

MATLAB Y LA ELECTRÓNICA

Parte I Conceptos Básicos

El presente material pretende presentar mediante ejemplos el alcance de la herramienta computacional matlab de gran difusión en el mundo de la ingeniería a nivel académico

El documento total se dividirá en etapas siendo esta la más básica donde se revisaran algunos conceptos básicos

MATLAB es la abreviatura de *Matrix Laboratory* (laboratorio de matrices). Es un programa de análisis numérico creado por *The MathWorks* en 1984. Está disponible para las plataformas Unix Windows y Mac OS X.

Se pueden ampliar sus capacidades con *Toolboxes*, algunas de ellas están destinadas al procesamiento digital de señal, adquisición de datos, economía, inteligencia artificial, lógica difusa... También cuenta con otras herramientas como Simulink, que sirve para simular sistemas.

La primera versión surgió con la idea de emplear unos paquetes de subrutinas escritas en Fortran en los cursos de álgebra lineal y análisis numérico, sin necesidad de escribir programas en Fortran.

Usa un lenguaje de programación creado en 1970 para proporcionar un sencillo acceso al software de matrices *LINPACK* y *EISPACK* sin tener que usar Fortran. También tiene su propio compilador.

Es un software muy usado en universidades, centros de investigación y por ingenieros. En los últimos años ha incluido muchas más capacidades, como la de programar directamente procesadores digitales de señal, crear código VHDL y otras.

MATLAB es un programa de cálculo numérico, orientado a matrices y vectores. Por tanto desde el principio hay que pensar que todo lo que se pretenda hacer con el, será mucho más rápido y efectivo si se piensa en términos de matrices y vectores

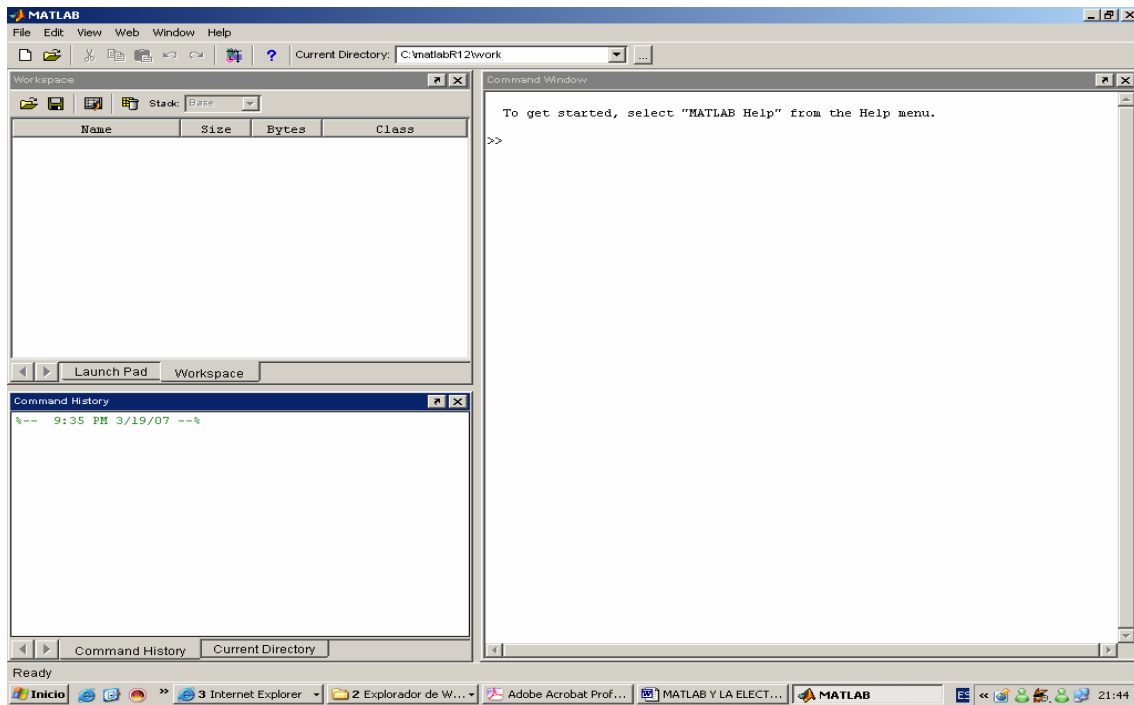
Para el presente documento se usará la versión 6 Release 12 pues a partir de ella al ser incorporadas nuevas funciones al paquete los requerimientos respecto a memoria se elevaron exponencialmente

Para usar este release se recomienda como mínimo :

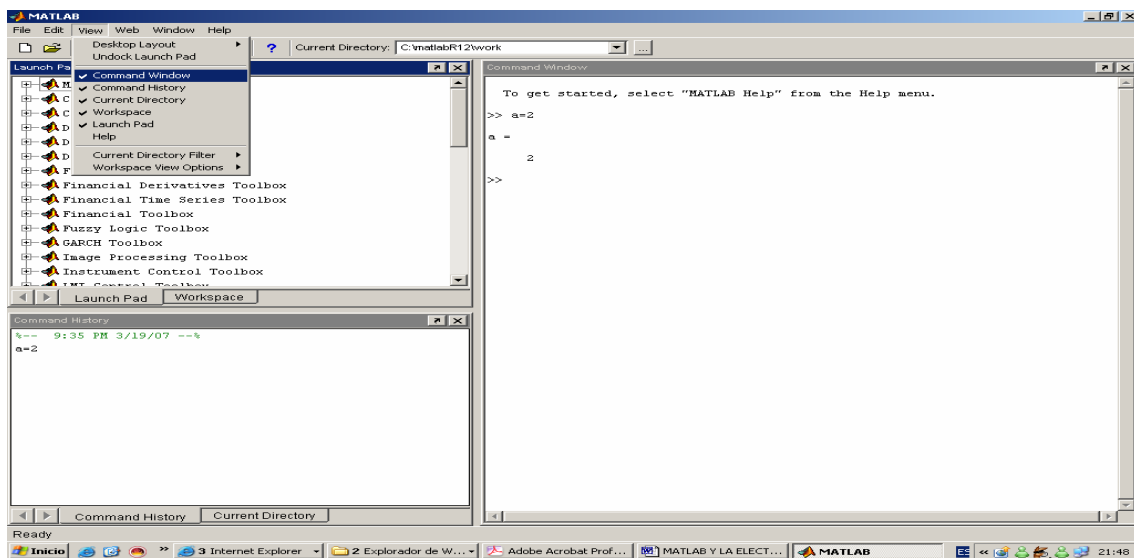
un pc con mmx 233 ,32 Mb de ram y un disco superior a 4 Gb, si desea usar las funciones implementadas en los paquetes (toolboxes) como la adquisición de datos será necesario anexar webcam (o cámara de video), micrófono y parlantes

Este release igual que los posteriores incorporan una serie de ayudas las cuales es posible invocar desde el prompt (>>) de matlab las cuales debieron ser instaladas previamente desde el cd 2 suministrado por la compañía Mathworks

Para hincar revisemos el entorno de trabajo de matlab



en la grafica vemos tres paneles (dos a la izquierda y uno en la derecha) revisémoslos cuidadosamente, estos paneles fueron señalados previamente en el menú view



En la parte superior izquierda vemos dos pestañas llamadas Launchpad y Workspace

En el launchpad observamos los contenidos de la instalación de Matlab (tanto el core la instalación básica como los paquetes (comúnmente llamados Toolboxes) seleccionados por el usuario durante la instalación

En el workspace (espacio de trabajo) aparecen tanto el nombre como las características de las diversas variables introducidas o usadas en el command Window (ventana de comandos)

En la parte inferior vemos otras dos pestañas llamadas command History (historial de comandos) y Current directory (directorio de trabajo)

Command History (historial de comandos) es una bitácora con los comandos introducidos en el command window (ventana de comandos) , si desea repetir una operación tan solo haga doble clic en la operación deseada en este espacio y voila

Ahora pasemos a la ventana derecha (la más grande) denominada command window (ventana de comandos) en ella se muestra el prompt >> (el cual esta predeterminado por defecto; aunque es posible cambiarlo).

En este espacio se escribirán los comandos y se observara el resultado de las diversas operaciones o los errores en caso de ocurrir estos. Si a una determinada operación no se le asigna una variable por defecto Matlab le asignara el valor de ans
Ej

Al escribir en el prompt la siguiente operación

```
>>34.356+345
```

obtendremos

```
ans =
```

```
379.3560
```

de otro lado si reemplazamos la sintaxis por

```
>>a=34.356+345
```

obtendremos

```
a =
```

```
379.3560
```

resultado almacenado en la variable a, si no desea la presentación de la respuesta en el prompt anéxele al comando introducido un punto y coma (;) al final de la sentencia lo

cual es recomendable al generar resultados iterativos pues reduce el tiempo de computo y el uso de memoria que realiza Matlab

realicemos un ejemplo deseamos determinar el valor de

$$V = 35 * e^{-4.5t} + 45 \text{ para } t > 0$$

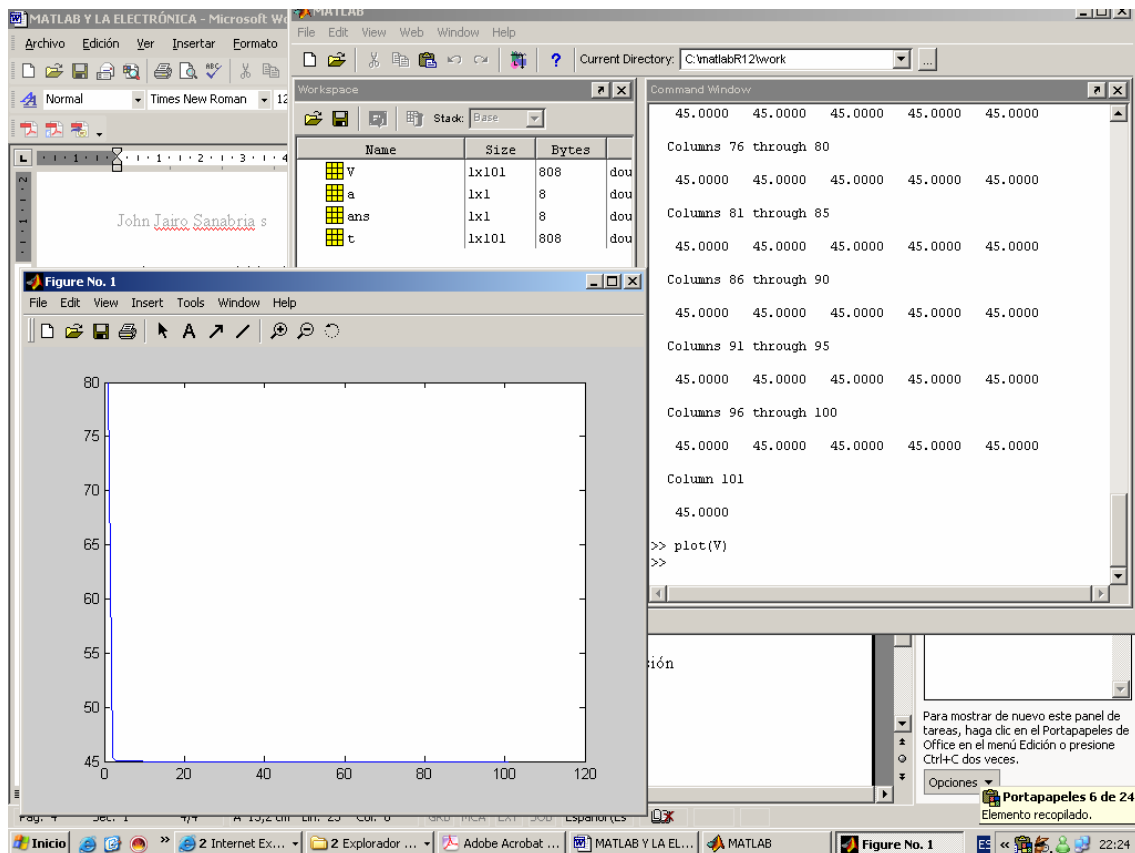
como la respuesta de un circuito electrónico de primer orden en el prompt de matlab escribimos entonces

```
>> t=0 : 1 : 100;
>> V=35*exp(-4.5*t)+45
```

como puede observar le hemos asignado a la variable t (tiempo) una serie de valores a partir de cero hasta 100 segundos moviendonos en un paso igual a un segundo (se le coloco punto y coma para no generar una lista de valores en la ventana de comandos en el calculo de los valores de la tensión V no se ha hecho lo mismo así que se observaran 101 valores para dicha tensión

Para poder emitir algún juicio de valor respecto a la variación de dicha tensión presentaremos los valores mediante una grafica mediante el comando

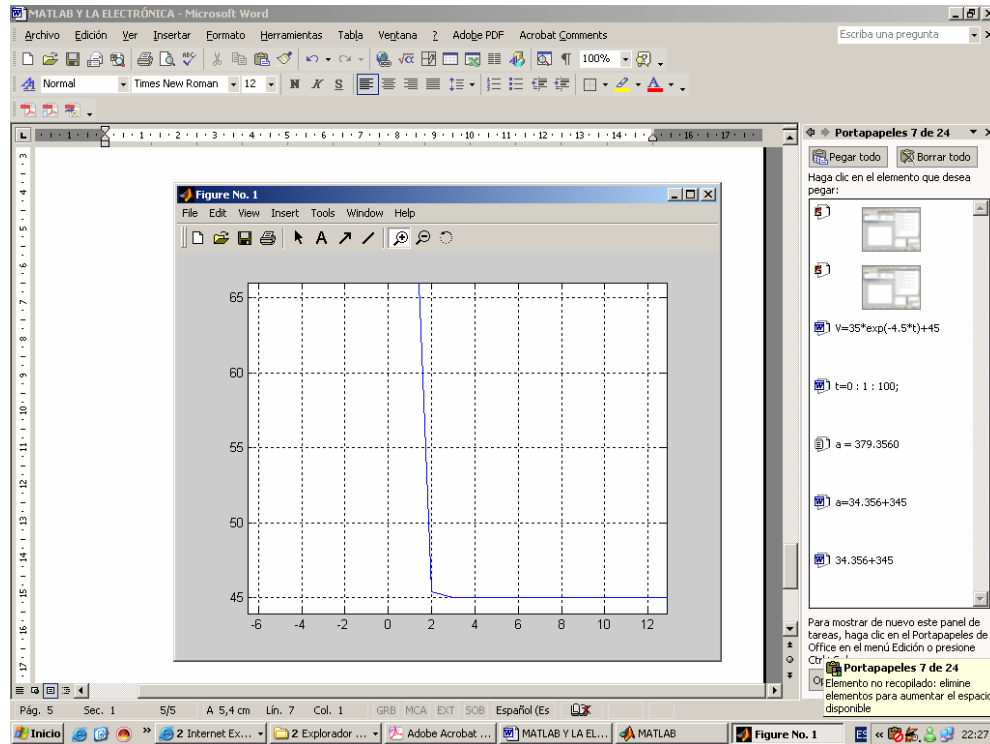
```
>> plot(V)
```



como en dicha grafica no se observan con claridad e comportamiento cerca de cero usaremos otros comandos con los cuales podremos observar mejor nuestra grafica

>>zoom

>>grid



revisemos algunos conceptos y comandos empleados en el documento

- si a una sentencia no se le asigna una variable automáticamente Matlab la asociara a **ans**
- Para evitar la presentación de resultado de un ciclo u otra operación el comando escrito en la ventana de comandos debe finalizarse con un punto y coma (;)
- Con el comando **x= - 0.15 : 0.5 : 1000** generaremos los valores entre -0.15 y 1000 con variaciones cada 0.5 y los almacenaremos en la variable x
- Mediante el comando **exp(c)** obtendremos el valor del Euler a la c
- Con **plot** obtendremos la grafica de una función continua para las variables discretas se usa el comando ítem
- Con **zoom** se nos permite seleccionar un area y realizar un acercamiento a ella
- Con el comando **grid** se puede implementar una rejilla sobre la grafica especificada

En el siguiente documento revisaremos algunos conceptos de vectores y matrices y los usaremos para resolver problemas relacionados con mallas presentes en circuitos eléctricos.