***1-Introducción***

Robots son aquellos sistemas electromecánicos conducidos por un cierto programa que les permite realizar una determinada tarea de forma autónoma.

***2-Programación de robots***

Para ser capaces de definir el comportamiento de un robot existen dos niveles de programación, gestual y textual:

-Gestual: En la programación gestual el robot interviene en el proceso como un digitalizador de posiciones por las que luego pasará a ejecutar la tarea. No es necesaria experiencia por parte del usuario, ya que la definición de su comportamiento se realiza con métodos sencillos:

\*Moviendo manualmente el robot a la posición deseada.

\*Mediante un sistema maestro-esclavo por lo que lo movimientos realizados por el operador en el maestro son seguidos por el esclavo.

\*Por medio de un mando que permita definir y registrar posiciones, es la opción más usada.

-Textual: La secuencia de posiciones del robot es definida mediante un programa que define el movimiento del robot y su relación con el entorno (ya que puede obtener información de él). En este caso si es necesaria experiencia en técnicas de programación y realización de estrategias de movimiento basadas en información sensorial. Dentro de la programación textual se pueden distinguir tres tipos:

\*Nivel robot: orientados a las operaciones y secuencia de ellas que el robot debe realizar para llevar a cabo la tarea (lectura de sensores y movimiento de actuadores).

\*Nivel objeto: las instrucciones se dan en función de los objetos a manejar, las instrucciones a nivel robot se conseguirán consultando una base de datos.

\*Nivel tarea: el programador debe establecer cuáles son las acciones que debe ejecutar el robot, pero no tiene necesariamente que detallar cómo hacerlo. Es el sistema el que decide qué movimientos y comprobaciones sensoriales debe realizar, y en qué orden. Las decisiones se toman en función de:

-> Los objetivos propuestos.

-> El estado en cada momento del mundo del robot.

***3-Relacion de los lenguajes de programación de robots y los habituales***

No es posible agrupar todos los lenguajes de programación de robots dentro de una lista de características generales, debido a su gran variedad, en algunos casos el lenguaje es simplemente un lenguaje habitual de programación, como c++, java, Python Matlab…, al que se añaden funciones específicas de programación de robots mediante librerías y en otros es un lenguaje desarrollado por la empresa fabricante de robots exclusivamente para sus unidades.

Por ejemplo :

-Matlab cuenta con la Robotics Toolbox que proporciona funciones para el estudio y simulación de los movimientos de las articulaciones de un robot, asi como generaciones de trayectorias y cinemáticas.

-Microsoft Robotics Developer Studio que es una herramienta para la creación de aplicaciones robóticas. Posee un soporte de tiempo real orientado a servicios y está desarrollado sobre el entorno .NET y soporta lenguajes de programación como VB.NET, Python , Visual Basic, VPL o C#.

***4-leguajes en empresas***

La mayor parte de las empresas tienen su propio lenguaje de programación de robots teniendo en cuenta solo sus necesidades en cuanto a software y variedad de protocolos de comunicación con los dispositivos hardware.

Como ejemplo tendríamos:

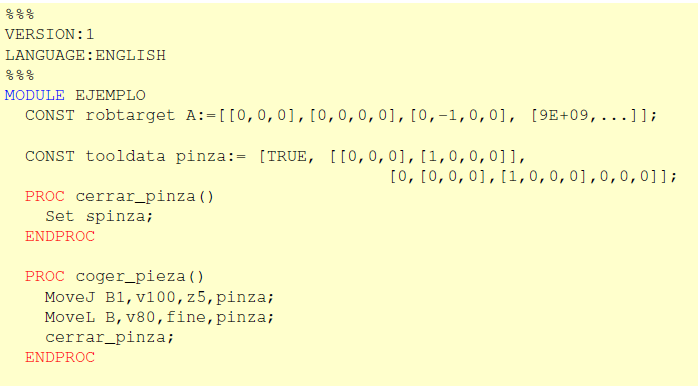
-ABB con el lenguaje RAPID

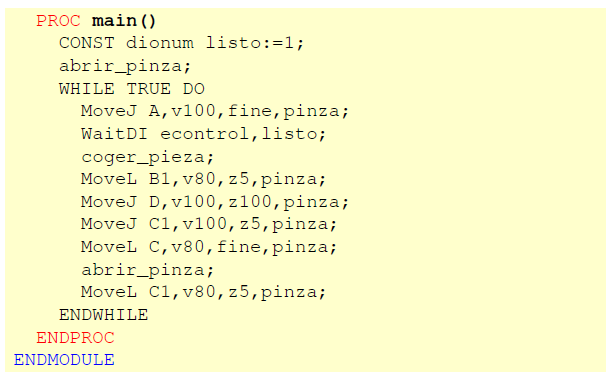
RAPID es un leguaje de programación textual de alto nivel desarrollado por la empresa ABB. Una aplicación RAPID consta de un programa y una serie de módulos del sistema. El programa es una secuencia de instrucciones que controlan el robot y en general consta de tres partes:

-Una rutina principal (main): Rutina donde se inicia la ejecución.

-Datos de programa : Definen posiciones, valores numéricos, sistemas de coordenadas, etc.

-Subrutinas : Sirven para dividir el programa en partes más pequeñas a fin de obtener un programa modular.





-Kuka con KRL

-Comau con PDL2

-Yaskawa con INFORM

# -Fanuc con Karel

# KAREL es un lenguaje de bajo nivel muy similar a Pascal. Cuenta con variables fuertemente tipadas, constantes, tipos personalizados, procedimientos, funciones, etc.KAREL es un lenguaje compilado,el fichero.KL debe traducirse a un fichero.PC antes de cargarse y ejecutarse en el controlador.

# 

# 

# 

***5-lenguajes en educación***

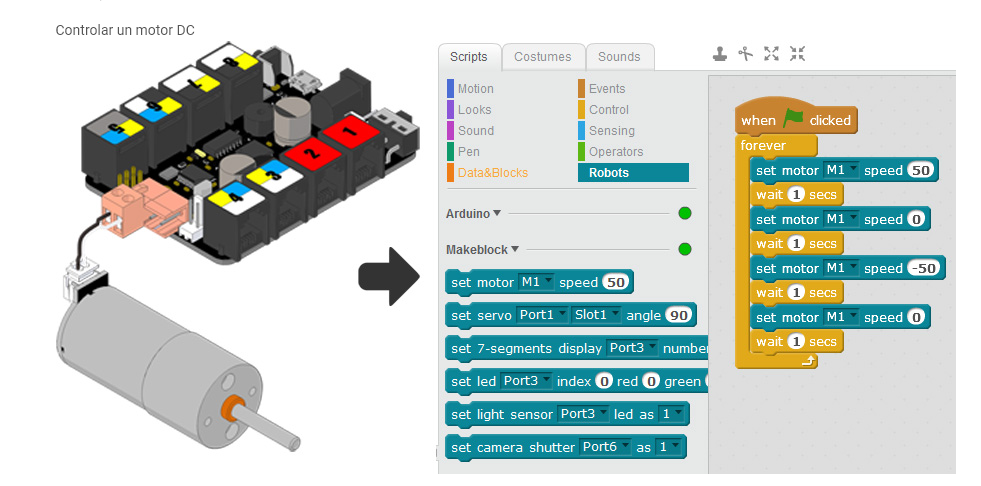
Actualmente los robots se han convertido en una forma educativa de interesar a la juventud en el mundo de la programación, la inteligencia artificial y la robótica.

Como sucedía en el mundo de las empresas, también hay compañías con su lenguaje propietario, pero la mayoría soportan lenguajes típicos en programación, además suelen soportar Scratch o poseer una interfaz similar a este último, pero compilando en otro lenguaje.

Algunos lenguajes en este ámbito (no normalmente utilizados):

- Scratch

-Derivado de Scratch tenemos Makeblock ha desarrollado mBlock, un software basado en Scratch especialmente diseñado para niños.Con él aprenderán a controlar su robot mediante sencilla programación gráfica de arrastrar y soltar bloques. Makeblock permite usar Scratch con placas Arduino ya que traduce los bloques de Scratch a código fuente de Arduino.



-**Scracth también es utilizado junto** LEGO Education WeDo, ques es un material de robótica corientado a los más jóvenes.

***6-conclusión de los 2 anteriores***

***7-Bibliografía***

<https://es.wikipedia.org/wiki/Robot>

<https://analisisyprogramacionoop.blogspot.com.es/2014/10/programacion-de-robots.html>

<http://platea.pntic.mec.es/vgonzale/cyr_0708/archivos/_15/Tema_5.6.htm>

<https://blog.robotiq.com/what-is-the-best-programming-language-for-robotics>

<https://en.wikipedia.org/wiki/Educational_robotics>

https://blog.robotiq.com/what-is-the-best-programming-language-for-robotics

<http://petercorke.com/wordpress/toolboxes/robotics-toolbox>

https://www.makeblock.es/soporte/mblock/

<http://codigo21.educacion.navarra.es/autoaprendizaje/primeros-pasos-con-scratch-y-lego-wedo/>

http://www.onerobotics.com/posts/2013/introduction-to-karel-programming/