

MEMORIA DE PRÁCTICAS  
Nº 24-A-225655

**Memoria de las Prácticas Curriculares en Zennio  
bajo la Escuela de Ingeniería Industrial y  
Aeroespacial**

David Morán Soriano

SEPTIEMBRE DE 2024

*Título:* Memoria de las Prácticas Curriculares en Zennio bajo la Escuela de Ingeniería Industrial y Aeroespacial

*Autor:* David Morán Soriano

*Grado:* Grado en Ingeniería Eléctrica

*Tutor Académico:* Jesus Rosado Linares

*Tutor Empresa:* Francisco

*E-mail:* david.moran@alu.uclm.es

*Dirección:* UCLM — Escuela de Ingeniería Industrial y Aeroespacial  
Campus Universitario de la Real Fábrica de Armas  
45071 Toledo – Spain

© 2024 David Morán Soriano

Esta obra está bajo una [Licencia Creative Commons Atribución 4.0 Internacional](#).

Muchos de los nombres usados por las compañías para diferenciar sus productos y servicios son reclamados como marcas registradas. Allí donde estos nombres aparezcan en este documento, y cuando el autor haya sido informado de esas marcas registradas, los nombres estarán escritos en mayúsculas o como nombres propios.

# Agradecimientos

Pon aquí tus agradecimientos, pero no te olvides de que se trata de un documento profesional.



Edita esta sección en el archivo `agradecimientos.tex`.

*Aquí va la dedicatoria.*



*Edita esta sección en el archivo `dedicatoria.tex`.*

# Resumen



Edita esta sección en el archivo `resumen.tex`.

Aquí va el resumen en español. Debería ocupar entre 1000 y 1500 palabras a modo orientativo.



Escribe el resumen en estilo periodístico, desde lo más general a lo más específico. Desde lo más importante a lo menos importante. Esto será especialmente útil si es necesario hacer un resumen más breve en otro contexto (presentación, herramienta, etc). Si el resumen está en formato periodístico bastará con seleccionar los primeros párrafos.

# Abstract

Write down the abstract in English here. It must be between 1000 and 1500 words long.

# Índice general

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Organización de la memoria . . . . .	1
1.2	Repositorio de información . . . . .	2
<b>2</b>	<b>Objetivos</b>	<b>3</b>
<b>3</b>	<b>Motivación y antecedentes</b>	<b>4</b>
<b>4</b>	<b>Desarrollo del TFG</b>	<b>5</b>
<b>5</b>	<b>Resultados y discusión</b>	<b>6</b>
<b>6</b>	<b>Conclusiones</b>	<b>7</b>
<b>A</b>	<b>Chuleta de <math>\text{\LaTeX}</math></b>	<b>8</b>
A.1	Texto . . . . .	8
A.1.1	Listas numeradas y con viñetas . . . . .	9
A.1.2	Acrónimos . . . . .	10
A.2	Figuras . . . . .	10
A.3	Tablas . . . . .	13
A.4	Ecuaciones . . . . .	15
A.5	Bibliografía, citas y referencias . . . . .	16
A.5.1	Lo que toda referencia debe tener . . . . .	17
A.6	Hojas de datos . . . . .	17
A.7	Código fuente . . . . .	21
A.8	Compilación local . . . . .	23
A.8.1	Configurar el entorno . . . . .	23
A.8.2	Configurar TeXStudio . . . . .	24
	<b>Bibliografía</b>	<b>26</b>

# Índice de cuadros

A.1	Resultados de la simulación . . . . .	14
A.2	Tabla muy sencilla. . . . .	14
A.3	Fusionando celdas. . . . .	14
A.4	Un ejemplo de tabla larga . . . . .	14



# Índice de figuras

A.1	Figurita ejemplo . . . . .	10
A.2	Figurita ejemplo 2 . . . . .	11
A.3	Figurita ejemplo 3. Extraída de la plantilla de TFG de Fernando Castillo. ©2018 Fernando Castillo. Reproducida con permiso. . . . .	11
A.4	Figura ejemplo que ocupa todo el ancho del texto. . . . .	12
A.5	Figura ejemplo de rotación. . . . .	13
A.6	Matriz de figuras . . . . .	13
a	Figurita 1 . . . . .	13
b	Figurita 2 . . . . .	13
c	Figurita 3 . . . . .	13
A.7	Figura ejemplo. Tomada de hoja de catálogo de <a href="#">motores DC con escobillas de grafito de Maxon</a> © 2014 Maxon Motors. Reproducida con permiso. . . .	20
A.8	Configuración de la compilación en TeXStudio . . . . .	24
A.9	Configuración de la orden <i>Latexmk</i> en TeXStudio . . . . .	25

# Lista de símbolos



Edita esta sección en el archivo `simbolos.tex`.

## Mayúsculas

- $T$  — Temperatura ( $^{\circ}\text{C}$ ).  
 $V$  — Volumen ( $\text{m}^3$ ).

## Minúsculas

- $c$  — Velocidad de la luz en el vacío ( $\text{m/s}$ ). La velocidad de la radiación electromagnética es independiente de la velocidad del emisor.  
 $i$  — Raíz de menos uno (-).

## Letras griegas

- $\alpha$  — El principio de todo (-).  
 $\pi$  — Pastel en inglés (-).

# Lista de acrónimos

**CD**

Compact Disc. [10](#)

# Capítulo 1

## Introducción



Edita esta sección en el archivo `introduccion.tex`.

En este capítulo debes introducir el problema sin divagar, sin copiar de otros documentos y sin utilizar un lenguaje excesivamente técnico. Tampoco utilices un lenguaje informal. Este capítulo debería convencer al cliente de que el proyecto merece la pena. Es decir, es un problema real y no está resuelto completamente.

Debe tenerse siempre presente que el cliente es el que paga. En el TFG el que paga es el tribunal, en forma de calificación. Así que a quien hay que convencer es a los miembros del tribunal. El tribunal no lo conocerás a priori. Por eso la memoria debe estar escrita para que la entienda alguien que no es especialista en el campo de aplicación. Pero eso no implica que se toleren la falta de rigor o la falta de argumentación técnica. Solo implica que los argumentos específicos hay que explicarlos o citar la fuente que los explica.

Redacta la introducción al final del TFG, cuando tengas elaborado el capítulo de antecedentes, los resultados y su discusión. De esta forma podrás evitar repetir argumentos que ya están en esos capítulos. La introducción debe introducir también el contexto en el que se desarrolla el TFG. Divide el documento en secciones y subsecciones para organizar el contenido del capítulo. Utiliza preferentemente frases cortas.

### 1.1 Organización de la memoria

La organización de este documento responde a un documento científico-técnico. Se descompone en los siguientes capítulos.

#### Capítulo 2

Enumera y justifica los objetivos del proyecto y establece los límites intrínsecos y extrínsecos de ejecución del TFG.

#### Capítulo 3

Analiza los antecedentes y estado del arte en relación al tema del proyecto.

#### Capítulo 4

Describe todo el proceso de desarrollo del TFG. Esto incluye la metodología de trabajo empleada y las diferentes etapas o iteraciones que se han llevado a cabo. No dudes en descomponer el capítulo en varios si aglutina demasiado material.

**Capítulo 5**

Describe en detalle los resultados obtenidos y las pruebas realizadas. Discute los resultados en relación a los objetivos del proyecto.

**Capítulo 6**

Recopila las principales conclusiones del proyecto y comenta las líneas de trabajo futuro, en caso de que se contemplen.

**Anexos**

Complementan la información del cuerpo del documento con información técnica útil para reproducir los resultados, pero innecesaria para comprender en su totalidad el TFG realizado.

**Bibliografía**

Recopila las referencias bibliográficas utilizadas en este documento.



La normativa de la EIIA no te obliga a usar una estructura fija. Adáptala como quieras a tu TFG. Puedes reordenar capítulos, mezclarlos, dividirlos, etc. Por ejemplo, es frecuente poner unos antecedentes antes de los objetivos, que definen el contexto del TFG, y desarrollar el estado de la cuestión después de los objetivos.



Al finalizar el resto de los capítulos revisa esta descripción del documento para que coincida con lo que realmente contiene la memoria. Por ejemplo, es frecuente fusionar varios capítulos en uno cuando son muy pequeños. También es frecuente lo contrario, dividir un capítulo en varios cuando es muy extenso.

## 1.2 Repositorio de información



Es muy útil tanto para la ejecución como para la evaluación disponer de un repositorio para almacenar las sucesivas versiones del documento y de todo el material generado durante el proyecto. La UCLM incorpora [GitHub](#) como servicio institucional. Si no utilizas un repositorio quita esta sección.

Todo el material generado durante la ejecución de este proyecto está disponible en el repositorio . El material incluye el código  $\LaTeX$  del presente documento, el código fuente de los programas realizados o modificados, y todos los datos generados en la evaluación de resultados.

## Capítulo 2

# Objetivos



Edita esta sección en el archivo `objetivos.tex`.

Primero enumera los objetivos, no los resumas ni los redactes en un párrafo. Cada uno de los objetivos de un proyecto debe ser SMART:

### **Simple**

Cada objetivo tiene que ser independiente, tener sentido por sí mismo y más o menos indivisible. Si no es suficientemente indivisible, pero tiene sentido como una entidad independiente, debes descomponerlo en subobjetivos.

### **Medible**

Tiene que ser posible medir el grado de consecución al final del TFG.

### **Acordado**

Los objetivos no los pones tú solo. Deben partir de un acuerdo con tu director.

### **Realista**

No pongas objetivos muy ambiciosos. Basta con que resuelva el problema de la forma más simple posible. Si superas los objetivos nadie se va a quejar. El director se encargará de que tampoco sean demasiado poco ambiciosos.

### **Temporalizado**

Un objetivo debe tener un marco temporal. Si no es así el objetivo podría no cumplirse nunca. Es difícil poner límites temporales muy estrictos en un primer proyecto de ingeniería, pero al menos acota.

Tras cada objetivo puedes añadir párrafos ampliando la descripción del objetivo, describiendo los límites y justificándolos. También puedes describir de qué se parte. Si es posible debería quedar plenamente justificado que se trata de objetivos SMART. Considera tanto límites intrínsecos (inherentes a la definición del proyecto) como extrínsecos (limitaciones presupuestarias, equipamiento disponible, etc).

## Capítulo 3

# Motivación y antecedentes



Edita esta sección en el archivo `antecedentes.tex`.

El problema que pretendes resolver está dentro de un contexto que el cliente debe conocer. Esta sección aporta información para conocer en detalle la importancia del problema y la dificultad para resolverlo con los productos y programas disponibles actualmente.

Este capítulo concentrará el grueso de las citas del TFG. Dado que se trata del primer trabajo profesional, el alumno no suele estar familiarizado con las citas bibliográficas. Pon toda tu atención en qué citas y cómo lo citas. Revisa la sección [A.5](#) para las reglas mínimas que deben cumplir las citas.



Es muy importante respetar la regla de atribuir correctamente. No es aceptable desde el punto de vista legal, ni tampoco desde el punto de vista ético, copiar trabajo de otros sin atribuirlo correctamente a los autores.

Esta sección debe estudiar de forma sistemática todas las opciones ya disponibles en la actualidad para resolver el problema. No basta con una mera enumeración, hay que estudiarlos mínimamente para explicar por qué no son una solución para el problema o qué podría aportar a la solución del problema.

Un método sistemático para realizar esta parte del TFG es la revisión sistemática de literatura, conocida habitualmente por sus siglas en inglés SLR (*Systematic Literature Review*). Un resumen muy sencillo de cómo realizar una SLR puede encontrarse en [\[Kof14\]](#). También encontrarás consejos prácticos en [\[Sch17\]](#). Para un proceso más detallado, especialmente si tu problema tiene mucho arte previo, puedes consultar [\[KC07\]](#).

En una tesis doctoral el análisis sistemático del estado del arte es esencial. En un TFG es importante, pero no hay que perder la cabeza. Un TFG son unas 300 horas de trabajo de un estudiante medio que ya posea los conocimientos generales necesarios (volveremos a esto más tarde). Considero que un buen análisis del estado del arte corresponde a un trabajo de entre 25 horas y 100 horas, dependiendo del tema del proyecto. Si el tema es muy específico es más fácil hacer el estudio del estado del arte.



Termina este capítulo con una sección que resuma el estado del arte e identifique las lagunas lo más claramente posible. Una tabla comparativa o un gráfico pueden ser formas interesantes de presentar la información.

## Capítulo 4

# Desarrollo del TFG



Edita esta sección en el archivo `desarrollo.tex`.

En general este capítulo debe describir cómo se ha llegado a la solución del problema incluyendo metodología, planificación y ejecución.

En primer lugar, este capítulo debe tratar con la metodología de trabajo empleada para elaborar el TFG. La metodología puede variar sustancialmente dependiendo del contexto o el área temática en la que se encuadre. Si es necesario, divide la descripción de la metodología como capítulo independiente.

A continuación, debe describir cómo se ha dividido el problema en una secuencia de fases, tareas o iteraciones y qué cantidad de recursos (tiempo fundamentalmente) se han planificado para cada una.

Por último, debe incluir información sobre la ejecución del proyecto. Cuáles han sido los problemas encontrados y qué desviaciones se han producido en la ejecución. Dependiendo de la naturaleza del proyecto es posible organizar esta información de manera conjunta con la planificación.

	Hola	
	Caracola	



## Capítulo 5

# Resultados y discusión



Edita esta sección en el archivo `resultados.tex`.

Escribe en este capítulo los resultados del proyecto. Este capítulo debería explicar los resultados de forma global, no los resultados de cada fase o iteración. Probablemente será el capítulo con más tablas y gráficas. Revisa las secciones [A.2](#) y [A.3](#) para aprender cómo se escriben en  $\text{\LaTeX}$ .

Tus contribuciones no tienen por qué limitarse al trabajo sistemático del TFG. Puede que hayas contribuido en aspectos metodológicos, en ideas novedosas, en la planificación de experimentos, en desarrollos matemáticos. Este capítulo está para agrupar todo eso. Describe con claridad todo lo que ha supuesto contribuciones originales por tu parte.



Es importante destacar que un resultado negativo es también un resultado. Es posible que el proyecto planteara abordar un problema con un método que ha demostrado ser no apto. Si el trabajo ha sido sistemático sigue teniendo mucho valor, puesto que excluye el método para cualquier otro trabajo futuro. Escribe este tipo de resultados con especial cuidado para destacar que el trabajo se ha realizado de manera sistemática.

## Capítulo 6

# Conclusiones



Edita esta sección en el archivo `conclusiones.tex`.

Las conclusiones deben cerrar el documento, destacando los aspectos más importantes de la ejecución del TFG. Debe analizar qué objetivos se han alcanzado y en qué grado, qué objetivos se han tenido que dejar fuera del proyecto y por qué, y qué líneas de trabajo futuro abre el TFG.

Fíjate en que los objetivos abren el trabajo personal y las conclusiones lo cierran. Procura mantener un orden que resalte esta relación, pero no te limites a parafrasear los objetivos.

## Apéndice A

# Chuleta de L<sup>A</sup>T<sub>E</sub>X

L<sup>A</sup>T<sub>E</sub>X es un formato textual para construir documentos complejos, cualquier tipo de documento complejo. No es difícil, y de hecho está pensado para que personas sin formación en tipografía y composición puedan redactar documentos estéticamente impecables. Pero necesita algo de tiempo para acostumbrarse a la filosofía.

Puedes acabar tu TFG simplemente imitando la redacción de este documento, pero seguro que puedes sacar más partido si aprendes algo de L<sup>A</sup>T<sub>E</sub>X. Hay muchas guías sencillas para empezar. Por ejemplo, [Ara13] o [Sim02] son resúmenes suficientemente completos y sencillos. En caso de que quieras ampliar tus conocimientos tendrás que empezar por el libro de Leslie Lamport [Lam94], el creador de L<sup>A</sup>T<sub>E</sub>X. Procura complementarlo con [MGB+04] y, especialmente, navegar por el CTAN (*Comprehensive T<sub>E</sub>X Archive Network*). CTAN es un repositorio enorme de paquetes, que facilitan la edición de todo tipo de detalles. Desde las cabeceras de página, hasta la creación de enlaces hipertexto o el manejo de colores.

En este anexo se incluye un pequeño recetario, extraído de la plantilla de Fernando Castillo, que puedes usar a modo de recordatorio rápido, pero procura utilizar una referencia más completa cuando empieces a editar el documento.



Este recetario no es un libro de L<sup>A</sup>T<sub>E</sub>X, ni es para sentarse a leerlo, es para usarlo cuando lo necesites. No te mostramos cómo se escribe en L<sup>A</sup>T<sub>E</sub>X cada ejemplo, sino que directamente lo hacemos. Cuando necesites hacer algo semejante, mira el código de este documento en [https://github.com/UCLM-eiia-to/eiia\\_doc\\_tfg\\_latex](https://github.com/UCLM-eiia-to/eiia_doc_tfg_latex)

### A.1 Texto

En esta sección se muestran ejemplos de efectos de texto.

#### **textbf**

**Texto en negrita.**

#### **textit**

*Texto en cursiva.*

#### **underline**

Texto subrayado.

**large**

Texto grande

**Large**

Texto más grande

**LARGE**

Texto mucho más grande

**textsc**

TEXTO EN VERSALITAS

**textsf**

Texto en fuente sans-serif

**texttt**

Texto en fuente mono-espaciada

A veces resulta útil este tipo de texto para funciones de programación o código fuente. Por ejemplo:

```
[x,y]=function(t,z)
```

**A.1.1 Listas numeradas y con viñetas**

Listas numeradas. Se utilizan cuando la posición que ocupan los elementos es importante, para contar los elementos, o cuando se necesita añadir referencias a algunos de los elementos.

1. diodos
2. transistores
  - (a) transistores pnp
  - (b) transistores npn
  - (c) operacionales
3. operacionales

Listas con viñetas. Se utilizan cuando la posición concreta de los elementos no es importante:

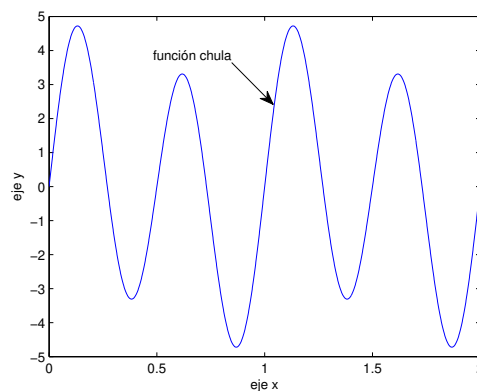
- diodos
- transistores
  - transistores pnp
  - transistores npn
  - operacionales
- operacionales

### A.1.2 Acrónimos

Si quieres lista de acrónimos debes marcar los acrónimos con una etiqueta especial. Por ejemplo, así se pondría la primera vez un **Compact Disc (CD)**. Una vez que se ha usado el acrónimo, ya se puede usar en forma abreviada, como en **CD**. Marca incluso en este caso los acrónimos, porque así el PDF permitirá navegar a la definición pinchando sobre él.

## A.2 Figuras

En esta sección se muestran algunos ejemplos de figuras. La fig. **A.1** es un ejemplo de *Encapsulated PostScript*. Se trata de un formato vectorial, que no se degrada al escalarlo. Este tipo de formatos son los ideales para usar con  $\text{\LaTeX}$ . Utiliza siempre que puedas imágenes en formato EPS o PDF.



**Figura A.1:** Figurita ejemplo

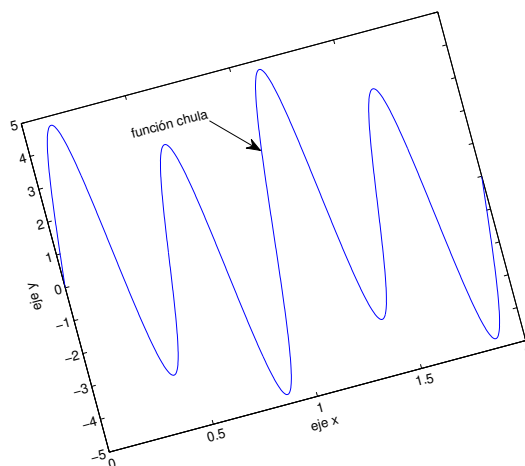
La forma de incluir la figura es simple:

```
\begin{figure}
\centering\includegraphics[width=6cm]{ejemplo.eps}
\caption{Figurita ejemplo}
\label{fig:figurita-ejemplo}
\end{figure}
```

El entorno `figure` crea un cuadro flotante, con todo el contenido de la figura, que  $\text{\LaTeX}$  coloca en el sitio menos malo. La `caption` es el pie de la figura y la `label` es la etiqueta que nos permitirá referirnos a ella en el texto (con la orden `ref`).

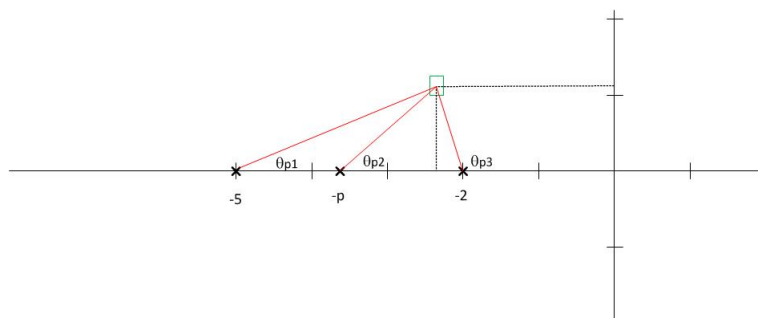
$\text{\LaTeX}$  utiliza un algoritmo nada evidente para colocar las figuras de manera que sea estéticamente agradable. Pero tú puedes influir en las preferencias de colocación. El entorno `figure` tiene un parámetro opcional entre corchetes que indica las opciones de colocación. Por defecto es `[tbp]`, que equivale a *top*, *bottom*, *page*. Eso quiere decir que intenta primero ponerla a comienzo de página. Si no lo consigue, al final de una página. Y si así tampoco lo consigue, en una página entera, solo para la figura. En este ejemplo utilizo las opciones `[hbt]` para que intente la secuencia *here*, *bottom*, *top*, *page*. En este caso prefiero que la ponga debajo antes que arriba, para que no aparezca en una sección anterior.

Con  $\text{\LaTeX}$  puedes conseguir que las figuras no se muevan en absoluto, pero eso deja documentos extremadamente descompensados. No lo hagas nunca. Es mejor mover ligeramente la figura en el texto o incluso re-escribir parte del texto, antes de forzar la posición. De todas formas, si no me quieres hacer caso, en la fig. **A.2** tienes un ejemplo que fuerza la posición.



**Figura A.2:** Figurita ejemplo 2

Para poder poner figuras que no son de elaboración propia es necesario primero obtener permiso del autor y, además, añadir la fuente al pie de foto. Hay muchas guías de estilo que explican en detalle cómo hacerlo. Por ejemplo, la *American Psychological Association* tiene un [capítulo específico de su manual de publicaciones](#). El manual de la APA se usa extensivamente en todo tipo de literatura científica.



**Figura A.3:** Figurita ejemplo 3. Extraída de la plantilla de TFG de Fernando Castillo. ©2018 Fernando Castillo. Reproducida con permiso.

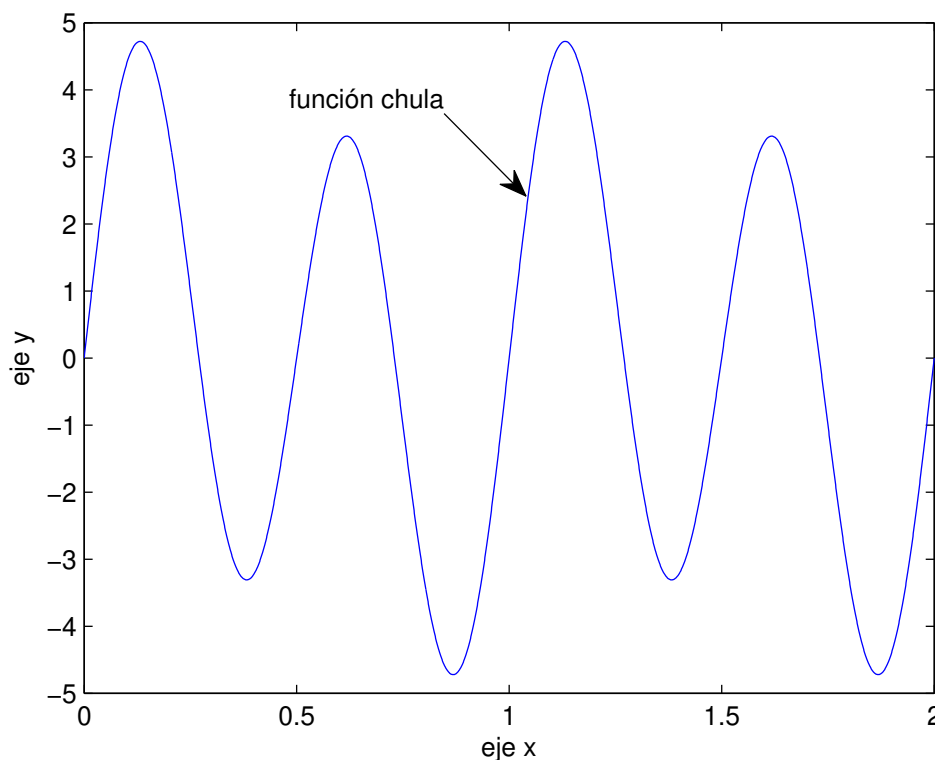


Fíjate bien. No se citan las imágenes como si se tratara de referencias bibliográficas. No debe haber pie de página (orden footnote) ni cita (orden cite) en un pie de foto (caption). La atribución de la obra debe estar al mismo nivel que la obra usada. Por eso debe atribuirse completamente en el pie de foto. Si no te gusta como queda haz tus propias imágenes.

Se puede controlar el escalado de la imagen y el ángulo de forma muy sencilla, con las opciones de la orden `includegraphics`. En la fig. A.3 se muestra un ejemplo de figura escalado a un 30%. La orden `includegraphics` ajusta los parámetros de la imagen para mantener la relación de aspecto original, si esto es posible. Esto hace que podamos especificar simplemente el ancho o el alto deseado, que va a ser lo más habitual. En mi opinión, las opciones más frecuentes de `includegraphics` son, por orden:

### **width**

Fija el ancho de la imagen. Puede ser un tamaño absoluto en centímetros (cm), milímetros (mm) o puntos PostScript (pt). Por ejemplo, `width=1.5cm`. También puede



**Figura A.4:** Figura ejemplo que ocupa todo el ancho del texto.

ser un tamaño relativo a cualquier medida del documento. Por ejemplo, `width=0.5\textwidth` sería una figura que ocupe la mitad del ancho del texto.

### height

Fija la altura de la imagen. Es similar a `width` pero con la altura. Se puede especificar tanto altura como anchura, de manera que se modifica la relación de aspecto original.

### scale

Utiliza un factor de escala para la imagen. Puede ser mayor de 1 para ampliar la imagen.

### angle

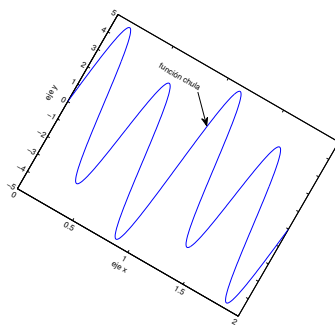
Gira la imagen un número de grados determinado. Si el número es negativo el giro es en sentido horario. Si es positivo el giro es antihorario.

En la fig. A.5 se muestra un ejemplo de figura girada 30.

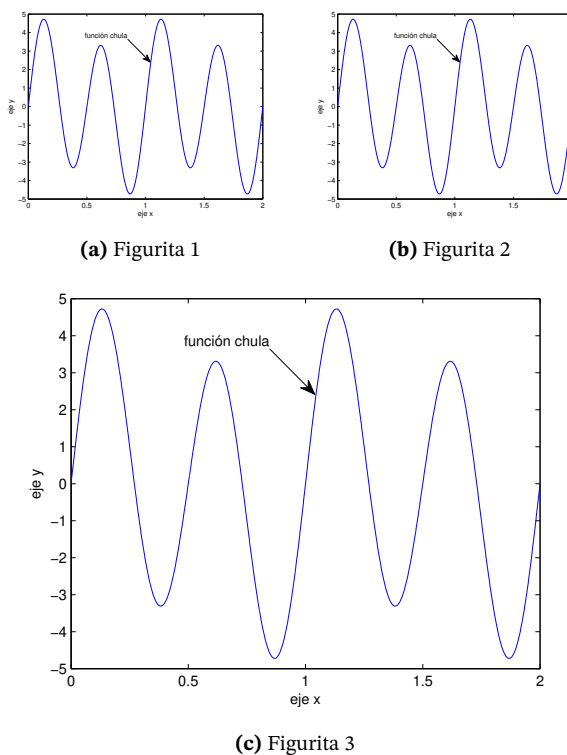


Hay muchos paquetes para definir subfiguras (`subfigure`, `subfig`, `subfloat`, `floatrow`, `subcaption`, ...). ¡Todos son incompatibles entre sí! En esta plantilla hemos usado el paquete más moderno, `subcaption`. No mezcles con otros paquetes de subfiguras.

Una característica interesante del entorno `figure` es que permite definir sub-figuras con la orden `subcaptionbox`. La colocación de las sub-figuras es prácticamente automática. Un ejemplo puede verse en la figura A.6. Las subfiguras pueden referenciarse eti-

**Figura A.5:** Figura ejemplo de rotación.

quetando la *subcaption* (primer argumento de `subcaptionbox`), como cualquier otro elemento del texto. Por ejemplo, la figura A.6a es igual que la figura A.6b.

**Figura A.6:** Matriz de figuras

### A.3 Tablas

En esta sección se muestran algunos ejemplos de tablas.

Las tablas en  $\text{\LaTeX}$  no son complejas pero puedes simplificar aún más usando un editor interactivo de tablas. Por ejemplo, en <https://truben.no/table/> hay una aplicación *online* para editar multitud de formatos de tablas. Es especialmente útil para tablas complicadas.

En  $\text{\LaTeX}$  es bastante frecuente separar la cabecera del cuerpo de la tabla poniendo dos `hline`, como en la tabla A.2.



Regulador	Función de Transferencia	orden
P	$\alpha_1$	2

**Cuadro A.1:** Resultados de la simulación

País	Ciudad
España	Madrid
España	Valencia
Francia	París

**Cuadro A.2:** Tabla muy sencilla.

La complejidad empieza cuando hay que expandir celdas para ocupar varias columnas o varias filas. Por ejemplo, la tabla (A.3) tiene una celda multi-columna y otra celda multi-fila. En estos casos un editor interactivo como el de [Peder Lång Skeidsvoll](#) puede ser de gran ayuda para un principiante. Explora las opciones, no son evidentes al principio.

Europa	
País	Ciudad
España	Madrid
	Valencia
Francia	París

**Cuadro A.3:** Fusionando celdas.

Las tablas, al igual que las figuras, tienen un parámetro opcional entre corchetes que indican las preferencias de posición. Se puede forzar pero, al igual que con las figuras, conduce a documentos muy descompensados. Procura evitarlo. Dentro de la tabla se define un entorno `tabular` que indica con su argumento obligatorio las columnas. Este entorno es muy útil en cualquier organización matricial. Se puede usar también para presentar las subfiguras de una figura, o para definir una matriz.

Las tablas muy largas deben dividirse en varias páginas. En el estilo de este TFG hemos incluido el paquete `longtable`, que facilita enormemente escribir este tipo de tablas largas. En ese caso, en lugar del entorno `table` y el entorno `tabular` se usaría solamente el entorno `longtable`, que es una especie de híbrido de los dos, con un montón de características opcionales. Para ilustrar su uso reproducimos [un ejemplo de TeXblog](#) en la tabla A.4.

**Cuadro A.4:** Un ejemplo de tabla larga

Primera	Segunda	Tercera	Cuarta
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4

*Continúa en la página siguiente*

Cuadro A.4 – Continúa de la página anterior

Primera	Segunda	Tercera	Cuarta
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4
1	2	3	4

A.4 Ecuaciones

Si hay algo donde  $\LaTeX$  es especialmente útil, es en las fórmulas matemáticas. Prácticamente no hay otra opción cuando las fórmulas son relativamente complejas. En esta sección se muestran algunos ejemplos de ecuaciones.

$$\mathbf{v} = \begin{bmatrix} 2 \\ 3 \\ -4 \end{bmatrix}$$

(A.1)

En  $\LaTeX$  es trivial el uso de cualquier notación de vectores. Tan solo hay que familiarizarse con las órdenes correspondientes. Por ejemplo, en esta ecuación:

$$\vec{F} = m\vec{a}$$

(A.2)

donde  $\vec{F}$  es la fuerza,  $\vec{a}$  es la actitud y  $m$  la masa.

Las ecuaciones pueden referenciarse igual que las figuras, las tablas o las secciones. Por ejemplo, la ecuación A.3 ...

$$\alpha_{inicial} = \beta^{final} + \gamma$$

(A.3)

Repara bien en cómo se escribe la ecuación anterior. Una palabra (o más de una) en una fórmula debe ponerse con ayuda de las órdenes auxiliares `\mathrm`, `\mathit`, etc. En

caso contrario parecerá un conjunto de letras que representan símbolos que se multiplican entre sí. Para más detalles consulta [Rob04].

$$G(s) = \frac{(s^2 + s + 1)^2}{s^3 + 1} \quad (\text{A.4})$$

El uso de letras griegas o símbolos matemáticos es también muy sencillo. Tan solo hay que familiarizarse con la orden que los inserta. Puede parecer difícil, pero basta con ojear una chuleta como [ésta](#)<sup>1</sup>.

La ecuación A.5 muestra un ejemplo de integral. También es muy sencillo, puesto que la notación de los límites coincide con la de los subíndices y superíndices.

$$F(y) = \int_{x_a}^{x_b} K(x, y) f(x) dx \quad (\text{A.5})$$

Cuando se necesita un entorno tabular dentro de un entorno matemático se utiliza el entorno `array`. La ecuación A.6 muestra un ejemplo.

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \quad (\text{A.6})$$

## A.5 Bibliografía, citas y referencias

Otro de los aspectos especialmente cuidados de  $\LaTeX$  es el manejo de bibliografía y citas. En esta plantilla utilizamos el paquete *biblatex*. Es un paquete muy flexible que permite adaptarse a casi cualquier estilo de citas existente. En los documentos de ingeniería no hay consenso en el estilo a utilizar. Nosotros hemos configurado en la plantilla el estilo del IEEE, pero dependiendo del área específica de trabajo puede ser necesario cambiarlo. Consulta con tu tutor.

La bibliografía en  $\LaTeX$  se hace con ayuda de unos archivos auxiliares escritos en formato BibTeX. Es otro formato textual, con una serie de campos que hay que rellenar. Para la composición de entradas BibTeX lo más sencillo es utilizar un editor online, como la página <http://truben.no/latex/bibtex/>. En principio todas las entradas de bibliografía que utilices en tu TFG deben ponerse en `bib/main.bib`.

En  $\LaTeX$  la forma más básica de cita consiste en emplear la orden `cite` con el campo clave que contiene todo registro de BibTeX. Por ejemplo, según el trabajo [AGBB11] ...mientras que según [CFRS10] el control es una cosa muy buena. Pero hay muchas otras opciones de cita. Consulta la [chuleta de Bib \$\LaTeX\$](#)  para ver todas las posibilidades. Entre las alternativas más frecuentes está la orden `parencite` con el campo clave que contiene el registro BibTeX y entre corchetes la página concreta. Por ejemplo, según [AGBB11, pág. 3] bla bla. Como ves, el aspecto visual es similar, pero añade la página a la que hacemos referencia. Otra posibilidad es usar la orden `textcite` que añade los autores. Por ejemplo, según Armas et al. [AGBB11] bla bla.

Se puede personalizar el aspecto de las citas cambiando los parámetros `style` y `citestyle` del paquete *biblatex* en `sty/eiitfg.cls`. En ingeniería el estilo más utilizado, con mucha diferencia, es el de IEEE, del que existen dos variantes, `ieeey` y `ieee-alphabetic`.

<sup>1</sup><https://tug.ctan.org/info/symbols/comprehensive/symbols-a4.pdf>

La única diferencia entre ambos es que en el primero se usan números para identificar las referencias y en el segundo se utilizan iniciales de los apellidos de los autores.

Otros valores muy utilizados son `alphabetic`, `authoryear`, `apa`, `chem-acs`, `mla`, `phys`, `nature`, `science`. Sin embargo te recomendamos que utilices una de las dos variantes del estilo IEEE porque incluye soporte de entradas BibTeX para patentes.

También se puede acceder a campos concretos del registro BibTeX, tales como el autor, el año o el título. Por ejemplo, Armas et al. escribió en 2011 el artículo «Estimation of opacity tendency of ethanol–and biodiesel–diesel blends by means of the smoke point technique».

Otro estilo más antiguo es el estilo de Vancouver, en el que se usan notas al pie. En  $\LaTeX$  puede hacerse con la orden `footcite`. Por ejemplo, según el profesor Armas<sup>2</sup> bla bla. Todavía se ve este estilo en libros de historia, pero tiende al desuso porque rompe la linealidad del texto. En general, lo más importante es ser consistente, usa un estilo y solo un estilo en todo el documento.

### A.5.1 Lo que toda referencia debe tener

Una referencia bibliográfica se utiliza como argumento de autoridad, para dar peso a tu propia argumentación. Por tanto, hay tres elementos clave que siempre deben estar:

- El autor, puesto que palabras anónimas no dan peso a nada. Recuerda que el autor es lo que da peso a tu argumento. No cites artículos divulgativos, ni autores sin un mínimo prestigio en el campo de lo que afirman.
- El título, puesto que el lector debe poder buscar por sí mismo el documento original.
- La fecha, puesto que un mismo autor puede cambiar de opinión a lo largo de su vida. Por ejemplo, John Maynard Keynes es Premio Nobel, pero tiene numerosos escritos contradictorios. Su opinión era bastante cambiante con el tiempo.

Si falta alguno de estos elementos no es una referencia y no se cita. Se puede poner como una nota a pie de página (`footnote`) o como una URL en el cuerpo del texto, pero no como una referencia.

Por cierto, es conveniente citar las fuentes. Es decir, debes tomarte la molestia de buscar quién dijo o inventó lo que citas y dónde lo publicó por primera vez. Es la mínima cortesía que se debe tener con los colegas de profesión. Supongo que tú también querrás crédito por tu trabajo en tu futuro profesional.

## A.6 Hojas de datos

Las hojas de datos normalmente son archivos PDF. En  $\LaTeX$  hay dos formas de insertar archivos PDF. La más sencilla es utilizar la orden `includegraphics` que ya hemos visto en las figuras, pero solo se puede incrustar una página en cada llamada a `includegraphics`.

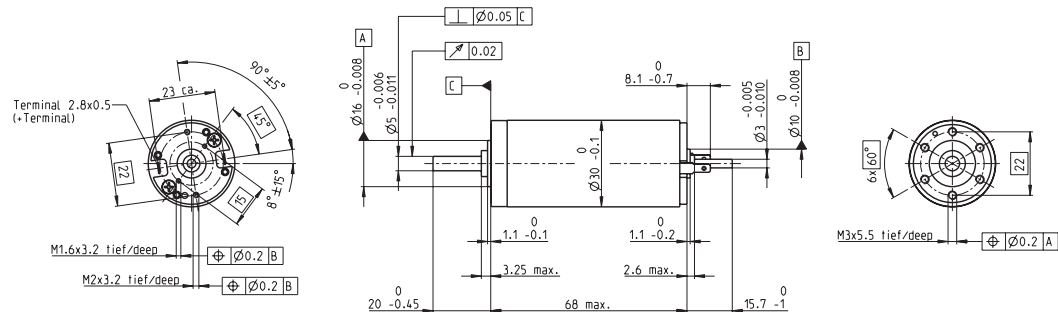
Si el documento tiene varias páginas habría que insertar cada una de las páginas de interés con órdenes `includegraphics` independientes. En el parámetro `page` podemos elegir la página a insertar. Este método nos da el máximo control para poder poner pies de figura y el posicionamiento. Sin embargo, cuando el documento tiene muchas páginas es

---

<sup>2</sup>AGBB11, pág. 3.

un poco engorroso. Para esos casos se puede utilizar la orden `includepdf` del paquete *pdfpages*.

Para ilustrarlo veamos el mismo ejemplo, primero con `includepdf` y luego con `includegraphics`.

**RE 30** Ø30 mm, Graphite Brushes, 60 Watt

M 1:2

- Stock program  
 Standard program  
 Special program (on request)

according to dimensional drawing  
shaft length 15.7 shortened to 8.7 mm

Part Numbers					
310005	310006	310007	310008	310009	
268193	268213	268214	268215	268216	

Motor Data		208193		208213		208214		208215		208216					
Values at nominal voltage															
1	Nominal voltage	V	12	18	24	36	48								
2	No load speed	rpm	8170	8590	8810	8590	8490								
3	No load current	mA	301	213	165	106	78.6								
4	Nominal speed	rpm	7630	7910	8050	7840	7760								
5	Nominal torque (max. continuous torque)	mNm	51.6	75.5	85.6	86.6	89.7								
6	Nominal current (max. continuous current)	A	4	4	3.47	2.28	1.74								
7	Stall torque	mNm	852	1000	1020	1000	1050								
8	Starting current	A	61.1	50.3	39.3	25.2	19.6								
9	Max. efficiency	%	85	87	87	87	88								
Characteristics															
10	Terminal resistance	Ω	0.196	0.358	0.611	1.43	2.45								
11	Terminal inductance	mH	0.0345	0.0703	0.119	0.281	0.513								
12	Torque constant	mNm/A	13.9	19.9	25.9	39.8	53.8								
13	Speed constant	rpm/V	685	479	369	240	178								
14	Speed / torque gradient	rpm/mNm	9.64	8.61	8.69	8.61	8.09								
15	Mechanical time constant	ms	3.4	3.24	3.05	2.98	2.94								
16	Rotor inertia	gcm²	33.7	35.9	33.5	33.1	34.7								

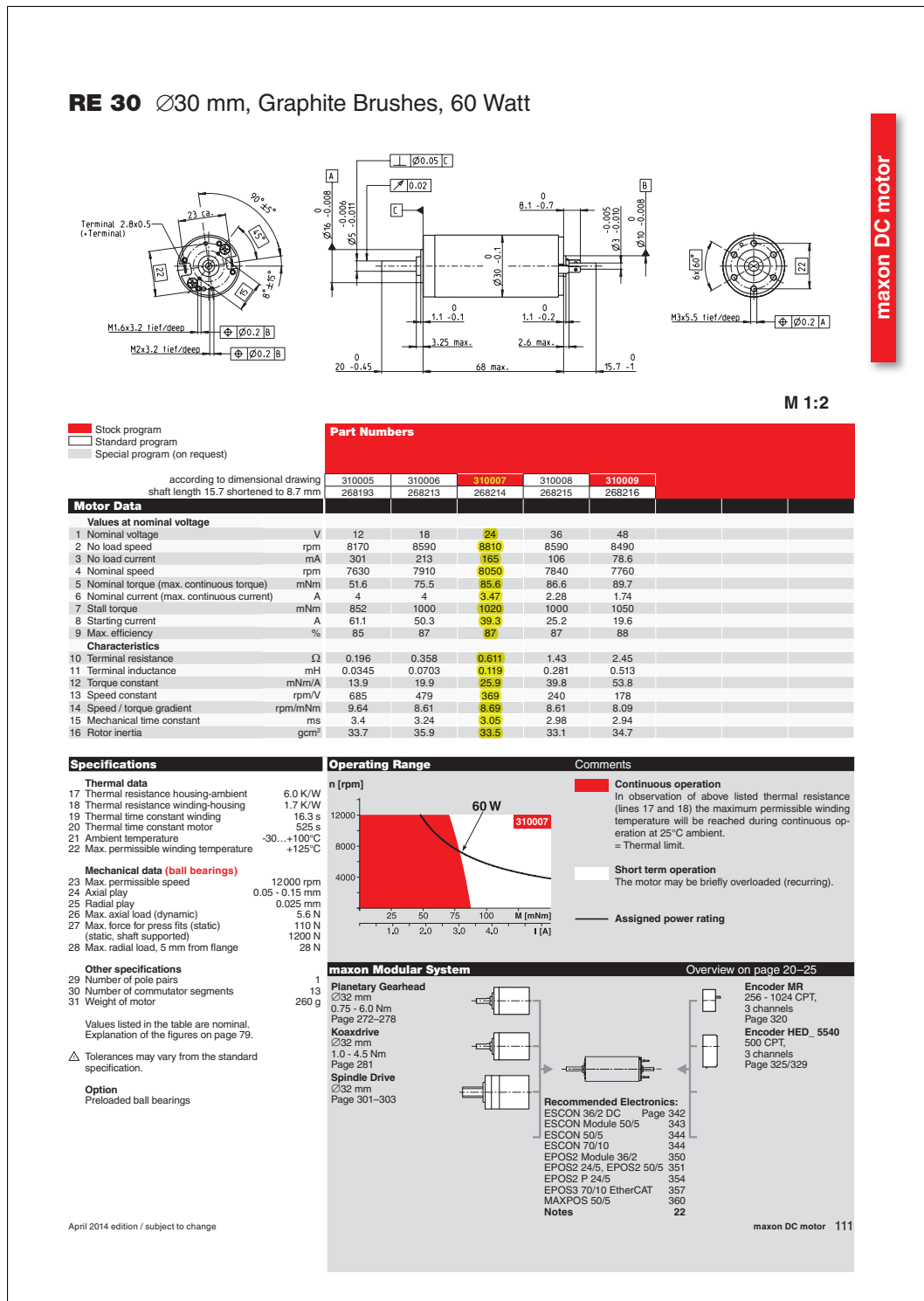
Specifications	Operating Range	Comments
<b>Thermal data</b> 17 Thermal resistance housing-ambient 6.0 K/W 18 Thermal resistance winding-housing 1.7 K/W 19 Thermal time constant winding 16.3 s 20 Thermal time constant motor 525 s 21 Ambient temperature -30...+100°C 22 Max. permissible winding temperature +125°C  <b>Mechanical data (ball bearings)</b> 23 Max. permissible speed 12000 rpm 24 Axial play 0.05 - 0.15 mm 25 Radial play 0.025 mm 26 Max. axial load (dynamic) 5.6 N 27 Max. force for press fits (static) 110 N (static, shaft supported) 1200 N 28 Max. radial load, 5 mm from flange 28 N	<p>Operating range graph showing speed <math>n</math> [rpm] vs torque <math>M</math> [mNm] and current <math>I</math> [A] for 60 W and 310007.</p>	<p><span style="background-color: red; color: white;"> </span> <b>Continuous operation</b> In observation of above listed thermal resistance (lines 17 and 18) the maximum permissible winding temperature will be reached during continuous operation at 25°C ambient. = Thermal limit.</p> <p><span style="background-color: white; border: 1px solid black;"> </span> <b>Short term operation</b> The motor may be briefly overloaded (recurring).</p> <p>— <b>Assigned power rating</b></p>

Other specifications	maxon Modular System	Overview on page 20–25
29 Number of pole pairs 1 30 Number of commutator segments 13 31 Weight of motor 260 g  Values listed in the table are nominal. Explanation of the figures on page 79.  ⚠ Tolerances may vary from the standard specification.  <b>Option</b> Preloaded ball bearings	<b>Planetary Gearhead</b> Ø32 mm 0.75 - 6.0 Nm Page 272–278 <b>Koaxdrive</b> Ø32 mm 1.0 - 4.5 Nm Page 281 <b>Spindle Drive</b> Ø32 mm Page 301–303	<b>Encoder MR</b> 256 - 1024 CPT, 3 channels Page 320 <b>Encoder HED_ 5540</b> 500 CPT, 3 channels Page 325/329

**Recommended Electronics:**

ESCON 36/2 DC	Page 342
ESCON Module 50/5	343
ESCON 50/5	344
ESCON 70/10	344
EPOS2 Module 36/2	350
EPOS2 24/5, EPOS2 50/5	351
EPOS2 P 24/5	354
EPOS3 70/10 EtherCAT	357
MAXPOS 50/5	360
Notes	22

La opción con `includegraphics` permite referirse a una página concreta y poner pies de figura. Por ejemplo, en la figura A.7 se muestra la hoja de especificaciones del motor empleado.



**Figura A.7:** Figura ejemplo. Tomada de hoja de catálogo de **motores DC con escobillas de grafito de Maxon**  
© 2014 Maxon Motors. Reproducida con permiso.

## A.7 Código fuente

Para insertar fragmentos o listados completos de código se puede usar el paquete `lstlistings`. Permite incluir archivos o parte de archivos directamente del proyecto con la orden `lstinputlisting`.

Listado A.1: Ejercicio 20 como texto incorporado

```

1 function [T]=Ejercicio20(f,c)
2
3 T = char('B'*ones(8,8));
4
5 for i=1:8
6     for j=1:8
7         if ( (i==f) || (j==c) || (i+j==f+c) || (i-j==f-c) )
8             T(i,j)='*';
9         elseif ( rem(i+j,2)~=0 )
10            T(i,j)='N';
11        end
12    end
13 end
14
15 T(f,c)='R';

```

O bien se puede copiar el texto del programa o fragmento en un entorno `lstlisting` con las mismas opciones que la orden `lstinputlisting`.

Listado A.2: Ejercicio 20 como texto en línea.

```

1 function [T]=Ejercicio20(f,c)
2
3 T = char('B'*ones(8,8));
4
5 for i=1:8
6     for j=1:8
7         if ( (i==f) || (j==c) || (i+j==f+c) || (i-j==f-c) )
8             T(i,j)='*';
9         elseif ( rem(i+j,2)~=0 )
10            T(i,j)='N';
11        end
12    end
13 end
14
15 T(f,c)='R';

```



Te recomendamos que incluyas los archivos o parte de los archivos directamente del código de tu proyecto, ya sea mediante `lstinputlisting` o mediante `inputminted`. De esta forma mantendrás sincronizado el documento con el código fuente.

El paquete `lstlisting` te permite quitar los números y el marco, cuando el código se incluye como parte del texto.

```

function [T]=Ejercicio20(f,c)

T = char('B'*ones(8,8));

for i=1:8
    for j=1:8

```



```

        if ( (i==f) || (j==c) || (i+j==f+c) || (i-j==f-c) )
            T(i,j)='*';
        elseif ( rem(i+j,2)~=0 )
            T(i,j)='N';
        end
    end
end

T(f,c)='R';

```

Otra forma de incluir código es mediante el entorno `verbatim`. Este método no tiene resalte de sintaxis ni facilidades de ningún tipo para definir etiquetas o numerar las líneas.

```

function [T]=Ejercicio20(f,c)

T = char('B'*ones(8,8));

for i=1:8
    for j=1:8
        if ( (i==f) || (j==c) || (i+j==f+c) || (i-j==f-c) )
            T(i,j)='*';
        elseif ( rem(i+j,2)~=0 )
            T(i,j)='N';
        end
    end
end

T(f,c)='R';

```

Otra forma alternativa a `lstlisting` es el paquete `minted`, que colorea el programa según el lenguaje empleado.

```

1  function [T]=Ejercicio20(f,c)
2
3  T = char('B'*ones(8,8));
4
5  for i=1:8
6      for j=1:8
7          if ( (i==f) || (j==c) || (i+j==f+c) || (i-j==f-c)
8              ↪ )
9              T(i,j)='*';
10             elseif ( rem(i+j,2)~=0 )
11                 T(i,j)='N';
12             end
13         end
14     end
15     T(f,c)='R';

```

O bien, usando la orden `inputminted` para incluir directamente un archivo Matlab.

```

1  function [T]=Ejercicio20(f,c)
2
3  T = char('B'*ones(8,8));
4
5  for i=1:8
6      for j=1:8
7          if ( (i==f) || (j==c) || (i+j==f+c) || (i-j==f-c)
8              ↪ )
9              T(i,j)='*';
10             elseif ( rem(i+j,2)~=0 )
11                 T(i,j)='N';
12             end
13         end
14     end
15     T(f,c)='R';

```

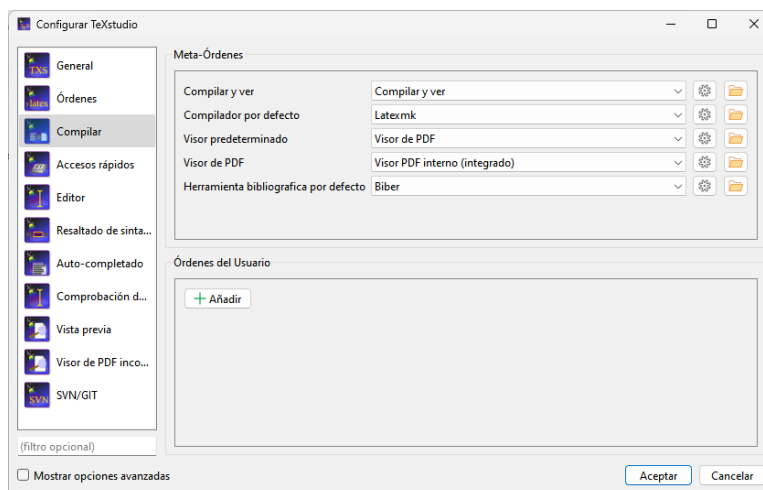
## A.8 Compilación local

Si utilizas Overleaf con una cuenta gratuita es muy posible que excedas el límite de tiempo para compilar. Nuestra recomendación es que tu tutor cree el proyecto correspondiente a tu TFG y te lo comparta como colaborador con permiso de escritura. Pero también puedes compilarlo localmente en tu propio ordenador instalando algo de software. Como ha sido una preocupación recurrente, hemos cambiado la plantilla para que sea más fácil de compilar en Microsoft Windows.

1. Instala [TexStudio](#). Es un editor de  $\text{\LaTeX}$  no tan sencillo como Overleaf pero mucho más amigable que editar los archivos en un editor de texto normal.
2. Instala [MikTeX](#). Si marcas la opción *Install missing packages on the fly* los paquetes de  $\text{\LaTeX}$  se instalarán automáticamente, que resulta muy conveniente.
3. Si usas Windows instala [Strawberry Perl](#). Es necesario para utilizar la orden *Latexmk* en la compilación.
4. Instala [Python](#). Lo necesitarás si usas el entorno **minted** para colorear trozos de código. A la hora de instalar Python no olvides marcar la casilla para añadir Python a la variable de entorno Path.

### A.8.1 Configurar el entorno

1. Abre la aplicación *MikTeX Console* e instala las fuentes de la plantilla. Para ello pincha en *Packages* y añade los paquetes `stix`, `stix2-type1`, `stix2-otf`, `arial` y `noto`.
2. Añade la carpeta que contiene `perl.exe` a la variable de entorno Path. Si instalaste Strawberry Perl en el lugar por omisión debes añadir `C:\Strawberry\perl\bin`.



**Figura A.8:** Configuración de la compilación en TeXStudio



3. Elimina el alias de ejecución de Python de Windows. Para ello pulsa la tecla de Windows y escribe `alias`, selecciona *Administrar alias de ejecución de aplicaciones*. Asegúrate que el alias del *Instalador de aplicaciones* no está activado para `python.exe` y `python3.exe`.

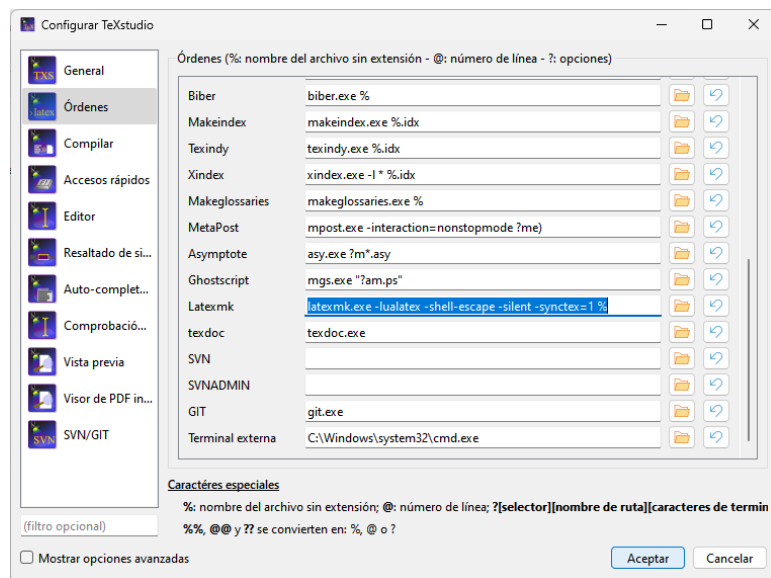
Comprueba que todo está correctamente configurado. Para ello abre una ventana de órdenes (PowerShell o Símbolo de sistema) y teclea `perl -v` y posteriormente `python -V`. Ambas órdenes deberían ejecutarse sin error y mostrar las versiones de Perl y Python respectivamente. Por último ejecuta `pip install Pygments` que instalará el software de coloreado de código.

### A.8.2 Configurar TeXStudio

Entra en *Opciones* → *Configurar TeXStudio* y configura la sección *Compilar* como se muestra en la [Figura A.8](#). El compilador por defecto debe ser **Latexmk** y la herramienta bibliográfica debe ser **Biber**.

Entra en la sección *Órdenes* y modifica las opciones de *Latexmk* como muestra la [Figura A.9](#). La opción `-pdf` debe ser sustituida por `-lualatex -shell-escape`.

Una vez configurado, para compilar el documento solo hay que pulsar en el botón **Compilar** . Para previsualizar el resultado hay que pulsar sobre el botón **Visualizar** .

Figura A.9: Configuración de la orden *Latexmk* en TeXstudio

# Bibliografía

- [Ara13] E. Aranda, *Curso de  $\text{\LaTeX}$* , oct. de 2013. dirección: <http://matematicas.uclm.es/earanda/wp-content/uploads/downloads/2013/10/latex.pdf>.
- [AGBB11] O. Armas, M. A. Gómez, E. J. Barrientos y A. L. Boehman, «Estimation of opacity tendency of ethanol–and biodiesel–diesel blends by means of the smoke point technique», *Energy & Fuels*, vol. 25, n° 7, págs. 3283-3288, 2011.
- [CFRS10] F. J. Castillo, V. Feliú, R. Rivas y L. Sánchez, «Design of a class of fractional controllers from frequency specifications with guaranteed time domain behavior», *Computers & Mathematics with Applications*, vol. 59, n° 5, págs. 1656-1666, 2010.
- [KC07] B. Kitchenham y S. Charters, «Guidelines for performing Systematic Literature Reviews in Software Engineering», Keele University y University of Durham, inf. téc. EBSE-2007-01, jul. de 2007. dirección: [https://www.elsevier.com/\\_\\_data/promis\\_misc/525444systematicreviewsguide.pdf](https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf).
- [Kof14] A. Kofod-Petersen, *How to do a Structured Literature Review in computer science*, ago. de 2014. dirección: [https://research.idi.ntnu.no/aimasters/files/SLR\\_HowTo.pdf](https://research.idi.ntnu.no/aimasters/files/SLR_HowTo.pdf).
- [Lam94] L. Lamport,  *$\text{\LaTeX}$ : A Document Preparation System*, second. Addison Wesley, 1994.
- [MGB+04] F. Mittelbach, M. Goossens, J. Braams, D. Carlisle y C. Rowley, *The  $\text{\LaTeX}$  Companion*, second. Addison Wesley, 2004.
- [Rob04] A. Roberts. «Adding text to equations». (ago. de 2004), dirección: <https://www.andy-roberts.net/res/writing/latex/textineqn.pdf>.
- [Sch17] S. Schulze, *Conducting Systematic Literature Reviews*, oct. de 2017. dirección: <https://pdfs.semanticscholar.org/presentation/63ca/d41c0d584f56461c28d8f2f89443fdebee9c.pdf>.
- [Sim02] G. Simone, *Tutorial de  $\text{\LaTeX}$ : Introducción básica*, jun. de 2002. dirección: <http://www.mat.uda.cl/hsalinas/cursos/2008/latex/doc-tutorial-latex.pdf>.