede estar seteado o reseteado, el ADC solo puede ser triggered por software
- **LOWRES** (Resolución) produce 8 bits de salidas digitales en el registro de conversión si está seteado, y produce 10 bit a sus salida si está reseteado.
- **SLEEP** (Modo Sleep) desactiva el ADC para ahorro de energía cuando este no esta funcionando
- **STARTUP** (Tiempo de Inicio) contiene el tiempo mínimo (en ciclos de reloj) necesario para garantizar el mejor valor de conversión después de que el ADC ha sido iniciado
- **SHTIM** (tiempo de espera y de muestra) contiene el tiempo mínimo (en ciclos de reloj) necesario para garantizar el mejor valor de conversión entre los canales de selección del ADC
- **PRESCAL** (Selección de la tasa de prescaler) define como el reloj maestro es convertido en el reloj del ADC
- **RATE [Hz]** es el valor del reloj de conversión del ADC

Estado del canal

- **ADC_CHSR** (Registro de estado del canal) habilita o deshabilita la conversión de los canales del 0 al 7
- **CH7-CH0** (Estados del canal) es seteado para habilitar un canal o reseteado para deshabilitar un canal

Estado de interrupción

- **ADC_IMR** (Registro de enmascaramiento de interrupción) controla el final de la conversión (EOCx) y las interrupciones excedentes (OVREx)
- **ADC_SR** (Registro de estado) Muestra el fin de la interrupción (EOCX)
- **OVRE7-OVRE0** (Enmascara la interrupción de error de desbordamiento) habilita la interrupción de error por desbordamiento para los canales del 0-7
- **EOC7 – EOC0** (Fin de la interrupción de conversión) habilita el fin de la interrupción de conversión para los canales del 0 al 7
- **RXBUFF** Habilita la interrupción cuando el buffer de recepción esta lleno
- **ENDRX** Deshabilita la interrupción del buffer de recepción
- **GOVRE** Deshabilita interrupciones por desbordamiento generales
- **DRDY** Deshabilita interrupción de dato listo

Entradas análogas

- **AD0 – AD7** (Canales de entradas análogas 0 - 7) muestra los valores de voltajes análogos para cada entrada del los canales

Voltaje de Referencia

- **ADVREF** (Referencia análoga) configura los limites de entradas de voltaje para AD0 – AD7

Último dato de conversión

- **ADC_LCDR** (Registro de último dato convertido) contiene los valores 8 o 10 bits (dependiendo del seteo de LOWRES) LDATA
- **LDATA** (Último dato convertido) es el 8 o 10 bit de valor de salida digital, basado en la resolución del seteo (LOWRES)

Datos convertidos

- **ADC_CDR_ADC_CDR7** (Dato convertidos) contiene 8 o 10 bit de valor de salida digital, basado en la resolución del seteo (LOWRES)

ADVREF VTREG

Es un tipo de dato flotante, setea el voltaje de referencia análogo para simular la conversión análoga digital.

ADX VTREG

Es de tipo flotante setea las entradas de voltaje análogas para simular la conversión análoga digital.

5.7 TUTORIAL BÁSICO PARA DESARROLLAR PROYECTOS; EDICIÓN, COMPILACIÓN, SIMULACIÓN.

6.3.1 Instalación:

Paso 1: Dar un clic en es icono



Paso 2: Dar un clic en *NEXT*



Figura. 6.12. Ventana de Instalación del uVision3

Paso 3: Poner una dirección para que se genera la carpeta del Keil, y dar un clic en *NEXT*

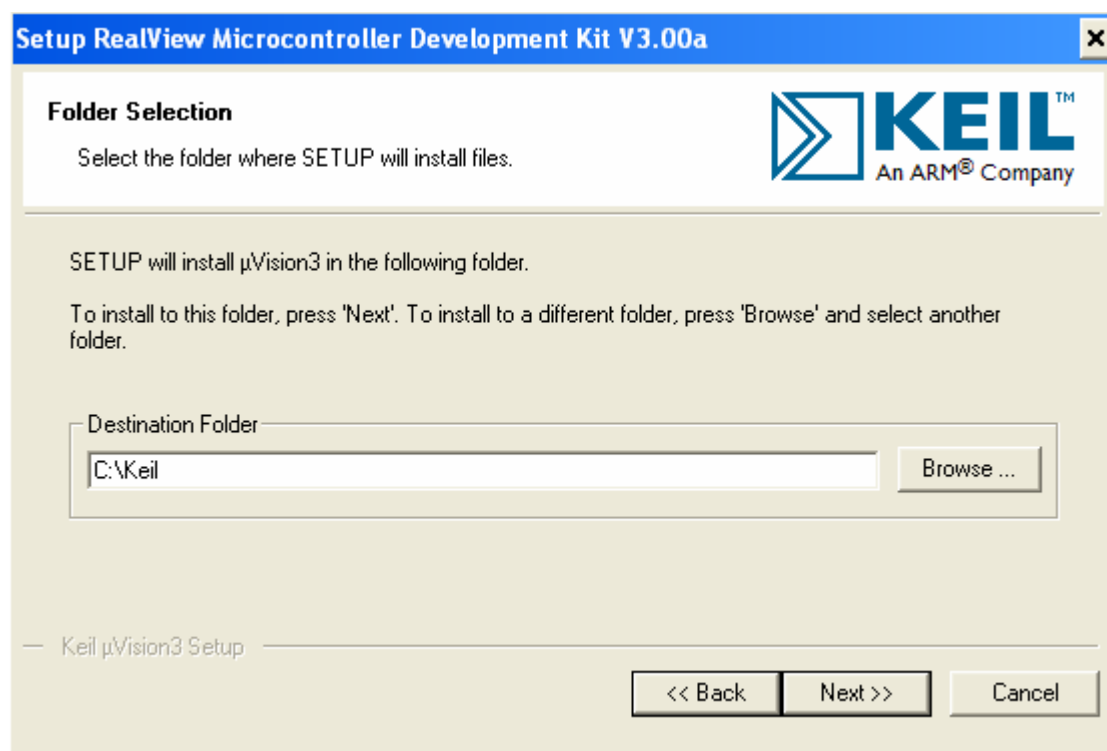
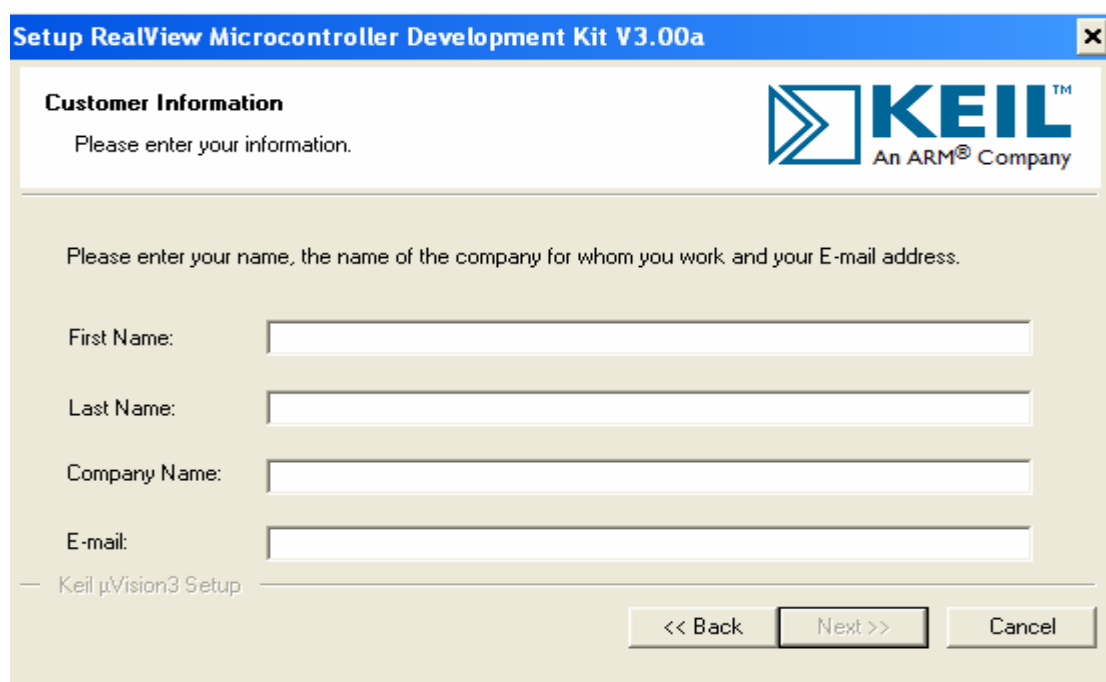


Figura. 6.13. Dirección de la Carpeta del Keil

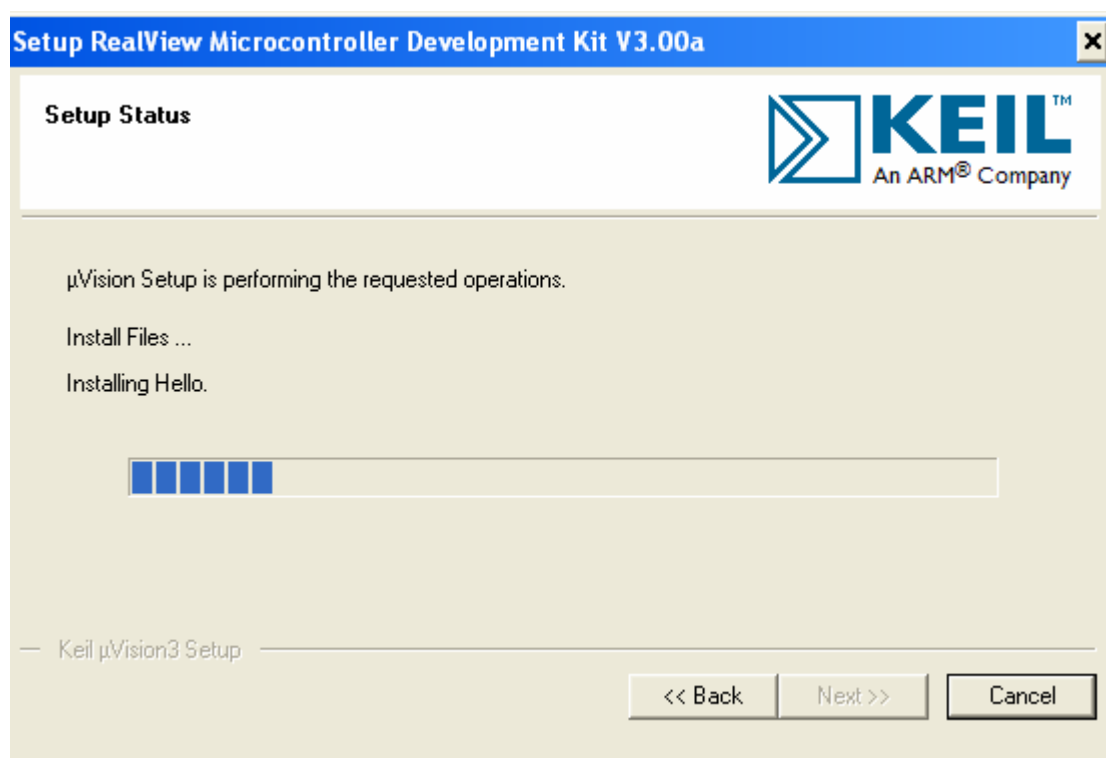
Paso 4: Llenar los datos de instalación pedidos y dar un clic en *NEXT*



The screenshot shows the 'Setup RealView Microcontroller Development Kit V3.00a' window. The title bar is blue with the text 'Setup RealView Microcontroller Development Kit V3.00a' and a close button. The window has a light beige background. At the top left, it says 'Customer Information' in bold. To the right is the KEIL logo, which consists of a blue square with a white 'K' inside, followed by the word 'KEIL' in bold blue letters and 'An ARM® Company' in smaller blue letters below it. Below the title, it says 'Please enter your information.' in a small font. Further down, it says 'Please enter your name, the name of the company for whom you work and your E-mail address.' in a small font. There are four text input fields: 'First Name:', 'Last Name:', 'Company Name:', and 'E-mail:'. Each field has a corresponding text box to its right. At the bottom left, there is a progress bar with the text 'Keil µVision3 Setup' below it. At the bottom right, there are three buttons: '<< Back', 'Next >>', and 'Cancel'. The 'Next >>' button is highlighted with a black border.

Figura. 6.14. Datos del Propietario

Paso 5: Esperar mientras se realiza la instalación del software.



The screenshot shows the 'Setup RealView Microcontroller Development Kit V3.00a' window. The title bar is blue with the text 'Setup RealView Microcontroller Development Kit V3.00a' and a close button. The window has a light beige background. At the top left, it says 'Setup Status' in bold. To the right is the KEIL logo, which consists of a blue square with a white 'K' inside, followed by the word 'KEIL' in bold blue letters and 'An ARM® Company' in smaller blue letters below it. Below the title, it says 'µVision Setup is performing the requested operations.' in a small font. Further down, it says 'Install Files ...' and 'Installing Hello.' in a small font. Below these, there is a progress bar with five blue squares. At the bottom left, there is a progress bar with the text 'Keil µVision3 Setup' below it. At the bottom right, there are three buttons: '<< Back', 'Next >>', and 'Cancel'. The 'Next >>' button is highlighted with a black border.

Figura. 6.15. Instalación del uVision3

Paso 6: Damos un clic en *Finís*, para terminar la instalación

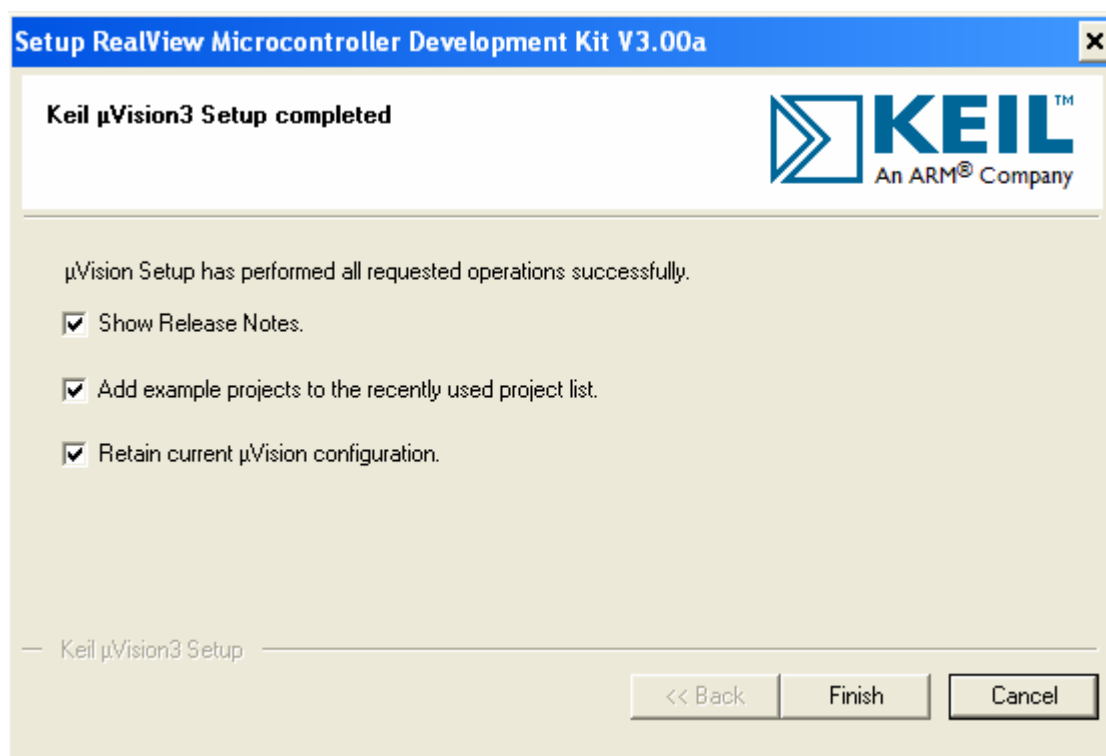


Figura. 6.16. Instalación Terminada

6.3.1 Edición:

Paso 1: Para abrir un proyecto se debe seleccionar la opción *Project*, y se da un clic en *Open Project*

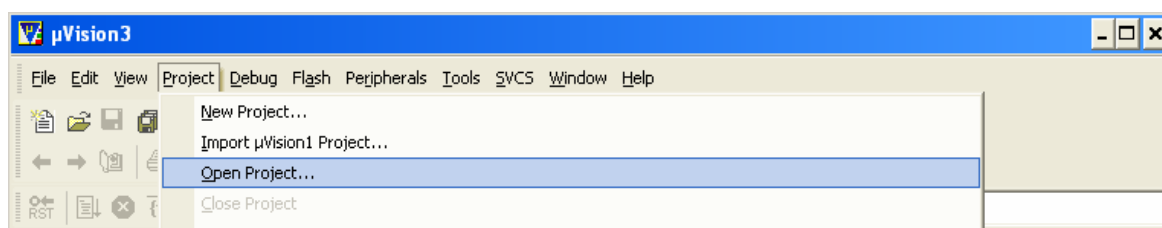


Figura. 6.17. Abrir proyecto

Paso 2: Seleccionar el archivo a ser ejecutado, y dar un clic en *Abrir*

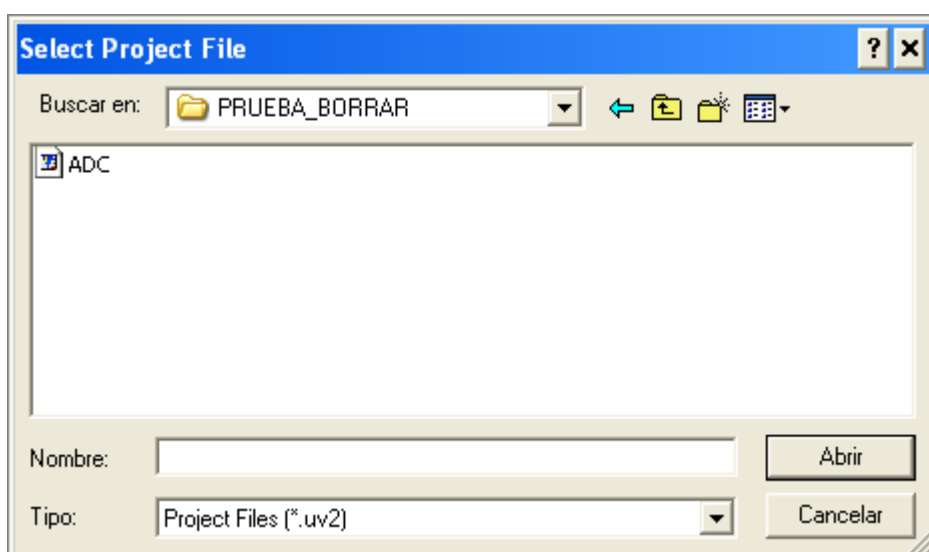


Figura. 6.18. Archivo que se va abrir

Edición de código fuente

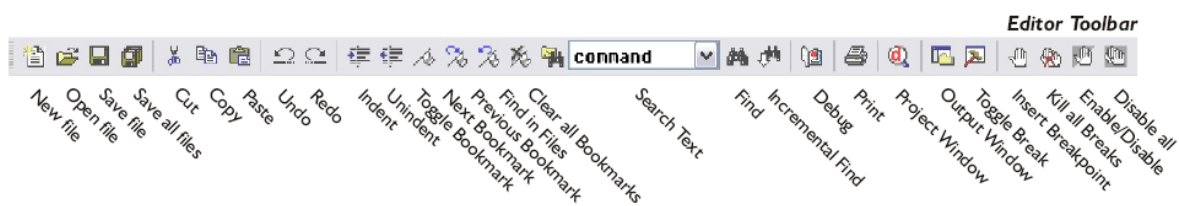


Figura. 6.19. Editor Toolbar

El editor del uVison3 ofrece algunos estándares y software modernos para la edición de programas. Sintaxis en colores vistosos y texto con sangría son optimizadas para la edición de código fuente en C.

La mayoría de funciones de edición pueden ser de acceso rápido desde la barra de herramientas o desde el menú de contexto. La depuración del programa es controlada directamente en el editor. Este ambiente de desarrollo permite al usuario rápidamente integrar y probar cambios en el código fuente.

```

01
02 #include <AT91SAM7S64.H>           /* AT91SAM7S definitions */
03 #include <stdio.h>                 /* standard I/O .h-file */
04
05 #define VREF 3                     /* number of records */
06
07
08 AT91S_ADC * pADC = AT91C_BASE_ADC; /* Global Pointer to ADC */
09
10
11 extern void init_serial (void);    /* Initialize Serial Interface */
12
13 /* Main Program */
14 void main (void) {                 /* main entry for program */
15
16     unsigned int adval;
17
18     /* Enable peripheral clocks */
19     *AT91C_PMC_PCER = (1<<AT91C_ID_ADC) | /* enable ADC, */
20                      (1<<AT91C_ID_US0); /* enable USART0 */
21
22     init_serial();                 /* Initialize Serial Interface */
23
24     /* Setup the A/D converter */
25     pADC->ADC_CR = AT91C_ADC_SWRST; /* reset ADC */
26     pADC->ADC_CHER = AT91C_ADC_CH0 |
27                    AT91C_ADC_CH1; /* enable ADC channels 0..3 */
28     pADC->ADC_MR = AT91C_ADC_LOWRES_10_BIT |
29                  (9<<AT91C_ADC_PRESCAL); /* set 10-bit resolution, prescale = 9 */
30

```

Figura. 6.20. Edición de código fuente

Código fuente del conversor análogo digital:

Conversión de una señal análoga de 2.99 V a digital

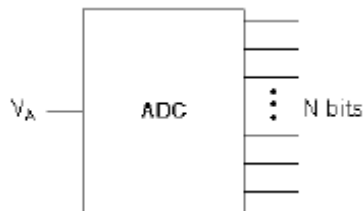


Figura. 6.21. Conversión Analógica - Digital

Definimos el archive de cabecera del chip con el que se va a trabajar

#include <AT91SAM7S64.H>

Ponemos el archivo estándar para que me active las entradas y salidas del chip

#include <stdio.h>

Definimos el valor de referencia igual a 3

#define VREF 3

Habilito un puntero global para trabajar igualando a la dirección base

AT91S_ADC * pADC = AT91C_BASE_ADC;

Llamo a una función externa para habilitar la comunicación serial

extern void init_serial (void);

Programa Principal

void main (void)

{

Creo la variable adval

unsigned int adval;

Habilito el conversor análogo digital por medio y la comunicación USART

***AT91C_PMC_PCER=(1<<AT91C_ID_ADC)|(1<<AT91C_ID_US0);**

Inicializo comunicación serial

init_serial();

Se produce un reset de software antes de la conversión análogo digital

pADC->ADC_CR = AT91C_ADC_SWRST;

Habilito el canal 0 y el canal 1 del conversor análogo digital

pADC->ADC_CHER = AT91C_ADC_CH0 | AT91C_ADC_CH1;

Ponemos el modo en el que trabajara el ADC, con una resolución de 10 bit y la selección de frecuencia del prescaler donde el prescaler va a tener un valor de 9

pADC->ADC_MR = AT91C_ADC_LOWRES_10_BIT |

(9<AT91C_ADC_PRESCAL);

while (1)

{

Se inicia la conversión análoga digital

pADC->ADC_CR = AT91C_ADC_START;

Esperamos a que la bandera del registro de estado del ADC (ADC_SR) cambie y se iguale con la de final de conversión (AT91C_ADC_EOC0) para dar por terminada la conversión.

while ((pADC->ADC_SR & AT91C_ADC_EOC0) == 0);

El dato convertido se lo almacena dentro de la variable ADVAL

adval = (unsigned int) pADC->ADC_CDR0 & AT91C_ADC_DATA;

}

}

Características Avanzadas

uVison3 ofrece algunas nuevas características al editor como son:

- **Buscador incremental** posiciona el curso mientras se continúa digitando el párrafo a buscarse.
- **Active brace checking** muestra el código anidado y resalta desigualdades las cuales se encuentran introducidas en paréntesis, corchetes o llaves.
- **Configuración del documento seleccionado** en la tabulación de espacios y el color de sintaxis tanto para Assembler, C, y otros.
- **Bosquejo** del documento para proporcionar una rápida visualización de los archivos fuentes completos.

Planillas y funciones

La etiqueta de funciones en el espacio de trabajo del proyecto muestra todas las funciones en el proyecto o abre los archivos del editor. El usuario puede dar clic en una función para saltar a su definición.


Las etiquetas de planillas muestran los bloques de texto definidos por el usuario (planillas).

Utilidades del editor

uVison3 contiene poderosas características e interfaces para trabajar con herramientas de desarrollo externas las cuales ayudan al usuario completamente en sus proyectos.

6.3.2 Compilación:

Una vez editado el programa se procede a la compilación del mismo.

Paso 1: Se procede a dar un clic sobre el icono **Translate current file** .

Si no existen errores en la *output window* nos mostrará que el programa esta correctamente editado, con 0 errores y 0 advertencias.

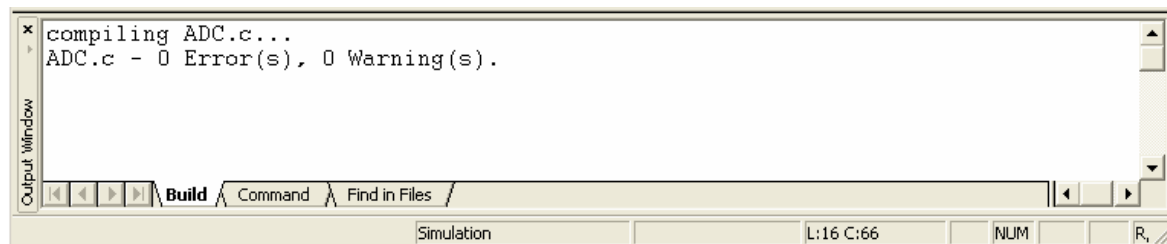


Figura. 6.22. Compilación ADC

Caso contrario si hubiera algún error o una advertencia en la *output window* nos indicará el número de errores y advertencias existentes y en que línea se encuentran.

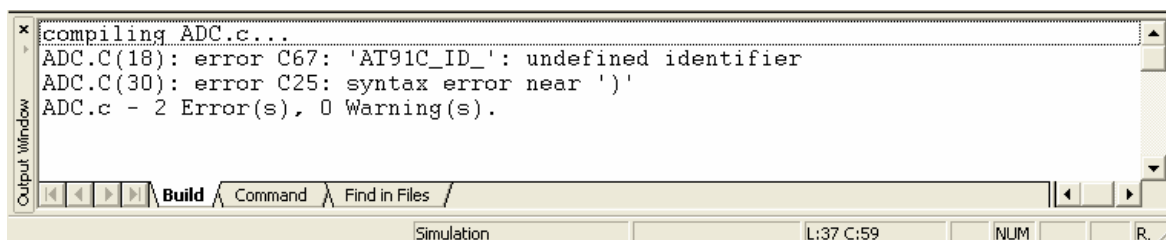


Figura. 6.23. Compilación ADC con error

Paso 2: Dar un clic sobre el icono *Build target* 

Paso 3: Dar un clic sobre el icono *Rebuild all target files*. 

Permite compilar otros códigos fuentes que tenga de apoyo el programa para ponerlo en funcionamiento con el ensamblador del programa.

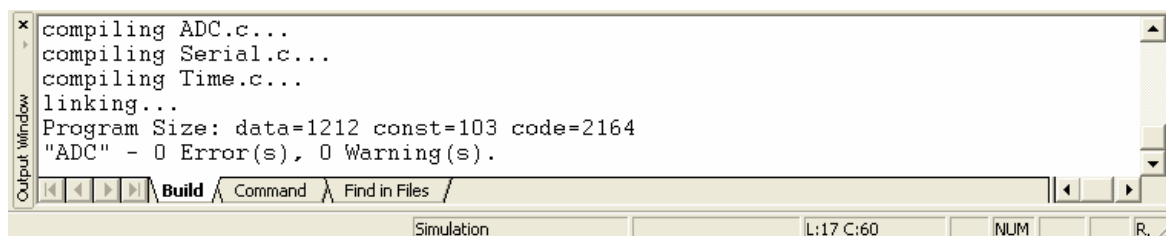


Figura 6.24. Build target

Como podemos ver en el ejemplo son también compilados los archivos Serial y Time aparte del archivo ADC.

6.3.3 Simulación:

Paso 1: Dar un clic en Debug, posteriormente dar un clic en Star/Debug Session, o también presionando CTRL+F5. Esto nos permite iniciar o finalizar la simulación.

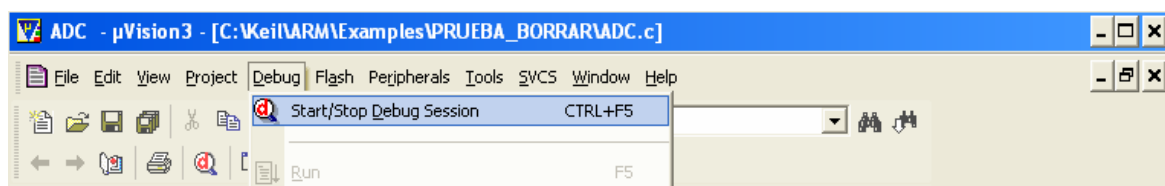


Figura. 6.25. Iniciar Simulación

Paso 2: Dar un clic sobre Run o caso contrario presionar F5.

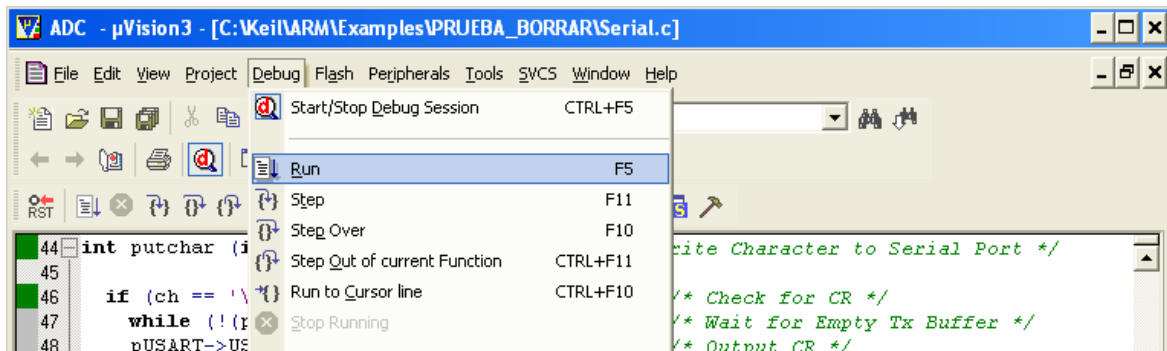


Figura. 6.26. Run programa

Paso 3: Escoger los periféricos que se quiere simular dando un clic en Peripherals.

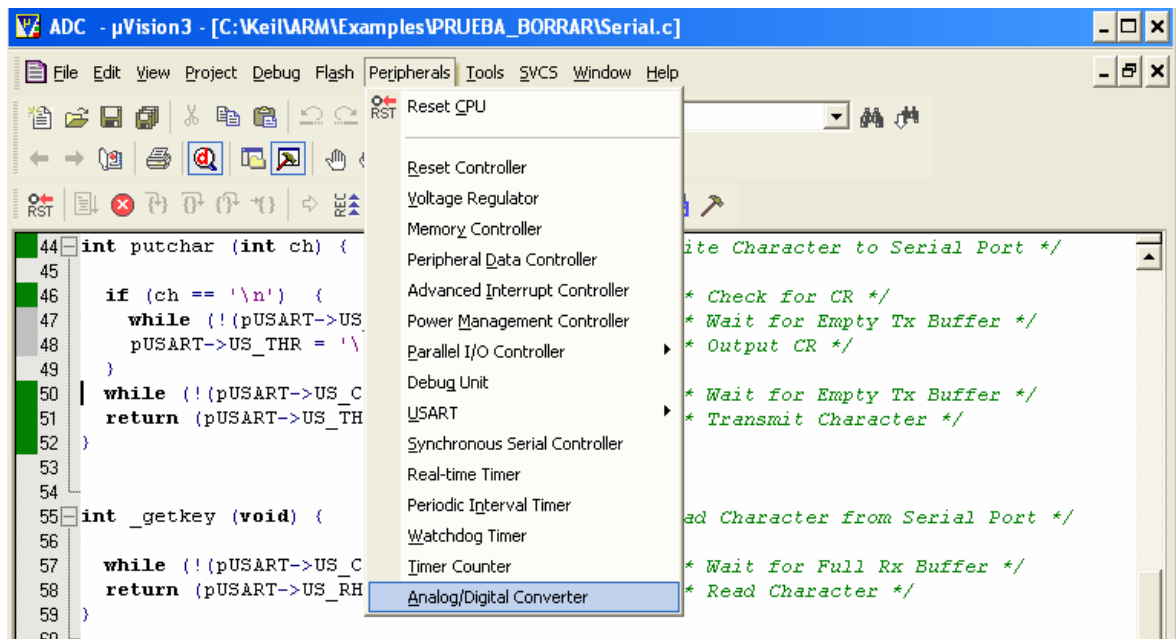


Figura. 6.27. Escoger simulador

Para la simulación se tomará como ejemplo el conversor análogo digital, al cual se lo selecciona dando un clic.

Para realizar la simulación del conversor análogo digital se debe tomar en cuenta el Toolbox para poder generar una señal.

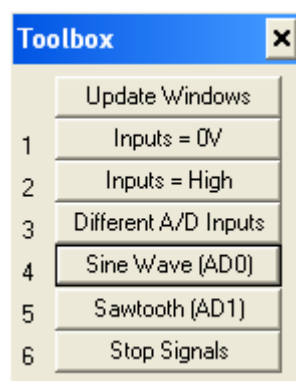


Figura. 6.28. Toolbox

Como podemos ver esta herramienta nos permite generar diferentes tipos de señales en diferentes puertos.

Paso5: Generar una señal sinusoidal por medio del Toolbox en la entrada AD0 del conversor análogo digital. Esta señal tiene una amplitud de 2.99 V, el cual es un valor propio de este generador de señales.

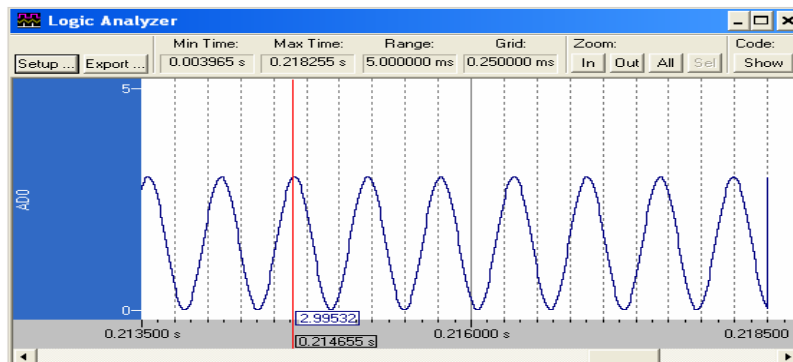


Figura. 6.29. Señal Sinusoidal Generada

Paso 6: Colocamos los principales parámetros que son necesarios para realizar la simulación como son: el prescales, colocamos el lowres si queremos una resolución con 8 bits o si no es puesto se tendrá una resolución de 10 bits, se señala también el canal por el que va a entrar la señal en este caso CH0 y se coloca el voltaje de referencia al cual va a trabajar.

ADC: Analog/Digital Converter

Control
 ADC_CR: 0x00000000 ☐ SWRST ☐ START External Trigger ☐ ADTRG

Mode
 ADC_MR: 0x00000010 ☐ TRGEN ☒ LOWRES STARTUP: 0x00 PRESCAL: 0x00
 TRGSEL: TIOA0 ☐ SLEEP SHTIM: 0x00 Rate[Hz]: 2363636

Channel Status
 ADC_CHSR: 0x00000001 CH7 CH6 CH5 CH4 CH3 CH2 CH1 CH0 ☒

Interrupt Mask & Status
 ADC_IMR: 0x00000000 EOC7 EOC6 EOC5 EOC4 EOC3 EOC2 EOC1 EOC0
 ADC_SR: 0x000C0000 ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐
 RXBUFF ENDRX GOVRE DRDY OVRE7 OVRE6 OVRE5 OVRE4 OVRE3 OVRE2 OVRE1 OVRE0
☒ ☒ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Channel Data
 ADC_CDR0: 0x00000000 ADC_CDR4: 0x00000000
 ADC_CDR1: 0x00000000 ADC_CDR5: 0x00000000
 ADC_CDR2: 0x00000000 ADC_CDR6: 0x00000000
 ADC_CDR3: 0x00000000 ADC_CDR7: 0x00000000

Last Converted Data
 ADC_LCDR: 0x00000000
 LDAT: 0x0000

Analog Inputs
 AD0: 0.0000 AD1: 0.0000 AD2: 0.0000 AD3: 0.0000
 AD4: 0.0000 AD5: 0.0000 AD6: 0.0000 AD7: 0.0000

Reference Voltage
 ADVREF: 3.0000

Figura. 6.30. Parámetros del Conversor Análogo Digital

El prescaler es de mucha importancia debido a que es un predivisor de frecuencia de la señal de entrada utilizado comúnmente para programar tiempos largos.

Por ejemplo para un prescaler 0x00 con una voltaje de referencia de 3, tenemos la siguiente señal de salida

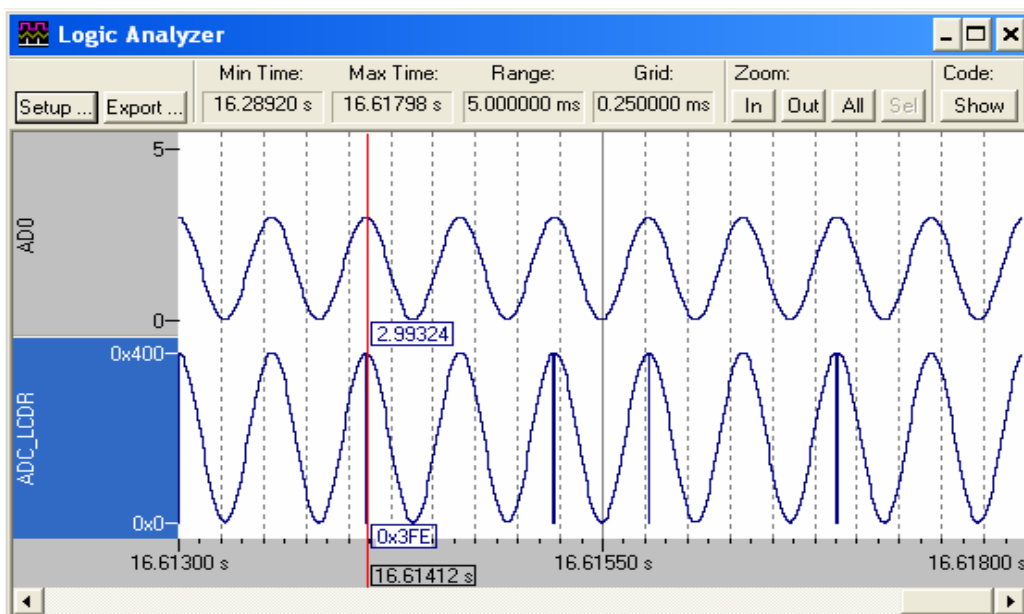


Figura. 6.31. Señal con Prescaler 0x00

Los valores digitales que va a tomar la señal para los diferentes valores análogos de entrada con un valor de referencia de 3 son:

| Valor de Referencia 3 o Mayor | |
|-------------------------------|---------------|
| Valor Análogo | Valor Digital |
| 0 | 0x1 |
| 0,1 | 0x20 |
| 0,37 | 0x6D |
| 0,49 | 90 |
| 0,62 | 0xB7 |
| 1,07 | 0x13B |
| 1,41 | 0x19D |
| 1,58 | 0x1CF |
| 1,75 | 0x200 |
| 2,22 | 0x25F |
| 2,37 | 0x2B6 |
| 2,72 | 0x31E |
| 2,99 | 0x3FF |

Tabla. 6.1. Valores Convertidos a Digitales

Como podemos ver la señal se digitaliza para todos los valores análogos de entrada.

Para un prescaler 0x1F podemos ver que se dan pasos o saltos entre frecuencias sucesivas, que quedan como el producto de la frecuencia de referencia por el factor de división del prescaler.

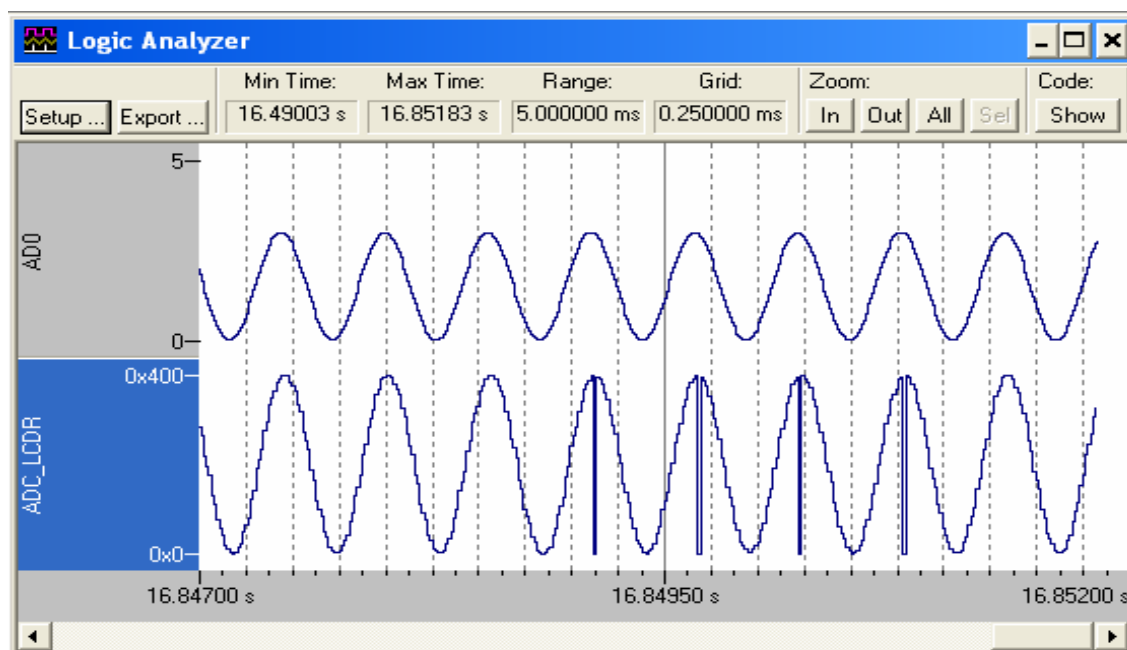


Figura. 6.32. Señal con Prescaler 0x1F

Para el valor máximo de prescaler 3F los pasos saltos de las frecuencias son mayores.

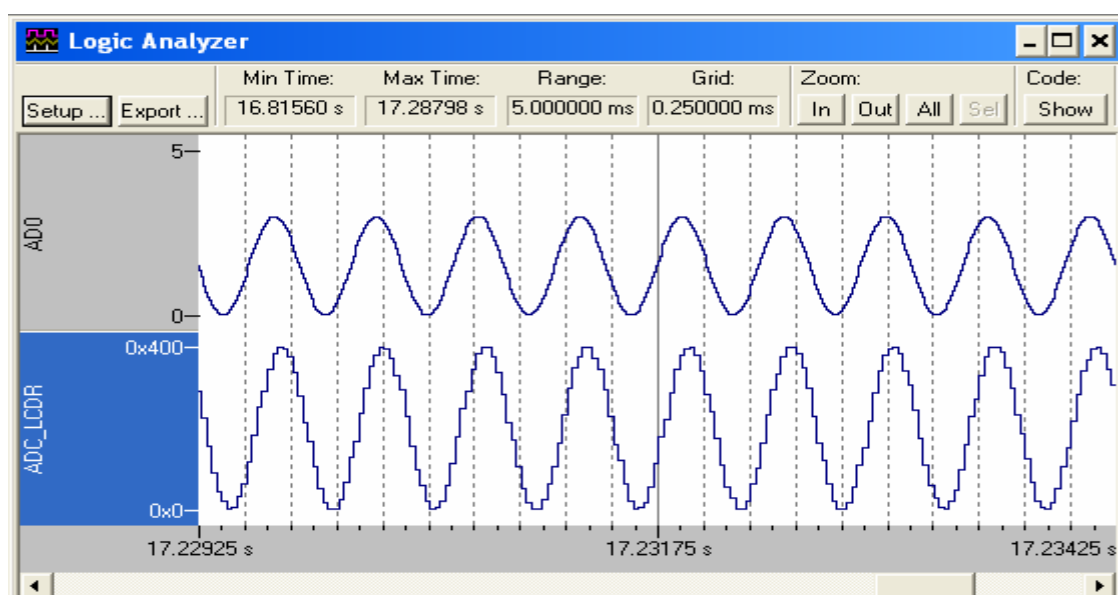


Figura. 6.33. Señal con Prescaler 0x3F

Paso 7: Ingresar voltaje de referencia.

Este valor determina el rango máximo en el cual se convertirá una señal de entrada. El dispositivo establece una relación entre su entrada (señal analógica) y su salida (digital) dependiendo de su resolución. Esta resolución se puede saber siempre y cuando conozcamos el valor máximo que la entrada de información utiliza y la cantidad máxima de la salida en dígitos binarios.

Por ejemplo si ingresamos un voltaje de referencia de 1 y con un prescaler de 0x00:

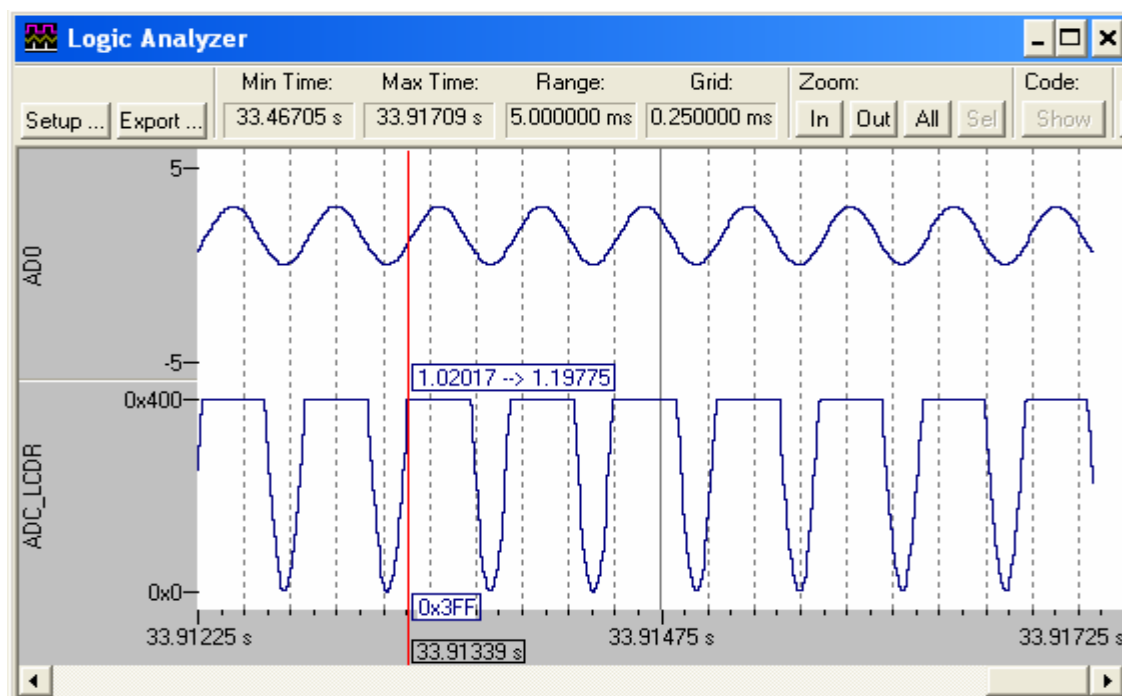


Figura. 6.34. Conversión ADC con Voltaje de Referencia 1V

Podemos ver en el gráfico que cuando tiene el valor del voltaje igual a 1 automáticamente la señal del registro ADC_LCDR toma el máximo valor del registro que es 0x400 y se mantiene con este valor durante todos los valores que sean superiores a 1.

A continuación se muestra una tabla de los valores digitales que están a la salida con los diferentes valores analógicos de entrada.

| Valor de Referencia 1 o Mayor | |
|-------------------------------|---------------|
| Valor Análogo | Valor Digital |
| 0,0063 | 0x6 |
| 0,21 | 0xDE |
| 0,23 | 0x54 |
| 0,54 | 0x231 |
| 0,833 | 0x355 |
| 0,989 | 0x3F6 |
| 1 | 0x3FF |
| 1,21 | 0x3FF |
| 1,65 | 0x3FF |
| 1,93 | 0x3FF |
| 2,21 | 0x3FF |
| 2,63 | 0x3FF |
| 2,99 | 0x3FF |

Tabla. 6.2. Valores Convertidos a Digitales con Vref 1

Para un valor de referencia igual a 2 y con un prescaler 0x00 tenemos:

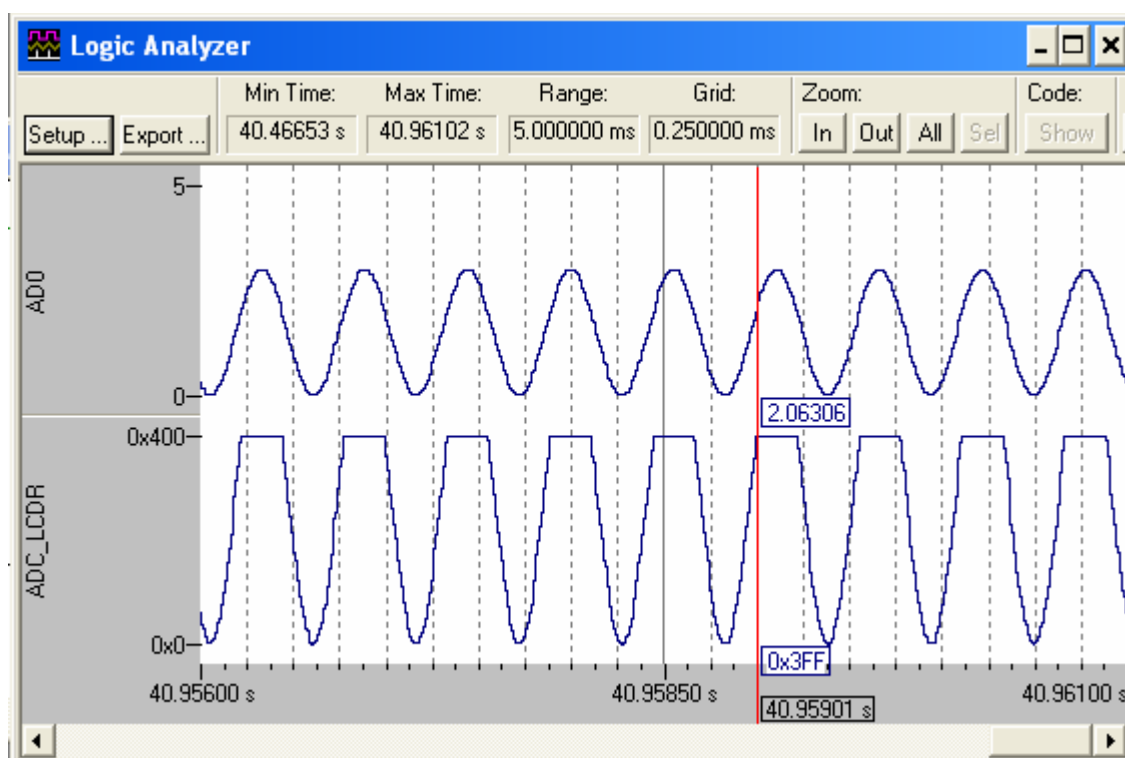


Figura. 6.35. Conversión ADC con Voltaje de Referencia 2V

La tabla de valores digitales quedaría:

| Valor de Referencia 2 o Mayor | |
|-------------------------------|---------------|
| Valor Analógico | Valor Digital |
| 0,0089 | 0x5 |
| 0,087 | 02D |
| 0,23 | 0x7A |
| 0,45 | 0xE7 |
| 1,02 | 0x20D |
| 1,07 | 0x13B |
| 1,18 | 0x261 |
| 1,69 | 0x364 |
| 2,02 | 0x3FF |
| 2,18 | 0x3FF |
| 2,21 | 0x3FF |
| 2,59 | 0x3FF |
| 2,99 | 0x3FF |

Tabla. 6.3. Valores Convertidos a Digitales con Vref 2

Como se puede ver en el grafico cuando la señal análoga tiene el valor de 2 toma el máximo valor del registro que es 0x400 y de la misma forma para todos los valores superiores a dos.

Para un voltaje de referencia igual a 0 se tendrá un valor digital constante cuyo valor será 0x400.

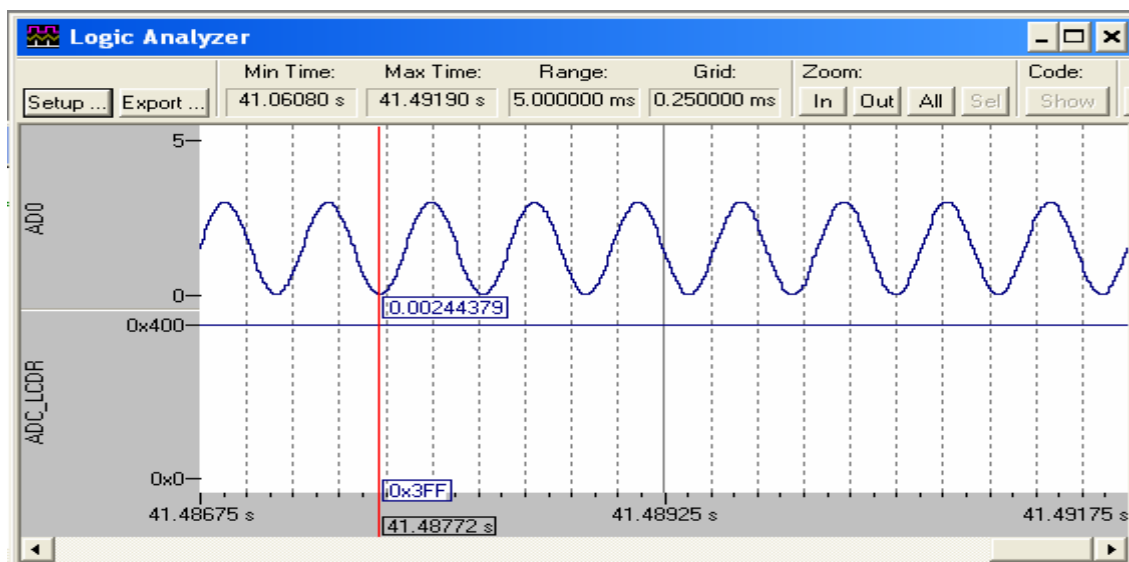


Figura. 6.36. Conversión ADC con Voltaje de Referencia 0V

Paso 8: Para detener la simulación se da un clic sobre *Halt*



CAPÍTULO VII

CONCLUSIONES Y RECOMENDACIONES

5.8 CONCLUSIONES

- Se estudió las familias por las que está conformada la arquitectura ARM, sus principales características y en las aplicaciones que se las utiliza.
- Se realizó un análisis comparativo entre sus características como son: espacio de memoria, número de periféricos, formas de comunicación entre otras que tienen los chips que están constituidos por la arquitectura ARM
- ARM es la principal industria en la producción de sistemas integrados RISC de 16/32 bits. La compañía licencia su alto rendimiento, costo económico, procesadores RISC eficientes, con diferentes periféricos y diseño de circuitos integrados para que compañías electrónicas internacionales puedan trabajar con ellos. ARM también ofrece un amplio soporte requerido para desarrollar sistemas completos.
- Los núcleos de los microprocesadores ARM están volviéndose rápidamente un estándar en los principales mercados como en el de las comunicaciones móviles, computación de manos libres, de consumo digital y multimedia y soluciones integradas.

- En la industria el núcleo ARM7TDMI es el más usado en microprocesadores RISC integrados de 32 bits. Estos núcleos ofrecen un bajo consumo de potencia, un tamaño pequeño y un alto rendimiento necesitados en aplicaciones móviles e integradas.
- La familia ARM9 probó reducción en el costo del sistema por medio de la habilitación de algunas funciones en un simple chip y reduciendo los requerimientos de memoria del sistema gracias a la alta densidad de código del set de instrucciones Thumb. La arquitectura ARM RISC de 16/32 bits ofrece bajo consumo de potencia esencial para aplicaciones que utilicen batería y aplicaciones SoC donde el alto consumo de potencia puede limitar la integración on-chip.
- Estos núcleos ARM son también ideales en sistemas donde el procesador debe compartir el ancho de banda del bus con otros dispositivos requiriendo un alto rendimiento de los datos. El procesador opera a una alta velocidad desde la cache dejando de esta manera bus del sistema libre para que sea usado por otros dispositivos.
- La convergencia de productos computarizados y tecnologías de comunicación están manejando una mezcla de código en tiempo real y con aplicaciones para el usuario dentro de un CPU simple y de alto rendimiento.
- La herramienta de desarrollo Keil para dispositivos ARM7, ARM9 y Cortex-M3 soportan todos los niveles de desarrollo desde aplicaciones profesionales de ingeniería para estudiantes que estén aprendiendo acerca de desarrollo de softwares de integración.
- Las herramientas de desarrollo estándares para la industria de ARM (herramientas de desarrollo de RealView incluyendo compiladores de C/C++, Assembler y Linker) y Keil (debuggers, aplicaciones en tiempo real

y emuladores) soporta los más populares microcontroladores de ARM y ayuda a terminar los proyectos en forma rápida y fácil.

- Los dispositivos ARM pueden ser complejos para instalar y configurar, por lo cual siempre se incluyen numerosos ejemplos de programas para facilitar el trabajo de estos.
- Con la simulación se puede escribir y probar programas antes de que sean cargados para ser utilizados en la tarjeta de hardware. El depurador del uVision es capaz de simular periféricos on-chip (I²C, CAN, UART, SPI, interrupciones, puertos de I/O, conversos A/D, conversor D/A y módulos PWM) del dispositivo ARM que se vaya a utilizar. La simulación ayuda a entender la configuración del hardware y a evitar problemas de instalación.
- ARM provee su propiedad intelectual (IP) en sus desarrollos en la forma de los diseños de los núcleos de sus procesadores, cache y diseños SoC, en productos con estándares de aplicaciones específicas (ASSPs), en softwares relacionados y herramientas de desarrollo, cualquier producto que se necesite para crear un diseño innovador basado en componentes estándares de la industria los cuales son compatibles con las siguientes generación.
- El uVision esta compuestos por la herramienta de desarrollo integrada Keil y el ambiente de depuración que ayudan a crear rápidamente y a probar aplicaciones integradas para ARM7, ARM9, Cortex-M3, C16x, ST10, XC16x, C251 y aplicaciones integradas con el C51. El uVision combina todos los aspectos de desarrollo de proyectos integrados incluyendo edición de código fuente, organización y manejo de proyectos, control de errores, depuración de la tarjeta, simulación y programación de la Flash.

5.9 RECOMENDACIONES

- Es recomendable trabajar con circuitos integrados que pertenezcan a las familias ARM7 y ARM9 debido a que son chips que pueden cumplir cualquier aplicación y sus herramientas para desarrollo de proyectos se las puede encontrar fácilmente.
- Es conveniente el momento de la programación del chip trabajar con el set de instrucciones Thumb por ser un lenguaje de alto nivel y ofrece una mejor densidad de código el momento de ejecutar el programa.
- Por mayor facilidad para encontrar en el mercado y la variedad de chips que existen cada uno con diferentes herramientas como son puertos, tamaño de memoria, herramientas de desarrollo con las que trabajan entre otras es preferible utilizar circuitos integrados marca ATMEL.
- El momento que se va escoger un circuito integrado para trabajar se debe tomar en cuenta el número de periféricos con el que se va a trabajar y si el espacio de memoria que posee el chip es el necesario.
- Para la edición del programa se debe trabajar con los correspondientes archivos de cabecera de cada circuito integrado debido a que son distintos para cada uno y de estos dependen que esté bien editado el programa.
- El momento de la simulación es recomendable tratar de visualizar mayor cantidad de simuladores debido a que son una herramienta muy útil para ver que se cumpla lo que se ha editado en el programa.
- Se recomienda que para un futuro el departamento de eléctrica y electrónica incursione con sus docentes y alumnos en el uso de esta tecnología, realizando prácticas para sustentar el estudio de esta tesis.

REFERENCIAS BIBLIOGRAFICAS

- Desarrollos tecnológicos ARM,
<http://www.cl.cam.ac.uk/~fms27/teaching/2001-02/arm-project/index.html>.
- ARM, Pagina Web de la compañía ARM, <http://arm.com>.
- WIKIPEDIA, Estructura general de la arquitectura ARM
http://en.wikipedia.org/wiki/ARM_architecture.
- Circuitos integrados y herramientas de desarrollo ARM,
<http://www.htsoft.com/products/armccompiler.php>.
- KEIL, Herramienta de desarrollo Keil Uvision,
<http://www.keil.com/arm/chips.asp>,
- ATMEL, Circuito integrado AT91SAM7S64,
<http://www.atmel.com/products/AT91/>,

ANEXO 1

Tabla Comparativa de Circuitos Integrados

Tabla 2. Tabla Comparativa Microcontroladores Philips

| | LPC2101 / 2102 / 2103 | LPC2104 / 2105 / 2106 | LPC2114 / 2124 | LPC2119 / 2129 | LPC2131 / 2132 / 2134 / 2136 / 2138 | LPC2141/2142 /2144/2146/2148 |
|--------------------------------|------------------------------|------------------------------|-----------------------|-----------------------|--|-------------------------------------|
| RAM (bytes) | 8192 / 16384 / 32768 | 16384 / 32768 / 65536 | 16384 | 16384 | 8K / 16K / 16K / 32K / 32K | 8K a 40K |
| Frecuencia (MHz) | 70 | 60 | 60 | 60 | 60 | 60 |
| ADCs | 1 de 10bits | - | 4 de 10bits | 4 de 10bits | 2 de 10bits | 2 de 10bits |
| Watchdog timers | - | x | x | x | x | x |
| Número de pines I / O | 32 | 32 | 46 | 46 | 47 | 45 |
| Tem operación (C°) | 0 - 70 | 0 a 70 | (-40) a 85 | (-40) a 85 | (-40) a 85 | (-40) a 85 |
| Timers | 2 de 16bits | 2 de 16bits | 2 de 16bits | 2 de 16bits | 2 de 16bits | 2 de 32bits |
| PWM | 1 de 6 canales | 1 de 6 canales | 1 de 6 canales | 1 de 6 canales | 1 de 6 canales | 1 de 6 canales |
| Interfaces seriales | UART, I2C, SPI, SSP. | 2UARTs, I2C,SPI | 2UARTs, I2C,SPI | 2UARTs, I2C,SPI,CAN | 2UARTs, I2C,SPI | USB 2.0, 2 UART, 2I2C, 2SPI |
| Interrupciones externas | 3 | 3 | 3 | 3 | 4 | 4 |

Tabla 3.Tabla Comparativa Microcontroladores Analog Devices

| ANALOG DEVICE | ADUC7019 | ADUC7020 | ADUC7021 | ADUC7022 | ADUC7024 | ADUC7025 | ADUC7026 | ADUC7027 | ADUC7128 |
|------------------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Velocidad de trabajo (MIPS) | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 | 40 |
| Memoria Flash (Kbytes) | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 62 | 126 |
| SRAM (Bytes) | 8192 | 8192 | 8192 | 8192 | 8192 | 8192 | 8192 | 8192 | 8192 |
| Número de I/O | 14 | 14 | 13 | 13 | 30 | 30 | 40 | 40 | 28 |
| Número de canales de ADC | 5 | 5 | 8 | 10 | 10 | 12 | 12 | 16 | 10 |
| Otros puertos | - | - | - | - | PWM | PWM | PWM | PWM | DAC y PWM |
| Números de salidas del DAC | 3 | 4 | 2 | - | 2 | - | 4 | - | - |

Tabla 4. Tabla Comparativa Microcontroladores Cirrus Logia

| ARM7 | | | | | | |
|---------------|-------------------|------------------------|-------------------|---------------------|------------------------|-----------------|
| | Procesador | Velocidad (MHz) | Cache (KB) | SRAM(Kbytes) | Controlador LCD | Ethernet |
| EP7309 | ARM720T | 74 | 8 | 48 | x | - |
| EP7311 | ARM720T | 90 & 74 | 8 | 48 | x | - |
| EP7312 | ARM720T | 90 & 74 | 8 | 48 | x | - |

| ARM9 | | | | | |
|---------------|------------------------|------------|-------------------------------|------------------------|-----------------|
| | Velocidad (MHz) | USB | Procesador de gráficos | Controlador LCD | Ethernet |
| EP9301 | 166 | 2 | - | x | x |
| EP9302 | 200 | 2 | - | x | x |
| EP9307 | 200 | 3 | x | | x |
| EP9312 | 200 | 3 | - | | x |
| EP9315 | 200 | 3 | x | | x |

Tabla 5. Tabla Comparativa Microcontroladores Sanyo

| | LC67F5006A / LC7F5104A | LC690123A |
|-------------------------------------|-------------------------------|-----------------------|
| ROM (KBits) | 768 / 512 | 768 |
| RAM (KBits) | 32 | 32 |
| Voltaje del CPU y las I/O(V) | 2,25 a 2,75 / 3,0 a 3,6 | 1,1 a 1,3 / 3,0 a 3,6 |
| Corriente (mA) | 25 | 25 |
| Num. de I/O | 68 | 40 |
| ADC de 8 bits | 1 | 1 |
| Timer | 5 de 16 bits | 1 de 32 bits |
| Periféricos Seriales | UART | SSP, UART |
| Dispositivo USB | - | 1 |
| Interfaz de Audio | - | 1 |
| Interfaz de Video | - | 1 |

ANEXO 2

Tabla Comparativa Entornos de Desarrollo

Tabla 1. Tabla De Entornos De Desarrollo

| Entornos de Desarrollo | | | | | |
|-------------------------------|---|------------------------------------|--|--|---------------|
| | Lenguaje de programación | Sistema operativo requerido | Características | Aplicaciones | Precio |
| AdaMULTI | Ada 95, C, C++ y la integración entra C++ y FORTRAN | Windows y UNIX | Es un completo ambiente de desarrollo utilizado para aplicaciones integradas. Sus componentes mantienen una constante comunicación entre ellos para saber como están trabajando y no se produzcan fallas de una manera muy rápida y eficiente. | Es un constructor de proyectos, depurador de Source-level, analizador de eventos, veriifcación de errores en tiempo real, análisis del código de respaldo, versiones del sistema de control, perfil de rendimiento, examinador gráfico | US \$1350.00 |
| Visual X-Tools | C++ y FORTRAN | Windows 98 y Windows NT | Código de navegación, Argumentos completos, Soporte para regrabado, Tiene soporte para seleccionar código por bloques, Reemplaza y lista los símbolos automáticamente, Busca y reemplaza directorios y archivos. | Permite combinar la edición de un programa, tomando la versión original, luego separa las modificaciones que se la hagan y lo guarda dentro de un archivo | US \$1000.00 |
| Embest IDE | ANSI C y ARM | Windows 98 y Windows NT | Un integrado ambiente de desarrollo, facilidad para manejar proyectos, integra un editor de código fuente, compiladores GNU, assembler y linker, librerías GNU para ANSI C, soporte para set de herramientas ARM, corrección de errores, simulador para el set | Permite manejar y construir proyectos, establece y dirige las comunicaciones entre el host y la tarjeta, corre y depura las aplicaciones realizadas | US \$1160.00 |

ANEXO 3

Tabla Comparativa Emuladores

Tabla 1. Emuladores

| EMULADORES | | | | | | | |
|------------------------------------|---|------------------|---|---|---|---|-------------|
| Keil ULINK USB-JTAG Adapter | Permite conectar el puerto USB a la tarjeta de hardware de la computadora para realizar la depuración de los mismos mientras estos se están ejecutando en la tarjeta de hardware. | | | | | | US \$300.00 |
| | Soporte | Rango de Voltaje | Lenguajes de programación con los que trabaja | Ofrece | Sistemas operativos con los que trabaja | Modo de Conexión al PC | Precio |
| Embest PowerICE | Da soporte para los núcleos ARM7TDMI, ARM720T, ARM9TDMI, ARM920T, ARM922T, ARM940T | De 3V a 5V | C, C++ | Reloj en tiempo real para generar interrupciones, el usuario puede controlar la tarjeta del procesador completamente con el Embest PowerICE, puede ser energizado desde una tarjeta de desarrollo o desde una fuente de potencia externa, pueden configurar diferente | Cualquier sistema operativo | Se conecta al puerto paralelo de la computadora | US \$360.00 |

| | | | | | | | |
|-----------------------|--|----------------|---|--|---|-----------------------|-------------|
| Embest UNetICE | Da soporte a las arquitecturas ARM7 y ARM9 | De 3V a 5V | C, C++ | Soporta interrupciones por software y por hardware, soporte de Internet para desarrollo y para la depuración remota, ejecución de programas en tiempo real | Windows 98/NT/2000/XP | Puerto USB o Ethernet | US \$560.00 |
| Nohau Emulator | Todos los núcleos de las familias ARM7 y ARM9 | De 3V a 5V | Trabaja con el set de instrucciones Thumb o ARM | Soporte para programación de la Flash, soporte para múltiples dispositivos JTAG | Windows 98/ME/2000/XP | USB de alta velocidad | US \$300.00 |
| TRACE32-FIRE | ARM7TDMI, ARM710T, ARM720T, ARM9TDMI, ARM920T, ARM940T, ARM9X6 | De 3,3V a 5,5V | C, C++, JAVA, ADA, Peral | Análisis de rendimiento en el programa, funciones o módulos bases en áreas definidas por el usuario | Trabaja con cualquier sistema operativo | USB de alta velocidad | US \$380.00 |

ANEXO 4

Tabla Comparativa Tarjetas de Desarrollo

Tabla 1. Tarjetas de Desarrollo

| Tarjetas de Desarrollo | | | | | | | | | | | | | | | |
|------------------------|-------------|---------------|----------------|--------------------|--------------------|--------------------|-------------------------|------------|-----|-----|-----|-----|------|---|----------|
| | Dimensiones | Temperatura | SDRAM (Mbytes) | Voltaje de entrada | Comunicación RS485 | Comunicación RS232 | Modulos Ethernet 10/100 | Puerto USB | PWM | CAN | ADC | LCD | JTAG | Extras | Precio |
| Embest TABSAM7S | 100x100 mm | (-45) - 85 °C | 32 | 5 V | | 1 | | 1 | 1 | | 1 | x | x | 1 botón de reset, 4 botones de propositos generales | US \$119 |
| Embest ATEB9200 | 180x180 mm | (-45) - 85 °C | 32 | 5 V | 1 | 1 | 1 | 1 | | | | | x | Interfaz IrDA, Interfaz SD/MMC, Interfaz CF, Interfaz SM | US \$370 |
| CSB637 | | (-45) - 85 °C | 64 | 5 V | | 3 | 1 | 1 | | | | x | x | Soporte para I2S (audio) | |
| Embest ATEB40x | | (-45) - 85 °C | 32/64 | 5 V | | 2 | | | | | | x | x | Botón de reset, y 8 de propositos generales, 8 leds indicadores | US \$119 |
| Embest AX4510 | 177x130mm | (-45) - 85 °C | 16 | 5 V | | 2 | 1 | 1 | | | | x | x | Botón de reset, y 4 de propositos generales, 4 leds indicadores, 2 puertos PS2 | US \$139 |
| Embest S3CEV40 | 190x190mm | (-45) - 85 °C | 8 | 5 V | | 1 | 1 | 1 | | | | x | x | Botón de reset, y 2 botones simalr interrupciones, 2 leds | US \$266 |
| Keil MCB2100 | | | | 5 V | | 2 | | | | 2 | | | x | | US \$199 |
| KB9202 | 100 x 160mm | | 64 | 5 V | 1 | 1 | 1 | 1 | | | | x | x | Interfaz para memoria externa, sensor de temperatura digital, RTC | US \$215 |
| Embest LPCEB2000 | 182x139mm | (-45) - 85 °C | | 9V | | 2 | | | | 1 | | | x | Teclado 4x4, Displa 8 dígitos de 8 segmentos, salida de señal de parlante análoga, boton de reset | US \$165 |
| Embest S3CEB2410 | 190x140mm | (-45) - 85 °C | 64 | 5 V | | 3 | 1 | 1 | | | 1 | x | x | 4 botones para simular interrupciones, 4 leds indicadores, 2 inicadores de potencia, RTC | US \$815 |
| Embest STDV710A | 163x1124mm | (-45) - 85 °C | 16 | 5 V | | 2 | 1 | 1 | | 1 | 4 | x | | 5 leds indicadores, 1 botón de reset, 1 botón de wake up, 5 botones de propositos generales | US \$179 |

INDICE DE FIGURAS

| | | |
|------|--|-----|
| 1.1 | Gráfica comparativa de las tecnologías ARM, Thumb, Thumb-2 | 16 |
| 2.1 | Diagrama de bloques de estructura interna ARM | 21 |
| 2.2 | Organización interna del procesador ARM | 22 |
| 2.3 | Almacenamiento de cada etapa del pipeline en los registros | 24 |
| 2.4 | Funcionamiento del pipeline | 25 |
| 2.5 | Formato del registro de estado | 31 |
| 2.6 | Formato de una instrucción | 34 |
| 2.7 | Instrucciones para transferencia en bloques | 37 |
| 2.8 | Instrucción Swap | 38 |
| 2.9 | Interrupción por software | 38 |
| 2.10 | Instrucción Branching Link | 39 |
| 3.1 | Encapsulado LQFP | 58 |
| 3.2 | Encapsulado PBGA | 58 |
| 3.3 | Encapsulado FBGA | 59 |
| 3.4 | Encapsulado TFBGA | 59 |
| 4.1 | Diagrama de bloques AT91SAM7S64 | 86 |
| 4.2 | Mapa de Memorias | 90 |
| 4.3 | Mapa de periféricos | 93 |
| 4.4 | Diagrama de bloques del controlador del sistema | 96 |
| 4.5 | Diagrama de bloques del controlador del sistema | 97 |
| 5.1 | Adaptador USB-JTAG | 110 |
| 5.2 | Embest PowerICE | 111 |
| 5.3 | Embest Emulator | 112 |
| 5.4 | UNetICE Emulator | 113 |
| 5.5 | UNetICE Emulator | 115 |
| 5.6 | Emulador TRACE32-FIRE | 115 |
| 5.7 | Tarjeta de desarrollo Embest ATEBSAM7S | 117 |

| | |
|--|-----|
| 5.8 Tarjeta de desarrollo Embest ATEB9200 | 118 |
| 5.9 Tarjeta de desarrollo Embest ATEB40x | 120 |
| 5.10 Tarjeta de desarrollo Embest AX4510 | 121 |
| 5.11 Tarjeta de desarrollo Embest S3CEV40 | 123 |
| 5.12 Tarjeta de desarrollo Keil MCB210 | 125 |
| 5.13 Tarjeta de desarrollo KB9202 | 126 |
| 5.14 Tarjeta de desarrollo Embest LPCEB2000 | 128 |
| 5.15 Tarjeta de desarrollo Embest S3CEB2410 | 128 |
| 5.16 Tarjeta de desarrollo Embest STDV710A | 130 |
| 6.1 Entorno de desarrollo Keil uVision3 | 132 |
| 6.2. Opciones de tarjeta | 134 |
| 6.3. Archivos del Workspace | 134 |
| 6.4. Libros y hojas técnicas | 135 |
| 6.5. Build Toolbar | 136 |
| 6.6. Debug Toolbar | 137 |
| 6.7. Window disassembly | 138 |
| 6.8. Ventana de memorias | 139 |
| 6.9. Controlador de entradas y salidas paralelas | 140 |
| 6.10. Timer Counter | 142 |
| 6.11. Conversor Análogo Digital | 145 |
| 6.12. Ventana de Instalación del uVision3 | 149 |
| 6.13. Dirección de la Carpeta del Keil | 149 |
| 6.14. Datos del Propietario | 150 |
| 6.15. Instalación del uVision3 | 150 |
| 6.16. Instalación Terminada | 151 |
| 6.17. Abrir proyecto | 151 |
| 6.18. Archivo que se va abrir | 152 |
| 6.19. Editor Toolbar | 152 |
| 6.20. Edición de código fuente | 153 |
| 6.21. Conversión Analógica – Digital | 153 |
| 6.22. Compilación ADC | 156 |
| 6.23. Compilación ADC con error | 156 |
| 6.24. Build target | 156 |
| 6.25. Iniciar Simulación | 157 |

| | |
|---|-----|
| 6.26. Run programa | 157 |
| 6.27. Escoger simulador | 158 |
| 6.28. Toolbox | 158 |
| 6.29. Señal Sinusoidal Generada | 159 |
| 6.30. Parámetros del Conversor Análogo Digital | 159 |
| 6.31. Señal con Prescaler 0x00 | 160 |
| 6.32. Señal con Prescaler 0x1F | 161 |
| 6.33. Señal con Prescaler 0x3F | 161 |
| 6.34. Conversión ADC con Voltaje de Referencia 1V | 162 |
| 6.35. Conversión ADC con Voltaje de Referencia 2V | 163 |
| 6.36. Conversión ADC con Voltaje de Referencia 0V | 164 |

INDICE DE TABLAS

| | |
|---|-----|
| 1.1 Aplicaciones familia ARM11 | 12 |
| 1.2 Características y beneficios de la familia NEON | 18 |
| 2.1 Modos de procesador ARM | 28 |
| 2.2 Set de registros ARM | 29 |
| 2.3 Bits de modo | 32 |
| 2.4 Modo de procesamiento de excepciones | 33 |
| 2.5 Códigos de campo ARM | 35 |
| 2.6 Instrucciones del procesamiento de datos | 36 |
| 2.7 Instrucciones de transferencia de datos | 37 |
| 3.1 Tipos de encapsulados chips LPC2101/2102/2103 | 69 |
| 3.2 Tipos de encapsulados chips LPC2104/2105/2106 | 70 |
| 3.3 Tipos de encapsulados chips LPC2114/2124 | 71 |
| 3.4 Tipos de encapsulados chips LPC2119/2129 | 72 |
| 3.5 Tipos de encapsulados chips LPC2131/2132/2134/2136/2138 | 72 |
| 3.6 Tipos de encapsulados chips LPC2141/2142/2144/2146/2148 | 74 |
| 4.1 Descripción de señales | 87 |
| 4.2 Multiplexión en el controlador de PIO | 94 |
| 4.3 Entradas de reloj | 102 |
| 5.1 Características emulador JTAG | 108 |
| 5.2. Características CSB637 | 119 |
| 6.1. Valores Convertidos a Digitales | 160 |
| 6.2. Valores Convertidos a Digitales con Vref 1 | 163 |
| 6.3. Valores Convertidos a Digitales con Vref 2 | 164 |

GLOSARIO

| | |
|-------------|--|
| ADC | Converso Análogo Digital Analog to Digital Converter |
| DMA | Acceso de memoria directo Direct-memory access |
| EPB | Bus de periféricos integrado Ebedded peripheral bus |
| FIFO | Lo que primero entra / primero sale First in / first out |
| GPIO | I/O de propósito general General purpose I/O |
| ICT | Prueba en el circuito In circuit test |
| IR | Comunicación Infrarroja Infrared |
| IRQ | Respuesta a Interrupción Standard interrupt request |
| JTAG | Puerto para prueba Joint Test Action Group |
| MIPS | Millón de instrucciones por segundo Millions of instructions per second |
| MMU | Unidad de memoria magnética Memory management unit |

| | |
|-------------|--|
| PCB | Printed circuit board Circuito impreso |
| PDA | Asistente digital personal Personal digital assistant |
| PLL | Lazo de bloqueo de fase Phase locked loop |
| RISC | Reduced instruction set computer Set de instrucciones reducido |
| RTC | Reloj en tiempo real Real-Time Clock |
| SRAM | Acceso de memoria random estática Static random access memory |
| SSI | Interfaz serial sincrónica synchronous serial interface |
| TAP | Puerto de acceso de prueba Test access port |
| TLB | Translation lookaside buffer Buffer de transmisión de datos |
| UART | Recepción asincrónica universal Universal asynchronous receiver |