



**COMILLAS**  
UNIVERSIDAD PONTIFICIA

ICAI

# GRADO EN INGENIERÍA DE TECNOLOGÍAS INDUSTRIALES

TRABAJO FIN DE GRADO

SISTEMA DOMÓTICO PARA EL CONTROL REMOTO DE ACCESO A UNA VIVIENDA

Autor: Eduardo Martínez Jiménez

Director: Juan Luis Zamora Macho

Madrid

Julio de 2019



## **AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESIS O MEMORIAS DE BACHILLERATO**

### ***1º. Declaración de la autoría y acreditación de la misma.***

El autor D.EDUARDO MARTÍNEZ JIMÉNEZ

DECLARA ser el titular de los derechos de propiedad intelectual de la obra: \_\_SISTEMA DOMÓTICO PARA EL CONTROL REMOTO DE ACCESO A UNA VIVIENDA\_\_, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

### ***2º. Objeto y fines de la cesión.***

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

### ***3º. Condiciones de la cesión y acceso***

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

- a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar “marcas de agua” o cualquier otro sistema de seguridad o de protección.
- b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
- c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
- d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
- e) Asignar por defecto a estos trabajos una licencia Creative Commons.
- f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente*).

### ***4º. Derechos del autor.***

El autor, en tanto que titular de una obra tiene derecho a:

- a) Que la Universidad identifique claramente su nombre como autor de la misma
- b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
- c) Solicitar la retirada de la obra del repositorio por causa justificada.
- d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

### ***5º. Deberes del autor.***

El autor se compromete a:

- a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
- b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la

intimidad y a la imagen de terceros.

- c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
- d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

**6º. Fines y funcionamiento del Repositorio Institucional.**

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusiva del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a ...13..... de .....Julio..... de ...2019...

**ACEPTA**



Fdo.....

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título .....SISTEMA DOMÓTICO PARA EL CONTROL REMOTO DE ACCESO A UNA VIVIENDA.....en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico ...2018/2019.... es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.

Fdo.: Eduardo Martínez Jiménez

Fecha: ...13.../ ...07.../ ...2019...



Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: Juan Luis Zamora Macho

Fecha: 14 / 7 / 2019







**COMILLAS**  
UNIVERSIDAD PONTIFICIA

ICAI

# GRADO EN INGENIERÍA DE TECNOLOGÍAS INDUSTRIALES

TRABAJO FIN DE GRADO

SISTEMA DOMÓTICO PARA EL CONTROL REMOTO DE ACCESO A UNA VIVIENDA

Autor: Eduardo Martínez Jiménez

Director: Juan Luis Zamora Macho

Madrid

Julio de 2019



# SISTEMA DOMÓTICO PARA EL CONTROL REMOTO DE ACCESO A UNA VIVIENDA

**Autor: Martínez Jiménez, Eduardo**

**Director: Zamora Macho, Juan Luis**

## RESUMEN DEL PROYECTO

### 1.- Introducción:

La domótica es uno de los campos de la electrónica que ha visto más evolución en los años recientes.

El acceso a la vivienda es aspecto importante de la automatización de los hogares porque es posible ofrecer productos convenientes y fáciles de usar para los dueños.

El objetivo del proyecto es crear un prototipo que permita ordenar la apertura de la vivienda enviando un SMS con un código preestablecido. Esto será posible cuando el timbre sea pulsado y se haya enviado al dueño un e-mail con una imagen adjunta tomada en el momento de llamada al timbre.

### 2.- Metodología:

Para cumplir con los objetivos del proyecto, se siguieron los siguientes pasos:

1. Determinación del software a utilizar para implementar el proyecto.
2. Determinación del hardware a utilizar, teniendo en cuenta compatibilidad con el software y los demás componentes del hardware.
3. Diseño y pruebas de cada subsistema por separado en escenarios similares al de funcionamiento habitual.
4. Diseño de la máquina de estados general del sistema.

5. Unificación de los subsistemas en un único modelo.

El software utilizado para este proyecto es Matlab/Simulink, por su practicidad y su compatibilidad con numerosos microcontroladores.

El microcontrolador elegido fue la Raspberry Pi 3B+, por cumplir con los requisitos del proyecto, la compatibilidad con Simulink a través de un paquete de bloques descargable, y su precio asequible. Además, el módulo GSM elegido (SIM800C) cuenta con modelos adaptados para la Raspberry Pi, y este microcontrolador cuenta además con una ranura donde integrar la cámara de Raspberry, pudiendo así juntar los componentes de forma compacta.

### 3.- Resultados

Las primeras pruebas realizadas fueron sobre la cámara. El principal obstáculo era que el bloque de Simulink que permite la interacción con la cámara emite vídeo en directo, y no existe un bloque que tome una captura única al ser activado. La primera alternativa planteada fue diseñar una función de Simulink que se activará en la transición del primer estado al segundo, pero al evaluar los resultados se puso de manifiesto que al recorrer varias veces el bucle, las capturas que se tomaban eran de instantes consecutivos desde la primera ejecución. Para arreglar el problema, se modificó el sistema para que funcionara en todo momento el

bloque de vídeo en directo. En esta situación, es necesario un estado auxiliar para tomar la foto, en la que una variable de salida de la máquina de estados activa un switch en el subsistema de la cámara. Este switch alterna la entrada a un "Data Store Write" entre un "Data Store Read" de sí mismo (no tiene efecto, funcionamiento por defecto), y el bloque de vídeo en directo. De esta manera, la imagen se actualiza correctamente en el momento necesario.

En lo que respecta al módulo GSM, se evaluaron dos opciones para el envío de la imagen: e-mail o MMS. Se eligió e-mail por parecer la opción más viable y tener menos restricciones en el tamaño de envío de archivos. Se diseñó un proceso en Simulink para el envío del e-mail con los comandos necesarios. Sin embargo, el comando de enviar el e-mail daba persistentemente un error de conexión a la red. No se consiguió determinar el error, puesto que la tarjeta SIM parecía funcionar adecuadamente en otros tests (envío y recepción de SMS) y la secuencia de comandos debía ser la correcta. La conclusión final es que el error debía estar en uno de los parámetros de conexión a la red GPRS de Movistar.

El otro apartado que requería el uso del módulo GSM fue más exitoso. Se consiguió diseñar un sistema capaz de leer los mensajes nuevos recibidos en el módulo y extraer de ellos el número de teléfono del remitente y el contenido del mensaje para, exitosamente, determinar si el conjunto de valores era válido y se podía accionar el cierre eléctrico de la puerta.

Finalmente, se diseñó un pequeño circuito para la detección del accionamiento de un pulsador, consistente en una resistencia, un condensador y un pin de lectura digital de la Raspberry. Las pruebas realizadas dieron los resultados esperados, por el pin se leía 1 con el botón pulsado, y 0 en otros casos, sin ningún efecto rebote.

#### 4.- Conclusiones y desarrollo futuro

Tras el proceso de diseño del prototipo, se pueden extraer las siguientes conclusiones:

- El principal problema a solucionar es conseguir la conexión a Internet para hacer eficaz el envío del e-mail.
- La apertura del cierre es segura, puesto que, incluso si alguien inesperado descubre el código que abre la puerta, necesitará también acceso al teléfono móvil del propietario para poder abrirla de forma remota.
- El diseño del pulsador es exitoso y cumple su función a un muy bajo coste. Se podría mejorar este aspecto incluyendo elementos como un altavoz para permitir la comunicación mediante llamada telefónica o una pantalla digital para mostrar mensajes como "Acceso denegado" o "Por favor espere. Solicitando acceso".
- El funcionamiento de la cámara es adecuado, pero si la primera foto tomada no es satisfactoria, se enviará una imagen poco representativa. Sería posible añadir filtros para detectar imágenes defectuosas y ordenar la toma de una nueva.

# AUTOMATED SYSTEM FOR REMOTE CONTROL ACCESS TO A HOUSE

**Author: Martínez Jiménez, Eduardo**

**Director: Zamora Macho, Juan Luis**

## 1. - Introduction:

Home automation is one of the fields of electronics that has seen more changes in the last few years.

Granting access to a house is a very important aspect of automation because it is possible to offer convenient, easy to use products for home owners.

This project's objective is to create a prototype capable of opening a house's front door by sending an SMS with a pre-established code. This will be possible when someone rings the front door bell and the owner has been sent a picture in the form of an e-mail attachment.

## 2. - Methodology:

To achieve the project's objectives, the following steps were taken:

1. Choosing the software that will be used
2. Choosing the hardware components, taking into account compatibility with the software and other hardware components.
3. Design and testing of each subsystem independently.
4. Design of the global status machine of the system.
5. Putting the subsystems together in one final model.

The software that will be used for this project is MatLab/Simulink, for its practicality and compatibility with most microcontrollers.

The microcontroller that was chosen for the project was the Raspberry Pi 3B+, because it meets the project requirements, it is compatible with Simulink through a downloadable blockset, and it has an affordable price. The chosen GSM module is available in setups for Raspberry Pi compatibility, and this microcontroller also has a camera input, ideal to keep the product's size to a minimum.

## 3. - Results

The first subsystem that was looked into was the camera. The main obstacle was the fact that Simulink's block that allows interaction with cameras outputs live video. The first attempt at extracting an image was to create a Simulink function that would only trigger on a state transition. However, the results that this test yielded were of consecutive frames from the first time the live video block was activated. To fix the problem, the system was modified to have continuous functioning of the live video block. This, in turn, required an auxiliary state where a state machine output would trigger a switch, allowing the Data Store Write block corresponding to the image to be updated with the current frame output of the live video block. This guarantees the update of the image at the correct time.

Concerning the GSM module, two alternatives were evaluated for the delivery of the image: e-mail and MMS. The first option was chosen, because it

seemed more viable and had less attachment size restrictions. A sequence of AT commands were designed in Simulink to be sent in order to the GSM module via the serial port. However, the e-mail send command kept giving a network connection error. The cause of the error was not found, because the SIM card seemed to work correctly in other tests (sending and receiving SMS), and the command sequence was correct. The final conclusion was that one of the network setup parameters for the connection to Movistar's GPRS network must have been wrong.

The other task that required the use of the GSM module was more successful. The designed subsystem was capable of reading the new message that was received and it correctly extracted the information relating to the sender's phone number and the SMS's content. It then correctly determined if these two values were correct or not, allowing for the door to be opened, or denying access to the house.

Finally, a small circuit was built to detect the pressing of a button, consisting in a resistor and a capacitor, with a Raspberry Pi digital read. Testing yielded the expected results: reading a 1 when the button is pressed, and a 0 otherwise.

#### 4. - Conclusions and future development

After the prototype's design process, the following conclusions can be extracted

- The main problem to solve is getting an Internet connection that allows for the e-mail to be sent
- The lock's opening is safe, because, even if an unexpected third party were to figure out the code that was established, it would still be necessary to use the owner's phone number to remotely open the door.
- The circuit's design was successful and achieves its function at a very low price. It could be enhanced by adding elements such as speakers and a microphone to allow for voice communication via phone call, or a digital screen that could display messages such as "Access denied" or "Please wait. Access has been requested"
- The camera works correctly, but if the first picture that is taken is not satisfactory, a non-ideal image will be sent to the owner: Filters can be added to detect poor quality images, and, if the picture taken is not sufficiently good, a new one may be taken.

# Índice

|  |    |
|--|----|
| Índice .....                                 | 1  |
| Índice de figuras .....                      | 3  |
| 1.- Introducción.....                        | 5  |
| 1.1 Introducción.....                        | 5  |
| 1.2 Estado del arte .....                    | 6  |
| 1.3 Motivación.....                          | 7  |
| 1.4 Objetivos .....                          | 7  |
| 1.5 Estructura del documento.....            | 8  |
| 2.- Hardware .....                           | 9  |
| 2.1.- Estructura general.....                | 9  |
| 2.2.- Microcontrolador .....                 | 10 |
| 2.3.- Módulo GSM .....                       | 10 |
| 2.3.1.- SIM800C .....                        | 11 |
| 2.3.2.- Comunicación serie.....              | 11 |
| 2.3.3.- Comandos de Hayes.....               | 11 |
| 2.4.- Cámara .....                           | 12 |
| 2.5.- Cierre eléctrico.....                  | 13 |
| 2.6.- Timbre.....                            | 14 |
| 3.- Software.....                            | 15 |
| 3.1.- Estructura general.....                | 15 |
| 3.2.- Módulo GSM .....                       | 16 |
| 3.3.- Cámara Raspberry. ....                 | 21 |
| 3.4.- Timbre.....                            | 22 |
| 3.5.- Conjunto batería/cierre.....           | 22 |
| 3.6.- Máquina de estados general. ....       | 22 |
| 4.- Resultados .....                         | 25 |
| 4.1.- Módulo GSM .....                       | 25 |
| 4.2.- Cámara de la Raspberry.....            | 28 |
| 4.3.- Pulsador.....                          | 29 |
| 5.- Conclusiones y futuros desarrollos ..... | 31 |
| 5.1.- Actuador: SIM800C.....                 | 31 |

|   |    |
|---|----|
| 5.2.- Sensor: Pulsador.....                 | 32 |
| 5.3.- Sensor: Cámara.....                   | 32 |
| 5.4.-Viabilidad económica del proyecto..... | 33 |
| A. Presupuesto.....                         | 34 |
| A.1: Sumas parciales.....                   | 34 |
| A.1.1: Hardware.....                        | 34 |
| A.1.2: Software.....                        | 34 |
| A.1.3: Herramientas y equipos.....          | 34 |
| A.1.4: Mano de obra directa.....            | 35 |
| A.2: Presupuesto general.....               | 35 |
| Bibliografía.....                           | 36 |

# Índice de figuras

|  |    |
|--|----|
| Figura 2.1: Esquema de conexión del hardware.....  | 8  |
| Figura 2.2: Raspberry Pi 3B+.....  | 9  |
| Figura 2.3: Circuito correspondiente al timbre.....                                      | 13 |
| Figura 3.1: Estructura del software.....   | 14 |
| Figura 3.2: Máquina de estados módulo GSM.....   | 16 |
| Figura 3.3: Lógica del estado “Esperar SMS” .....  | 19 |
| Figura 3.4: Funcionamiento del buffer de lectura.....                                    | 20 |
| Figura 3.5: Diagrama de flujo de software de la cámara.....                              | 21 |
| Figura 3.6: Máquina de estados general.....  | 22 |
| Figura 4.1: Respuesta al comando AT.....   | 24 |
| Figura 4.2: Respuesta al comando AT+CMGL=“REC UNREAD” sin mensajes nuevos.....           | 25 |
| Figura 4.3: Extracto de la respuesta al comando AT+CMGL=“ALL” .....                      | 25 |
| Figura 4.4.1: Transiciones más destacables de ESTADO_SIM, ABRIR_PUERTA y TOMAR_FOTO..... | 26 |
| Figura 4.4.2: Transiciones más destacables de OUTPUT_SIM, TIMBRE, ESTADO GENERAL.....    | 27 |
| Figura 4.5: Imagen tomada por la cámara de la Raspberry (Tamaño 320 x 240).....          | 28 |
| Figura 4.6: Lectura del GPIO 21 para el circuito del timbre.....                         | 29 |



# 1.- Introducción

## 1.1 Introducción

La electrónica es uno de los aspectos de la ingeniería con una mayor evolución en las últimas décadas. A lo largo de estos años, se han producido numerosos cambios en la sociedad moderna por los avances electrónicos que han sucedido. Estos avances, desde mediados del Siglo XX, han mejorado y optimizado el funcionamiento de numerosas máquinas y procesos. Gradualmente, elementos mecánicos han sido sustituidos por componentes electrónicos [1]. Inicialmente, el desarrollo de la electrónica tuvo lugar en el ámbito analógico, pero en los últimos años se ha dado una gran evolución de la electrónica digital y el manejo de datos en tiempo discreto.

En el año 1975, el ingeniero Gordon Moore, cofundador de Intel, predijo que el número de transistores presentes en un microprocesador se duplicaría cada dos años. Esta predicción difiere muy poco de los datos obtenidos en este aspecto hasta la fecha [2]. Este progreso en el número de transistores por microprocesador ha permitido un gran avance en el desarrollo de componentes electrónicos más potentes. Unido a la reducción en los precios de los transistores, la electrónica ha ido evolucionando en diferentes sectores del mercado.

La domótica es la parte de la electrónica aplicada a las viviendas. Los sistemas domóticos tienen la función de regular, monitorizar, o automatizar procesos que ocurren en una vivienda, como la subida o bajada de persianas, la iluminación o la calefacción [3]. Las mejoras en la electrónica en general, y en la domótica en particular, tienen además un impacto medioambiental (especialmente en el aspecto energético) reducido, respecto a las viviendas tradicionales [4].

Uno de los aspectos con un mayor desarrollo en la domótica es el de los porteros electrónicos. Las comodidades que aportan y la variedad de métodos de acceso a las viviendas que se están desarrollando son un gran incentivo para los clientes que buscan integrar estas nuevas tecnologías en su hogar. En este proyecto, se propone el desarrollo de un sistema domótico que permita el acceso a la vivienda de forma remota.

Para ello, se ha utilizado una Raspberry Pi 3B+ como microprocesador, con un módulo GSM (SIM800C) para comunicarse con el usuario a través de correo electrónico y SMS. Además, para implementar el proyecto en su totalidad, será necesaria una cámara, un pulsador (timbre), y un conjunto batería-relé que actúa como cierre eléctrico de la puerta.

## 1.2 Estado del arte

En años recientes, los avances en domótica han sido muy numerosos; la investigación y el desarrollo de productos en esta área es cada vez más extendida y existen numerosas soluciones parciales al problema que se plantea.

Por una parte, existen varios tipos de cámaras que se pueden conectar a un Smartphone a través de una aplicación asociada a la propia cámara. Esto permite el acceso visual a la zona que está grabando la cámara en cualquier momento desde el propio Smartphone, siempre que se disponga de conexión a Internet. Un ejemplo muy conocido de esto son las conocidas “vigilabebés”, que graban en todo momento al niño para que los padres puedan estar pendiente de él sin estar en la misma habitación. Pero en este campo también se han desarrollado cámaras de exterior, con resistencia a cambios medioambientales y con funciones como giro 360º, visión nocturna, o detección de movimiento [5].

Por otra parte, también existen módulos GSM que permiten la apertura de puertas de casas y garajes de forma remota a través de llamada o SMS [6]. Generalmente llevan integrado un microcontrolador y un módem GSM y funcionan como interruptores que solo pueden ser operados mediante llamada o SMS de números de teléfono autorizados previamente. Otros métodos de apertura sin llave incluyen reconocimiento facial, por huellas dactilares, o través de algún panel con código preestablecido [7].

## 1.3 Motivación

Hay varias conclusiones a las que se puede llegar a partir del apartado previo sobre el estado de la cuestión actualmente. Una de las primeras conclusiones podría ser que es un campo en el que ya se están dando grandes avances y hay una gran competencia en el desarrollo de producto: algunas empresas ya han lanzado sus productos, con más recursos tanto de fabricación como de marketing.

Sin embargo, la motivación para la realización de este proyecto viene de un análisis algo más cercano al cliente. Los artículos de domótica deben tener como primer objetivo funcional la practicidad: deben ser fáciles de usar y tener un objetivo claro, sin nunca descuidar la seguridad de la vivienda.

El desarrollo del sector, como he mencionado previamente, es uno de gran competitividad y especialización. Precisamente por esto, lo más común para un cliente es poder encontrar cada uno de los componentes de este proyecto por separado. Es posible instalar una cámara que se puede controlar desde el móvil, que envíe alertas de movimiento e imagen en directo, para luego poder utilizar por separado una llamada telefónica para operar la puerta. Sin embargo, no es tan fácil encontrar un artículo que cumpla todas estas funciones integradas en un único producto fácil de usar.

Por consiguiente, la motivación de este proyecto es crear un único sistema domótico que integre estos componentes de forma que sea fácil de usar y requiera una intervención mínima por parte del usuario (solo la necesaria).

## 1.4 Objetivos

Los principales objetivos del proyecto son:

1. Reaccionar a la llamada al timbre (accionamiento del pulsador) tomando una imagen para enviar al dueño de la vivienda.
2. Enviar la imagen al dueño de la vivienda y esperar a su respuesta por SMS, que deberá ser un código alfanumérico de 4 caracteres con sensibilidad mayúscula/minúscula.
3. Permitir la apertura del cierre eléctrico solo cuando se ha recibido un SMS nuevo, se ha verificado que el SMS proviene del dueño de la vivienda (por

su número de teléfono), y se ha comprobado que el texto del SMS se corresponde con el código alfanumérico preestablecido.

4. Que la estructura del proyecto concluya de forma que se faciliten futuras actualizaciones y mejoras del software.

## 1.5 Estructura del documento

El documento se estructura en cinco capítulos. El primero consiste en una introducción, que detalla el estado del arte y los objetivos del proyecto. El segundo capítulo desarrolla los componentes hardware elegidos para el prototipo y su funcionamiento. La tercera sección indaga en el funcionamiento del hardware implementado para el funcionamiento del proyecto, y se desarrolla la lógica de operación de cada componente. La sección de resultados muestra las pruebas realizadas hasta la obtención del resultado final y la monitorización realizada a lo largo del desarrollo del proyecto. El capítulo final, de conclusiones, explica el proceso de desarrollo del producto y las dificultades afrontadas, y sienta las bases para futuros desarrollos y posibles mejoras del prototipo.

## 2.- Hardware

### 2.1.- Estructura general

Los elementos que componen el hardware del sistema son:

- **Microcontrolador:** Es el centro del proyecto, porque determina los estados de funcionamiento de cada componente interpretando las entradas que recibe para decidir lo que debe hacer el producto en todo momento. El microcontrolador elegido es una Raspberry Pi 3B+.
- **Módulo GSM:** Es el componente encargado del envío del e-mail y la recepción del SMS.
- **Cámara:** Se encarga de tomar la imagen de la persona que haya accionado el timbre. Para este proyecto, se ha elegido la cámara de Raspberry.
- **Cierre eléctrico:** Es el cierre de la puerta, que se debe accionar cuando se verifique que el código recibido es válido.
- **Timbre:** Un pulsador cuya función es generar una señal que inicie el proceso diseñado.

En la Figura 2.1 se muestra un esquema de las conexiones entre los distintos componentes del prototipo final.

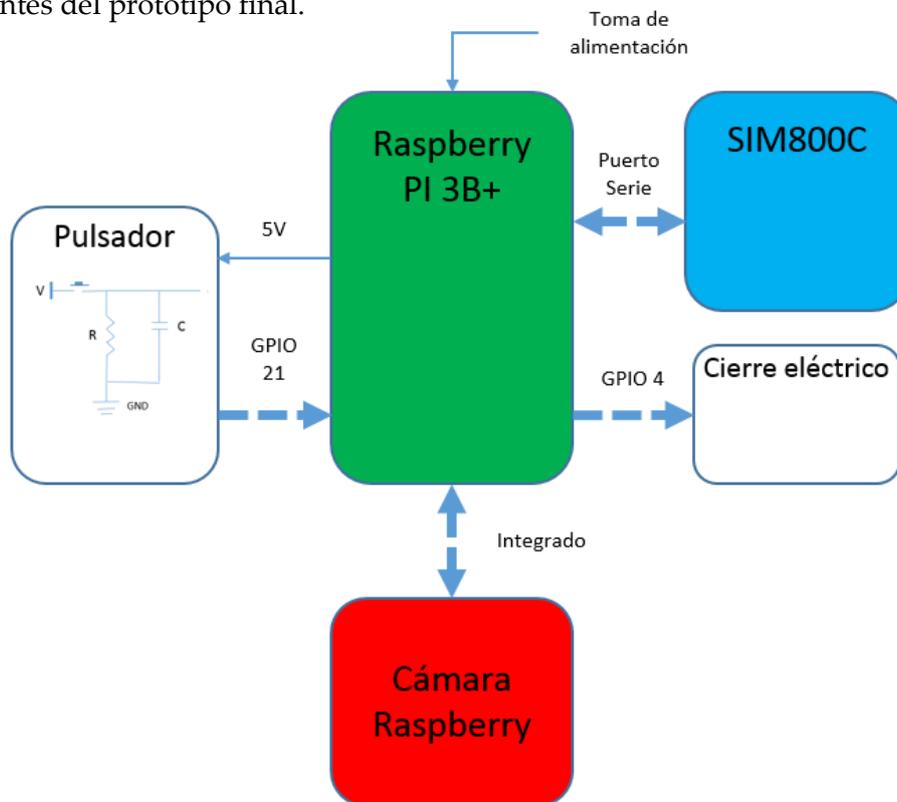


Figura 2.1: Esquema de conexión del hardware. Fuente: Elaboración propia.

## 2.2.- Microcontrolador

El microcontrolador elegido para este proyecto es una Raspberry Pi 3B+, el último modelo de este tipo de microprocesadores de la marca Raspberry, exceptuando el modelo 4B, anunciado en Junio de 2019.

La primera Raspberry fue comercializada en 2012, con una gran demanda en todo el mundo puesto que daba la opción a aficionados a la electrónica a desarrollar sus propios proyectos desde casa. Posteriormente, en los años 2014, 2016 y 2018, salieron al mercado las versiones 2B, 3B y 3B+ respectivamente, todas ellas mejorando a su predecesora en aspectos como el procesador, la RAM, los puertos USB, o la conectividad Ethernet. [8]



Figura 2.2: Raspberry Pi 3B+. Fuente: [8]

El modelo 3B+ cuenta con un procesador de 1.4Ghz, Wifi, Bluetooth, Ethernet mejorado, 4 puertos USB, HDMI, y la posibilidad de acoplar una cámara de Raspberry, como se muestra en la Figura 2.1 [8] [9]. Es un microprocesador adecuado para este proyecto porque cumple con los requisitos mínimos (puerto serie para utilizar el módulo GSM, capacidad para conectar a internet, y opción para acoplar una cámara), y ofrece una buena calidad a un bajo precio (33,85€) [10]. Además, es compatible con el software MatLab/Simulink utilizado en este proyecto a través de un paquete de bloques y comandos adicional disponible de forma gratuita [11].

## 2.3.- Módulo GSM

El GSM (Sistema Global de comunicaciones Móviles), conocido también como 2G, es una red que opera en numerosos países, basada en el uso de tarjetas SIM (siglas del inglés Subscriber Identity Module). Cada SIM tiene un número de teléfono asociado, que actúa como identificador del dispositivo al que conecta a la red GSM [12] [13].

### 2.3.1.- SIM800C

El módulo GSM elegido para este proyecto es el SIM800C, conectado a la Raspberry a través del puerto serie mediante una versión adaptada del módulo (Sección 2.3.2). Cuenta con capacidad para enviar y recibir SMS en modo texto, MMS, y conexión GPRS.

Además, la banda de frecuencia a la que opera la red GSM difiere en distintos países (en Europa 900/1800 MHz, en Estados Unidos 1900 MHz) [12], y este módulo es compatible con todas ellas [14].

El módulo elegido para adaptarse a la Raspberry permite además la alimentación eléctrica a través de esta conexión, y dispone de una ranura para tarjeta SIM fácilmente accesible. El conjunto GSM/Raspberry/cámara ocupa así el mínimo espacio posible.

### 2.3.2.- Comunicación serie

El módulo está conectado a la Raspberry mediante el protocolo de conexión serie UART. La Raspberry dispone de los pines Tx y Rx, que se utilizan para enviar y recibir comandos al dispositivo conectado por el puerto serie. En este caso, la comunicación se realizará a través de comandos de Hayes (o comandos AT), explicados en mayor profundidad en el apartado 2.3.3.

Las instrucciones enviadas desde el microcontrolador se transmiten por el pin Tx, y la respuesta del módulo GSM (errores, fallos de conexión, mensajes no leídos, etc.) se recibe por el pin Rx. En ambos casos, la transmisión se realiza en ASCII. Simulink dispone de bloques que permiten esta comunicación por el puerto serie, en los cuales se debe especificar ciertos parámetros como la paridad, el número de bits por palabra, el baudrate (bits por segundo) o el puerto que se conecta. En el modelo 3B, el puerto serie es el /dev/ttyS0 [15].

### 2.3.3.- Comandos de Hayes

Los comandos de Hayes (también conocidos como contactos AT, abreviatura de "attention"), se utilizan para interactuar con módulos GSM y Módems GPRS para utilizar los servicios de llamada, SMS o e-mail [16].

Se caracterizan por dar instrucciones precedidas siempre de "AT", llamando así la atención del dispositivo con el que se desea establecer una conexión. Al término de cada instrucción, el dispositivo responde de forma estandarizada, con un "OK", si el comando se ha ejecutado correctamente, un "ERROR", si ha habido algún fallo en la ejecución del comando, y, si aplica, la información relevante que se ha solicitado al módulo (por ejemplo, el texto del último SMS recibido). Mediante estos comandos es

posible introducir el PIN para desbloquear una SIM, enviar SMSs, MMSs, e-mails, leer los mensajes recibidos, eliminar mensajes, y muchas otras funciones que no se utilizan en este proyecto.

Para la implementación de este proyecto, se utilizan los siguientes comandos de Hayes:

- AT: Llama la atención del módulo y permite enviar los comandos sucesivos. El módulo responde con un "OK", indicando que está activo.
- AT+CPIN: Permite introducir el PIN para desbloquear la tarjeta SIM.
- AT+CMGF: Permite activar el modo texto para la función de SMS.
- AT+CMGD: Elimina los mensajes de texto seleccionados.
- AT+CMGL: Permite leer los SMS que se hayan recibido que cumplan las características especificadas.
- AT+CPOWD: Se utilizar para reiniciar el módulo GSM.
- AT+SAPBR: Utilizado para establecer los parámetros de GPRS y APN.
- AT+EMAILCID, AT+EMAILTO: Parámetros de inicialización del e-mail.
- AT+SMTPSRV: Parámetros del servidor de correo electrónico.
- AT+SMTPAUTH: Usuario y contraseña del remitente del e-mail.
- AT+SMTPFROM: Información de procedencia del correo electrónico al destinatario.
- AT+SMTPRCPT: Información sobre el/los destinatario/s del e-mail.
- AT+SMTPSUB: Asunto del e-mail
- AT+SMTPBODY: Determina el número de caracteres que formarán parte del cuerpo del e-mail. Los caracteres deben enviarse a continuación.
- AT+SMTPFILE: Indica la existencia de un archivo en el e-mail y su tipo.
- AT+SMTPSEND: Inicia el proceso de envío del correo
- AT+SMTPPFT: Se utiliza para indicar el tamaño del archivo adjunto, que debe enviarse acto seguido. En la forma AT+SMTPPFT=0 indica que se ha finalizado el proceso de adjuntar archivos al e-mail.

## 2.4.- Cámara

La elección de la cámara en este proyecto se fundamenta en tener un producto lo más compacto posible. El módulo de la cámara que se integra por la ranura dedicada de la Raspberry Pi 3B+ es idónea para ello, por ser muy compacto y tener una calidad que cumple con los requisitos del proyecto.

Su peso es inferior a los 5 gramos, tiene una forma cuadrada de 2,5 cm de lado, 8 Megapíxeles y un precio de 9€. La adaptación del módulo GSM a la Raspberry deja también abierta la ranura de la Raspberry para conectar la cámara por lo que queda perfectamente integrada [17].

Para operar con la cámara, existe un bloque en la librería adicional de Raspberry para Simulink que permite la grabación de vídeo en directo. El bloque tiene como parámetro la resolución de la imagen, y tiene como salida la imagen en formato RGB (3 señales separadas, matrices de dimensiones correspondientes a la resolución (Dxd)). Para generar una única señal con toda la información de la imagen en formato RGB, se genera una matriz F de dimensiones Dxdx3, donde, por ejemplo, el valor de F (100, 100, 1) corresponde al valor del color rojo en el píxel 100,100 de la imagen (valores entre 0 y 1 [18]).

Este bloque proporciona las 3 señales mencionadas como vídeo en directo, es decir, actualiza constantemente los valores de la matriz a lo que está grabando la cámara en ese momento. Para obtener una imagen, basta con tomar un conjunto de valores de las matrices en el instante en el que se acciona el timbre, y guardarlo en un Data Store hasta que se envía al módulo GSM el comando para incluir archivos adjuntos en el e-mail.

Sería posible utilizar otras cámaras para realizar la misma función puesto que el bloque de Simulink tiene un parámetro que, por defecto, toma como cámara el puerto adaptado para la de Raspberry, pero que se puede modificar en caso de utilizar una cámara externa.

## 2.5.- Cierre eléctrico

Los cierres eléctricos se componen de un relé y una batería que operan normalmente a una tensión de entre 10 y 24 V y se activan a través de impulsos recibidos externamente, en este caso, de la Raspberry. Los componentes más baratos que se pueden encontrar para cumplir con los objetivos del proyecto rondan los 40€. Al no ser necesario realizar ninguna prueba sobre estos componentes, y quedar su funcionamiento determinado por la salida de un puerto de la Raspberry, no se realizó la compra de estos componentes para el montaje final del prototipo, y su papel queda modelado en Simulink con un bloque que permite visualizar su estado: 0 cuando el conjunto deba mantenerse apagado, y 1 cuando deba activarse para que la puerta se abra.

## 2.6.- Timbre

Este componente consiste en un pequeño circuito para cuyo montaje se utiliza una placa externa. Los componentes necesarios son una fuente de tensión  $V$  (5V, de la Raspberry), una resistencia  $R$  (10 k $\Omega$ ) y un condensador  $C$  (10nF). Utiliza además el Pin GND de la Raspberry, una entrada digital, y el cableado necesario para el montaje (Figura 2.3).

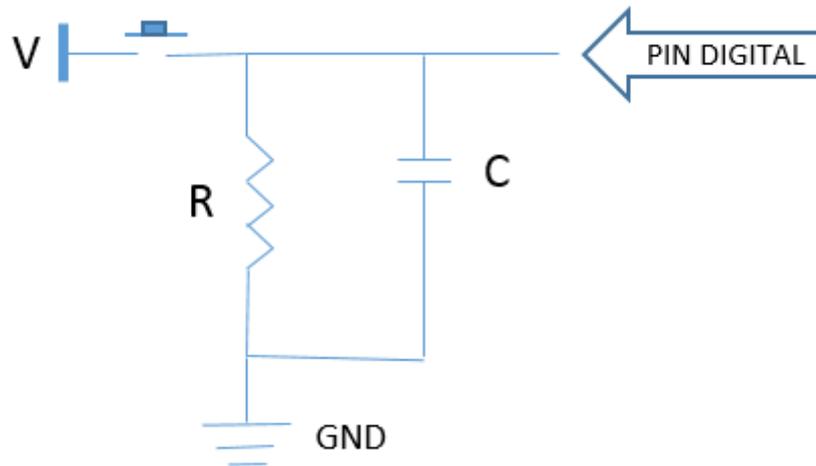


Figura 2.3: Circuito correspondiente al timbre. Fuente: Elaboración propia

La corriente teórica por el pulsador se mantendrá entre 0 y 0.5 mA. El valor mínimo corresponde al régimen permanente con el pulsador abierto, y el valor máximo corresponde al régimen permanente con el pulsador cerrado. Los valores de intensidad son adecuados para todos los componentes del circuito.

## 3.- Software

### 3.1.- Estructura general.

Las señales de entrada y salida de las máquinas de estado se almacenan en bloques de Simulink llamados “Data Store”, que permiten reservar espacios de memoria para estas variables. La Figura 3.1 muestra la estructura de estas señales en el software final.

Es posible leer o escribir en estos bloques en todos los subsistemas que queden por debajo del emplazamiento del bloque de memoria.

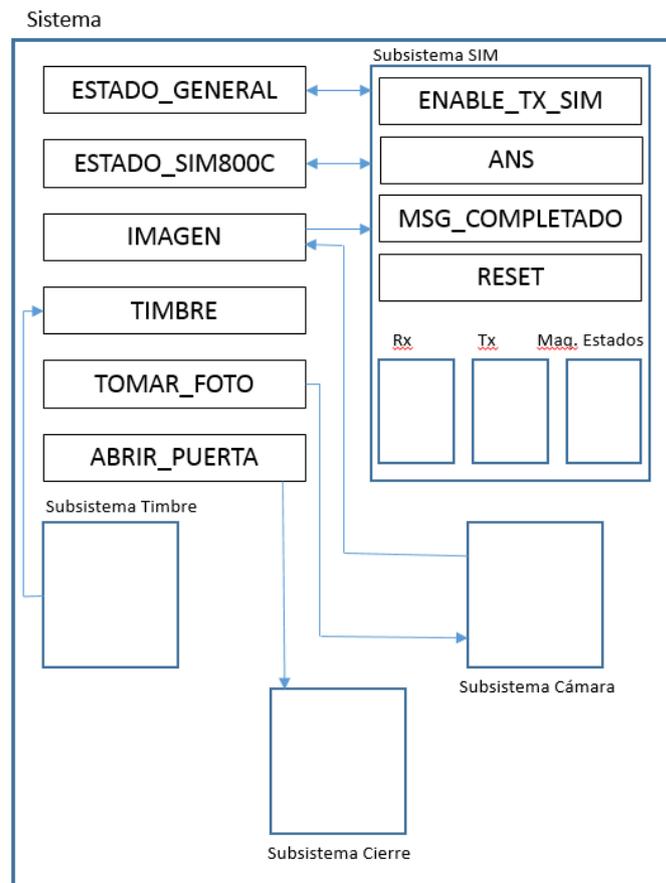


Figura 3.1: Estructura del software. Fuente: Elaboración propia

El sistema contiene una máquina de estados que coordina el funcionamiento global de todos los subsistemas (Figura 3.6), mediante las variables guardadas en los espacios de memoria establecidos en el nivel más alto.

- **ESTADO\_GENERAL:** Es el estado actual de la máquina de estados general, que se actualiza en cada ciclo.

- **ESTADO\_SIM800C:** Es el estado actual de la máquina de estados del módulo GSM.
- **IMAGEN:** Almacena la imagen más reciente tomada por la cámara. No se utiliza en la máquina de estados general, pero debe estar ubicada en el nivel más alto porque su información se escribe en el subsistema de la cámara, cuya función es tomar la fotografía, pero se utiliza en el subsistema Tx del módulo GSM como archivo adjunto al e-mail.
- **TIMBRE:** Se determina por lectura digital de un PIN y se utiliza como transición del estado de reposo al inicio del proceso.
- **TOMAR\_FOTO:** Es una salida de la máquina de estados que se activa en un único ciclo para tomar la fotografía. Es necesaria porque la única interacción de Simulink con la cámara es imagen en directo.
- **ABRIR\_PUERTA:** Es una salida de la máquina de estados que se activa durante 5 segundos si se recibe el código correcto.

Además, el módulo GSM tiene una máquina de estados propia cuya función es determinar el comando a enviar por el puerto serie y, en el momento adecuado, leer y procesar el mensaje recibido con el código. Para ello, utiliza las siguientes señales adicionales.

- **ENABLE\_TX\_SIM:** Determina el comando que se envía por el puerto serie.
- **ANS:** Contiene las respuestas recibidas por el módulo SIM.
- **MSG\_COMPLETADO:** Notifica a la máquina de estados que el buffer de lectura está completo (ANS es una respuesta completa)
- **RESET:** Vacía el buffer de lectura. Se utiliza en dos situaciones: cuando no es necesario escuchar a las respuestas del módulo GSM, o cuando se ha llenado el buffer y es necesario vaciarlo para poder seguir leyendo información.

### 3.2.- Módulo GSM

La transmisión de información pertinente a los comandos AT se produce a través del puerto serie (Sección 2.3). El envío de estos comandos debe realizarse en un orden específico, y el conjunto de comandos a enviar depende también de si se desea mandar un e-mail o leer el código recibido por SMS. Por ello es necesaria una máquina de estados secundaria del módulo GSM que se activa cuando el sistema general necesita usar el módulo (Figura 3.3)

Los comandos deben ser enviados en ASCII y deben ir seguidos de un 13, que indica el final de una instrucción. Las respuestas del módulo a las instrucciones se reciben también en ASCII, y vienen tanto precedidas como seguidas de la secuencia 13 10 (Carriage Return / New Line). Esto es especialmente relevante al interpretar el código, puesto que, por ejemplo, el código '1 2 3 4' en ASCII se lee como '49 50 51 52'. Para interpretar el mensaje recibido y decidir si se debe abrir la puerta o no, es necesario, o bien traducir el código a ASCII, o decodificar el mensaje recibido en el buffer. Además, las instrucciones enviadas deben ser del mismo tamaño, por lo que es necesaria la función del buffer de envío de instrucciones, que rellena cada instrucción a enviar con ceros.

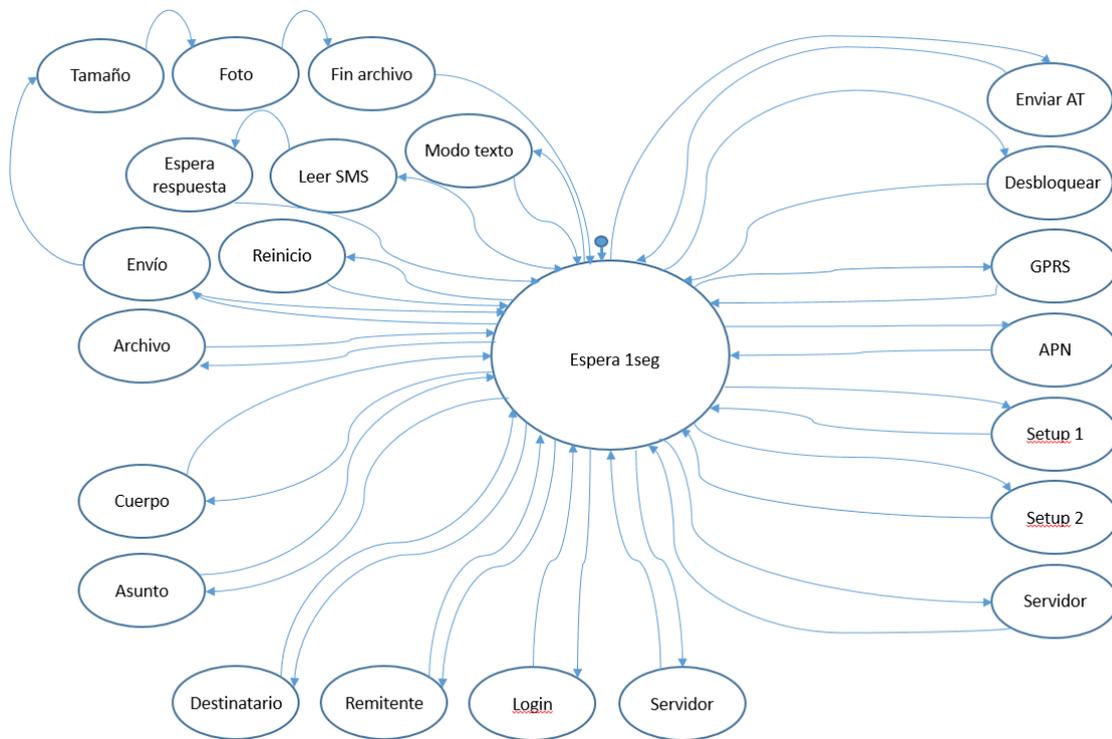


Figura 3.2: Máquina de estados módulo GSM. Fuente: Elaboración propia

La Figura 3.2 esquematiza la máquina de estados y su funcionamiento general. Su funcionamiento es el siguiente.

- **ESPERA 1 SEG:** Es un estado que tiene dos funciones principales. Por una parte, es el estado por defecto del sistema cuando no se está utilizando la SIM. Por otra parte, sirve como tiempo de espera entre el envío de cada comando. Esto hace que sea necesario una variable "SIGUIENTE\_MENSAJE", que determina a qué estado se debe pasar una vez que pasa 1 segundo. En la primera situación, vale 0, puesto que la

transición al principio del proceso viene forzada por la máquina de estados global. En el segundo caso, es el valor del estado correspondiente a la siguiente instrucción.

- **ENVIAR AT:** Envía el comando AT, que despierta al módulo
- **DESBLOQUEAR:** Utiliza el comando AT+CPIN=6588 para permitir utilizar las funciones de la tarjeta SIM.
- **GPRS:** Envía el comando AT+SAPBR=3,1,"Contype","GPRS" para establecer ajustes de GPRS.
- **APN:** Envía el comando AT+SAPBR=3,1,"APN","movistar.es" para establecer ajustes APN.
- **SETUP 1:** El comando AT+EMAILCID=1 establece el ajuste de conexión del e-mail a lo establecido previamente.
- **SETUP 2:** El comando AT+EMAILTO=30 establece el valor de timeout del servidor en segundos. 30 es el valor usado por defecto.
- **SERVIDOR:** Establece los ajustes del servidor utilizado para enviar el email, en este caso, Gmail. El comando enviado es AT+SMTPSRV="smtp.gmail.com",587.
- **LOGIN:** Envía al módulo la información de inicio de sesión de la cuenta con la que se debe mandar el e-mail. El comando es AT+SMTPAUTH=1,"elimbretfg@gmail.com","password"
- **REMITENTE:** Se introduce la información del remitente asociada al e-mail mediante el comando AT+SMTPFROM ="elimbretfg@gmail.com" , "MiTimbre"
- **DESTINATARIO:** Determina la dirección a la que se debe enviar el e-mail usando el comando AT+SMTPRCPT=0,0, "201503601@alu.comillas.edu". Es posible establecer más de un destinatario, y poner destinatarios en copia utilizando este comando varias veces.
- **ASUNTO:** AT+SMTPSUB= "Asunto" establece el asunto del e-mail.
- **CUERPO:** AT+SMTPBODY=25 establece el número de caracteres que incluye el cuerpo del e-mail. Estos caracteres se deben transmitir acto seguido.
- **ARCHIVO:** AT+SMTPFILE=2, "foto.jpg",1 indica la existencia de un archivo adjunto llamado "foto.jpg", su tipo (binario en este caso), y el tipo de codificación (que depende del tipo de archivo a enviar).

- **ENVÍO:** Se transmite el comando AT+SMTPSEND, que inicia el proceso de envío del e-mail.
- **TAMAÑO:** Se transmite el comando AT+SMTPFT que indica el tamaño del archivo a transmitir. Directamente se procede al estado siguiente para, acto seguido, enviar todos los bits del archivo.
- **FOTO:** Se transmite la imagen más reciente que existe en la memoria.
- **FIN ARCHIVO:** El comando AT+SMTPF=0 se utiliza para indicar al módulo que ya se ha transmitido toda la información de archivos adjuntos y se puede finalizar el envío.
- **MODO TEXTO:** Se envía el comando AT+CMGF=1 para utilizar el modo texto en el SMS.
- **LEER SMS:** En este estado se envía el comando solicitando al módulo la lectura de los mensajes nuevos que tenga en memoria. Para ello se transmite el comando AT+CMGL="REC UNREAD".
- **ESPERA RESPUESTA:** En este estado, se espera a recibir una respuesta completa del módulo GSM. Se permanece en este estado mientras no se haya recibido una respuesta completa, o mientras la respuesta recibida sea a un comando distinto al anterior. Se sale de este estado para volver al anterior cuando no se detecten mensajes nuevos. Se sale de este estado hasta el siguiente ciclo cuando se detecta un nuevo SMS y se realizado la transición de la máquina de estados general.
- **REINICIO:** Se utiliza el comando AT+CPOWD=1 para reiniciar el módulo GSM. Esto se hace para vaciar el buffer del puerto serie.

Durante todo el proceso, es necesario que el módulo GSM realice dos funciones claramente diferenciadas: enviar un e-mail y recibir un SMS. El envío del e-mail consiste en el envío de ciertas instrucciones de forma consecutiva, y no requiere atención a la respuesta del módulo GSM siempre que no se reciba ningún error. En cambio, la recepción del SMS sí que necesita leer la información que recibe la Raspberry por el puerto serie como parte vital para su funcionamiento.

En realidad, la única respuesta cuya información hay que almacenar es la que transmite la información de los SMS recibidos. Sin embargo, por el funcionamiento de la máquina de estados, que en cada estado envía una instrucción de tamaño fijo, cuando se llega al estado de lectura de mensajes, la mayoría de las instrucciones se encuentran aún en espera a ser transmitidas por el puerto serie. Por lo tanto, el estado correspondiente a la espera de la recepción de un SMS tiene una estructura más compleja que los demás.

Para importar la información, se utiliza un buffer (Figura 3.4) que detecta cuando el módulo comienza a transmitir información por el puerto y rellena “mensajes” (conjuntos de datos de un tamaño preestablecido para poder almacenar la respuesta más larga esperada). Cuando se completa un mensaje (se rellena el buffer), la máquina de estados general determina si la respuesta se corresponde con el comando AT+CMGL. En caso contrario, se continúa en este estado esperando al comando. Cuando la máquina detecta que ha llegado la respuesta a este comando, existen dos posibles respuestas: que aún no se detecte una respuesta o que haya un mensaje nuevo. En el primer caso (la respuesta recibida es ‘OK’), la máquina de estados vuelve al estado anterior (envía el comando de lectura de mensajes nuevos). En el segundo caso, se extrae la información correspondiente al texto recibido y el número de teléfono del que proviene, y se determina el siguiente estado de la máquina de estados general (Figura 3.3)

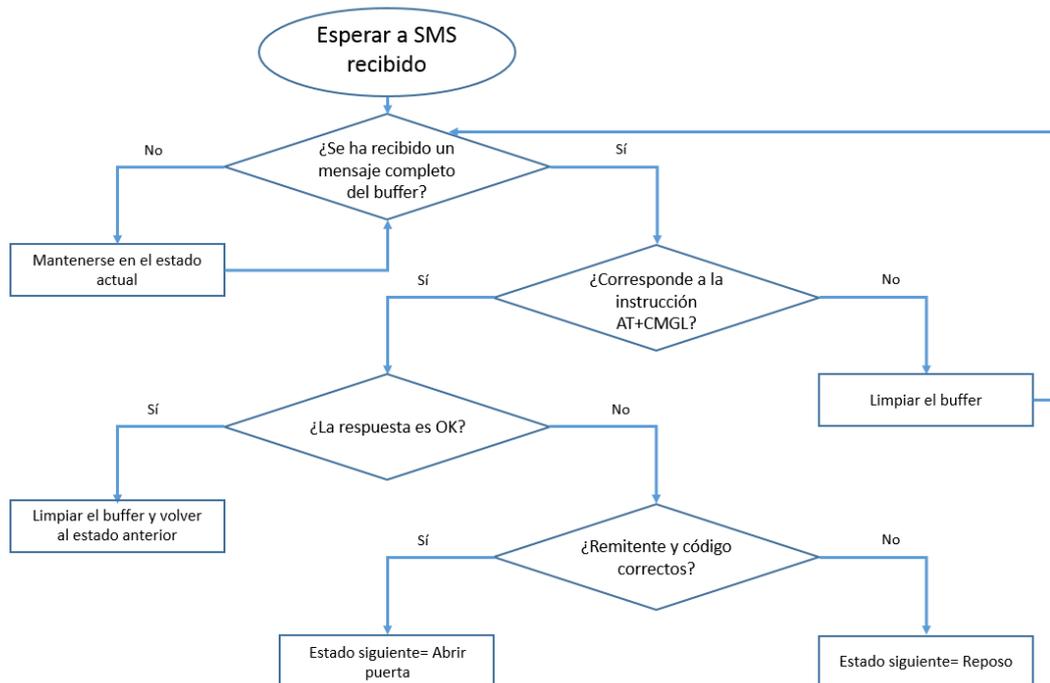


Figura 3.3: Lógica del estado “Esperar SMS”. Fuente: Elaboración propia.

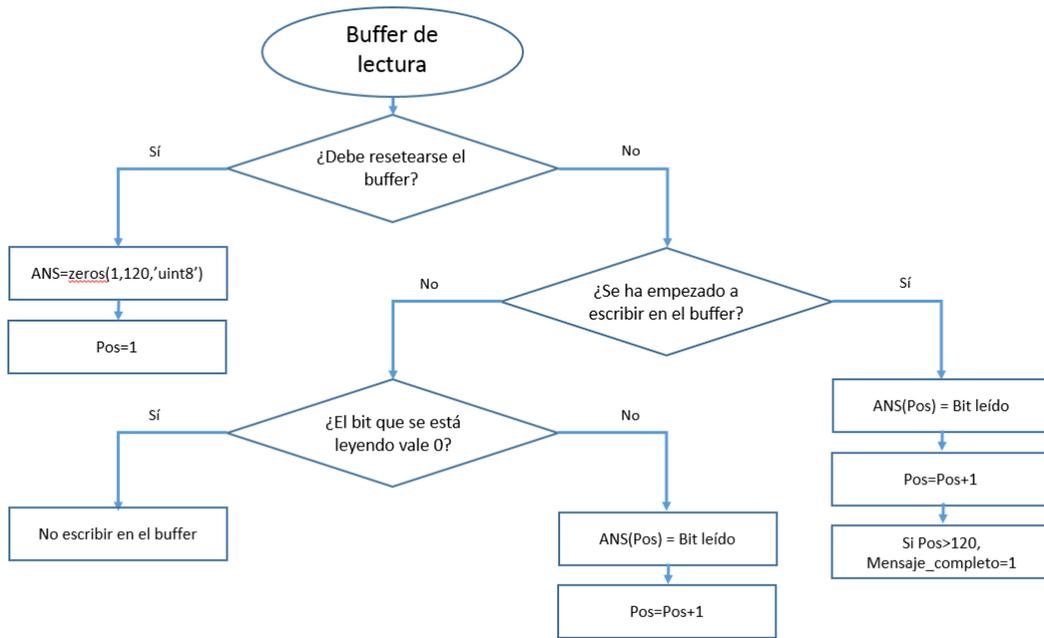


Figura 3.4: Funcionamiento del buffer de lectura. Fuente: Elaboración propia.

### 3.3.- Cámara Raspberry.

Como se explica en la Sección 2.4, el bloque del paquete de compatibilidad de Raspberry para Simulink tiene como salida 3 señales correspondientes a las matrices R, G y B de la imagen en cada instante.

Para obtener una señal de imagen en formato RGB, se puede escribir una función con 3 entradas (R, G, B), y una salida (imagen). La función genera una matriz  $m \times n \times 3$  a partir de tres matrices  $m \times n$ , donde  $(m, n, 1) = R$ ,  $(m, n, 2) = G$  y  $(m, n, 3) = B$ . Esta matriz estará actualizándose en cada momento. Como se desea tomar tan solo una imagen de la persona que ha accionado el timbre, esta matriz solo debe ser guardada en el bus dedicado a su almacenamiento una vez, cuando se pulse el timbre. Es decir, los contenidos del bus tan solo se modifican en el estado dedicado a tomar la foto de la máquina de estados general (Figura 3.5)

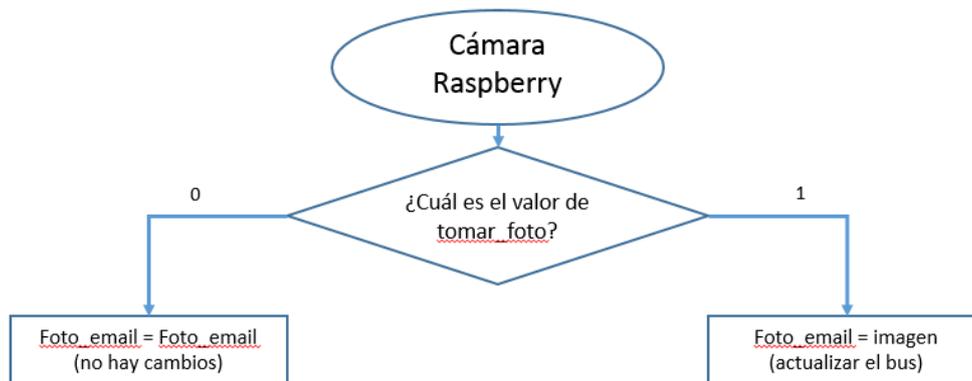


Figura 3.5: Diagrama de flujo de software de la cámara. Fuente: Elaboración propia

### 3.4.- Timbre

Para la instalación del timbre, se montó el circuito mostrado en la Figura 2.3, consistente en la fuente de tensión de la propia Raspberry conectada en serie con un pulsador y una resistencia y cerrando el circuito el tierra. De esta manera, cuando el pulsador está accionado, se lee 1 en la entrada digital, y cuando no se está pulsando, se lee 0, al no haber corriente por la resistencia.

El condensador tiene como función evitar el “efecto rebote” provocado por los elementos mecánicos que forman parte del pulsador. El circuito funcionaría adecuadamente sin él, puesto que una vez se activa el timbre, se inicia un proceso que no necesita leer esta entrada digital hasta un tiempo muy superior al del efecto rebote. Sin embargo, esta implantación es mejor y viable para cualquier uso que se quiera dar a este pulsador en futuros desarrollos.

### 3.5.- Conjunto batería/cierre

Como se explica en la Sección 2.5, los componentes del hardware de este componente no se montaron en el prototipo final. Consecuentemente, en lugar de estar representados en el modelo de Simulink como un puerto de la Raspberry, sus valores se monitorizan en modo externo con un Scope para verificar que se activan únicamente en el estado asignado para la apertura de la puerta.

### 3.6.- Máquina de estados general.

La máquina de estados global del sistema se encarga de coordinar el funcionamiento de todas las partes de sistema, y de activar o desactivar cada componente del proceso en los momentos adecuados. Su estructura se muestra en la Figura 3.6.

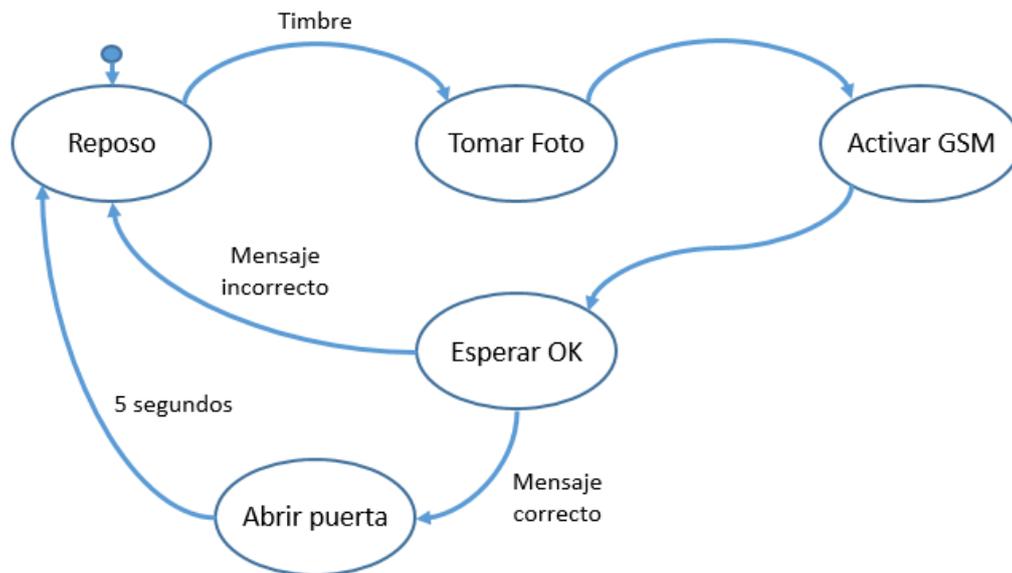


Figura 3.6: Máquina de estados general. Fuente: Elaboración propia

- **REPOSO:** Este estado comprueba si es necesario iniciar el proceso para una posible apertura remota de la puerta. Mientras el timbre se mantenga sin pulsar, no se produce ninguna transición. Al accionarse el timbre, se procede al estado en el que se toma la foto.
- **TOMAR FOTO:** Se actualiza el bus en el que se almacena la foto que debe ser enviada al dueño de la vivienda. En cuanto este bus ha sido sobrescrito, se procede al siguiente estado, enviar email.
- **ACTIVAR GSM:** Se activa la máquina de estados del módulo GSM (Sección 3.2) y se procede al siguiente estado.
- **ESPERAR OK:** Se espera a la finalización del proceso de enviar el email y recibir la respuesta por SMS. Una vez recibido el código, la máquina de estados de la SIM habrá determinado si se debe proceder a abrir la puerta o volver al estado inicial de reposo.
- **ABRIR PUERTA:** Solo se debe acceder a este estado cuando el código recibido sea el correcto y provenga del número de teléfono preestablecido. Una vez en él, se activa la batería que acciona el cierre eléctrico durante un tiempo que se considere suficiente (5 segundos) y al finalizar este contador, se vuelve al estado original (se espera a una nueva llamada al timbre)



## 4.- Resultados

### 4.1.- Módulo GSM

Para comprobar el funcionamiento del módulo GSM utilizado, se realizaron numerosas pruebas hasta llegar al funcionamiento final. El primer paso es el funcionamiento correcto de la comunicación por el puerto serie. El test consiste en mandar el comando AT y leer la respuesta del módulo. Si todo funciona como debería, la respuesta obtenida en el Scope al simular en modo externo debe ser 13 10 'OK' 13 10, como se muestra en la Figura 4.1 (donde 'OK' viene dado por su equivalente 79 75 en ASCII) [19].



Figura 4.1: Respuesta al comando AT. Fuente: Scope de Simulink

La secuencia de comandos enviados en la simulación de la versión final del prototipo es mucho más extensa, pero es conveniente enfatizar en la respuesta al comando `AT+elquerecibaSMSnoleido`, puesto que su estructura es clave en la formulación de la máquina de estados de la Sección 3.2 (Figuras 4.2 y 4.3).

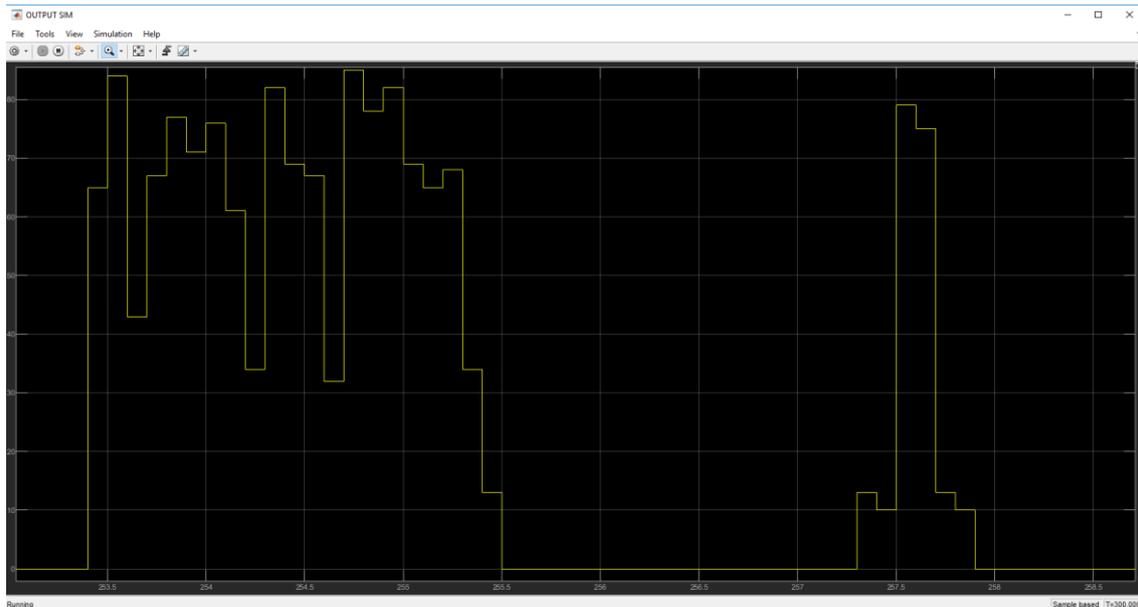


Figura 4.2: Respuesta al comando AT+CMGL="REC UNREAD" sin mensajes nuevos. Fuente: Scope de Simulink

La Figura 4.2 muestra la respuesta cuando no se encuentra ningún mensaje con los criterios especificados ('OK'). En el proyecto, cuando se recibe esta respuesta, la máquina de estados interpreta que aún no se ha recibido el mensaje con el código, y que por lo tanto debe seguir esperando hasta recibirlo.

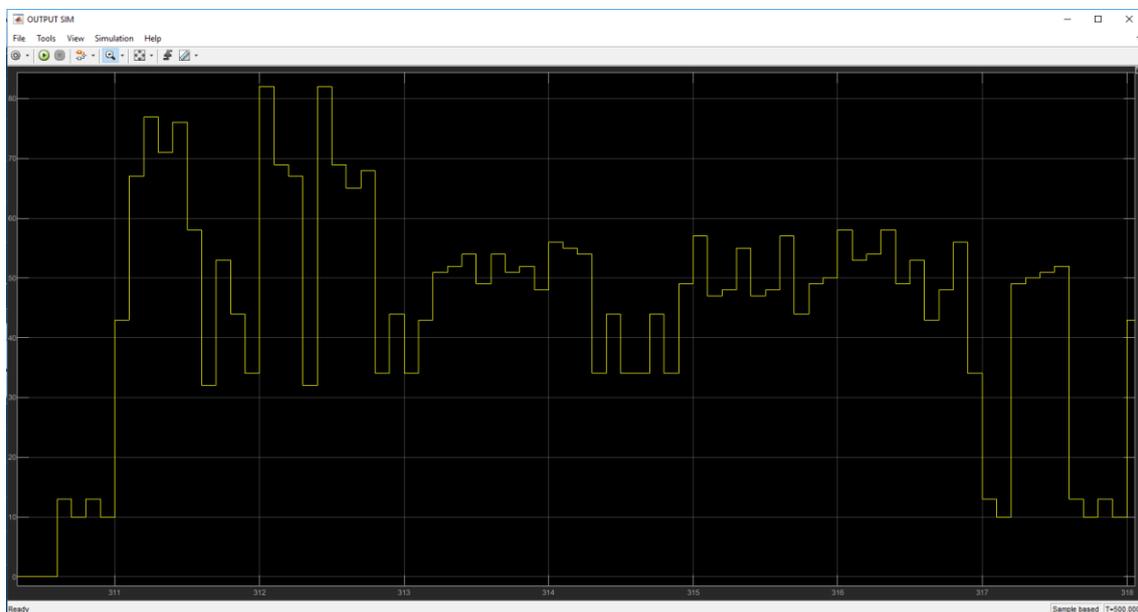


Figura 4.3: Extracto de la respuesta al comando AT+CMGL="ALL". Fuente: Scope de Simulink.

La Figura 4.3 muestra la respuesta cuando se encuentra algún mensaje. La información que proporciona es el índice de memoria en el que está guardado el mensaje (en este caso, 5), el estado del mensaje (“REC READ”, que significa que el mensaje ya ha sido leído), número de teléfono del remitente (incluye código de país), fecha (formato aa/mm/dd), hora (hh/mm/ss), zona horaria (donde el valor se expresa en cuartos de hora, es decir, +8 corresponde a GMT+2, la zona horaria en España), y el texto del mensaje (1234). En el proyecto, los dos términos más relevantes son el número de teléfono del remitente y el texto del mensaje. Para que se abra la puerta, el texto del mensaje debe ser un código numérico de 4 dígitos preestablecido, y debe ser recibido desde el número de teléfono del propietario.

En el modelo de Simulink final, es de especial importancia verificar que todas las transiciones de estado se realizan adecuadamente, en especial la transición que depende del SMS recibido. La Figura 4.4 muestra la monitorización de los apartados de memoria más relevantes durante la ejecución de un ciclo exitoso de apertura de la puerta, funcionando de la manera esperada.

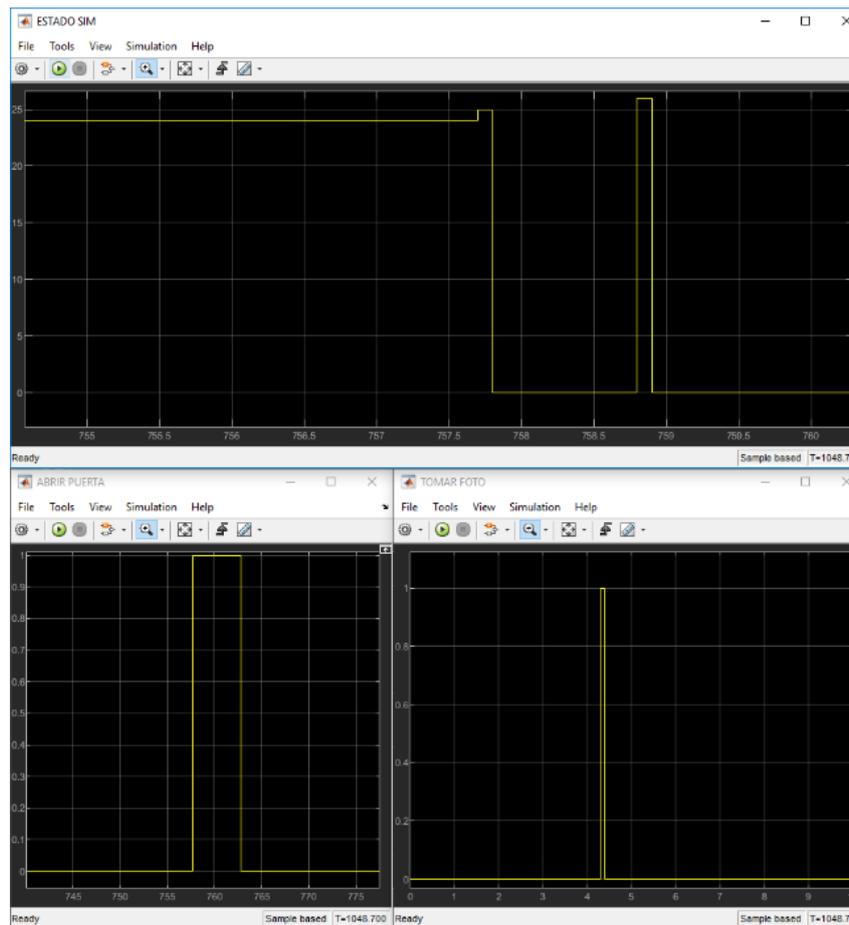


Figura 4.4.1: Transiciones más destacables de ESTADO\_SIM, ABRIR\_PUERTA y TOMAR\_FOTO. Fuente: Scope de Simulink

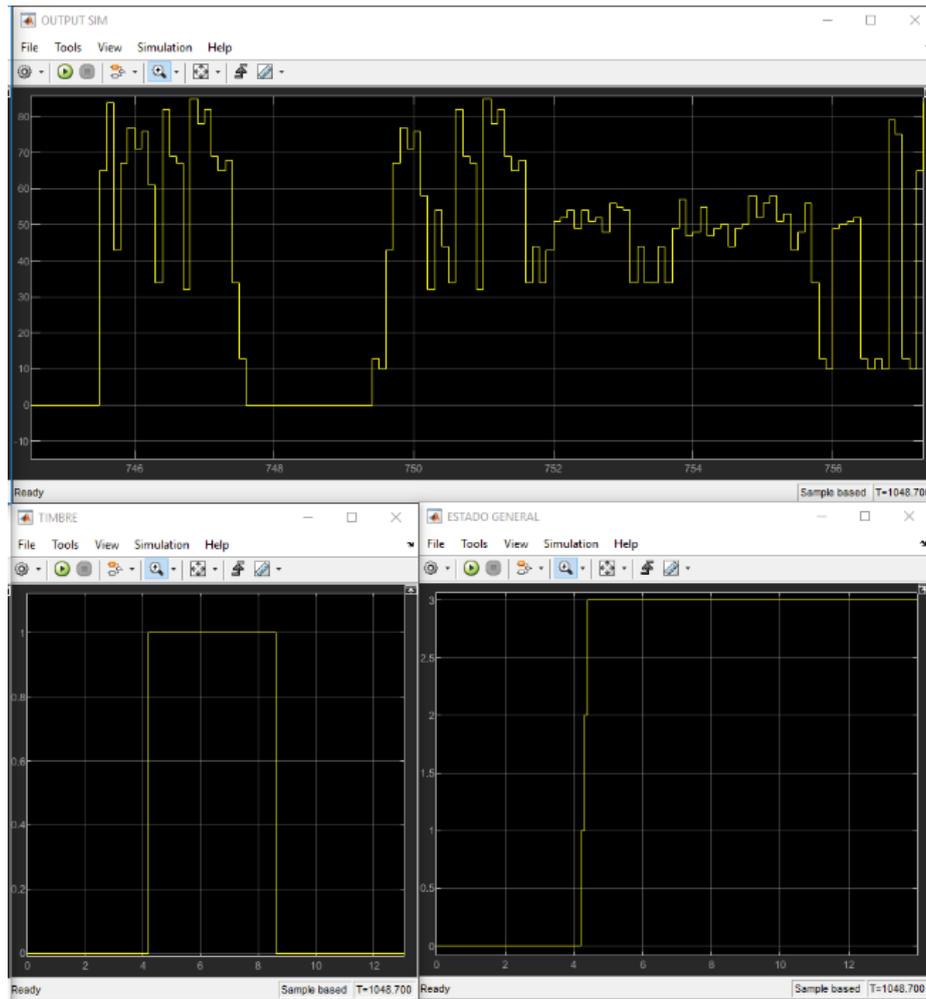


Figura 4.4.2: Transiciones más destacables de OUTPUT\_SIM, TIMBRE, ESTADO GENERAL. Fuente: Scope de Simulink.

## 4.2.- Cámara de la Raspberry

Para comprobar el funcionamiento adecuado de la cámara integrada en la Raspberry y de la función destinada a crear la señal RGB de la imagen, se diseñó un pequeño proyecto de Simulink. Inicialmente, se comprobó el funcionamiento de la grabación de vídeo, simulando en modo externo el bloque que obtiene la imagen en directo seguido del bloque que permite visualizar la imagen creada. Una vez obtenida esa señal de vídeo en directo, se modificó para obtener imágenes (capturas de ese vídeo en directo) al accionarse un “switch” manual (en modo externo). Una de las imágenes tomadas se muestra en la Figura 4.5.

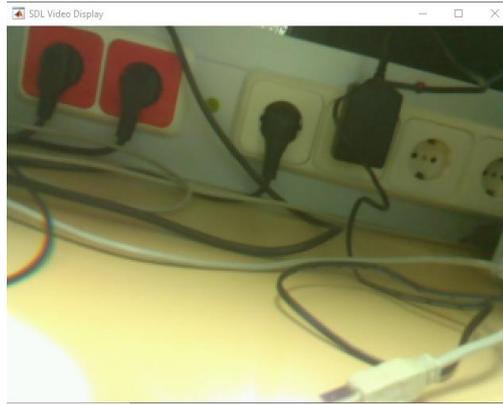


Figura 4.5: Imagen tomada por la cámara de la Raspberry (Tamaño 320 x 240). Fuente: SDL Display Simulink

Para verificar que, en efecto, el modelo era capaz de actualizar la foto recibida en el momento de accionamiento del interruptor en modo externo, se mantuvo la cámara grabando en una dirección. Acto seguido, se tomó la imagen y se modificó la dirección en la que apuntaba la cámara. Se volvió a accionar el pulsador en la nueva orientación, y la imagen mostrada en el SDL Display se actualizó, como se esperaba.

### 4.3.- Pulsador

Para verificar el funcionamiento del circuito diseñado para detectar el accionamiento del timbre (Figura 2.3), se creó un modelo de Simulink para monitorizar la entrada al PIN digital sobre el que se dispuso la salida del circuito (GPIO 21). Los resultados obtenidos para los valores que se obtienen en el PIN de la Raspberry asignado al circuito del pulsador se muestran en la Figura 4.6.

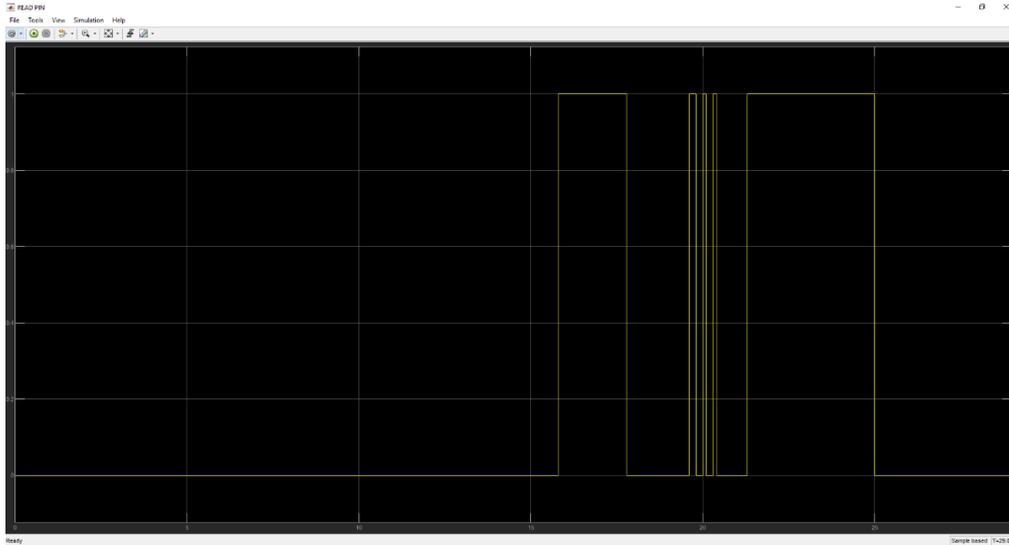


Figura 4.6: Lectura del GPIO 21 para el circuito del timbre. Fuente: Scope de Simulink.

Las lecturas del PIN son los valores esperados, cuando se acciona el pulsador, la entrada digital vale 1, y el resto del tiempo vale 0. Además, es destacable la ausencia de efecto rebote gracias al funcionamiento del condensador en el régimen transitorio, y el funcionamiento de la lectura del PIN para distintas duraciones de pulsado.

## 5.- Conclusiones y futuros desarrollos

### 5.1.- Actuador: SIM800C

El proceso de diseño relacionado con el módulo GSM fue el más laborioso y complejo por varias razones. La comunicación mediante comandos AT es fácil de aprender utilizando software como Python o Arduino que permiten una interacción más directa con el módulo GSM. La comunicación por Simulink tiene la desventaja de tener que simular un proceso con máquina de estados, previa conversión a ASCII. Esto dificulta ligeramente el proceso de pruebas, puesto que cada cambio realizado precisa de una nueva simulación en modo externo, y al extenderse la secuencia de comandos, se llegan a tiempos de simulación de varios minutos. La dificultad añadida es que la respuesta del módulo GSM por el puerto serie es una secuencia de caracteres en lenguaje ASCII, por lo que descifrarla lleva también bastante tiempo.

Otro inconveniente de la comunicación por el puerto serie en simulación es la necesidad de enviar constantemente información. En los tiempos en los que el módulo no está activo, se envían ceros, que no tienen efecto visible. Sin embargo, la información se envía en un buffer de 40 caracteres ASCII, que son acumulados y enviados uno a uno por el PIN Tx. Esto hace que, en cada ciclo, se acumulen 40 caracteres a enviar, pero solo se puede recibir uno por el puerto serie. Por lo tanto, cada instrucción enviada (de un ciclo de duración) requiere 40 veces ese tiempo para recibir una respuesta. Esto hace que cuando se envíen, durante tiempos muy prolongados, ceros al buffer, este se llene y empiece a funcionar de forma extraña. Por eso es necesario reiniciar el módulo GSM al acabar cada ciclo de uso. Además, si no se necesitara recibir información del PIN Rx, no habría mayor problema, pero, en la ejecución, se llega al estado de espera a lectura de SMS nuevo cuando aún se está recibiendo la respuesta a los primeros comandos de la secuencia enviada. Esto hizo necesario una modificación del buffer de lectura, que fuera capaz de identificar el ciclo en el que empieza cada instrucción para empezar a rellenar el buffer en ese instante, y que fuera también capaz de detectar si la respuesta que se estaba leyendo era la correspondiente al comando `AT+CMGL="REC UNREAD"`.

Finalmente, no se consiguió enviar el e-mail utilizando el módulo GSM por problemas de ajustes de servidor. El módulo respondía con 'OK' a todos los comandos relativos al e-mail, tanto de ajustes de servidor como de parámetros del propio mensaje. Sin embargo, al llegar a la fase final de envío, respondía con: 'OK ERROR 61', que es el código de error para "network error".

## 5.2.- Sensor: Pulsador

A lo largo del desarrollo, se utilizó principalmente un interruptor manual en Simulink para modelar un pulsador real. Se circuito final se diseñó y montó sobre el microcontrolador una vez verificado el funcionamiento en simulación del resto del software.

Se trata de un pequeño circuito que no necesita una placa externa para su implementación y que, por consiguiente, ocupa poco espacio y mantiene el objetivo de acabar el proyecto con un producto compacto y lo más pequeño posible.

En el futuro, sería interesante estudiar la posibilidad de añadir nuevas funciones al sistema domótico, como el uso de un altavoz, un micrófono, o un pequeño display. Los primeros permitirían la comunicación de voz entre el dueño de la vivienda y la persona que está en la puerta mediante, por ejemplo, una llamada telefónica. El Display permitiría mostrar mensajes como: "Solicitando acceso al propietario. Por favor espere" en el estado de espera a una respuesta, o "Acceso denegado", como transición entre el estado de espera y el estado de reposo.

En caso de implantar estas funcionalidades, sí que se utilizaría algún tipo de placa externa para montar los componentes necesarios

## 5.3.- Sensor: Cámara

Una vez acostumbrado al funcionamiento de la cámara en Simulink, la mayor dificultad afrontada fue tomar una única imagen en lugar de un vídeo en directo, que es la única opción que se maneja con los bloques de Simulink. Se valoró la posibilidad de guardar la imagen en la SD del microcontrolador, pero finalmente se decidió utilizar un espacio de memoria reservado para la imagen más reciente, que se actualizaría mediante un switch condicional cuyo input viene determinado por el valor de otra variable externa.

En cuanto al futuro desarrollo, es posible diseñar un gran número de funciones en MatLab para detectar, por ejemplo patrones en las imágenes. Las versiones actualizadas del software podrían clasificar las imágenes en "aptas" o no "aptas", siguiendo criterios como la detección de un rostro o la nitidez de la imagen. En el caso de tomar una fotografía "no apta", se procedería a tomar una nueva imagen, repitiendo el proceso hasta conseguir una imagen válida, y pasando entonces al estado de envío de la foto.

## 5.4.-Viabilidad económica del proyecto

Como se muestra en el Apéndice A, el precio completo de todos los componentes de hardware del proyecto asciende a 80.40€. Hay que tener en cuenta que los precios reflejados se corresponde con compra al por menor en tiendas on-line. Los productos comercializados a gran escala tienen precios inferiores, por comprar grandes cantidades de los componentes. Es conveniente mencionar también los costes asociados al uso continuado de este producto por el precio de la tarifa contratada para la tarjeta SIM del módulo GSM.

## A. Presupuesto

### A.1: Sumas parciales

#### A.1.1: Hardware

| <i>Componente</i>          | <i>Cantidad</i> | <i>Precio Unitario (€/u)</i> | <i>Coste (€)</i> |
|----------------------------|-----------------|------------------------------|------------------|
| <i>Raspberry Pi 3B+</i>    | 1               | 33.85                        | 33.85            |
| <i>GSM SIM800C</i>         | 1               | 14.91                        | 14.91            |
| <i>Cámara Raspberry</i>    | 1               | 8.99                         | 8.99             |
| <i>Circuito pulsador</i>   | 1               | 0.10                         | 0.10             |
| <i>Cierre eléctrico</i>    | 1               | 7.10                         | 7.10             |
| <i>Alimentación cierre</i> | 1               | 15.45                        | 15.45            |
| <i>Total</i>               | -               | -                            | 80.40            |

#### A.1.2: Software

| <i>Componente</i>        | <i>Cantidad</i> | <i>Precio Unitario (€/u)</i> | <i>Coste (€)</i> |
|--------------------------|-----------------|------------------------------|------------------|
| <i>MatLab</i>            | 1               | 700*                         | 700*             |
| <i>Simulink Blockset</i> | 1               | 0                            | 0                |
| <i>Raspbian</i>          | 1               | 0                            | 0                |
| <i>Total</i>             | -               | -                            | 700              |

\*Proyecto realizado con licencia MathWorks estudiante.

#### A.1.3: Herramientas y equipos

| <i>Componente</i>     | <i>Cantidad</i> | <i>Precio Unitario (€/u)</i> | <i>Coste (€)</i> |
|-----------------------|-----------------|------------------------------|------------------|
| <i>Ordenador</i>      | 1               | 700                          | 700              |
| <i>Cable Ethernet</i> | 1               | 5                            | 5                |
| <i>Total</i>          | -               | -                            | 705              |

### A.1.4: Mano de obra directa

| <i>Tarea</i>                                    | <i>Horas</i> | <i>Precio por hora (€/h)</i> | <i>Coste (€)</i> |
|---|--------------|------------------------------|------------------|
| <i>Implementación del circuito del pulsador</i> | 2            | 20                           | 40               |
| <i>Implementación módulo GSM</i>                | 100          | 50                           | 5000             |
| <i>Implementación de la cámara</i>              | 15           | 35                           | 525              |
| <i>Software y máquina de estados</i>            | 10           | 35                           | 350              |
| <i>Puesta en conjunto</i>                       | 20           | 45                           | 900              |
| <i>Documentación</i>                            | 50           | 30                           | 1500             |
| <i>Total</i>                                    | -            | -                            | 8315             |

### A.2: Presupuesto general

| <i>Componente</i>             | <i>Coste (€)</i> |
|-------------------------------|------------------|
| <i>Hardware</i>               | 80.40            |
| <i>Software</i>               | 700              |
| <i>Herramientas y equipos</i> | 705              |
| <i>Mano de obra directa</i>   | 8315             |
| <i>Total</i>                  | 9800.40          |

# Bibliografía

- [1] Sciencedirectcom. (2019). Sciencedirectcom. Retrieved 2 July, 2019, from <https://www.sciencedirect.com/science/article/pii/S0186104214712537>
- [2] Wikipediaorg. (2019). Wikipediaorg. Retrieved 2 July, 2019, from [https://es.wikipedia.org/wiki/Ley\\_de\\_Moore](https://es.wikipedia.org/wiki/Ley_de_Moore)
- [3] Sciencedirectcom. (2019). Sciencedirectcom. Retrieved 2 July, 2019, from <https://www.sciencedirect.com/science/article/abs/pii/S0378512209002606>
- [4] Sciencedirectcom. (2019). Sciencedirectcom. Retrieved 2 July, 2019, from <https://www.sciencedirect.com/science/article/pii/S0301421517300393>
- [5] npirtube. (2019). Xiaomi MiJia 360° 720P: a fondo - npirtube. Retrieved 2 July 2019, from <https://npirtube.com/xiaomi-mijia-360o-720p-fondo/>
- [6] MagiDeal Gsm Puerta Abridor Interruptor Relé Controlador Acceso Remoto Deslizante Abridor Compuerta- Motores para puertas de garaje. Retrieved 2 July, 2019, from <https://motoresparapuertas.net/producto/magideal-gsm-puerta-abridor-interruptor-rele-controlador-acceso-remoto-deslizante-abridor-compuerta/>.
- [7] Consultores y auditores de eficiencia energética. (2019). Control de acceso para una vivienda domótica | Loxone. Retrieved 2 July 2019 from: <https://www.ensaco.es/teclado-numerico-para-una-vivienda-domotica/>
- [8] Wikipediaorg. (2019). Wikipediaorg. Retrieved 3 July, 2019, from [https://en.wikipedia.org/wiki/Raspberry\\_Pi](https://en.wikipedia.org/wiki/Raspberry_Pi)
- [9] Raspberrypiorg. (2019). Raspberrypiorg. Retrieved 3 July, 2019, from <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>
- [10] Kubiifr. (2019). Kubiifr. Retrieved 3 July, 2019, from <https://www.kubii.fr/raspberry-pi/2119-raspberry-pi-3-modelo-b-1-gb-713179640259.html?src=raspberrypi>
- [11] Mathworkscm. (2019). Mathworkscm. Retrieved 3 July, 2019, from <https://es.mathworks.com/hardware-support/raspberry-pi-matlab.html>
- [12] Universidadviues. (2019). Universidadviues. Retrieved 4 July, 2019, from <https://www.universidadviu.es/que-es-gsm-y-como-funciona/>
- [13] Lifewirecom. (2019). Lifewire. Retrieved 4 July, 2019, from <https://www.lifewire.com/gsm-explained-3426452>

- [14] Simcomee. (2019). Simcomee. Retrieved 4 July, 2019, from <https://simcom.ee/modules/gsm-gprs/sim800c/>
- [15] Mathworkscom (2019). Mathworkscom. Retrieved 4 July, 2019, from [https://es.mathworks.com/help/supportpkg/raspberrypi/ref/serialwrite.html?s\\_tid=doc\\_t a](https://es.mathworks.com/help/supportpkg/raspberrypi/ref/serialwrite.html?s_tid=doc_t a)
- [16] Electronicsforucom. (2019). Electronics For You. Retrieved 4 July, 2019, from <https://electronicsforu.com/resources/cool-stuff-misc/gsm-at-commands>
- [17] Amazonas. (2019). Amazonas. Retrieved 13 July, 2019, from <https://www.amazon.es/AZDelivery-cámara-para-Raspberry-Pi/dp/B01M6UCEM5>
- [18] Mathworkscom. (2019). Mathworkscom. Retrieved 8 July, 2019, from [https://es.mathworks.com/help/matlab/creating\\_plots/image-types.html#f2-12468](https://es.mathworks.com/help/matlab/creating_plots/image-types.html#f2-12468)
- [19] Ascitablecom. (2019). Ascitablecom. Retrieved 13 July, 2019, from <http://www.asciitable.com/>
- [20] Inés Arnáiz Lázaro-Carrasco. (2018). Sistema domótico de notificación de alarmas por SMS. Retrieved 10 January, 2019.