



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

# **Sistema de Visualización Electrónico para Intubación**

**INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de

**Ingeniero Eléctrico-Electrónico**

**P R E S E N T A**

José Mario Iyañez Bolaños

**ASESOR(A) DE INFORME**

Dra. Fatima Moumtadi



**Ciudad Universitaria, Cd. Mx., 2019**

# Índice

1.	Introducción .....	3
1.1.	Objetivo .....	3
1.2.	Justificación.....	3
2.	Marco teórico.....	7
2.1.	Módulo de carga de batería .....	7
2.2.	Convertidores DC-DC .....	8
2.3.	Sensores de toque capacitivos .....	9
2.4.	Cámara Web .....	9
2.5.	Minicomputadoras .....	10
2.6.	Interfaz de Visualización .....	11
3.	Desarrollo .....	14
3.1.	Elevador de Voltaje .....	14
3.2.	Sensores de toque capacitivos .....	16
3.3.	Mini-computadora Raspberry Pi Zero W .....	18
3.4.	Display LCD .....	19
3.5.	Diseño de PCB.....	20
3.6.	Firmware .....	24
3.7.	Hardware.....	26
4.	Resultados.....	29
4.1.	Fuente Boost.....	29
4.2.	Sensores de toque capacitivos .....	30
5.	Conclusiones .....	31
6.	Fuentes de información .....	32
7.	Anexos.....	33
7.1.	Nota de Aplicación SNVA255A .....	33
7.2.	Datasheet IQS127D .....	34
7.3.	Datasheet LM2735.....	37
7.4.	Adafruit PITFT .....	38
7.5.	Datasheet AT42QT1011 .....	41
7.6.	Datasheet MCP73831 .....	43

# 1. Introducción

## 1. Objetivo

Realizar el diseño de un sistema de video electrónico que permita la visualización indirecta de la tráquea para asistir en la intubación durante los procedimientos de anestesia general.

## 2. Justificación

Al momento de incorporarme al proyecto, ya se contaba con un prototipo funcional, el cual no lograba el desempeño deseado, ya que la velocidad de reproducción del video mostrado era de 15 fps (cuadros por segundo o *frames per second*), además que al momento de encender no siempre reconocía la cámara, por lo que entraba en un ciclo de reinicio hasta que se sincronizaba.

Para este prototipo se utilizaba el microprocesador LinkIt Smart 7668 de MediaTek, el cual corría OpenWrt, una distribución de Linux comúnmente empleada en módems, el cual funciona bien para aplicaciones de conectividad de dispositivos en una red de área local (LAN), o a internet, al ser fácilmente configurable como punto de acceso o repetidor de señal WiFi; sin embargo debido a la poca memoria con la que cuenta (128 MB en RAM) y a la velocidad de su procesador, no contaba con los recursos necesarios para mostrar video a una velocidad superior a los 24 fps, que es el mínimo para que el ojo humano observe fluidez de movimiento; debido principalmente a este inconveniente y a que otras partes del desarrollo del prototipo aún no habían podido ser implementadas, como lo son el poder tomar fotos, o una galería para poder observar dichas fotos, se sugirió un cambio de procesador.

En esta versión tanto el microprocesador, como la cámara operaban con un voltaje de 3.3V, por lo que se utilizaban reguladores de voltaje para mantenerlo constante, siempre y cuando la carga de la batería fuera superior a 3.3V.

La conexión del microcontrolador con el display en este prototipo se hacía mediante un protocolo de 8 bits en paralelo, como se muestra en la figura 1.

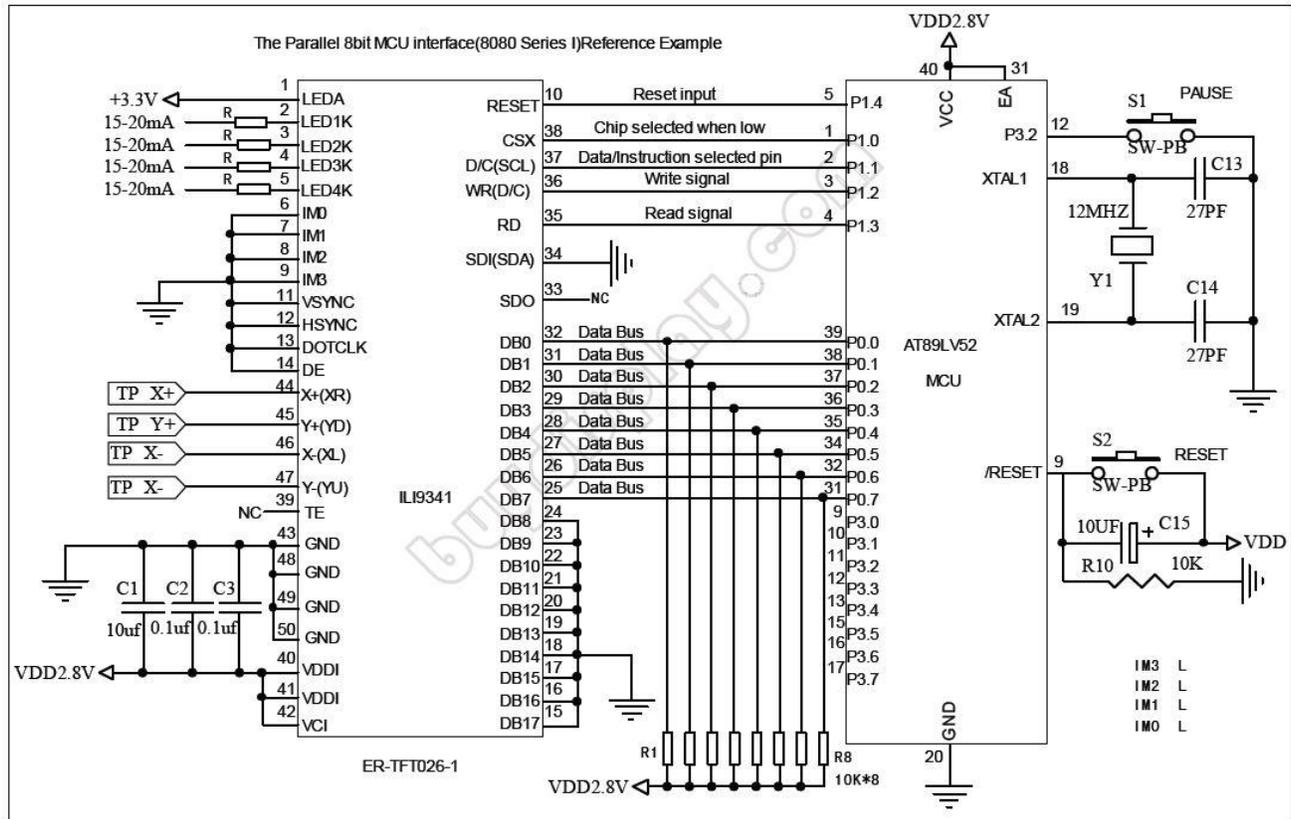


Figura 1: Conexión 8 bit paralela

Los botones táctiles estaban habilitados usando el sensor AT42QT1010 de MICROCHIP. La configuración básica para este integrado se muestra en la figura 2.

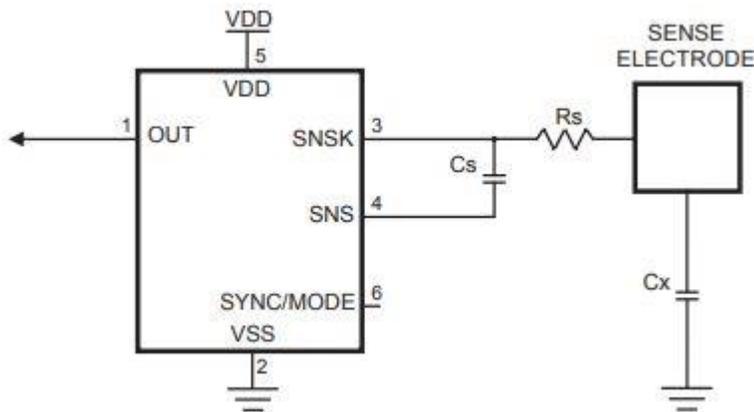


Figura 2: Circuito típico AT42QT1010 [Anexo 7.5]

Para un funcionamiento adecuado del dispositivo se tiene que  $C_s \gg C_x$ , por lo que de acuerdo a la hoja de datos del dispositivo [5],  $C_x$  debe estar en el rango de 5-20pF, mientras que  $C_s$  se recomienda que esté entre 2-50nF [Anexo 7.5]. La sensibilidad del dispositivo depende del valor de  $C_s$ , el tamaño, la forma, la orientación y la capacitancia de los electrodos a usar, además del objeto que se va a detectar como pulsador, por lo que si se desea aumentar la sensibilidad  $C_s$  debe aumentar, ajustando de manera correspondiente  $R_s$  para mantener la respuesta deseada.

Para solucionar estos problemas se propone como primera medida, cambiar el procesador por una Raspberry Pi Zero W, ya que al ser una minicomputadora y tener más RAM y velocidad de procesamiento que el LinkIt Smart, nos permite un desarrollo más rápido y un mejor desempeño de las funciones deseadas, las cuales consisten en:

- Velocidad de video superior a 24 fps.
- Capacidad de tomar fotos.
- Visualización nítida de la tráquea.
- Facilidad de transferir las fotos del dispositivo a una computadora.

Debido a que la Raspberry Pi funciona con 5V, se necesita implementar un convertidor DC-DC para elevar el voltaje de la batería.

Para solucionar los problemas de sensibilidad presentados por los sensores ya utilizados, se propone cambiarlos por los sensores programables ISQ127D de Azoteq, los cuales además de tener un tamaño compacto y de utilizar pocos componentes externos, permiten mediante su programación ajustar la configuración de sensibilidad y tipo de respuesta de los mismos.

El sistema electrónico propuesto se muestra en la figura 3.

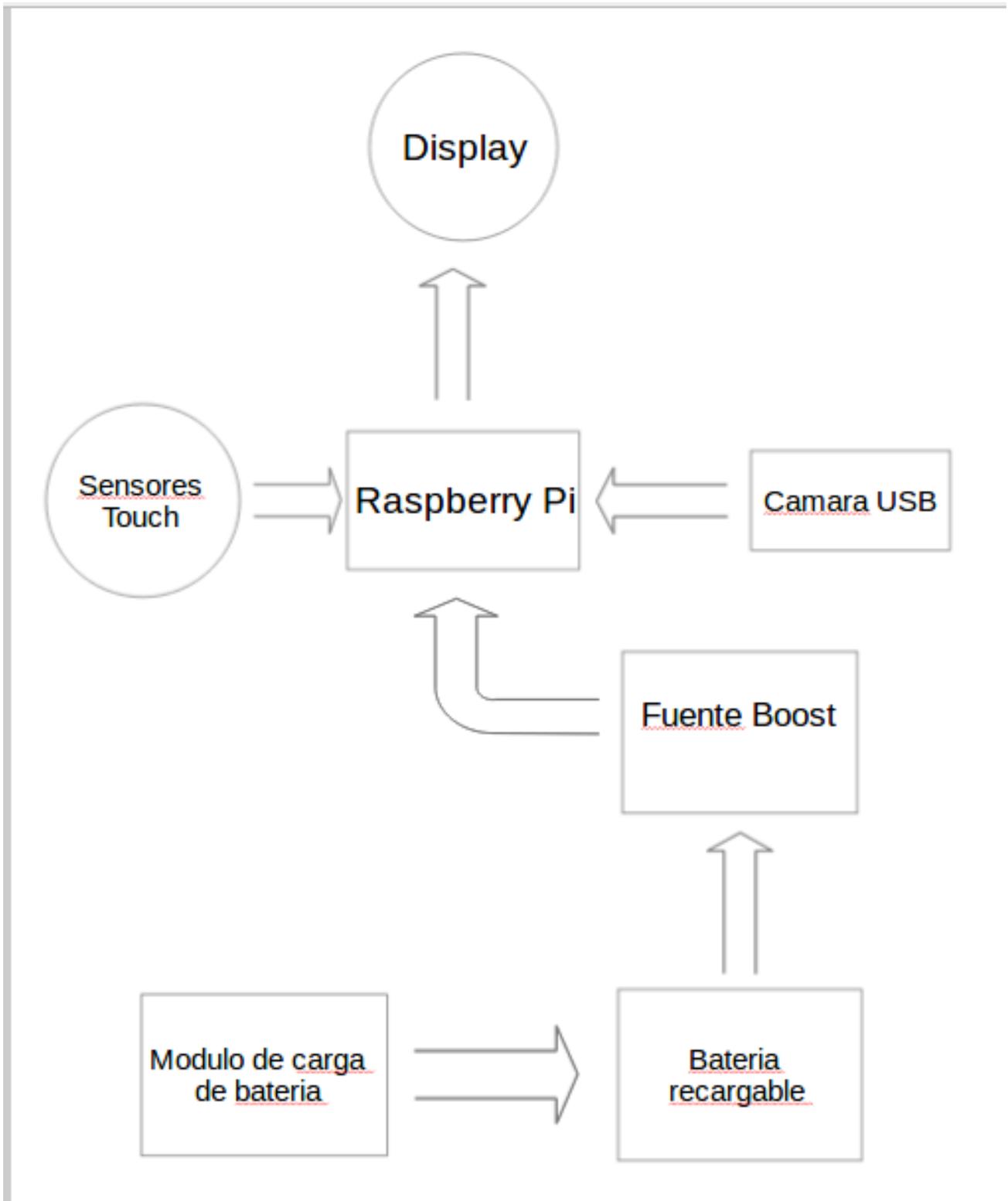


Figura 3: Diagrama de bloques del sistema propuesto

## 2. Marco teórico

### 1. Módulo de carga de batería

Las baterías a menudo sirven como el método para suministrar energía a los dispositivos portátiles; debido al costo a lo largo de la vida útil del producto las baterías recargables son usadas de manera común para los dispositivos portátiles.

Existen dos tipos de cargadores, uno suministra voltaje constante a la batería, y el otro, pulsos de voltaje [3].

Para esta aplicación se seleccionó el integrado MCP73831, el cual permite ajustar la salida de voltaje a 4.2V que es el voltaje de la batería seleccionada, el valor de la corriente de carga se ajusta mediante un resistor externo.

El circuito de aplicación típico se muestra en la figura 4.

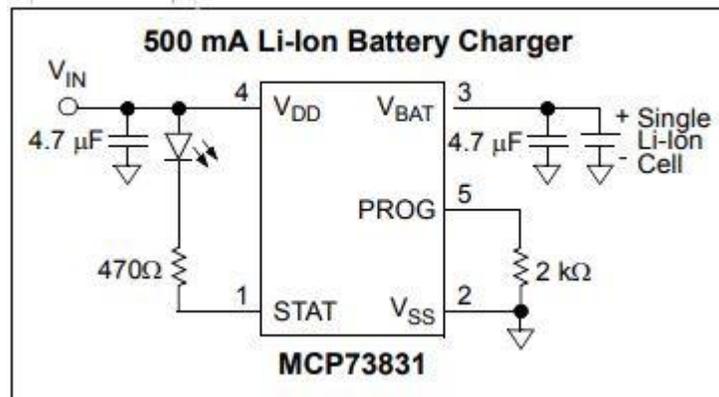


Figura 4: Circuito de aplicación típica [Anexo 7.6]

Para regular la corriente de carga, se hace uso de una resistencia externa conectada de la terminal PROG a V<sub>SS</sub>, y está se puede calcular de la siguiente manera [Anexo 7.6]:

$$I_{REG} = \frac{1000V}{R_{PROG}} \quad (1)$$

## 2. Convertidores DC-DC

Al ser un dispositivo alimentado con una batería de Li-Ion con un voltaje máximo de 4.2V, y que el procesador seleccionado requiere un voltaje de 5V y una corriente de 1A para funcionar, se requiere elevar el voltaje de la pila, empleando para esto un convertidor DC-DC de topología Boost.

En un convertidor elevador o Boost, mostrado en la figura 5 [1], el voltaje de salida es mayor al de entrada, de acuerdo con el circuito mostrado en la figura; su principio básico de funcionamiento consiste en dos estados distintos, dependiendo de si el MOSFET (representado en la figura 5 por un interruptor), conduce o no en ese momento.

Cuando el MOSFET se polariza el inductor L1 almacena energía, mientras que el capacitor C1 se encarga de alimentar a la carga.

Cuando el MOSFET deja de conducir, la corriente fluye a través del diodo D1 y circula por el capacitor y la carga.

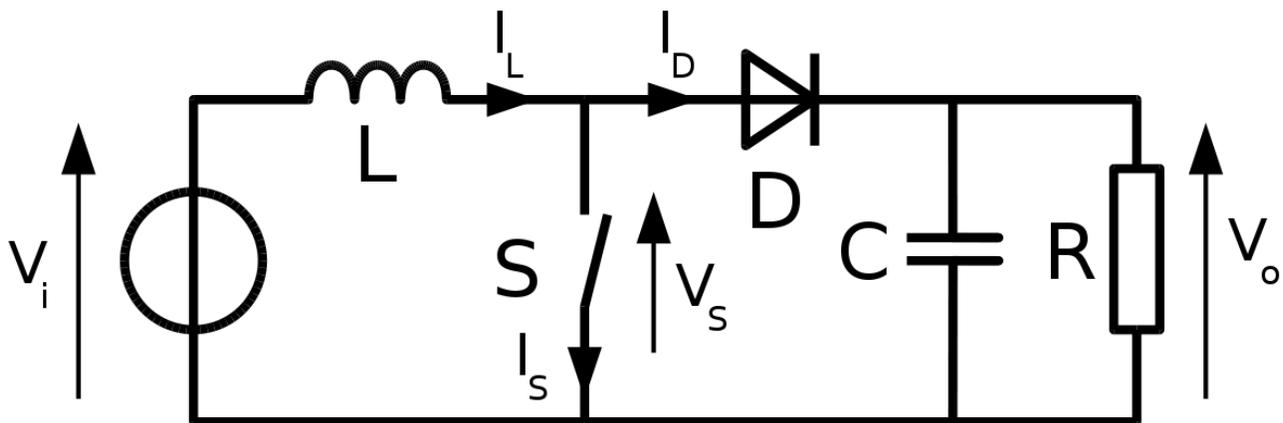


Figura 5: Circuito típico de un convertidor Boost

### **3. Sensores de toque capacitivos**

Si un dedo, o cualquier otro objeto capaz de modificar la carga se aproxima suficientemente a un sensor de toque capacitivo, éste actúa como otro capacitor. Esto ocurre por la naturaleza dieléctrica del sensor, que varía la capacitancia efectiva del sistema utilizado para detectar el contacto táctil [4].

Un sistema de detección táctil estándar se compone de tres bloques funcionales principales:

- Un bloque analógico para detección capacitiva.
- Un controlador para procesar los datos.
- Un bloque de interfaz para la comunicación con un procesador.

Las soluciones de detección táctil capacitiva basadas en la variación de la tensión pueden ser llevadas a la práctica con efectividad mediante técnicas que emplean los siguientes componentes:

- Una unidad de medida del tiempo de carga (CTMU) periférica integrada en un microcontrolador
- Un divisor de tensión capacitivo (CVD), utilizando el convertidor de analógico a digital (ADC), que no necesita de un periférico dedicado para la detección capacitiva.

### **4. Cámara Web**

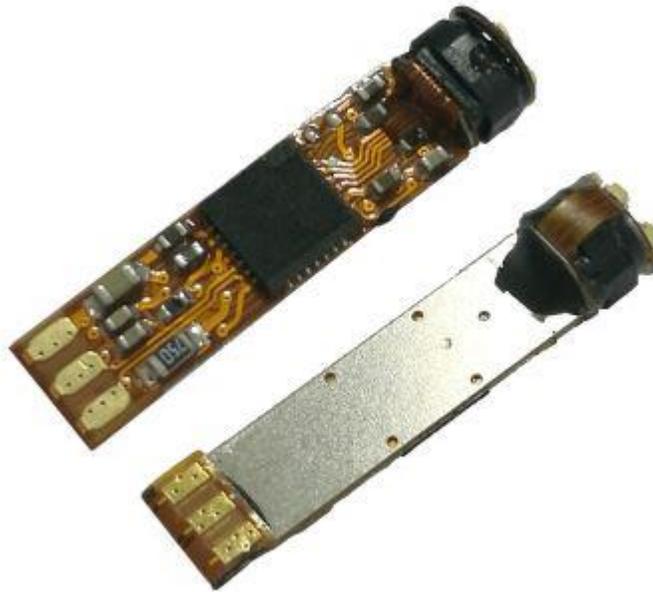
Las cámaras web normalmente están formadas por una lente, un sensor de imagen y la circuitería necesaria para manejarlos.

Existen distintos tipos de lentes, siendo las lentes plásticas las más comunes. Los sensores de imagen pueden ser CCD (charge coupled device) o CMOS (complementary metal oxide semiconductor). Este último suele ser el habitual en las cámaras más económicas, aunque eso no signifique necesariamente que cualquier cámara CCD sea mejor que cualquiera CMOS. Dependiendo de la resolución de las cámaras encontramos los modelos de gama baja, que se sitúan alrededor de 320x240 píxeles. Las cámaras web para usuarios medios suelen ofrecer una resolución VGA (640x480) con una tasa de captura de unos 30 fps, si bien en la actualidad están ofreciendo resoluciones medias de 1 a 1,3 MP, actualmente las cámaras de gama alta cuentan con 3, 5, 8, 10 y hasta 15 megapíxeles y son de alta definición.

La circuitería electrónica es la encargada de leer la imagen del sensor y transmitirla a la computadora. Algunas cámaras usan un sensor CMOS integrado con la circuitería en un

único circuito integrado de silicio para ahorrar espacio y costos. El modo en que funciona el sensor es equivalente al de una cámara digital normal.

En la figura 6 se muestra la cámara usada para este diseño.



*Figura 6: Cámara USB ©*

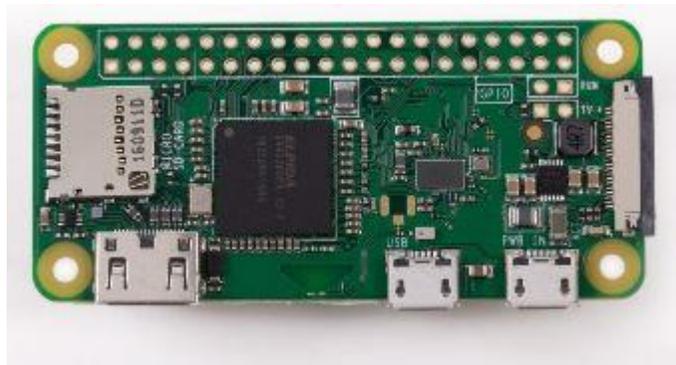
## **5. Minicomputadoras**

Las mini-computadoras han ganado un gran éxito en los últimos años. Desde el lanzamiento de la primera minicomputadora de bajo costo de Raspberry Pi, muchos fabricantes han creado sus propios equipos, más o menos potentes, buscando, entre otros aspectos, reducir su costo y abaratar el precio.

Por su costo y tamaño se seleccionó para esta aplicación la Raspberry Pi Zero W; las características de esta minicomputadora no son las mismas que las que ofrece la versión más potente, Raspberry Pi 3 B+, sin embargo, cuenta con una potencia a la altura de la primera versión de la mini-computadora capaz de correr el sistema operativo Raspbian (o cualquiera de los sistemas operativos de NOOBS) sin demasiada dificultad, así como ejecutar ciertas aplicaciones como Scratch y jugar Minecraft.

Las principales características de la Raspberry Pi Zero, mostrada en la figura 7, son:

- Procesador Broadcom BCM2835 @ 1GHz ARM 11
- 512 MB de memoria RAM LPDDR2
- Ranura para tarjeta microSD
- Salida de video mini HDMI a 1080p
- 802.11 b/g/n LAN inalámbrica
- Bluetooth 4.1
- Bluetooth Low energy (BLE)
- Conector CSI para cámara
- Header de 40 pines compatible con HAT



*Figura 7: Raspberry Pi Zero ©*

## **6. Interfaz de Visualización**

Para este proyecto se ocupó un display TFT, este tipo de pantalla es una variante de la pantalla de cristal líquido (LCD) que usa tecnología de transistor de película delgada (TFT) para mejorar su calidad de imagen. Las LCD de TFT son un tipo de LCD de matriz activa. Son usados en televisores, visualizadores de pantalla plana y proyectores. En informática, los monitores de TFT han desplazado la tecnología de CRT, y están comúnmente disponibles en tamaños de 12 a 30 pulgadas. En el 2006 han entrado en el mercado de las televisiones [5].

Las pantallas de cristal líquido normales, como las de las calculadoras, presentan elementos de imagen excitados en forma directa se puede aplicar una tensión a través de un segmento sin que interfiera con otros segmentos de la pantalla. Esto no es posible en pantallas grandes con un gran número de píxeles, puesto que se requerirían millones de conexiones en la parte superior e inferior para cada uno de los tres colores (rojo, verde y azul) de cada píxel. Para evitar esto, los píxeles son direccionados en filas y columnas, lo que reduce el número de conexiones de millones a miles. Si todos los píxeles de una fila son excitados mediante una tensión positiva y todos los píxeles de una columna son excitados con una tensión negativa, entonces el píxel que se encuentra en la intersección tiene el voltaje aplicado más elevado y es conmutado. El inconveniente de esta solución es que todos los píxeles de la misma columna reciben una fracción de la tensión aplicada, como ocurre con todos los píxeles de la misma fila, así a pesar de que no sean conmutados completamente, tienden a oscurecerse. La solución al problema es proporcionar a cada píxel su propio transistor conmutador, esto permite controlar a cada píxel por separado. La baja corriente de fuga del transistor implica que la tensión aplicada al píxel no se pierde durante las actualizaciones de refresco de la imagen en la pantalla. Cada píxel es un pequeño condensador con una capa transparente de óxido de indio y estaño en el frontal, una capa transparente en la parte posterior, y entre medio una capa aislante de cristal líquido.

La distribución de los circuitos en un TFT-LCD es muy similar a la utilizada en la memoria DRAM. Sin embargo, en vez de realizar los transistores usando obleas de silicio, estos son fabricados depositando una película delgada de silicio sobre un panel de vidrio. Los transistores ocupan sólo una pequeña fracción del área de cada píxel y la película de silicio de la superficie remanente es eliminada permitiendo que la luz pase a través de ella.

La capa del silicio para TFT-LCD se deposita generalmente usando el proceso denominado PECVD (Plasma Enhanced Chemical Vapor Deposition) de un precursor de gas silano ( $\text{SiH}_4$ ) para producir una película amorfa de silicio. El silicio policristalino también se utiliza en algunas pantallas donde se requieren TFTs con un rendimiento más alto, típicamente en pantallas donde se requiere una resolución muy alta o en aquellas donde se desea realizar algún procesamiento de datos en sí mismo. Ambos tipos de TFTs, los de silicio amorfo y los de silicio policristalino presentan una prestación muy pobre frente a los transistores fabricados a partir de cristales de silicio simples, normalmente durante la construcción quedan residuos que posteriormente con el tiempo pueden presentar píxeles apagados y se pueden propagar en líneas horizontales y verticales de manera progresiva.

En la figura 8 se muestra una imagen del display seleccionado para este prototipo, tiene un tamaño de 2.6 pulgadas y una resolución de 320x240.



Figura 8: Display LCD ©

### 3. Desarrollo

#### 1. Elevador de Voltaje

En mi primera intervención en el proyecto, comencé por diseñar un convertidor DC-DC mostrado en la figura 9, para elevar el voltaje de alimentación de la cámara de 3.3V a 5V, esto con la finalidad de comprobar si con un mayor voltaje aumentaba la velocidad con la que capturaba las imágenes, para ello se optó por una topología boost, y por el integrado LM2735Y de Texas Instruments, debido a su tamaño pequeño y a la poca cantidad de componentes externos necesarios para su funcionamiento.

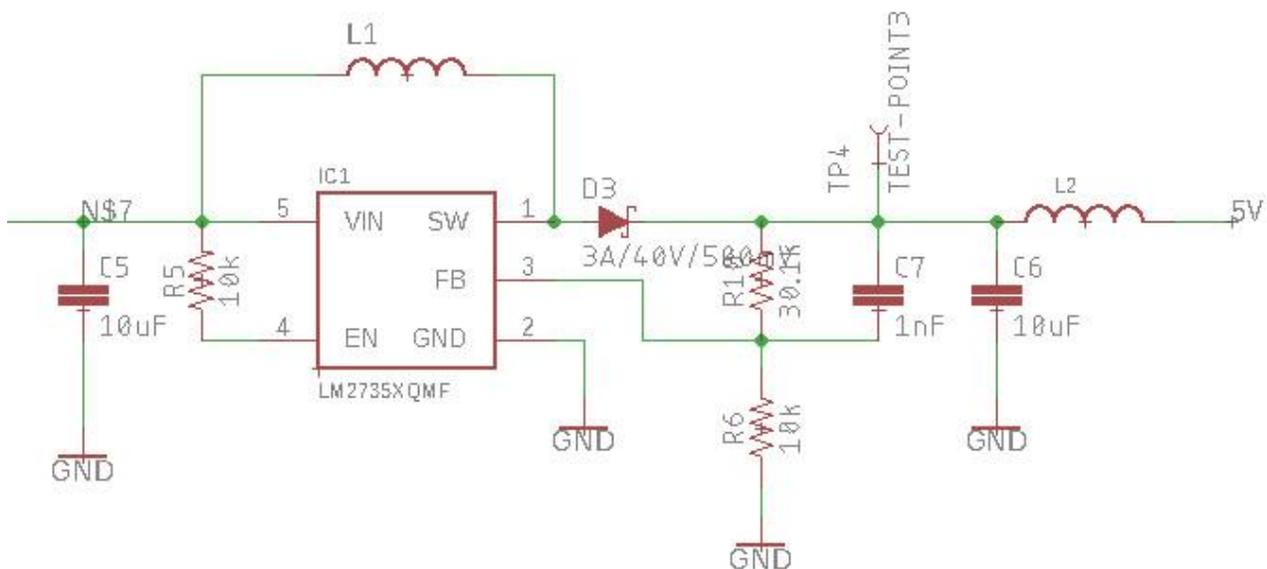


Figura 9: Fuente Boost

De acuerdo con lo mostrado en la hoja de datos del dispositivo [Anexo 7.1, Anexo 7.3], se calculó de los componentes necesarios para tener un voltaje de salida de 5V y una corriente de 800mA.

Como parámetros de entrada se tiene:

$$V_{in} = 3,8V; V_{out} = 5V; f_{osc} = 520KHZ; I_{out} = 1A; \Delta_{iL} = 100mA$$

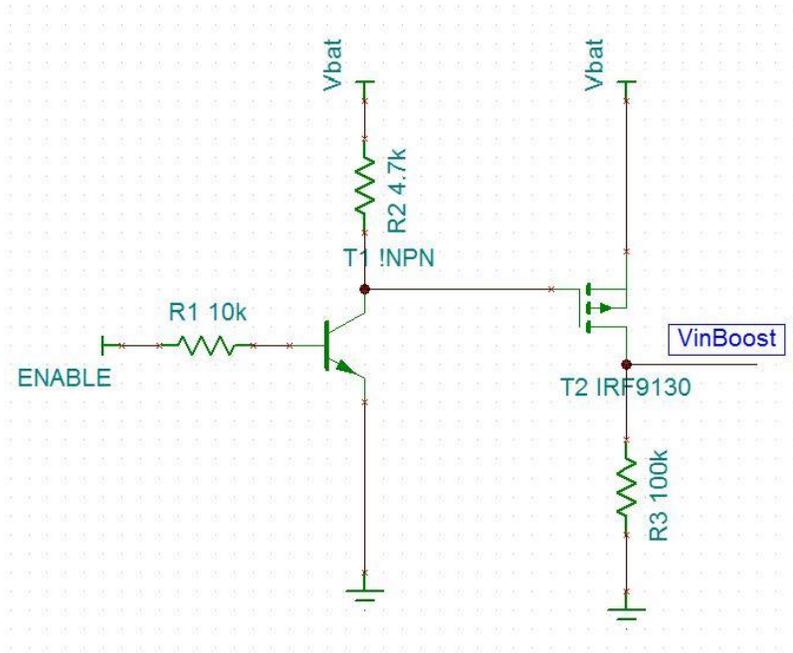
El inductor necesario se calcula como:

$$L = \left(\frac{V_{in}}{2\Delta_{iL}}\right)\left(\frac{V_{out} - V_{in}}{V_{in}}\right)\left(\frac{1}{f_{osc}}\right) = \frac{3,8V}{2(100mA)}\left(\frac{5V - 3,8V}{3,8V}\right)\left(\frac{1}{520KHz}\right) \quad (2)$$
$$L = 11,53\mu H$$

Para fijar el voltaje de salida se usa R1, la cual se puede dejar fija, si se le da un valor de R1=10KΩ, R2 se calcula de la siguiente manera:

$$R_2 = \left(\frac{V_{out}}{V_{ref}} - 1\right)(R_1) = \left(\frac{5}{1,255} - 1\right)(10K\Omega) \quad (3)$$
$$R_2 = 29,84K\Omega$$

El integrado LM2735Y funciona de manera excelente como fuente Boost, sin embargo presentó un problema, uno de sus pines tiene la función de ENABLE, este pin se activa con un nivel lógico alto, que pueden ser 3.3V o 5V, y es cuando comienza a funcionar el integrado, pero cuando no se habilita, la fuente presentaba a la salida el mismo voltaje que tenía a la entrada, por lo que fue necesario implementar el arreglo de transistores a manera de interruptor mostrado en la figura 10[2].



*Figura 10: Arreglo de transistores como interruptor*

Con este arreglo, el BJT se comporta como un inversor, al tener un nivel alto en el ENABLE, el BJT entra en la zona de saturación por lo que el voltaje que recibe en el colector se va a tierra, esto produce que tengamos un nivel lógico bajo en el gate del MOSFET, por lo que el voltaje que este recibe en el source pasa a la entrada de la fuente Boost; al tener un nivel bajo en el ENABLE del BJT, este entra en la zona de corte, por lo que el voltaje del colector se va al gate del MOSFET polarizándolo y haciendo circular la corriente de la batería hacia tierra.

Al terminar esta adaptación y realizar las pruebas correspondientes se concluyó que el voltaje de alimentación de la cámara no era el problema, por lo que se comenzó a buscar cómo aumentar la velocidad de reproducción del video por medio del software.

## 2. Sensores de toque capacitivos

Para esta aplicación se seleccionó el sensor capacitivo IQS127D de Azoteq, el cual presenta las siguientes ventajas:

- Ajuste automático para un rendimiento óptimo.
- Capacitor de referencia interno.
- Voltaje de alimentación de 2.95 a 5.5V.

- Componentes externos mínimos para su funcionamiento.
- Configuración programable.

El sensor cuenta con dos bancos de opciones configurables, las cuales se modifican asignándole bits de configuración [Anexo 7.2].

La implementación de este diseño se realizó de acuerdo al circuito mostrado en la figura 11.

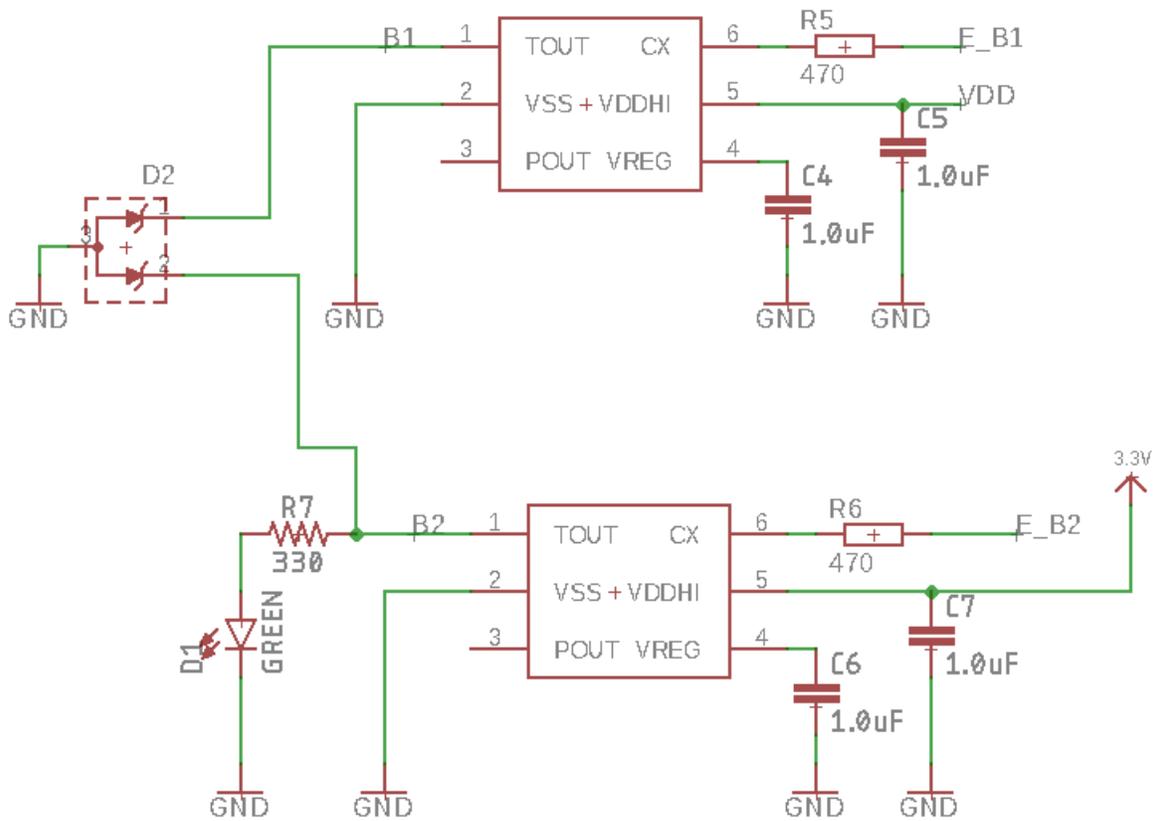


Figura 11: Circuito de aplicación del CI IQS127

### **3. Mini-computadora Raspberry Pi Zero W**

El microprocesador LinkIt Smart corría una distribución de Linux llamada OpenWrt la cual es muy común en módems y routers, por lo que permitía realizar la instalación de algunos paquetes de software para el procesamiento de imágenes de manera relativamente fácil, sin embargo al no estar pensado para aplicaciones de reproducción de video, y mucho menos para procesar imágenes en tiempo real como se esperaba que lo hiciera, este no contaba con el software necesario, por lo que las modificaciones necesarias para hacerlo funcionar nos llevaban a tener que compilar desde cero un kernel nuevo de Linux que pudiera contener los paquetes de software necesarios para la aplicación, al ser esto demasiado costoso en tiempo, se optó por cambiar el microprocesador por una Raspberry Pi Zero, la cual además de correr también una distribución de Linux, cuenta con un procesador más potente y mayor memoria RAM, y también cuenta con un puerto USB Host nativo y permite entablar de manera sencilla la comunicación con el Display por medio de un protocolo SPI.

Se procedió a desarrollar el firmware del dispositivo para la Raspberry, utilizando software como fswebcam para poder tomar fotografías y gstreamer para realizar la reproducción del video, además se configuro una galería para la visualización de las fotos utilizando un servidor web Apache.

La Raspberry Pi necesita 5V y 2A para su correcto funcionamiento, al medir la corriente consumida por la misma y por los demás circuitos se observó que consumía 500mA durante su operación por lo que fue necesario cambiar el inductor y el diodo de la fuente boost para que pudieran soportar hasta 1A y de esta manera garantizar el correcto funcionamiento de la misma.

## 4. Display LCD

El siguiente punto a corregir fue modificar la conexión del display, ya que se tuvo que modificar de una conexión de 8 bits en paralelo a una conexión por SPI, mostrada en la figura 11, que resulta más simple y rápida de implementar en una Raspberry.

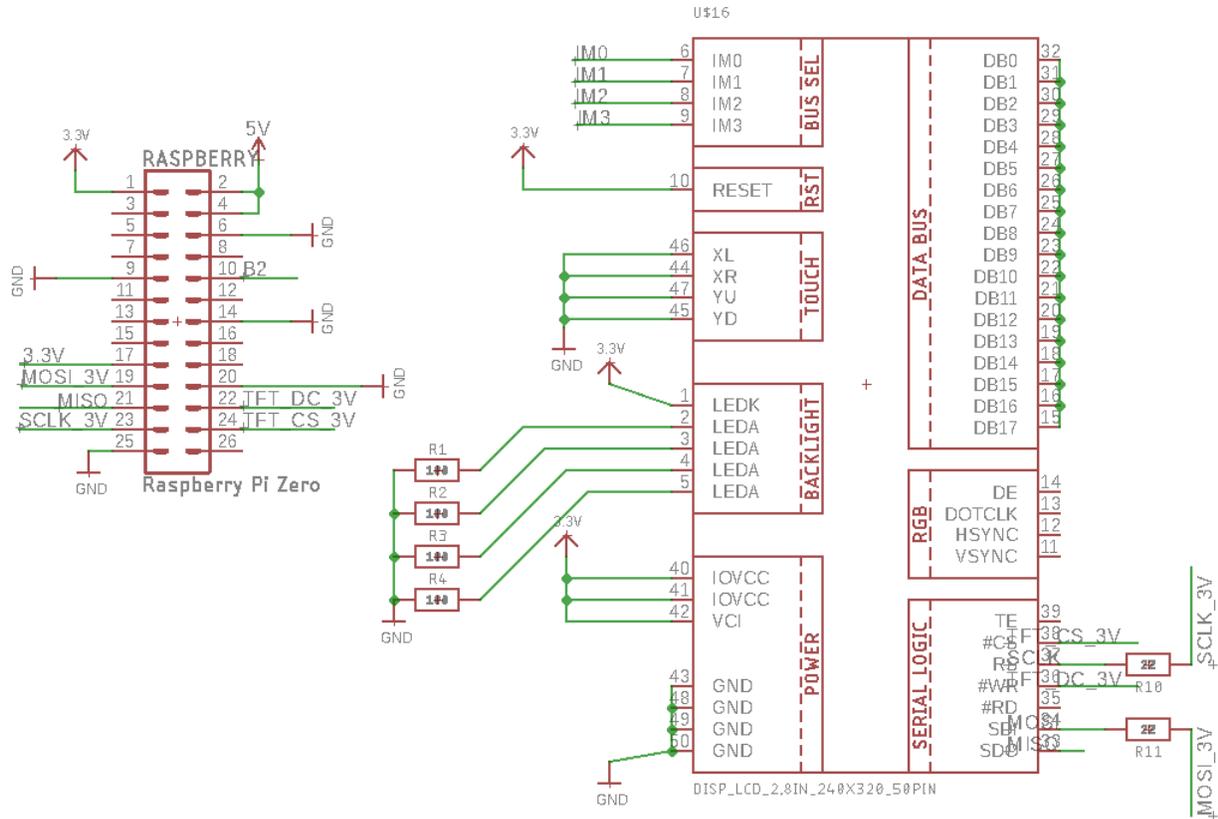


Figura 12: Conexión SPI

## 5. Diseño de PCB

Con estas modificaciones se procedió a rediseñar los PCB utilizando el software EAGLE, se rediseñaron los PCB para mandarlas a hacer y poder ensamblar un prototipo, de acuerdo a los circuitos mostrados en las figuras 13 y 14.

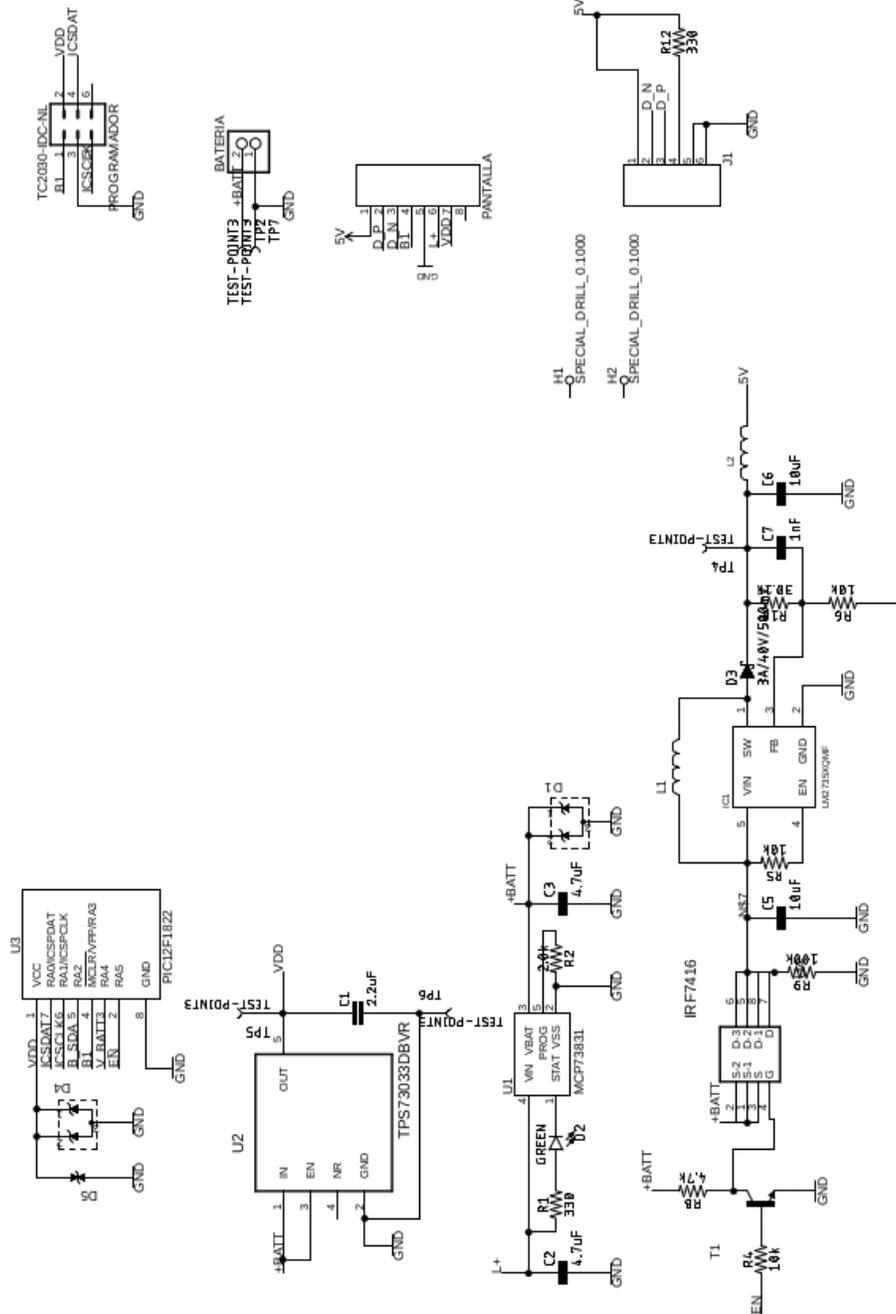


Figura 13: Esquemático PCB Mango

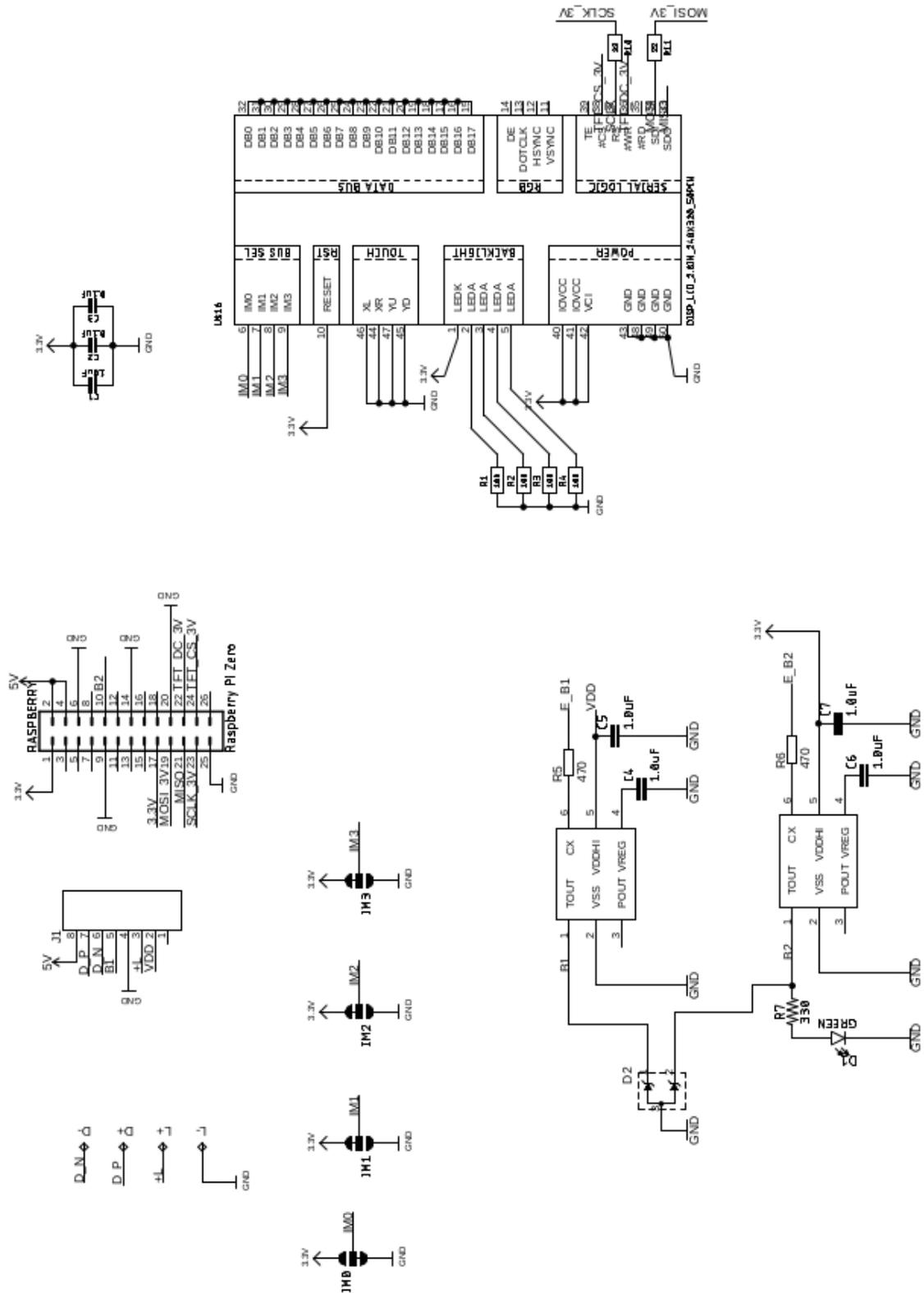


Figura 14: Esquemático PCB Pantalla

Los diseños más recientes de las tarjetas electrónicas, se muestran en las figuras 15, 16, 17 y 18.

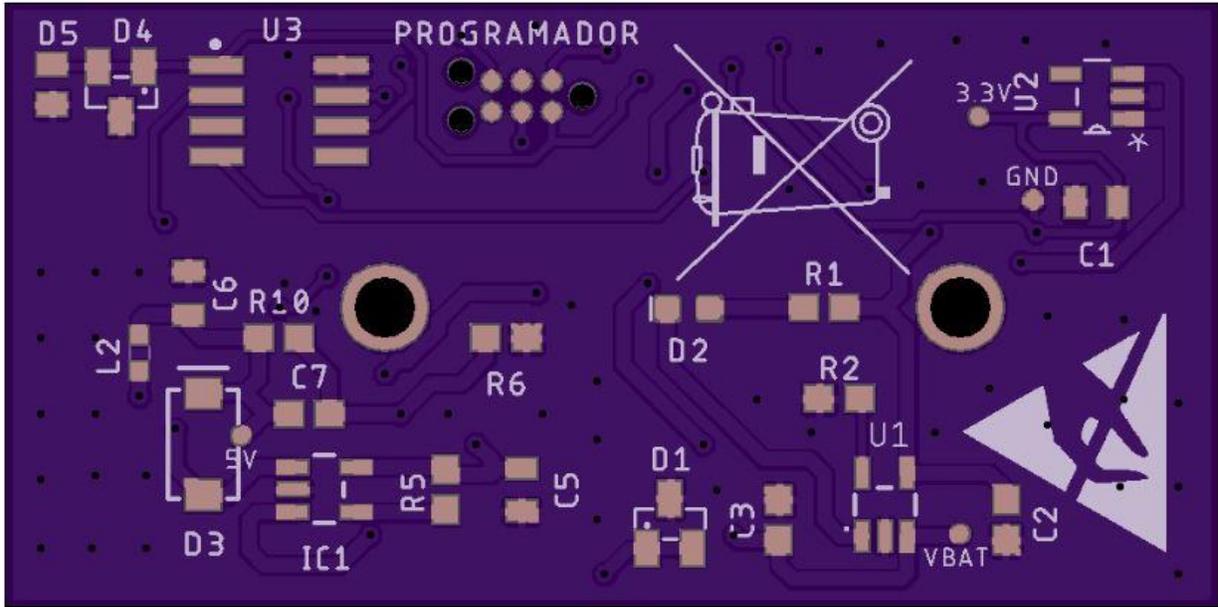


Figura 15: PCB mango top

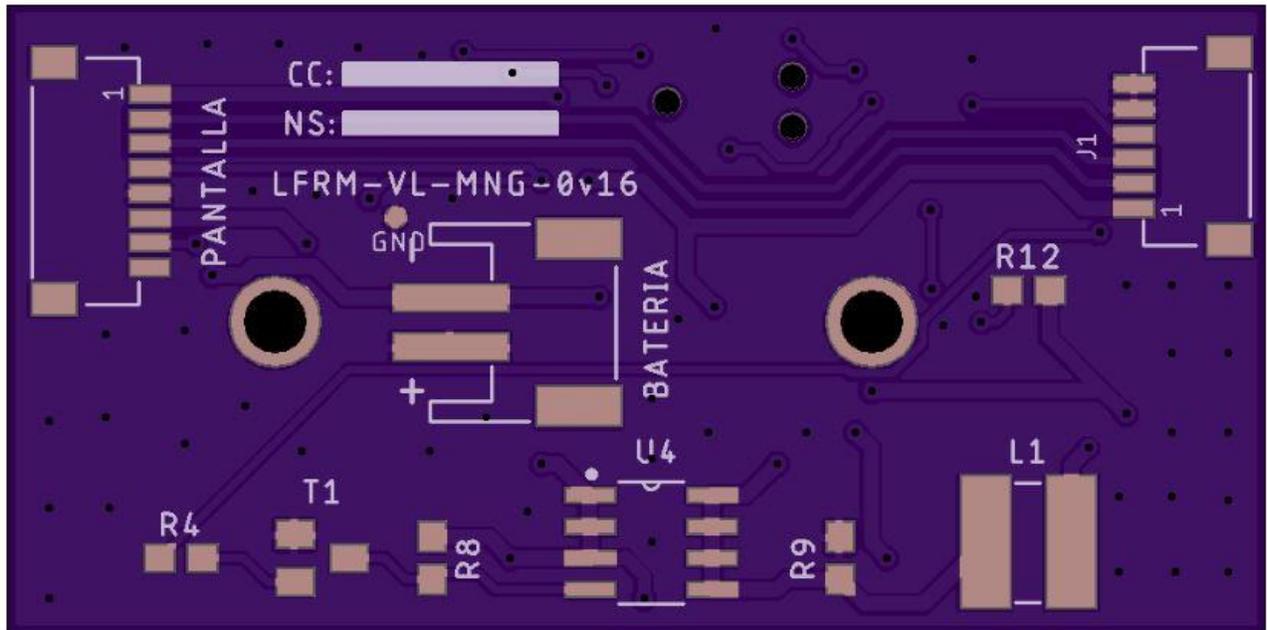


Figura 16: PCB mango bottom

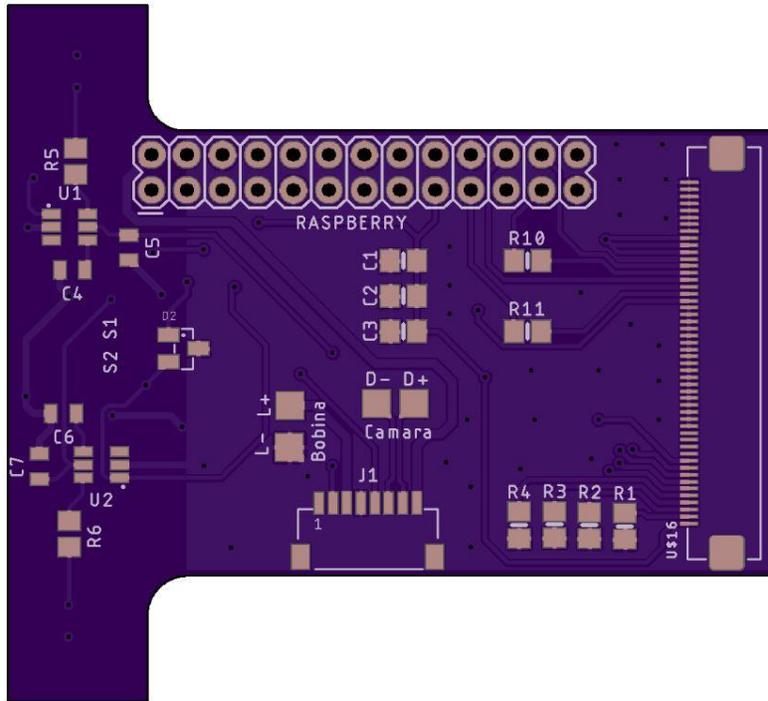


Figura 17: PCB pantalla top

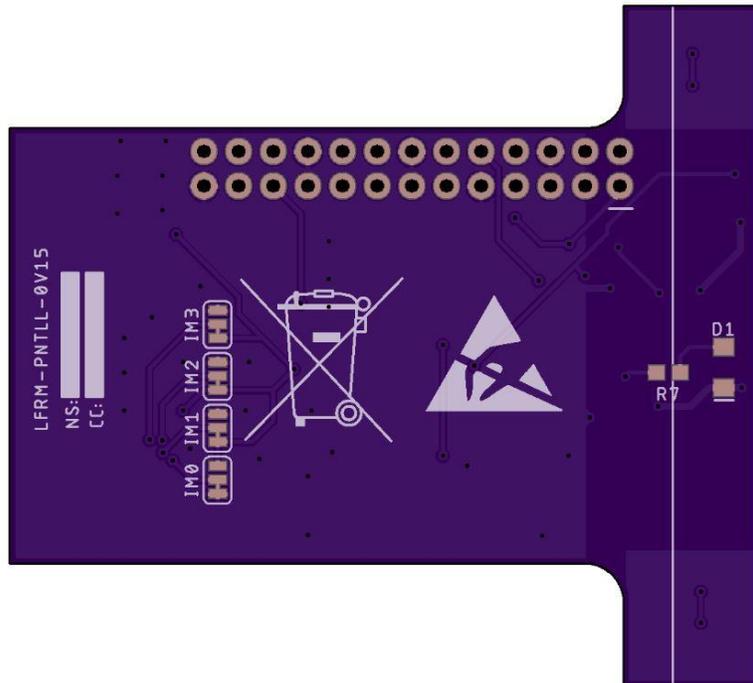


Figura 18: PCB pantalla bottom

## 6. Firmware

Ya con estas modificaciones se comenzó a trabajar en el software del dispositivo, en la Raspberry Pi se instaló Raspbian, el cuál es un sistema operativo basado en Debian que distribuye la Raspberry Pi Foundation, el primer paso fue entablar la comunicación con el display, ya que por default la salida de video del dispositivo es a través del puerto HDMI, para lo cual se comienza por habilitar la comunicación por SPI desde las opciones de interface de la Raspberry; después se descargó el software mediante los comandos:

```
cd ~
```

```
wget https://raw.githubusercontent.com/adafruit/Raspberry-Pi-InstallerScripts/master/adafruit-pitft.sh
```

```
chmod +x adafruit-pitft.sh
```

```
sudo ./adafruit-pitft.sh
```

Al ejecutar el script de instalación del display, se procede a configurar el hardware, de acuerdo a lo mostrado en la figura 19, seleccionando el tamaño del display a utilizar, la orientación del display, si se quiere que se ejecute la terminal o que se ejecute la interfaz gráfica.

```
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 2
Select rotation:
1. 90 degrees (landscape)
2. 180 degrees (portait)
3. 270 degrees (landscape)
4. 0 degrees (portait)

SELECT 1-4: 1
[PITFT] Checking init system...
Found systemd
/boot is mounted
[PITFT] System update
Updating apt indexes...
.....
Reading package lists...
.....
[PITFT] Installing Python libraries & Software...
Installing Pre-requisite Software...This may take a few minutes!
```

Figura 19: Configuración display LCD [Anexo 7.4]

Ya con el display funcionando de manera correcta, se creó un script en python que tomara fotos al presionar un botón, como primer paso se instala el paquete de fswebcam mediante el comando:

```
sudo apt-get install fswebcam
```

Con lo cual, ya está lista la Raspberry para usar la cámara USB con el comando:

```
fswebcam nombredelaimagen.jpg
```

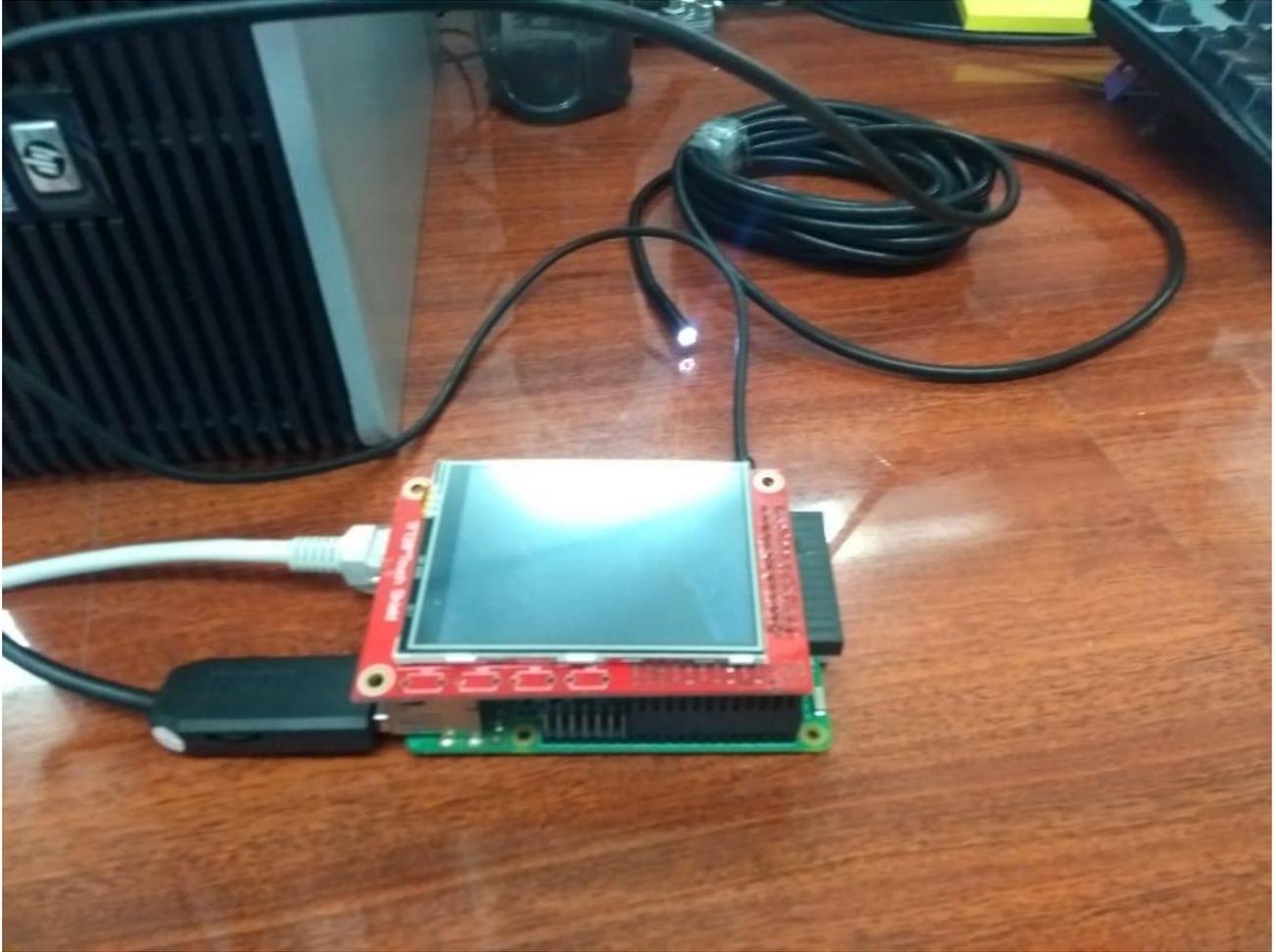
Esto se implementa en un script de python de manera que cuando en uno de los GPIO se detecta un nivel lógico alto, se tome de manera automática la foto.

Para poder ver el video que captura la cámara en el display, se analizaron varias opciones, como hacer el stream a una página web y visualizarlo desde el navegador Chromium, al final se optó por hacer uso del paquete gstreamer, el cual hace la captura y transmisión del video mediante tuberías, en los cuales se pueden agregar filtros, seleccionar la resolución y ajustes del video, seleccionar las entradas, especificar las salidas, etc, para ver el video, se hace el siguiente comando:

```
gst-launch-1.0 -v v4l2src device=/dev/video0 ! video/x-raw, format=BGR, width=320, height=240, framerate=30/1 ! videoconvert ! fbdevsink device=/dev/fb1
```

## 7. Hardware

En las figuras 20, 21 y 22 se observa el prototipo del sistema de video en distintas etapas; en la figura 20 se puede observar el primer prototipo realizado, esto para demostrar que era factible la realización de este proyecto con los componentes seleccionados, se utilizó una Raspberry Pi 3B, un adaptador con un display LCD, y una cámara USB para endoscopia.



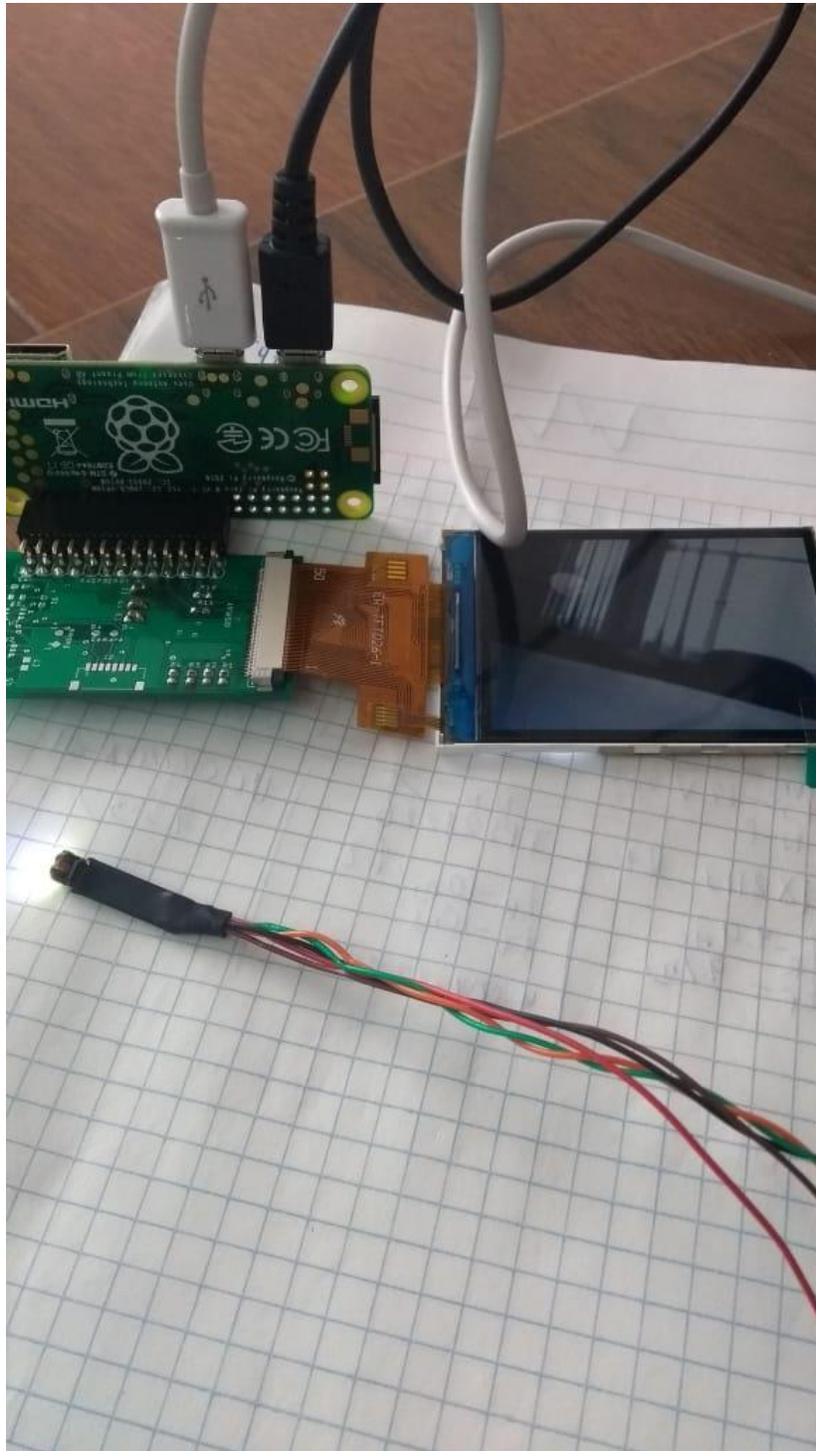
*Figura 20: Prototipo 1 del sistema de video*

En la figura 21 se observa el siguiente prototipo, para este se siguió utilizando la Raspberry Pi 3B, pero ya implementando el diseño propuesto de las tarjetas electrónicas y una cámara USB de propósito general.



Figura 21: Prototipo 2 del sistema de video

En la figura 22, se observa un prototipo cercano al final, donde ya se utiliza la Raspberry Pi Zero W, la cual se utilizara en el prototipo final.



*Figura 22: Prototipo 3 del sistema de video*

## 4. Resultados

### 1. Fuente Boost

Se diseñó y ensambló el convertidor DC-DC planteado, y se obtuvieron los resultados mostrados en la figura 23. La línea azul es el voltaje de entrada, y la línea verde es el voltaje de salida.

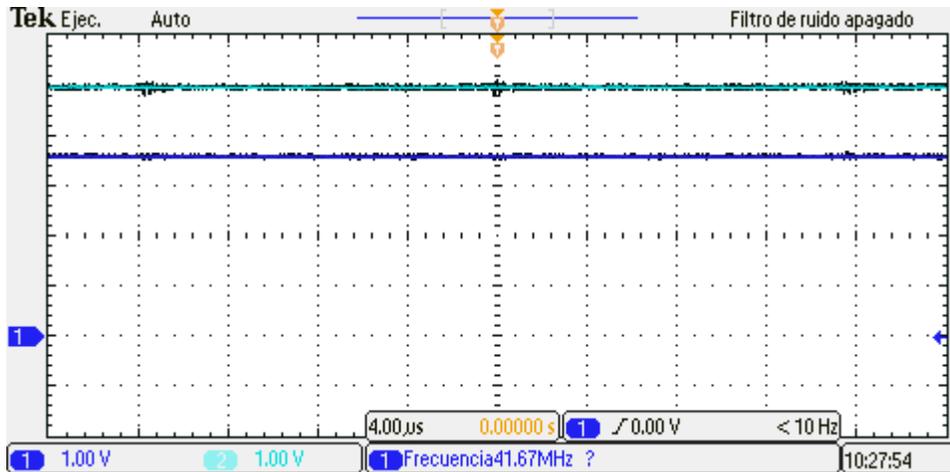


Figura 23: Funcionamiento de la fuente Boost

Se ensambló el arreglo de transistores propuesto para encender la fuente Boost y de esa manera evitar que aun cuando no está funcionando se tenga el voltaje de la batería en la salida de la misma. En la figura 24, se muestra el funcionamiento de dicho arreglo, teniendo que la línea azul es el voltaje del Enable que manda el microprocesador al encender el dispositivo, y la línea verde es el voltaje de salida en el source del MOSFET.

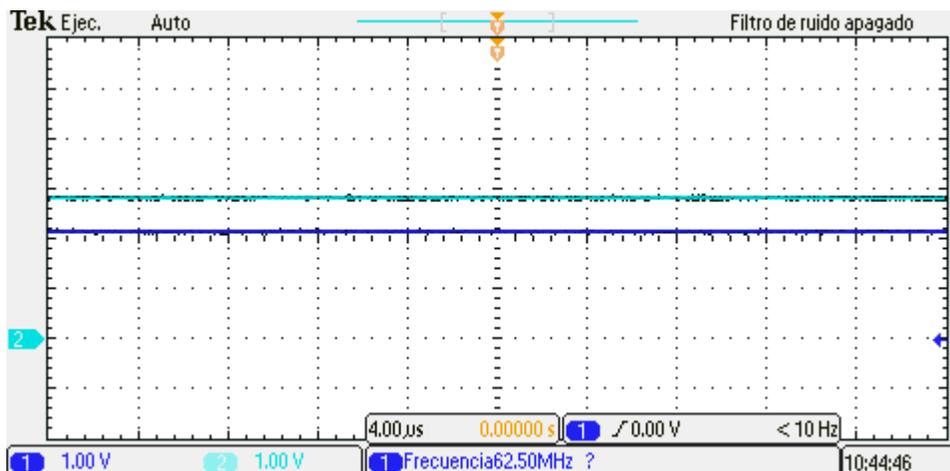


Figura 24: Funcionamiento de los transistores de switcheo

## 2. Sensores de toque capacitivos

Se ensamblaron los sensores capacitivos para implementarlos como botones táctiles, en la figura 25, se observa la respuesta de este sensor al ser presionado con el dedo a través de un acrílico de 4 mm. de espesor.

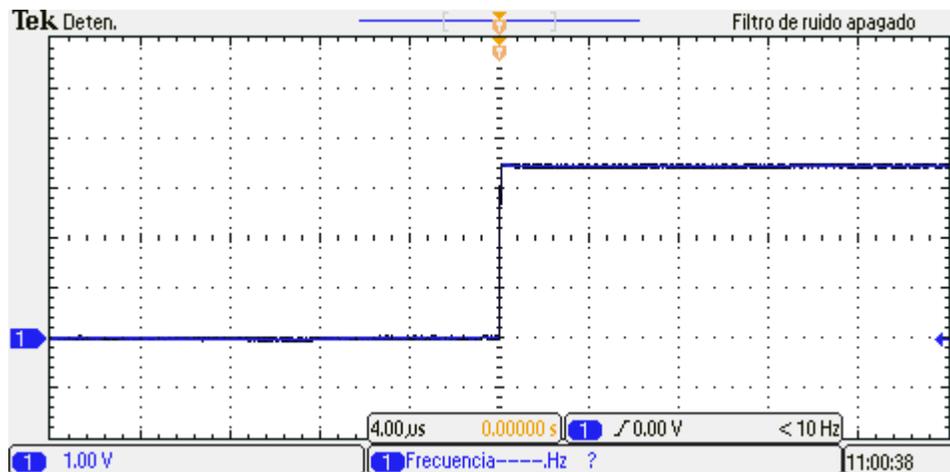


Figura 25: Encendido de sensor de toque

## 5. Conclusiones

- Se diseñó una fuente boost que eleve el voltaje de la batería a los 5 V necesarios para que el sistema funcione de manera adecuada. Esto mediante la implementación del circuito propuesto en las notas de aplicación del LM2735 [Anexo 7.1].
- Se implementaron los botones táctiles de acuerdo a lo mostrado en la hoja de datos del IQS127D [Anexo 7.2] y en la figura 14. Se ensambló el circuito y se comprobó su funcionamiento adecuado, tanto para encender y apagar el sistema, como para tomar fotos.
- Se realizó la correcta conexión del display a la micro-computadora Raspberry Pi, permitiendo de esta manera el correcto despliegue del video capturado por la cámara, además de proporcionar una interfaz gráfica en la cual fue posible trabajar de manera más eficiente en el desarrollo del sistema.
- Se realizaron las configuraciones necesarias en el sistema operativo Raspbian para que al encender el sistema se muestre un logo y después inicialice la visualización del video capturado por la cámara en el display, además de habilitar la opción de tomar fotos.
- Se ensamblaron todos los subsistemas en las PCB y se tuvo un funcionamiento correcto de estas como sistema completo.
- Al final se logró tener un prototipo ensamblado y listo para realizar pruebas, el resultado de estas pruebas dieron como resultado que se debe mejorar el tema de la resolución y la nitidez de la pantalla, aunque esto implica el cambio del display por uno de mayor resolución.
- Otra de las áreas en las que se debe mejorar es en el tiempo de encendido del sistema, ya que, desde que se alimenta el sistema, hasta que se muestra el video en el display transcurren alrededor de 30 segundos, se desea que este tiempo sea como máximo de 15 segundos.

## 6. Fuentes de información

- [1] MUHAMMAD Harunur Rashid  
Electrónica de Potencia, circuitos, dispositivos y aplicaciones  
2a. edición  
México  
Prentice-Hall, 1995
- [2] BOYLESTAD, Robert y Nashelsky, Louis  
Electronic devices and circuit theory  
5a. edición  
New Jersey  
Prentice Hall, 1992
- [3] Intelligent Power Ics  
Melbourne, FLA  
Harris semiconductors, 1992
- [4] Larry K. Baxter (1996).  
Capacitive Sensors.  
John Wiley and Sons.
- [5] "TFT LCD - Electronic Aspects of LCD TVs and LCD Monitors". Plasma.com  
Revisado el 13/09/2018

# 7. Anexos

## 1. Nota de Aplicación SNVA255A



### 4.3 Soft-Start

This function forces  $V_{OUT}$  to increase at a controlled rate during start up. During soft-start, the error amplifier's reference voltage ramps to its nominal value of 1.255V in approximately 4.0ms. This forces the regulator output to ramp up in a more linear and controlled fashion, which helps reduce inrush current.

### 4.4 Inductor Selection

The Duty Cycle (D) can be approximated quickly using the ratio of output voltage ( $V_O$ ) to input voltage ( $V_N$ ):

$$\frac{V_{OUT}}{V_N} = \left( \frac{1}{1-D} \right) = \frac{1}{D} \tag{1}$$

Therefore:

$$D = \frac{V_{OUT} - V_{IN}}{V_{OUT}} \tag{2}$$

Power losses due to the diode (D1) forward voltage drop, the voltage drop across the internal NMOS switch, the voltage drop across the inductor resistance ( $R_{DCR}$ ) and switching losses must be included to calculate a more accurate duty cycle (See Calculating Efficiency and Junction Temperature for a detailed explanation). A more accurate formula for calculating the conversion ratio is:

$$\frac{V_{OUT}}{V_N} = \frac{\eta}{D} \tag{3}$$

Where  $\eta$  equals the efficiency of the LM2735 application.

The inductor value determines the input ripple current. Lower inductor values decrease the size of the inductor, but increase the input ripple current. An increase in the inductor value will decrease the input ripple current.

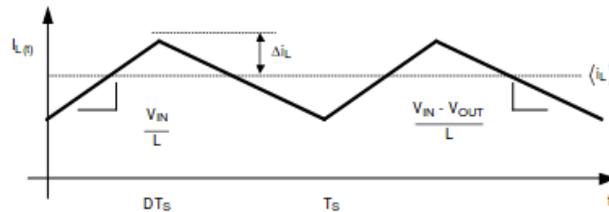


Figure 5. Inductor Current

$$\frac{2\Delta i_L}{DT_S} = \left( \frac{V_N}{L} \right)$$

$$\Delta i_L = \left( \frac{V_N}{2L} \right) \times DT_S \tag{4}$$

A good design practice is to design the inductor to produce 10% to 30% ripple of maximum load. From the previous equations, the inductor value is then obtained.

$$L = \left( \frac{V_N}{2 \times \Delta i_L} \right) \times DT_S \tag{5}$$

Where:  $1/T_S = F_{SW}$  = switching frequency

## 2. Datasheet IQS127D



IQ Switch®  
ProxSense® Series



### 3 Packaging and Pin-out

The IQS127D is available in a TSOT23-6 package.

#### 3.1 Pin-out

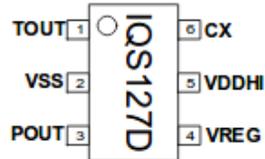


Figure 3.1 Pin-out of IQS127D package

Table 3.1 Pin-out description

IQS127D			
Pin	Name	Type	Function
1	TOUT	Digital Out	Touch Output
2	VSS	Ground	GND Reference
3	POUT	Digital Out	Proximity Output
4	VREG	Analogue Output	Internal Regulator Pin
5	VDDHI	Supply Input	Supply Voltage Input
6	CX	Analogue I/O	Sense Antenna

#### 3.1.2 Schematic

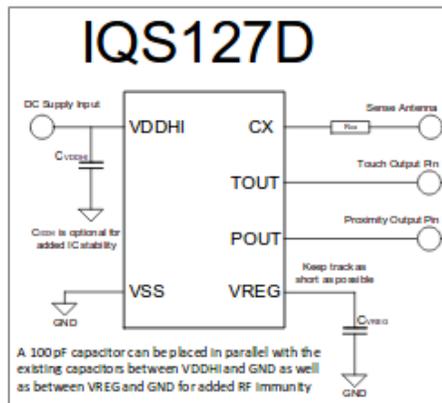


Figure 3.2 Typical application schematic of IQS127D.

#### 3.1.3 Typical values

Component	Value
$C_{VREG}$	1 $\mu$ F
$R_{CX}$	470 $\Omega$ (typical)
$C_{VDDHI}$	1 $\mu$ F



## 4 User Configurable Options

The IQS127D provides One Time Programmable (OTP) user options (each option can be modified only once). The device is fully functional in the default (unconfigured) state. OTP options are intended for specific applications.

The configuration of the device can be done on packaged devices or in-circuit. In-circuit configuration may be limited by values of external components chosen.

A number of standard device configurations are available (refer to Table 10.1). Azoteq can supply pre-configured devices for large quantities.

### 4.1 Configuring of Devices

Azoteq offers a Configuration Tool (CTxxx) and accompanying software (USBProg.exe) that can be used to program the OTP user options for prototyping purposes. More details regarding the configuration of the device with the USBProg program is explained by application note: "AZD007 – USBProg Overview" which can be found on the Azoteq website.

Alternate programming solutions of the IQS127D also exist. For further enquiries regarding this matter please contact Azoteq at ProxSenseSupport@azoteq.com or the local distributor.

**Table 4-1: User Selectable Configuration Options: Bank 0**

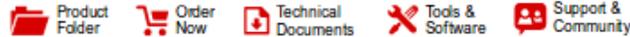
T <sub>THR1</sub>	T <sub>THR0</sub>	P <sub>THR1</sub>	P <sub>THR0</sub>	FUNC1	FUNC0	LOGIC	~
bit 7							bit 0
Bank 1: bit 0, Bank 0: bit 7-6	<b>T<sub>THR</sub>&lt;2:0&gt;</b> : Touch Thresholds				-Section 6.4		
	000 = 1/16 001 = 1/32 010 = 2/16 011 = 3/16 100 = 4/16 101 = 6/16 110 = 8/16 111 = 10/16						
Bank0: bit 5-4	<b>P<sub>THR</sub>&lt;1:0&gt;</b> : Proximity Thresholds				-Section 6.3		
	00 = 2 01 = 4 10 = 8 11 = 16						
Bank0: bit 3-2	<b>FUNC&lt;1:0&gt;</b> : OUTPUT Pins' functions				-Section 6.2		
	00 = POUT active, TOUT active 01 = POUT latch, TOUT active 10 = POUT active, TOUT toggle 11 = POUT latch, TOUT toggle						
Bank0: bit 1	<b>LOGIC</b> : Output logic select -				-Section 6.1		
	0 = Active Low 1 = Active High						
Bank0: bit 0	<b>Not Used</b>						



Table 4-2: User Selectable Configuration Options: Bank 1

STREAMING	-	SHORT STREAMING	t <sub>HALT1</sub>	t <sub>HALT0</sub>	P <sub>MODE1</sub>	P <sub>MODE0</sub>	T <sub>THR2</sub>
bit 7							bit 0
Bank 1: bit 7	<b>STREAMING:</b> 1-wire streaming mode 0 = disabled 1 = enabled				-Section 7		
Bank1: bit 6	<b>Not used</b>						
Bank1: bit 5	<b>SHORT STREAMING:</b> Short word streaming enable (Function enabled if this bit together with STREAMING bit is set) –Section 7						
Bank1: bit 4-3	<b>t<sub>HALT&lt;1:0&gt;</sub>:</b> Halt time of Long Term Average 00 = 18.6 seconds 01 = 74.5 seconds 10 = Never 11 = Always				-Section 6.6		
Bank1: bit 2-1	<b>P<sub>MODE&lt;1:0&gt;</sub>:</b> Power Modes 00 = Boost Mode 01 = Normal Power Mode 10 = Low Power Mode 1 11 = Low Power Mode 2				-Section 6.5		
Bank1: bit 0	<b>T<sub>THR&lt;2:0&gt;</sub>:</b> Touch Thresholds See Table 4-1				-Section 6.4		

### 3. Datasheet LM2735



LM2735

SNVS4851 – JUNE 2007 – REVISED 5 SEPTEMBER 2018

## LM2735 520-kHz and 1.6-MHz Space-Efficient Boost and SEPIC DC/DC Regulator

### 1 Features

- Input Voltage Range: 2.7 V to 5.5 V
- Output Voltage Range: 3 V to 24 V
- 2.1-A Switch Current Over Full Temperature Range
- Current-Mode Control
- Logic High Enable Pin
- Ultra-Low Standby Current of 80 nA in Shutdown
- 170-mΩ NMOS Switch
- ±2% Feedback Voltage Accuracy
- Ease-of-Use, Small Total Solution Size
  - Internal Soft Start
  - Internal Compensation
  - Two Switching Frequencies
  - 520 kHz (LM2735-Y)
  - 1.6 MHz (LM2735-X)
  - Uses Small Surface Mount Inductors and Chip Capacitors
  - Tiny SOT-23, WSON, and MSOP-PowerPAD™ Packages
- Create a Custom Design Using the LM2735 With WEBENCH® Power Designer

### 2 Applications

- LCD Display Backlighting For Portable Applications
- OLED Panel Power Supply
- USB-Powered Devices
- Digital Still and Video Cameras
- White LED Current Source
- For Automotive see LM2735-Q1

### 3 Description

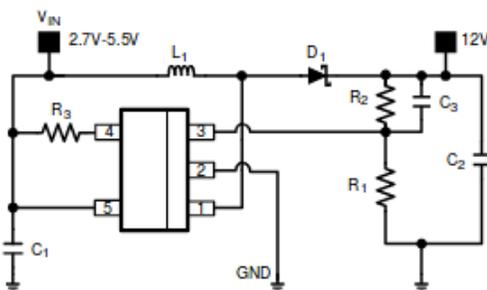
The LM2735 device is an easy-to-use, space-efficient 2.1-A low-side switch regulator, ideal for Boost and SEPIC DC/DC regulation. The device provides all the active functions to provide local DC-DC conversion with fast-transient response and accurate regulation in the smallest PCB area. Switching frequency is internally set to either 520 kHz or 1.6 MHz, allowing the use of extremely small surface mount inductor and chip capacitors, while providing efficiencies of up to 90%. Current-mode control and internal compensation provide ease-of-use, minimal component count, and high-performance regulation over a wide range of operating conditions. External shutdown features an ultra-low standby current of 80 nA, ideal for portable applications. Tiny SOT-23, WSON, and MSOP-PowerPAD packages provide space savings. Additional features include internal soft start, circuitry to reduce inrush current, pulse-by-pulse current limit, and thermal shutdown.

#### Device Information<sup>(1)</sup>

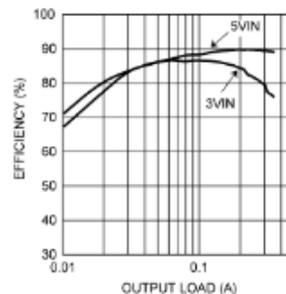
PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM2735	WSON (6)	3.00 mm × 3.00 mm
	SOT-23 (5)	1.60 mm × 2.90 mm
	MSOP PowerPAD (8)	3.00 mm × 3.00 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Typical Boost Application Circuit



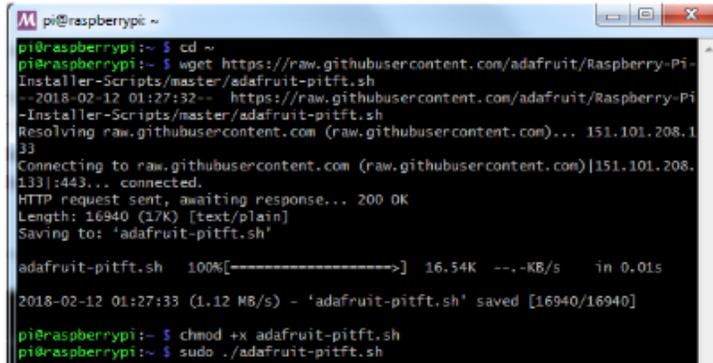
Efficiency vs Load Current  $V_O = 12\text{ V}$



**⚠** An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

## 4. Adafruit PiTFT

```
cd ~
wget https://raw.githubusercontent.com/adafruit/Raspberry-Pi-Installer-Scripts/master/adafruit-pitft.sh
chmod +x adafruit-pitft.sh
sudo ./adafruit-pitft.sh
```



```
pi@raspberrypi:~$ cd ~
pi@raspberrypi:~$ wget https://raw.githubusercontent.com/adafruit/Raspberry-Pi-
Installer-Scripts/master/adafruit-pitft.sh
--2018-02-12 01:27:32-- https://raw.githubusercontent.com/adafruit/Raspberry-Pi
-Installer-Scripts/master/adafruit-pitft.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.208.1
33
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)[151.101.208.
133]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 16940 (17K) [text/plain]
Saving to: 'adafruit-pitft.sh'

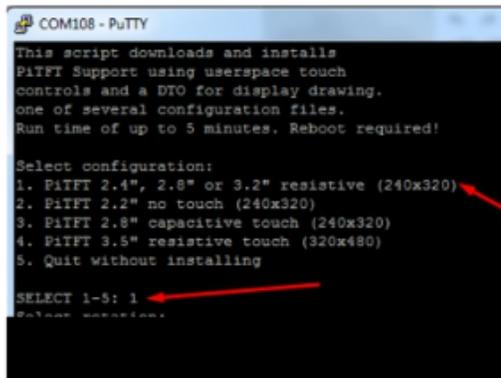
adafruit-pitft.sh  100%[=====] 16.54K  --,-KB/s   in 0.01s

2018-02-12 01:27:33 (1.12 MB/s) - 'adafruit-pitft.sh' saved [16940/16940]

pi@raspberrypi:~$ chmod +x adafruit-pitft.sh
pi@raspberrypi:~$ sudo ./adafruit-pitft.sh
```

### PiTFT Selection

Once you run it you will be presented with menus for configuration.



```
COM108 - PuTTY
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 1
```

For the 2.4", 2.8" and 3.2" PiTFT with resistive  
touchscreen overlay select #1

```
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 2
```

For the 2.2" PiTFT select #2

```
pi@raspberrypi: ~
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 3
```

For the 2.8" Capacitive PiTFT select #3

```
COM108 - PuTTY
This script downloads and installs
PiTFT Support using userspace touch
controls and a DTO for display drawing.
one of several configuration files.
Run time of up to 5 minutes. Reboot required!

Select configuration:
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 4
Select rotation:
```

For the 3.5" PiTFT select #4

### Rotation

Next you will be asked for the rotation you want, don't worry if you're not 100% sure which you want, you can always change this later by re-running the script

```
SELECT 1-5: 2
Select rotation:
1. 90 degrees (landscape)
2. 180 degrees (portrait)
3. 270 degrees (landscape)
4. 0 degrees (portrait)
SELECT 1-4: 1
```

It will take a few minutes to install the software and download all the things...

```
pi@raspberrypi: ~
1. PiTFT 2.4", 2.8" or 3.2" resistive (240x320)
2. PiTFT 2.2" no touch (240x320)
3. PiTFT 2.8" capacitive touch (240x320)
4. PiTFT 3.5" resistive touch (320x480)
5. Quit without installing

SELECT 1-5: 2
Select rotation:
1. 90 degrees (landscape)
2. 180 degrees (portrait)
3. 270 degrees (landscape)
4. 0 degrees (portrait)

SELECT 1-4: 1
[PITFT] Checking init system...
Found system
/boot is mounted
[PITFT] System update
Updating apt indexes...
.....
Reading package lists...
.....
[PITFT] Installing Python libraries & Software...
Installing Pre-requisite Software...This may take a few minutes!
```

## Configuring what shows where

You have a few different ways to set up the PiTFT, we ask 2 questions to figure out what you want

### PiTFT as Text Console (best for Raspbian 'Lite')

This is the simplest to set-up type of use. Its great if you have a simple text based or pygame/SDL based interface. If you want the PiTFT to act as a text console you can expect:

- HDMI will be 'deactivated' - nothing appears on the HDMI output but a black screen
- The login prompt appears on the Pi
- The Pi is all text, not a GUI (no PIXEL desktop)
- Keyboard and mouse are used only by the PiTFT interface
- Framebuffer-capable software (such as **fbi** for displaying images, **mplayer** for videos, or pygame software, etc) appear on the PiTFT
- OpenGL accelerated software *will not appear on the PiTFT* (it is unaccelerated framebuffer only)
- But, non-OpenGL-accelerated graphics software is a bit faster than using HDMI mirroring (not tons faster but you're not running **fbcp** which will always make it faster)

If you want that say **Yes** to the question **Would you like the console to appear on the PiTFT display**

## 5. Datasheet AT42QT1011

### AT42QT1011

**Legend:** I = Input only, O = Output only, push-pull, I/O = Input/output, OD = Open drain output, P = Ground or power

#### 1.2.2 8-pin UDFN/USON

**Table 1-2. Pin Listing**

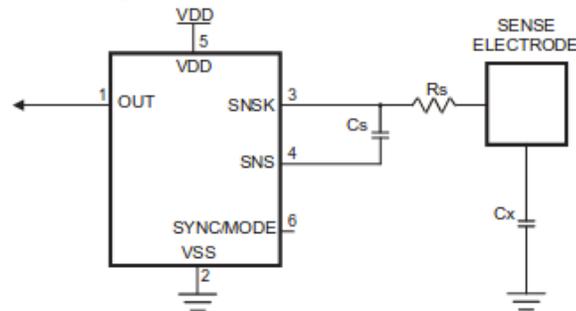
Name	Pin	Type	Comments	If Unused, Connect To...
SNSK	1	I/O	Sense pin	Cs + Key
N/C	2	—	No connection	—
N/C	3	—	No connection	—
VSS	4	P	Supply ground	—
OUT	5	O	Output state	—
SYNC/MODE	6	I	SYNC and Mode Input	Pin is either SYNC/Slow/Fast Mode, depending on logic level applied (see <a href="#">Section 3.1</a> )
VDD	7	P	Power	—
SNS	8	I/O	Sense pin	Cs

**Legend:** I = Input only, O = Output only, push-pull, I/O = Input/output, OD = Open drain output, P = Ground or power

### 1.3 Schematics

#### 1.3.1 6-pin SOT23-6

**Figure 1-1. Basic Circuit Configuration**



Note: A bypass capacitor should be tightly wired between Vdd and Vss and kept close to pin 5.

## **2. Overview of the AT42QT1011**

### **2.1 Introduction**

The AT42QT1011 is a digital burst mode charge-transfer sensor that is capable of detecting near-proximity or touch, making it ideal for implementing touch controls.

With the proper electrode and circuit design, the self-contained digital IC will project a touch or proximity field to several centimeters through any dielectric like glass, plastic, stone, ceramic, and even most kinds of wood. It can also turn small metal-bearing objects into intrinsic sensors, making them responsive to proximity or touch. This capability, coupled with its ability to self-calibrate, can lead to entirely new product concepts.

The QT1011 is designed specifically for human interfaces like control panels, appliances, toys, lighting controls, or anywhere a mechanical switch or button may be found. It includes all hardware and signal processing functions necessary to provide stable sensing under a wide variety of changing conditions. Only a single low-cost capacitor is required for operation.

### **2.2 Basic Operation**

Figure 1-1 and Figure 1-2 show basic circuits.

The QT1011 employs bursts of charge-transfer cycles to acquire its signal. Burst mode permits power consumption in the microamp range, dramatically reduces RF emissions, lowers susceptibility to EMI, and yet permits excellent response time. Internally the signals are digitally processed to reject impulse noise, using a “consensus” filter which requires four consecutive confirmations of a detection before the output is activated.

The QT switches and charge measurement hardware functions are all internal to the QT1011.

### **2.3 Electrode Drive**

For optimum noise immunity, the electrode should only be connected to SNSK.

In all cases, the rule  $C_s \gg C_x$  must be observed for proper operation; a typical load capacitance ( $C_x$ ) ranges from 5–20 pF while  $C_s$  is usually about 2–50 nF.

Increasing amounts of  $C_x$  destroy gain; therefore, it is important to limit the amount of stray capacitance on both SNS terminals. This can be done, for example, by minimizing trace lengths and widths, and keeping these traces away from power or ground traces or copper pours.

The traces and any components associated with SNS and SNSK will become touch sensitive and should be treated with caution to limit the touch area to the desired location.

A series resistor,  $R_s$ , should be placed in line with SNSK to the electrode to suppress ESD and EMC effects.

### **2.4 Sensitivity**

#### **2.4.1 Introduction**

The sensitivity on the QT1011 is a function of things like the value of  $C_s$ , electrode size and capacitance, electrode shape and orientation, the composition and aspect of the object to be sensed, the thickness

## 6. Datasheet MCP73831



# MCP73831/2

## Miniature Single-Cell, Fully Integrated Li-Ion, Li-Polymer Charge Management Controllers

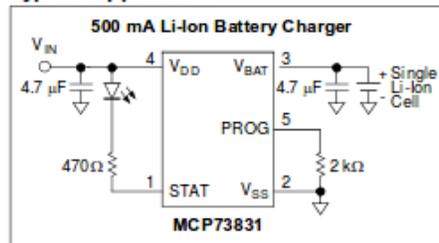
### Features

- Linear Charge Management Controller:
  - Integrated Pass Transistor
  - Integrated Current Sense
  - Reverse Discharge Protection
- High Accuracy Preset Voltage Regulation:  $\pm 0.75\%$
- Four Voltage Regulation Options:
  - 4.20V, 4.35V, 4.40V, 4.50V
- Programmable Charge Current: 15 mA to 500 mA
- Selectable Preconditioning:
  - 10%, 20%, 40%, or Disable
- Selectable End-of-Charge Control:
  - 5%, 7.5%, 10%, or 20%
- Charge Status Output
  - Tri-State Output - MCP73831
  - Open-Drain Output - MCP73832
- Automatic Power-Down
- Thermal Regulation
- Temperature Range:  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$
- Packaging:
  - 8-Lead, 2 mm x 3 mm DFN
  - 5-Lead, SOT-23

### Applications

- Lithium-Ion/Lithium-Polymer Battery Chargers
- Personal Data Assistants
- Cellular Telephones
- Digital Cameras
- MP3 Players
- Bluetooth Headsets
- USB Chargers

### Typical Application



### Description:

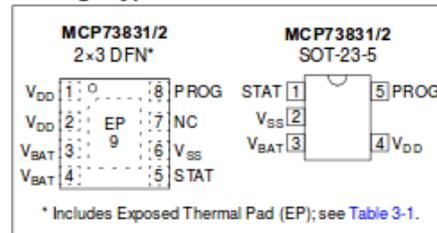
The MCP73831/2 devices are highly advanced linear charge management controllers for use in space-limited, cost-sensitive applications. The MCP73831/2 are available in an 8-Lead, 2 mm x 3 mm DFN package or a 5-Lead, SOT-23 package. Along with their small physical size, the low number of external components required make the MCP73831/2 ideally suited for portable applications. For applications charging from a USB port, the MCP73831/2 adhere to all the specifications governing the USB power bus.

The MCP73831/2 employ a constant-current/constant-voltage charge algorithm with selectable preconditioning and charge termination. The constant voltage regulation is fixed with four available options: 4.20V, 4.35V, 4.40V or 4.50V, to accommodate new, emerging battery charging requirements. The constant current value is set with one external resistor. The MCP73831/2 devices limit the charge current based on die temperature during high power or high ambient conditions. This thermal regulation optimizes the charge cycle time while maintaining device reliability.

Several options are available for the preconditioning threshold, preconditioning current value, charge termination value and automatic recharge threshold. The preconditioning value and charge termination value are set as a ratio, or percentage, of the programmed constant current value. Preconditioning can be disabled. Refer to **Section 1.0 "Electrical Characteristics"** for available options and the **"Product Identification System"** for standard options.

The MCP73831/2 devices are fully specified over the ambient temperature range of  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$ .

### Package Types



## 5.0 DETAILED DESCRIPTION

### 5.1 Analog Circuitry

#### 5.1.1 BATTERY MANAGEMENT INPUT SUPPLY (V<sub>DD</sub>)

The V<sub>DD</sub> input is the input supply to the MCP73831/2. The MCP73831/2 automatically enter a Power-Down mode if the voltage on the V<sub>DD</sub> input falls below the UVLO voltage (V<sub>STOP</sub>). This feature prevents draining the battery pack when the V<sub>DD</sub> supply is not present.

#### 5.1.2 CURRENT REGULATION SET (PROG)

Fast charge current regulation can be scaled by placing a programming resistor (R<sub>PROG</sub>) from the PROG input to V<sub>SS</sub>. The program resistor and the charge current are calculated using the following equation:

$$I_{REG} = \frac{1000V}{R_{PROG}}$$

Where:

R<sub>PROG</sub> = kOhms  
I<sub>REG</sub> = milliampere

The preconditioning trickle charge current and the charge termination current are ratiometric to the fast charge current based on the selected device options.

#### 5.1.3 BATTERY CHARGE CONTROL OUTPUT (V<sub>BAT</sub>)

The battery charge control output is the drain terminal of an internal P-channel MOSFET. The MCP73831/2 provide constant current and voltage regulation to the battery pack by controlling this MOSFET in the linear region. The battery charge control output should be connected to the positive terminal of the battery pack.

### 5.2 Digital Circuitry

#### 5.2.1 STATUS INDICATOR (STAT)

The charge status output of the MCP73831 has three different states: High (H), Low (L), and High-Impedance (Hi-Z). The charge status output of the MCP73832 is open-drain, and, as such, has two different states: Low (L), and High-Impedance (Hi-Z). The charge status output can be used to illuminate 1, 2, or tri-color LEDs. Optionally, the charge status output can be used as an interface to a host microcontroller.

Table 5-1 summarize the state of the status output during a charge cycle.

**TABLE 5-1: STATUS OUTPUT**

Charge Cycle State	STAT 1	
	MCP73831	MCP73832
Shutdown	Hi-Z	Hi-Z
No Battery Present	Hi-Z	Hi-Z
Preconditioning	L	L
Constant-Current Fast Charge	L	L
Constant Voltage	L	L
Charge Complete – Standby	H	Hi-Z

#### 5.2.2 DEVICE DISABLE (PROG)

The current regulation set input pin (PROG) can be used to terminate a charge at any time during the charge cycle, as well as to initiate a charge cycle or initiate a recharge cycle.

Placing a programming resistor from the PROG input to V<sub>SS</sub> enables the device. Allowing the PROG input to float or by applying a logic-high input signal, disables the device and terminates a charge cycle. When disabled, the device's supply current is reduced to 25 μA, typically.