

# Bluetooth Speaker

FYI: I am free to change some of the designs/ICs below as long as It does the same job

I am working on a Bluetooth speaker please check [GitHub](#) the goal is to make it open source. I have done most of the Hardware part but still facing some issues with Software. I have done some of the software check [here](#) I have figure out some of the issues. I am using Arduino IDE to program the **atmega328**.

1. First, I am going to use the [TPR54](#) to turn ON/OFF the system **atmega328** [I/O\_2] if it pressed for <2s . This module can send either active High/Low signal output as desire. I will explain later why I need this specific module.

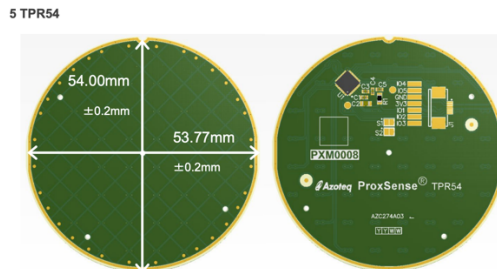


Figure 5-1: TPR54 – Front and Back View.

## 5.1 Outputs of TPR54

The pin mapping for the TPR54 gesture outputs are shown in below.

Table 5-1: Module Gesture Output Pin Mapping.

Gesture	Output Pin	Output Type	Typical Feature
Swipe Upward	I/O_1	Single Pulse	Volume Increase
Swipe Upward & Hold	I/O_1	Continuous Pulse	Continuous Volume Increase
Swipe Downward	I/O_3	Single Pulse	Volume Decrease
Swipe Downward & Hold	I/O_3	Continuous Pulse	Continuous Volume Decrease
Swipe Forward	I/O_5	Single Pulse	Skip / Next Track
Swipe Forward & Hold	I/O_5	Continuous Pulse	Fast Forward
Swipe Backward	I/O_4	Single Pulse	Skip / Previous Track
Swipe Backward & Hold	I/O_4	Continuous Pulse	Reverse/Rewind
Tap	I/O_2	Single Pulse	Play/Pause (Call Answer)
Tap & Hold	I/O_2	Continuous Pulse	Start/STOP

2. After the wake up the **atmega328** going to send high signal to a load switch [TPS22810](#) to supply the [TAS5825](#). Also, another high signal to turn on the [LT3042](#) LDO 3V3 to power the Bluetooth Module [BT-806](#) and the [TAS5825](#) circuitry.

- Startup Procedures: [I/O\_2] pressed for <2s
  - The **atmega328** wakes up
  - Bring up power supplies (it does not matter if [TPS22810](#) or [LT3042](#) comes up first).
  - Send 3V3 high signal more than 100ms to Bluetooth Module [BT-806](#) through VREG\_IN (PLAY/PAUSE) pin to boot the module then stop the signal
  - Once power supplies are stable, bring up PDN of [TAS5825](#) to High and wait 5ms (Keep in mind PDN is pulled up to 3V3 normally)
  - Set the [TAS5825](#) into HiZ state and enable DSP via the I2C control port.
  - Wait 5ms at least. Then initialize the DSP Coefficient, then set the [TAS5825](#) to Play state.
  - Display the BAT SOC with 4 LEDs with help of either the [BQ25883](#) or [MAX17044](#)

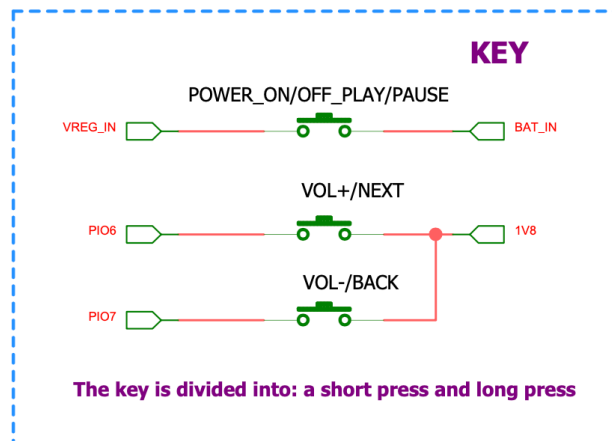
# Bluetooth Speaker

- Shutdown Procedures: **[I/O\_2] pressed for <2s**
  - Configure the Register 0x03h -D[1:0]=10 (Hiz) via the I2C control port or Pull PDN low.
  - Wait at least 6ms (this time depends on the LRCLK rate ,digital volume and digital volume ramp down rate).
  - Bring down power supplies
  - The **atmega328** goes to deep sleep waiting for the next event
- Charging Procedures:
  - Once the charge input is present the **BQ25883** will send an interrupter either through INT, STAT, PG pins
  - Display the BAT SOC with 4 LEDs with help of either the **BQ25883** or **MAX17044** as long as the charge input is present.

[Keep in mind that I seccseuflly check [here](#)]

3. Once the speaker is ON the **TPR54** is responsible to interact with Bluetooth Module **BT-806** to change the song and volume. (Keep in mind that the **TAS5825** can also change the just volume)

There is two ways to do that either directly with pin to pin or through TX/RX UART pin



As you can see just by sending High signal preform specific task. For example: I/O\_2 [swipe up] from the **TPR54** to the PIO6 of the Bluetooth Module **BT-806** should increase the volume [Short press] and so on.

The other way is through UART at 115200 rate data read this [web](#) or download the related [docs](#) . For example: Once the **atmega328** is connected to **BT-806** It can send a command to change the volume say

```
Example: Increase audio speaker volume
<< AT+SPKVOL=+
>> OK
```

# Bluetooth Speaker

And so on. Also, if the BAT SOC is low alert the user with an LED blinking and start the Shutdown Procedures once the BAT is too low to operate the system > 6V or 10%.

## Considerations:

I am not sure yet what charging IC to use I like the [BQ25883](#) as it has OTG to charge phones and I2C communications to read the BAT voltage. Or we can use standalone IC along with a buck converter and a Fuel Gauge [MAX17044](#) to read the SOC state of charge to Alert the **atmega328** of low battery