



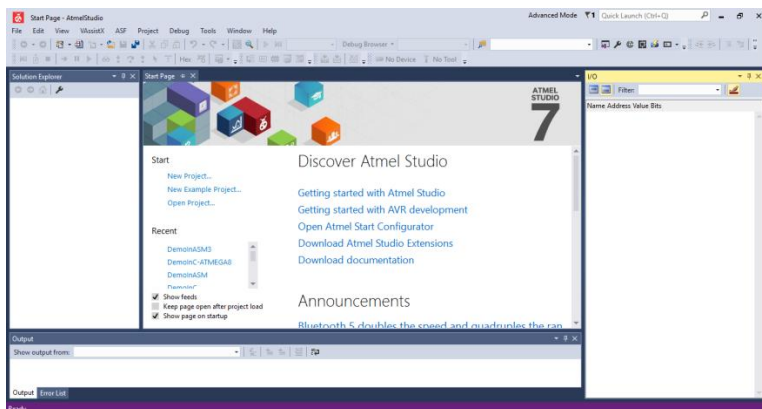
Alumno:	
Fecha de Presentación:	Observaciones:
Calificación:	Visado:
	<i>Res. Práctica Ing. Gustavo Maggiolo / Ing. Rubén Britos</i>
Objetivos: Que el alumno empiece a conocer el software Atmel Studio 7.0, la herramienta de Atmel para trabajar con sus microcontroladores.	
Requisitos: Haber asistido a la primera clase de microcontroladores Atmel.	
Elementos a Utilizar: Computadora personal. Software Atmel Studio 7.0.	

TEMA: Inicio en Atmel Studio 7.0

Actividades:

- ✓ Creación de proyectos
 - ✓ Configuración de la placa Arduino
 - ✓ Primer programa en ensamblador para la placa Arduino
 - ✓ Como descargar documentación (ayuda)
 - ✓ Anexo
-
- ✓ Creación de proyectos

1. Acceda a Atmel Studio 7.0. Para comenzar vamos a hacer un proyecto.



En la ventana de inicio del software acceda a:

Getting started with Atmel Studio

Y observe que información se suministra.

Hacer lo mismo con:

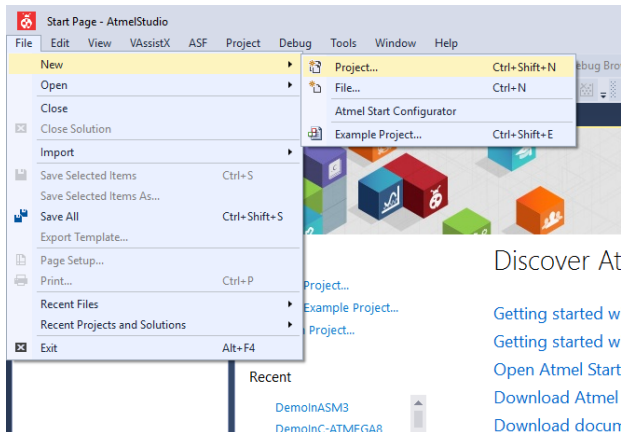
Getting started with AVR development

Nota: es necesario estar conectados a internet.

Link de descarga: <https://www.microchip.com/mplab/avr-support/atmel-studio-7>



MICROCONTROLADORES ATMEL



Para generar un nuevo proyecto en Atmel Studio accedemos a **File → New → Project...**

Esto nos abrirá una ventana donde podremos configurar el tipo de proyecto, el nombre y la ubicación del mismo.

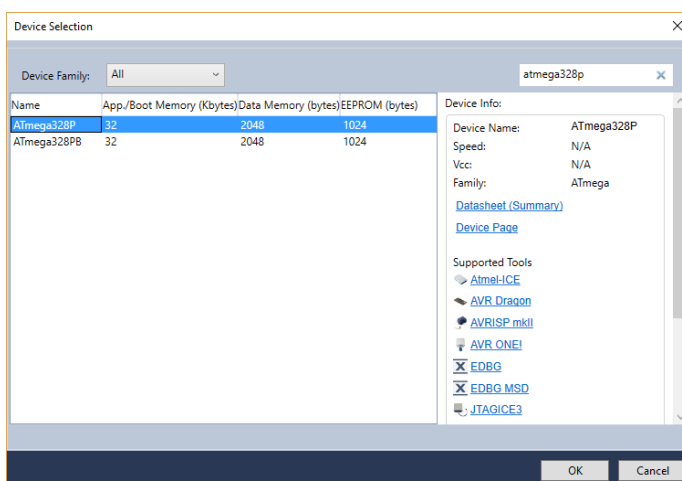
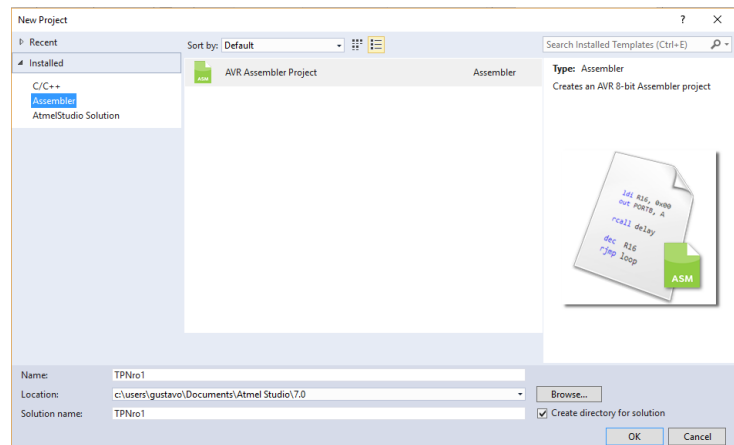
En la imagen siguiente se observa la ventana de generación de un nuevo proyecto en Atmel Studio.

En este caso seleccionamos el lenguaje *Assembler*.

Como nombre le asignamos TPNro1 (observar que no se ingresa ningún tipo de extensión al proyecto).

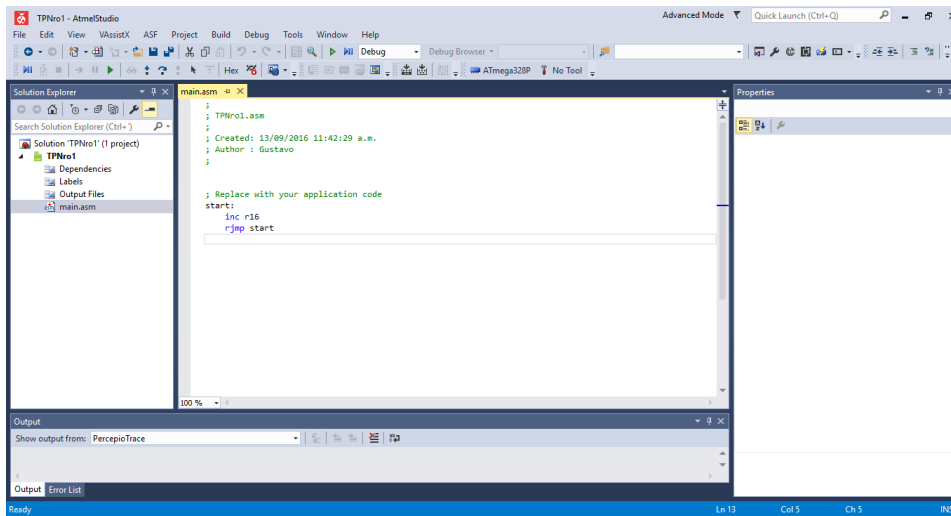
La ubicación del proyecto es el "por defecto" (recomiendo no modificarlo) de este modo, todos los proyectos se encuentran en el mismo lugar, facilitando la búsqueda de los mismos.

Nota: recomiendo dejar el tilde en "Create directory for solution".



Luego de presionar **OK** en la pantalla anterior, nos mostrara la pantalla de selección del microcontrolador del proyecto. En la parte de búsqueda ponemos atmega328p y nos busca los micros que tenga ese comienzo. Seleccionamos ATmega328P y damos **OK**.

Nos mostrara una ventana del proceso de creación del proyecto, para luego finalizar y mostrarnos lo siguiente:

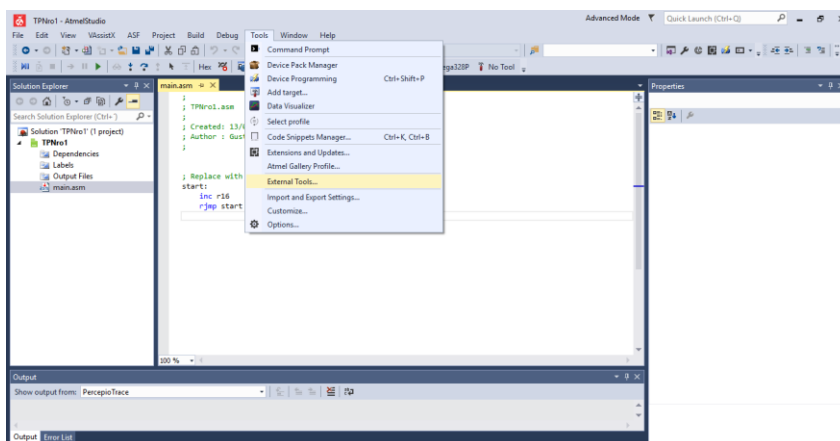


Identifique cada una de las ventanas que se observan.

Tarea: buscar que es el *ASF* en el “mundo Atmel”.

✓ Configuración de la placa Arduino

- Lo que realizaremos a continuación es la configuración de la placa Arduino en el entorno Atmel Studio 7.0, de manera que al escribir nuestros programas en el entorno Atmel Studio lo podamos descargar en la placa Arduino y podamos probarlo.



Para comenzar vamos al menú *Tools* y accedemos a la opción *External Tools*.

Esto nos abrirá una ventana como se observa en la siguiente imagen de la izquierda.

Completamos los campos como se observa en la imagen de la derecha. Es decir:



MICROCONTROLADORES ATMEL

Title¹:	Arduino Programador
Command²:	C:\Program Files (x86)\Arduino\hardware\tools\avr\bin\avrdude.exe
Arguments³:	-C "C:\Program Files (x86)\Arduino\hardware\tools\avr\etc\avrdude.conf" -v -patmega328p -c arduino -P COM8 -b 115200 -U flash:w:"\$(ProjectDir)Debug\\$(TargetName).hex":i

Nota: No olvidar seleccionar el checkbox **"Use Output Windows"**.

Los argumentos (que son pasados al programa avrdude.exe al momento de ejecutarlo) son:

-C <ruta al archivo de configuración avrdude.conf>
-v Muestra la versión al programa
-p <microcontrolador que se quiere grabar>
-c <interfaz de programación>
-P <puerto COM donde está conectada la placa Arduino>
-b <velocidad del puerto COM>
-U <especificaciones de operación de la memoria: flash (tipo), w (write), "ruta al archivo a grabar.hex", i (formato del archivo, en este caso Intel)>

Para terminar, presionamos *Apply* y luego *OK*.

Nota: Si lo desea, puede en una consola ejecutar el programa avrdude.exe y ver las diversas opciones que tiene el programa.

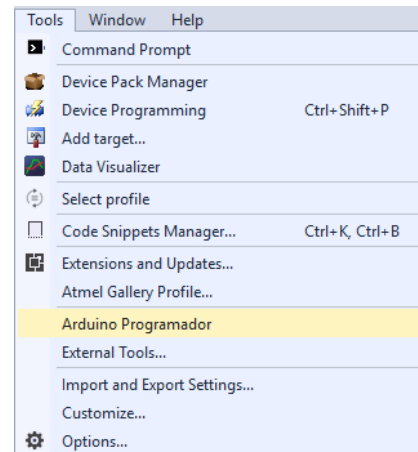
- 1- Poner el nombre que a uno le parezca mejor
- 2- Poner la ruta que corresponda al programa avrdude.exe
- 3- Poner la ruta que corresponda al archivo de configuración avrdude.conf



MICROCONTROLADORES ATMEL

De este modo, ahora nos aparecerá una nueva herramienta para trabajar, como se observa en la imagen de la derecha.

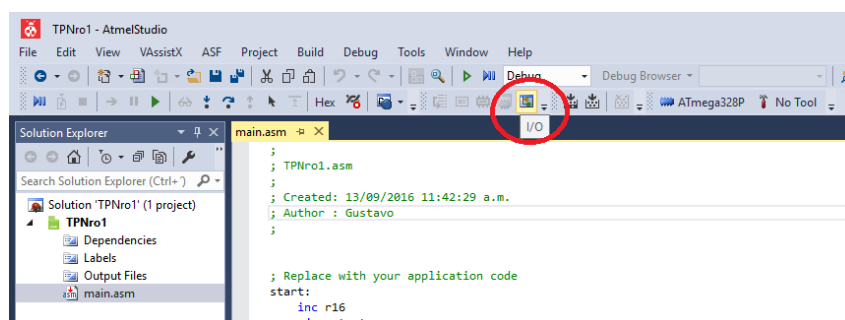
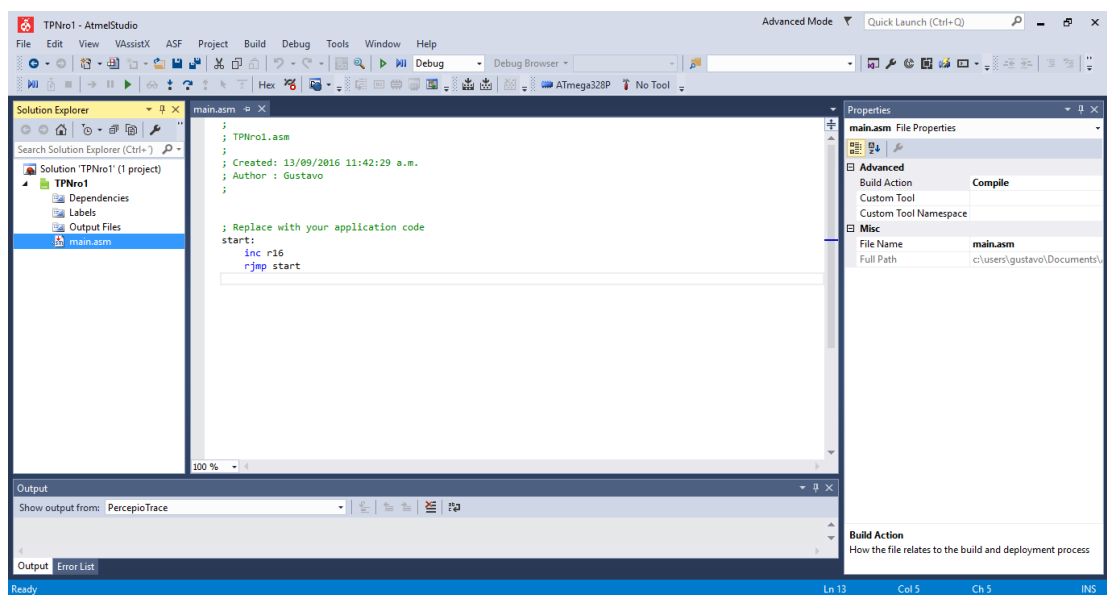
Ahora cada vez que deseamos grabar nuestro arduino, una vez compilado el archivo con el que estemos trabajando, y la compilación se haya realizado correctamente, hacemos clic en el menú *Tools* → *Arduino Programador* y el software avrduide programara el Arduino con nuestro código.




✓ Primer programa en ensamblador para la placa Arduino

3. Ahora veremos cómo hacemos nuestro primer programa en ensamblador en el micro Atmega328P que es el que trae el Arduino UNO.

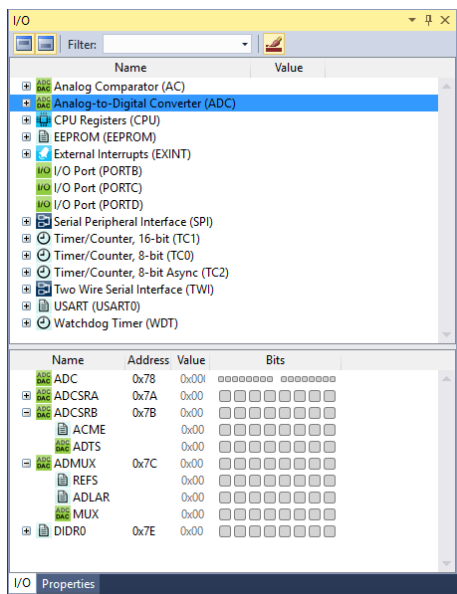
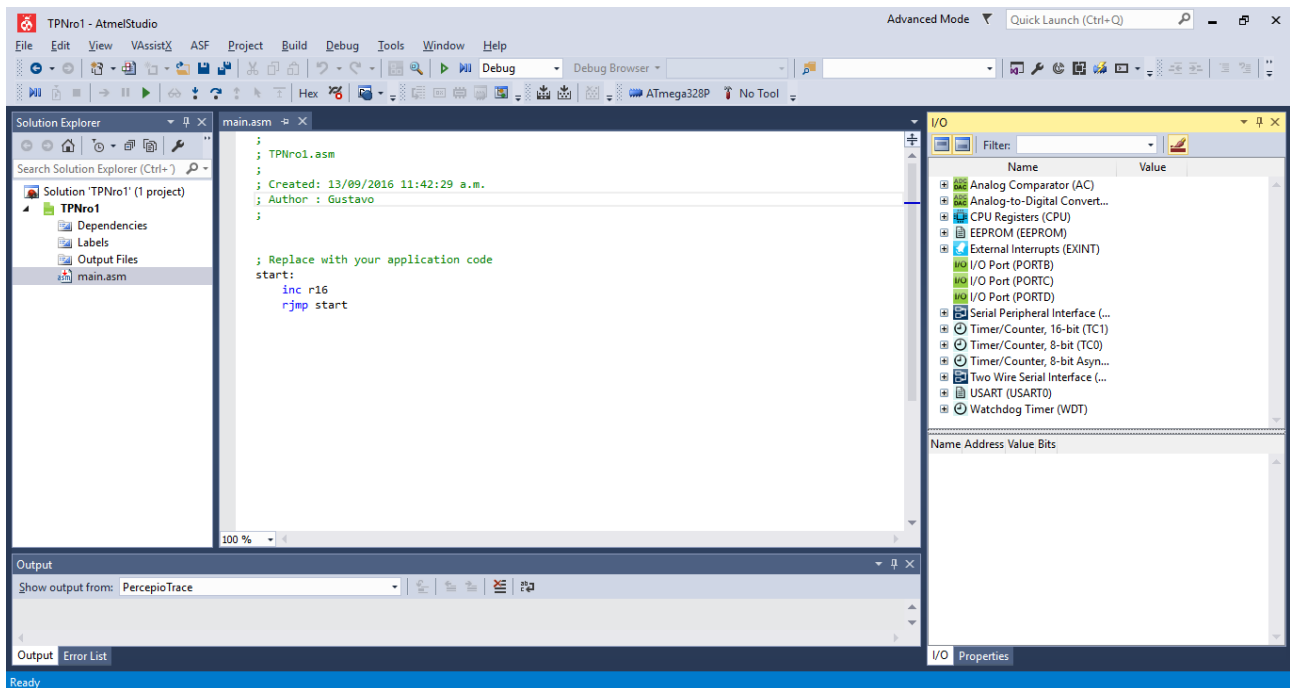
Como primer paso copie la distribución de pantalla que se muestra a la derecha. Esta es la última pantalla luego ve haber creado el proyecto en ensamblador



Ahora habilitaremos la vista de I/O, haciendo clic en el icono , como se muestra en la imagen de la izquierda. Reproducir la distribución que se muestra en la siguiente imagen.



MICROCONTROLADORES ATMEL



Como puede observarse, de la imagen anterior, al agregar la vista de I/O, se nos muestra todos los módulos internos del microcontrolador, como ser CPU, ADC, Puertos de I/O, Timers, USART, etc..

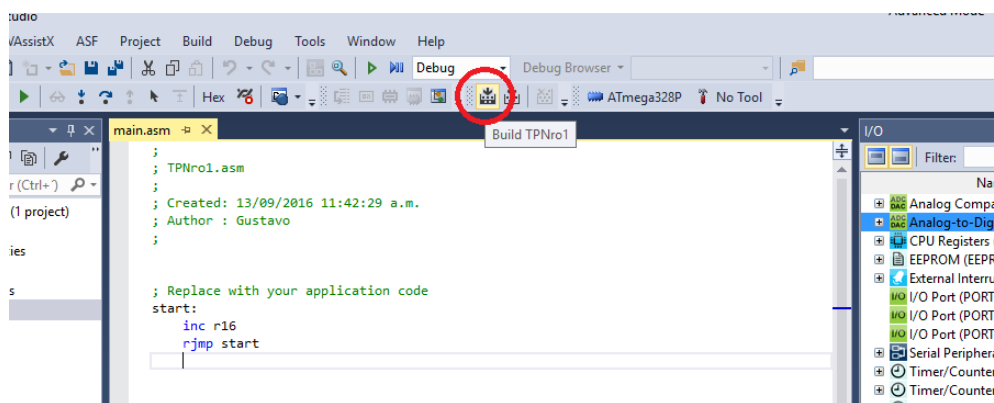
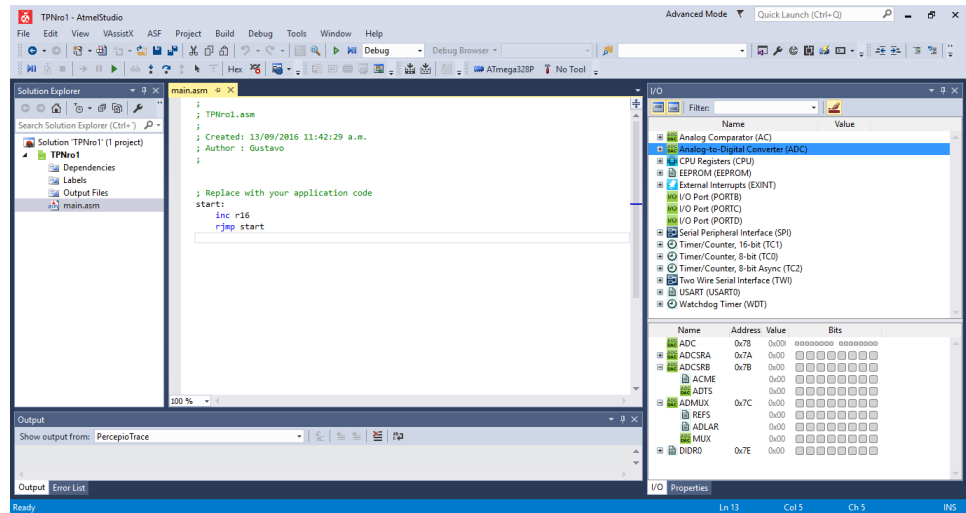
De este modo, si seleccionamos uno cualquiera, por ejemplo el ADC, se nos muestra en la parte inferior de la ventana los registros asociados a este módulo (imagen de la izquierda).


De esta manera es más sencillo trabajar con los diversos módulos del microcontrolador, ya que en esta vista podemos ver cualquier módulo, sus registros asociados y la denominación de cada bit en particular.

Ahora estamos listos para empezar a programar. El ejemplo que vamos a implementar en primer lugar es el blink de un LED en el PIN 5 del PUERTO B (PB5 - Pin Nro 19 del uC).

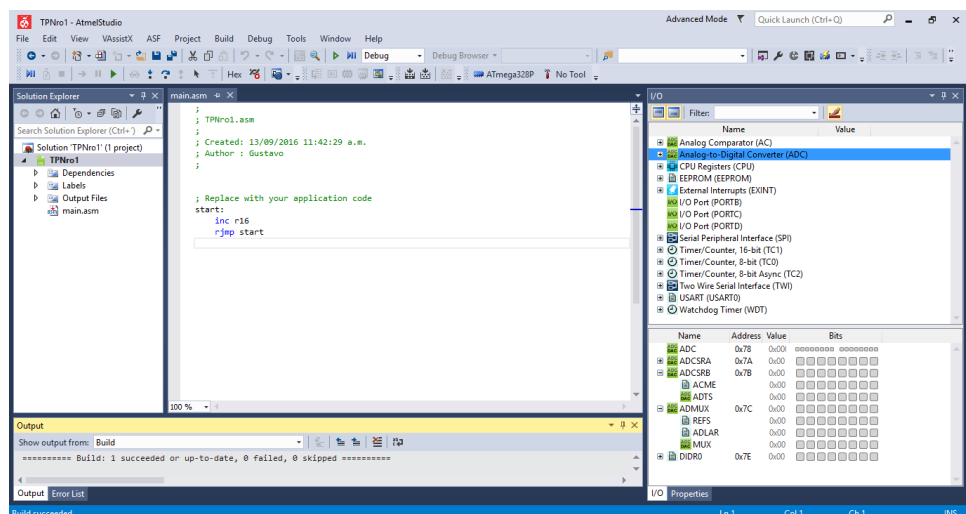


Hasta ahora lo único que realizamos fue generar el proyecto para el uC atmega328p y luego se realizaron configuraciones del entorno Atmel Studio, principalmente. Ahora procederemos a compilar el código generado por el asistente cuando se creó el proyecto.



Para proceder con la compilación del código, hacemos clic en el icono , como lo indica la imagen de la izquierda.

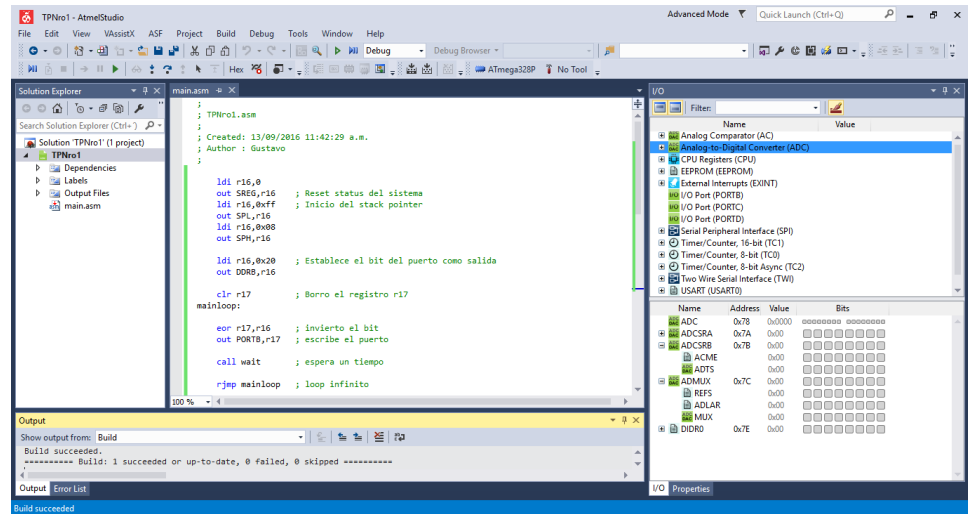
Esto nos producirá una salida similar a la mostrada en la imagen de la derecha. Se observa, en la parte inferior de la imagen, que la compilación terminó satisfactoria y que no ha habido errores.



Tarea: buscar las "Dependencias", "Etiquetas" y "Archivos de Salida" del código compilado hasta ahora.



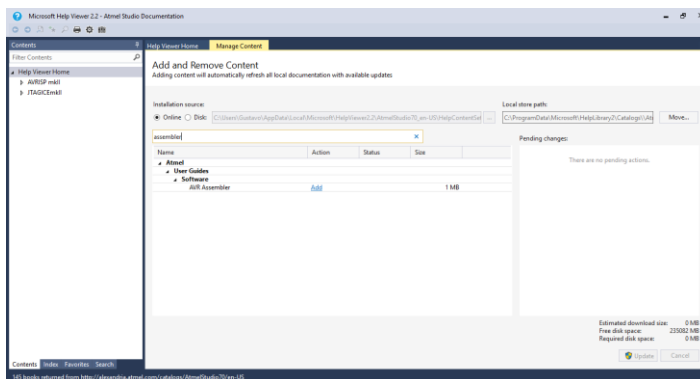
Ahora escribiremos el código de nuestro programa. Borramos el código generado por el asistente y escribimos nuestro código. Lo compilamos y nos dará como resultado lo que podemos observar en la imagen de la derecha.



Tarea: Escribir el código de la función **"wait"** para el programa anterior.

✓ Como descargar documentación (ayuda).

4. Ahora veremos como descargar documentación.

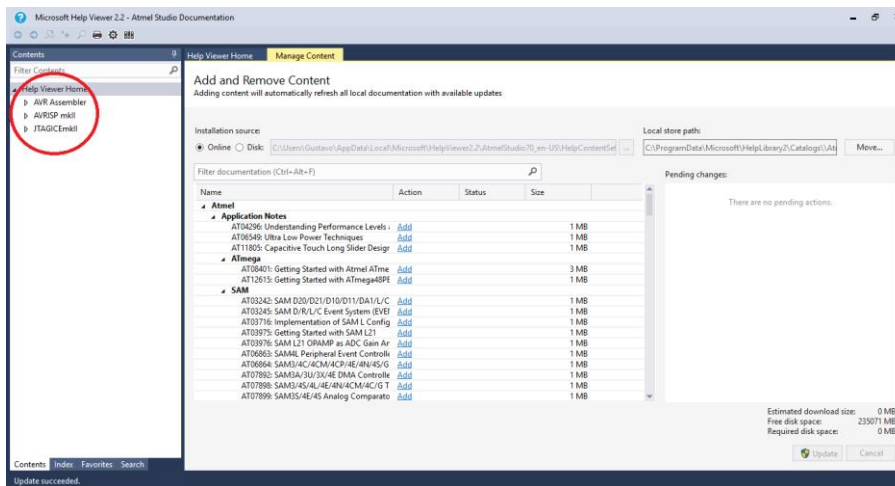
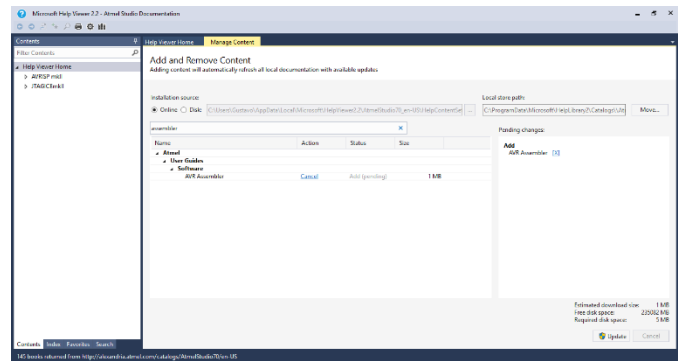


Esto es útil para los momentos que no disponemos una conexión a internet. En la pantalla principal, mostrada en el punto 1 de esta guía, accedemos a *Download documentation*. Esto nos abrirá el visualizador de ayuda de Windows, en este caso para el software Atmel Studio. En la sección de filtrado ingresamos assembler, y la ayuda nos mostrara algo similar a lo que se observa en la imagen de la izquierda.



Hacemos clic en el link *Add*, lo que nos llevara a lo que se observa en la imagen de la derecha. Ahora debemos presionar el botón Update (en la esquina inferior derecha de la pantalla) para que se proceda con la descarga de la documentación seleccionada (y todo lo que hayamos seleccionado para descargar).

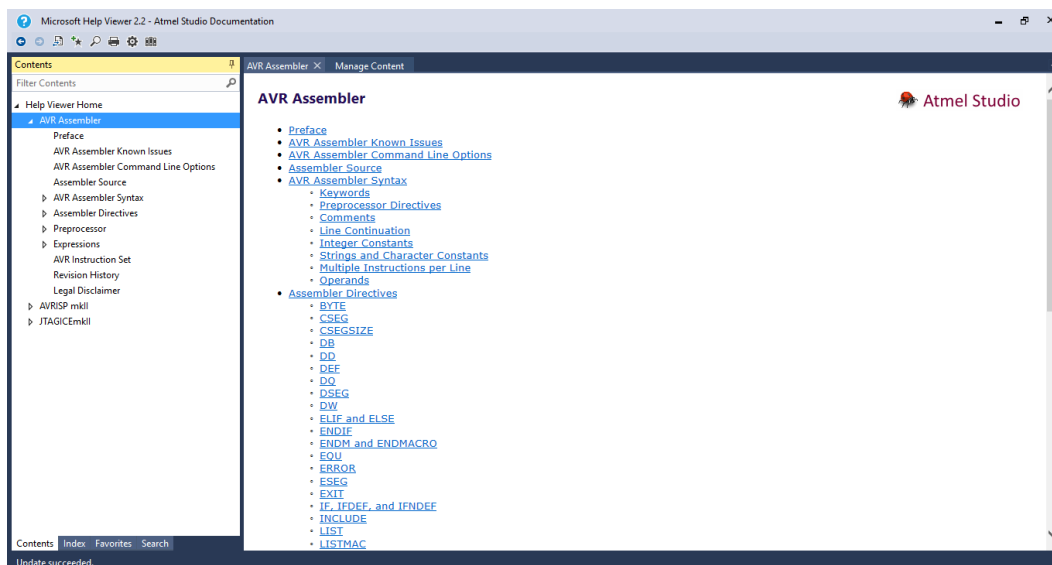
MICROCONTROLADORES ATMEL



Luego de aceptar las advertencias de Windows por las modificaciones y las licencias, se nos mostrara una ventana similar a la imagen de la izquierda.

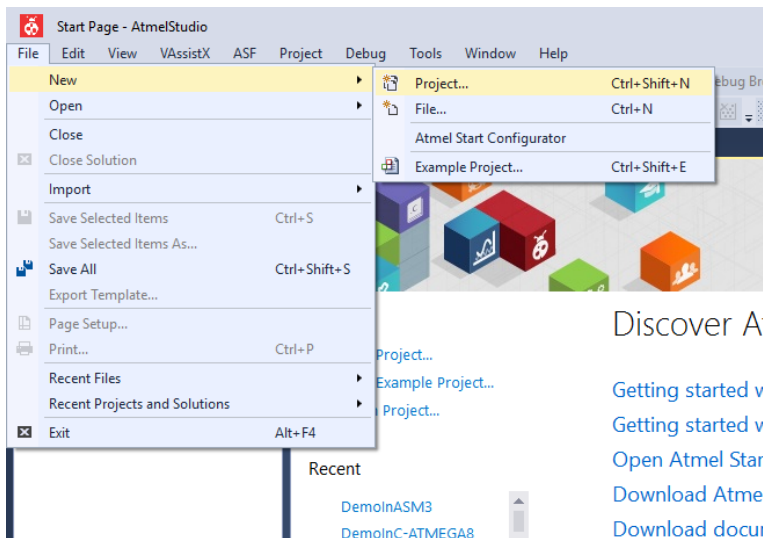
En un círculo rojo se observa las distintas documentaciones descargadas en mi caso. En la primera posición se encuentra la ayuda descargada recién denominada *AVR Assembler*.

Si abrimos esta ayuda nos mostrara:





❖ Anexo: Como crear un proyecto en C en Atmel Studio

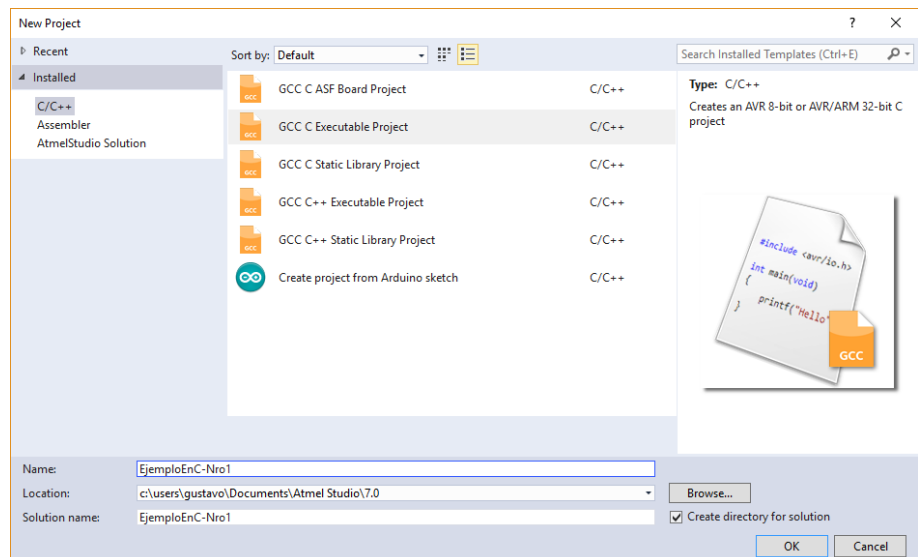


Para generar un nuevo proyecto en C en Atmel Studio accedemos a **File → New → Project...**

Esto nos abrirá una ventana donde podremos configurar el tipo de proyecto, el nombre y la ubicación del mismo.

En la imagen siguiente se observa la ventana de generación de un nuevo proyecto en Atmel Studio.

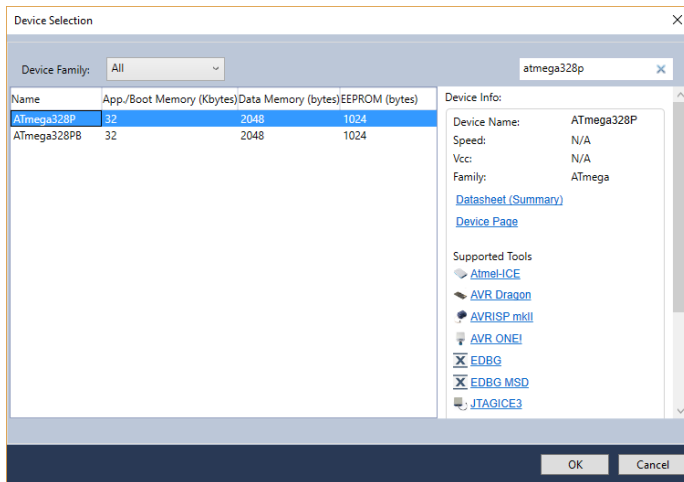
En este caso seleccionamos C/C++ y el tipo de proyecto seleccionamos "GCC C Executable Project". Como nombre le asignamos, en este caso, EjemploEnC-Nro1 (nuevamente observar que no se ingresa ningún tipo de extensión al proyecto). La ubicación del proyecto es el "por defecto" (no modificar) de este modo, todos los proyectos se encuentran en el mismo lugar, facilitando la búsqueda de los mismos. Nota: recomiendo dejar el tilde en "Create directory for solution".



Tarea: ¿En que caso cree que no hay que crear la Carpeta para la solución? Justifique su respuesta.

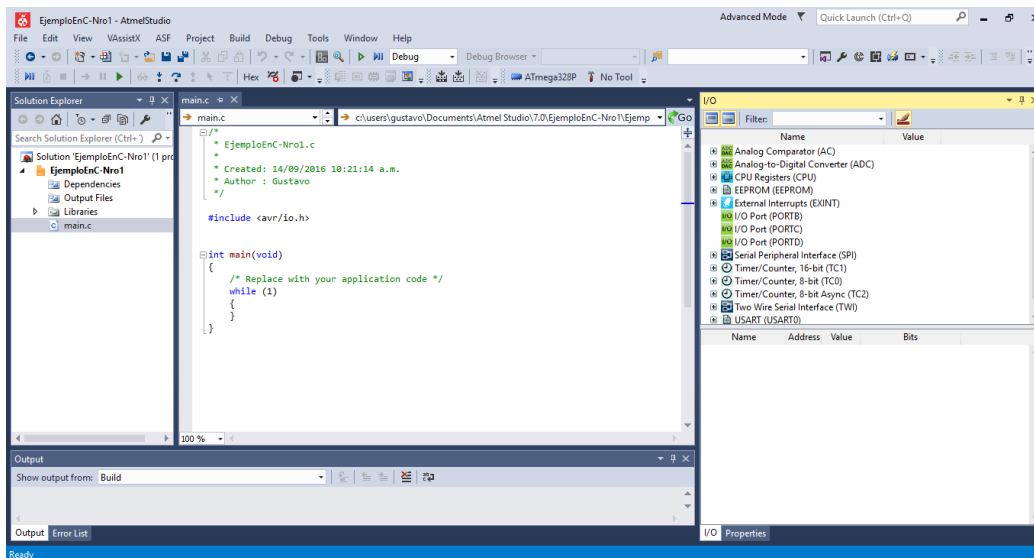


MICROCONTROLADORES ATMEL



Luego de presionar **OK** en la pantalla anterior, nos mostrara la pantalla de selección del microcontrolador del proyecto. En la parte de búsqueda ponemos **atmega328p** y nos busca los micros que tenga ese comienzo. Seleccionamos **ATmega328P** y damos **OK**.

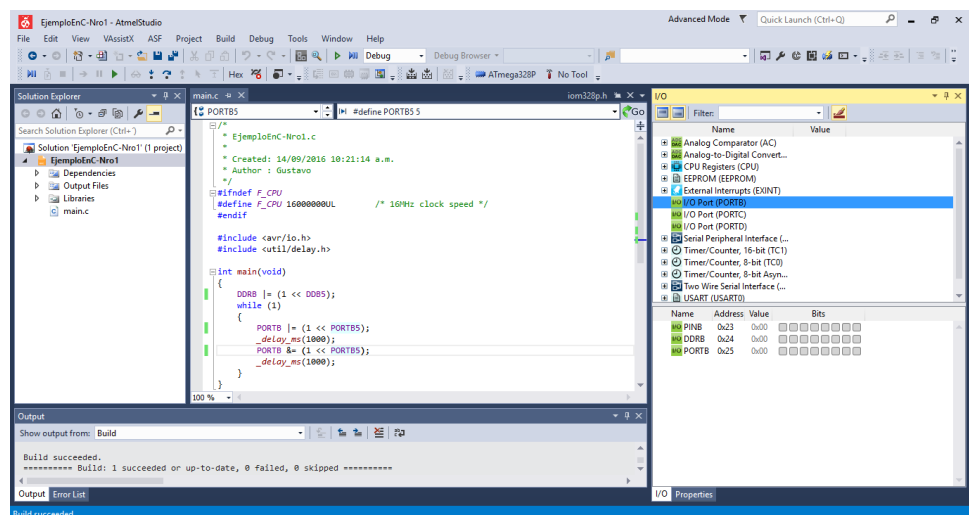
Nos mostrara una ventana del proceso de creación del proyecto, para luego finalizar y mostrarnos lo siguiente:



Observar la pantalla e identificar que es distinto a cuando generamos un proyecto en ensamblador. Compile el proyecto. Ahora agregaremos el código para hacer titilar un LED (similar al ejemplo del proyecto en ensamblador).

Observar el código del programa. Hay varios aspectos a tener en cuenta.

El programa en sí, lo que primero hace es configurar el puerto como salida y luego, en forma repetitiva, pone el pin en ALTO, hace una espera, pone el pin en BAJO, y por ultimo hace otra espera.





MICROCONTROLADORES ATMEL

Pero vamos a realizar un análisis más en detalle.

Por un lado tenemos la definición de F_CPU . Esta etiqueta es requerida al utilizar la biblioteca ***delay.h*** (notar donde se encuentra esta biblioteca), la cual hay que agregar al código para poder hacer uso de la función de retardo ***_delay_ms(<valor>)***.

Observar que la función *main* no tiene argumentos, y retorna un entero.

Dentro de la función principal, tenemos una parte que se ejecutará una sola vez, que es la parte que configura el puerto como salida.

El fragmento de código que se repetirá indefinidamente, que se logra por medio del ***while*** (1), es la secuencia de activar el pin, esperar, borrar el pin y esperar.