



**Universidad Autónoma de Baja California
FAC. DE CS. QUIM. E INGENIERIA
INGENIERIA EN COMPUTACION**

PRACTICA 6

Laboratorio de: Microprocesadores y microcontradores

Equipo:

López Madrigal Leonardo

Maestro:

García López Jesús Adán

Tijuana, B. C.

30 Marzo, 2017

Práctica 6

Programación en Lenguaje Ensamblador del ATmega1280

Objetivo:

Mediante esta práctica el alumno aprenderá el uso básico de programa AVR Studio como herramientas de programación en lenguaje Ensamblador para el sistema ATmega1280.

Equipo:

- Computadora Personal

Teoría:

- Arquitectura interna y conjunto de instrucciones del ATmega1280

Teoría:

Los AVR son una familia de [microcontroladores RISC](#) del fabricante estadounidense [Atmel](#). La arquitectura de los AVR fue concebida por dos estudiantes en el Norwegian Institute of Technology, y posteriormente refinada y desarrollada en Atmel Norway, la empresa subsidiaria de [Atmel](#), fundada por los dos arquitectos del chip. Cuenta con bastantes aficionados debido a su diseño simple y la facilidad de programación. Se pueden dividir en los siguientes grupos :

- **ATxmega:** procesadores muy potentes con 16 a 384 kB de [memoria flash](#) programable, encapsulados de 44, 64 y 100 pines (A4, A3, A1), capacidad de DMA, eventos, criptografía y amplio conjunto de periféricos con DACs.
- **ATmega:** microcontroladores AVR grandes con 4 a 256 kB de [memoria flash](#) programable, encapsulados de 28 a 100 pines, [conjunto de instrucciones](#) extendido (multiplicación y direccionamiento de programas mayores) y amplio conjunto de periféricos.
- **ATtiny:** pequeños microcontroladores AVR con 0,5 a 8 kB de [memoria flash](#) programable, encapsulados de 6 a 20 pines y un limitado set de periféricos.
- **AT90USB:** ATmega integrado con controlador [USB](#)
- **AT90CAN:** ATmega con controlador de [bus CAN](#)
- Tipos especiales: algunos modelos especiales, por ejemplo, para el control de los cargadores de baterías, pantallas LCD y los controles de los motores o la iluminación.
- **AT90S:** tipos obsoletos, los AVR clásicos

Microcontrolador AVR 1280/2560

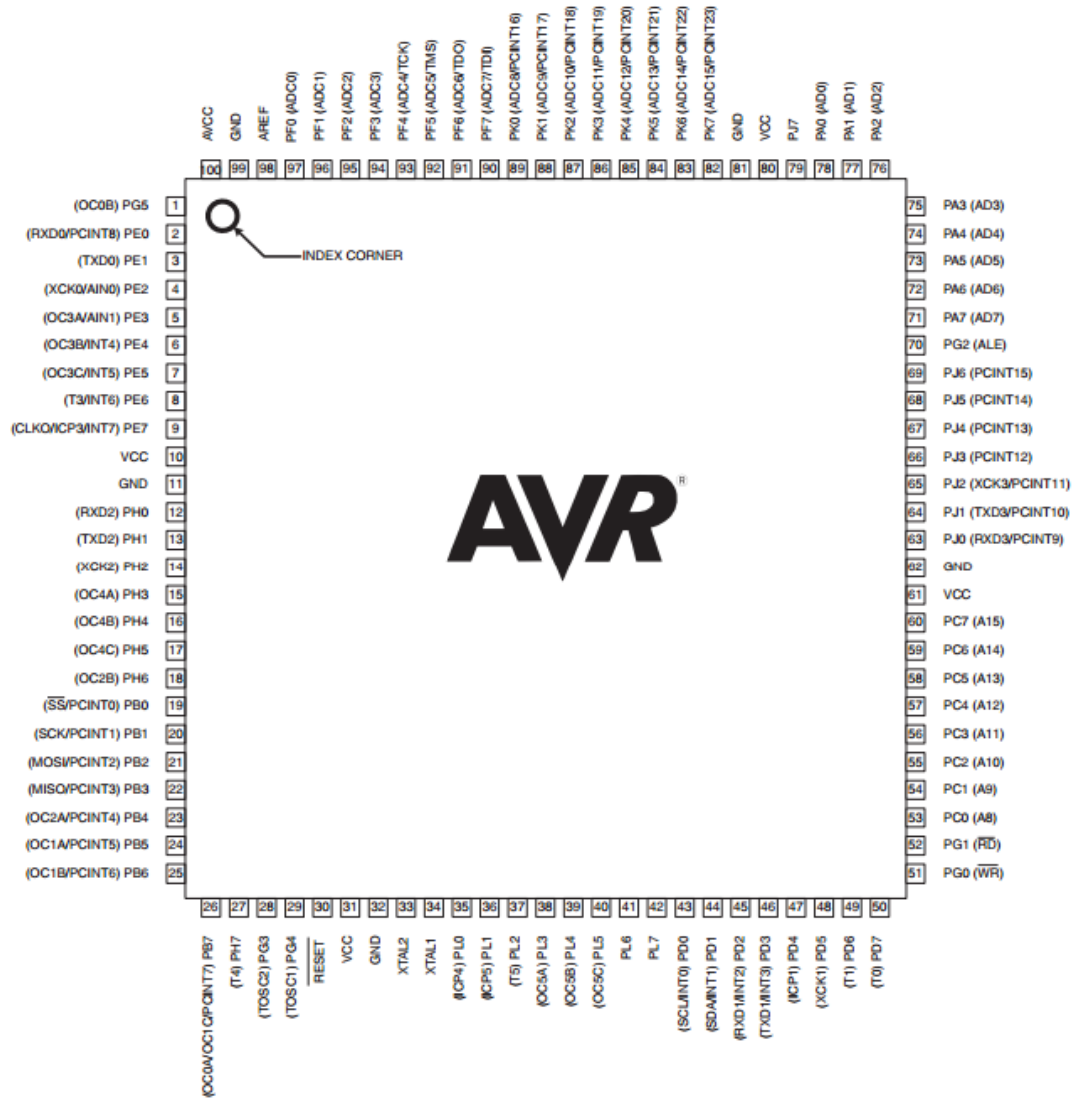
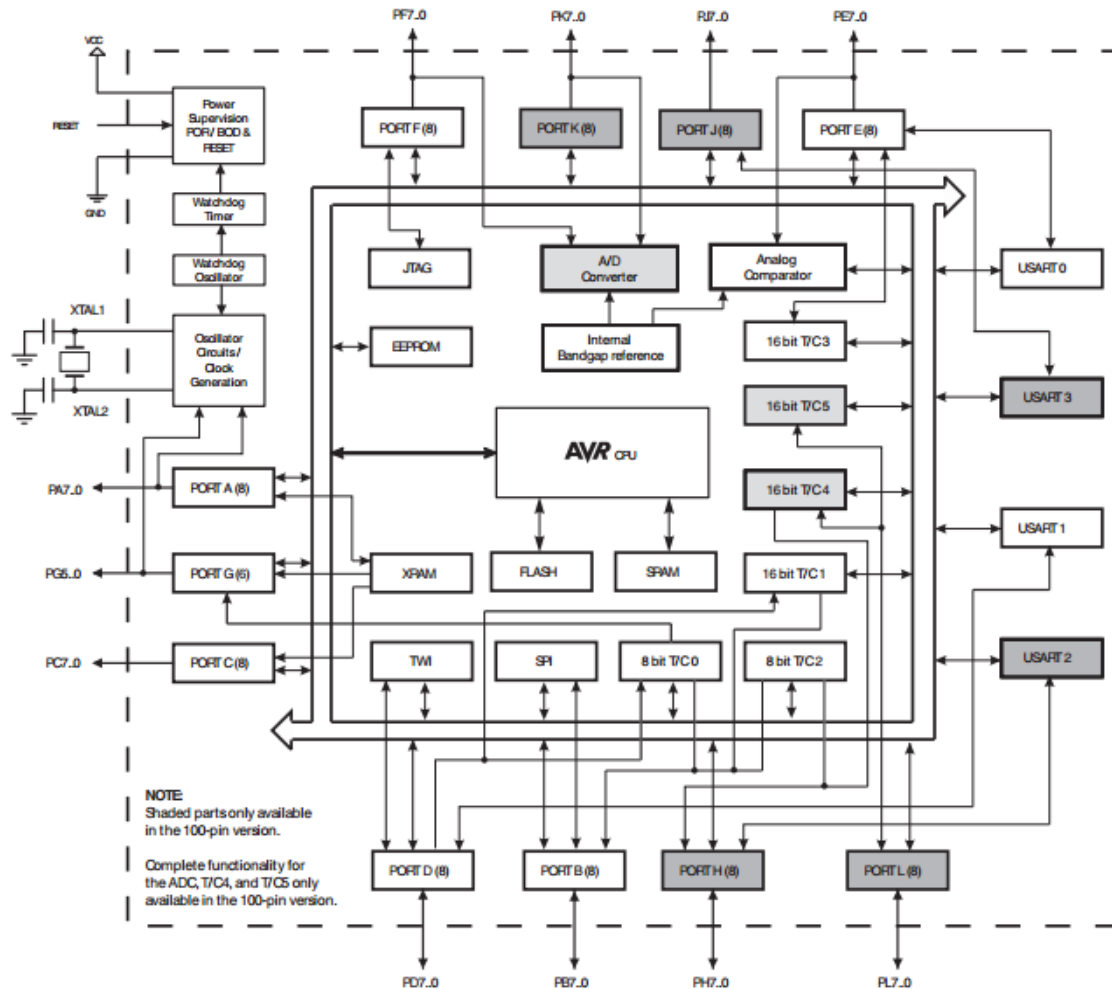


Diagrama de Bloques:



Conjunto de instrucciones del ATmega1280

Cuenta con 32 registros de 8 bits para uso general, 6 de estos pueden funcionar como 3 registros de 16 bits para direccionamiento indirecto.

Instrucciones aritmeticas y logicas

ADD Rd, Rr	Suma entre dos registros
ADC Rd, Rr	Suma con acarreo entre dos registros
ADIW Rdl, K	Suma de palabra con constante
SUB Rd, Rr	Resta entre dos registros
SUBI Rd, K	Resta constante a un registro
SBC Rd, Rr	Resta con préstamo entre dos registros

SBCI Rd, K	Resta con préstamo entre dos registros
SBIW Rd, K	Resta de palabra con constante
AND Rd, Rr	AND entre registros
ANDI Rd, K	AND entre registro y constante
OR Rd, Rr	OR entre registros
ORI Rd, K	OR entre registro y constante
EOR Rd, Rr	OR exclusivo entre registros
COM Rd	Complemento a 1
NEG Rd	Complemento a 2
SBR Rd, K	Activar bit(s) en registro (OR)
CBR Rd, K	Desactivar bit(s) en registro (AND)
TST Rd	Test para cero o menor
CLR Rd	Desactivar bits de registro
SER Rd	Activar bits de registro
MUL Rd, Rr	Multipliación sin signo
MULS Rd, Rr	Multipliación con signo
MULSU Rd, Rr	Multipliación con signo y sin signo
FMUL Rd, Rr	Multipliación fraccional sin signo
FMULS Rd, Rr	Multipliación fraccional con signo
FMULSU Rd, Rr	Multipliación fraccional con signo y sin signo

Instrucciones Branch

RJMP k	Salto relativo $PC \leftarrow PC + k + 1$
IJMP	Salto indirecto a (Z) $PC \leftarrow Z$
JMP k	Salto directo $PC \leftarrow k$
RCALL k	Llamada relativa a subrutina $PC \leftarrow PC + k + 1$
ICALL	Llamada indirecta a (Z) $PC \leftarrow Z$
CALL k	Llamada directa a $PC \leftarrow k$
RET	Retorno de subrutina $PC \leftarrow STACK$
RETI	Retorno de interrupcion $PC \leftarrow STACK$
CPSE Rd, Rr	Comparar, salto si es igual
CP Rd, Rr	Comparar
CPC Rd, Rr	Comparar con acarreo
CPI Rd, K	Comparar con constante
SBRC Rr, b	Omitir si bit desactivdo
SBRs Rr, b	Omitir si bit activado
SBIC P, b	Omitir si bit de I/O desactivado
SBIS P, b	Omitir si bit de I/O activado
BRBS s, k	Branch si bandera de estatus activa
BRBC s, k	Branch si bandera de estatus desactiva
BREQ k	Branch si igual
BRNE k	Branch si no igual
BRCS k	Branch si acarreo activo
BRCC k	Branch si acarreo desactivo
BRSH k	Branch si igual o mayor
BRLO k	Branch si menor

BRMI k	Branch si menos
BRPL k	Branch si mas
BRGE k	Branch si es mayor o igual a cero, con signo
BRLT k	Branch si es menor a cero, con signo
BRHS k	Branch si bandera de medio acarreo activa
BRHC k	Branch si bandera de medio acarreo desactiva
BRTS k	Branch si bandera T activa
BRTC k	Branch si bandera T desactiva
BRVS k	Branch si bandera overflow activa
BRVC k	Branch si bandera overflow desactiva
BRIE k	Branch si bandera de interrupciones activa
BRID k	Branch si bandera de interrupciones activa

Instrucciones de bit y prueba de bit

SBI P,b	Activar bit en registro I/O
CBI P,b	Desactivar bit en registro I/O
LSL Rd	Desplazamiento logico a la izquierda
LSR Rd	Desplazamiento logico a la derecha
ROL Rd	Rotar a la izquierda a travez de acarreo
ROR Rd	Rotar a la derecha a travez de acarreo
ASR Rd	Desplazamiento a la derecha aritmetico
SWAP Rd	Intercambiar nibbles
BSET s	Activar bandera
BCLR s	Desactivar bandera
BST Rr, b	Guardar bit de registro en T
BLD Rd, b	Cargar bit T a registro
SEC	Activar acarreo
CLC	Desactivar acarreo
SEN	Activar bandera negativa
CLN	Desactivar bandera negativa
SEZ	Acvtivar bandera cero
CLZ	Desactivar bandera cero

Conclusión:***López Madrigal Leonardo***

En esta práctica aprendí a utilizar el AVR Studio, el ensamblador del micro-controlador del Arduino/t-juino atmega1280, es muy parecido al x86 incluso te hace a hacer mejores problemas y contiene unas nuevas instrucciones a nivel bit que el otro no tenía el procesador 8088, la sintaxis es casi idéntica solo que aquí al parecer se pueden usar unas cosas de C, la practica consistió en poner en reversa un número del menos significativo al más significativo estuvo sencilla porque en la práctica pasada con puertos hicimos lo mismo pero en lenguaje C, ha sido de las practicas más sencillas porque aprendí a usar una nueva herramienta.

Bibliografía o referencias:

- <http://www.atmel.com/Images/Atmel-0856-AVR-Instruction-Set-Manual.pdf>