BITCOIN PRICE ALERTING SYSTEM

# Project Scenario

With the coming of cryptocurrencies in this fast-pacing tech world, it is necessary to ride along with it and not be left behind. Bitcoin is one of the most used crypto-currency whose market value often changes. To get regular checks of the prices, so that one can trade the currency in time to make money and get tremendous profits, this project constantly maintains and sees, if the selling price set is lower than the current market price. If so, then it alerts the user via a buzzer and also updates using Telegram.

## Hardware Connections

* Bolt Wi-Fi Module
* Buzzer
* Two Female-Male wires
* USB cable
* Power Adapter

## Software, Apps and Online Services

* Bolt Cloud
* Linux (Ubuntu)
* Telegram Messaging Application

## System Requirements

* A compatible 64-bit x86/AMD64 CPU launched in 2011 or later \*
* 1.3GHz or faster core speed
* 2GB RAM minimum/ 4GB RAM or more recommended

## Virtual OS: VMware

A Virtual Machine (VM) is a compute resource that uses software instead of a physical computer to run programs and deploy apps. One or more virtual “guest” machines run on a physical “host” machine. Each virtual machine runs its own operating system and functions separately from other VMs, even when they are all running on the same host. Virtual machine technology is used for many use cases across on-premises and cloud environments. VMware Workstation Player is an ideal utility for running a single virtual machine on a Windows or Linux PC. Organizations use Workstation Player to deliver managed corporate desktops, while students and educators use it for learning and training.

# Buzzer Pin ConfigurationBuzzer

An audio signaling device like a beeper or buzzer may be electromechanical or [piezoelectric](https://www.elprocus.com/what-is-a-piezoelectric-material-working/) or mechanical type. The main function of this is to convert the signal from audio to sound. Generally, it is powered through DC voltage and used in timers, alarm devices, printers, alarms, computers, etc.

# Telegram

Telegram is a popular cross-platform messaging app that is widely used because it offers some enhanced privacy and encryption features as well as support for large group chat features. It also has no ties to other social media platforms. The app is multiplatform, with versions available for iOS, Android, Windows, Mac, and Linux. It can also be accessed from a web browser. Telegram was founded by Russian social media entrepreneur Pavel Durov and the service is free to use.

PROS

* **End-to-end encryption**
* **Self-destructing messages**
* **Large file sizes**

CONS

* **Limited user base**
* **New user announcements may violate privacy**
* **It's a gathering place for conspiracy theorists and hate groups**

1. Bitcoin

Bitcoin uses peer-to-peer technology to operate with no central authority or banks; managing transactions and the issuing of bitcoins is carried out collectively by the network. Bitcoin is open-source; its design is public; nobody owns or controls it and everyone can take part. Through many of its unique properties, Bitcoin allows exciting uses that could not be covered by any previous payment system.

* Fast peer-to-peer transactions
* Worldwide payments
* Low processing fees

Pseudonym Satoshi Nakamoto, is said to be behind this digital currency, invented in 2009. Bitcoin protocol is built on the blockchain. It can be defined as the transactions that are held between parties, is done in blocks and each block has a timestamp that cannot be altered or tampered with, making it the world’s safest.

1. Setup

**Step 1:** Setup your Bolt Wi-Fi module.

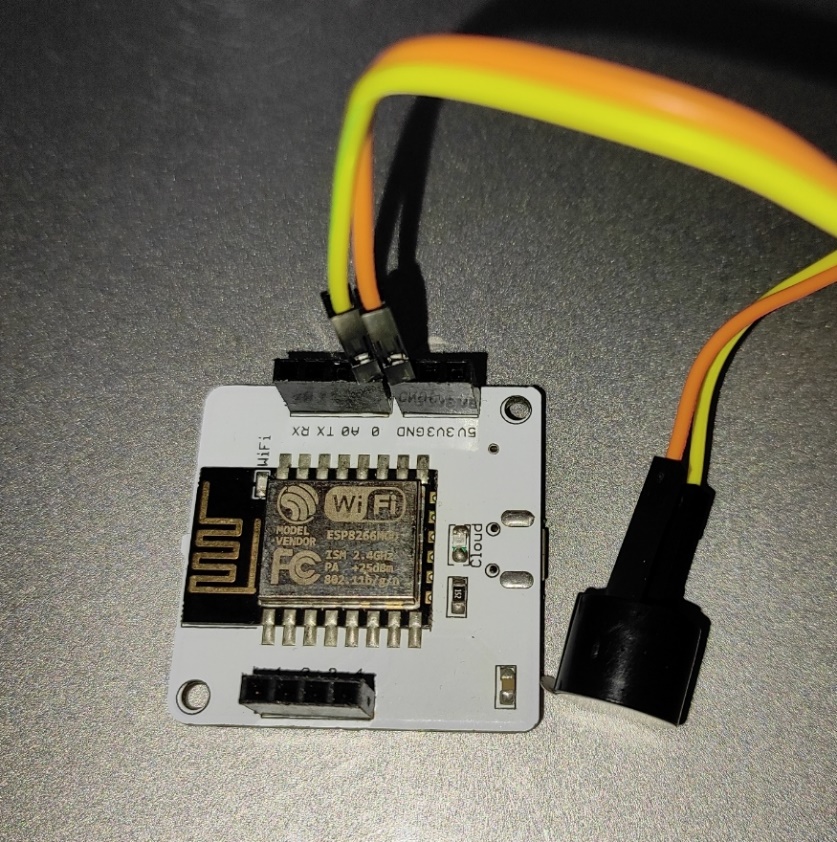
Register your Bolt Wi-Fi Module on Bolt Application.

1. It should show a stable blue and green light, indicating a proper connection.
2. Login to Bolt Cloud.
3. Here, note down the Device ID and API Key of your Bolt Wi-Fi Module that are present under the Devices and API tabs, respectively.

**NOTE:** Keep the Device ID and API Key somewhere safe, as they have to be used later on in the programming section.

**Step 2:** Make the hardware connections.

**NOTE:** Make sure to do the connections before turning ON the power supply for Bolt Wi-Fi Module.

1. Take the Buzzer. The long leg is positive and the shorter leg is negative.
2. Plug the positive leg in pin 0 and the negative leg in GND, of the bolt Wi-Fi module.

**Safety Precaution:** Double check your connections so that there are no loose ends, as it can cause harm to you, the buzzer and your module.

# Implementation

**Step 1:** Setup Telegram Channel

1. Open Telegram application and click on the "Pencil" icon in the bottom-right corner.
2. Click on "New Channel".
3. Enter a Name and Description for your channel.
4. In the next screen, set the channel as "Public" and enter a permanent link for your channel.
5. This link will be used later on to send alerts to this channel.

**Step 2:** Create a Bot

1. Click on the "Magnifying Glass" icon to search for the bot named "BotFather".
2. Click on the verified account of BotFather and type "**/start**" and send the message.
3. Then type "**/newbot**" to create a new bot.
4. Follow the instructions specified in the chat and enter the Bot name and Bot username.
5. After successfully creating a bot, it will provide you with A Token to access the HTTP API. Save it, as it will be used later on the python code.

**Step 3:** Link the Telegram Channel and the Bot.

1. Open your new Telegram Channel. Click on the channel name and go to Administrators tab.
2. Here, search for your newly created bot and select it.

**NOTE:** You have successfully created a telegram channel and a bot for your Bitcoin price alert system.

**Step 4:** Setup the Bolt Python Library

1. Open your Ubuntu Server. It can be VMware, Digital Ocean or Virtual Machine Box. You can also open the terminal in Ubuntu Operating System.

**NOTE:** In case you do not have these software or Operating System then you can download them for free on the internet.

2. Type in the following commands.

sudo apt-get -y update #update packages on Ubuntu

sudo apt install python3-pip #install python3 pip3

sudo pip3 install boltiot #install boltiot library using pip

## Reading from Bolt Device

* 1. Create a new directory.

sudo mkdir crypto\_alert #new directory

cd crypto\_alert #move into directory

* 1. Create a new python file.

sudo nano crypto\_alert.py #new file

import requests # for making HTTP requests

import json # library for handling JSON data

import time # module for sleep operation

from boltiot import Bolt # importing Bolt from boltiot module

SELLING PRICE-3086743.81

API\_KEY = "XXXX"

# This is your Bolt Cloud API Key

DEVICE\_ID= "XXXX"

# This is the device ID and will be similar to BOLTXXXX where XXXX is some numbers

telegram\_chat\_id = "@XXXX"

# This is the channel ID of the created Telegram channel. Paste after @ symbol.

telegram\_bot\_id = "botXXXX"

# This is the bot ID of the created Telegram Bot. Paste after bot text.

mybolt = Bolt(API\_KEY, DEVICE\_ID)

## Check Current Bitcoin Price

## Append this code to "crypto\_alert.py" to send message via Telegram.

def price\_check():

url="https://min-api.cryptocompare.com/data/price"

query\_string=("fsyn":"BTC","tsyns":"INR")

response=requests.request(“GET”, url, params=query\_string)

response=json. loads (response.text) current\_price=response["INR"]

return current\_price

## Send Message via Telegram

Append this code to "crypto\_alert.py" to send message via Telegram.

def send\_telegram\_message(message):

"""Sends message via Telegram"""

url = "https://api.telegram.org/" + telegram\_bot\_id + "/sendMessage"

data = {"chat\_id": elegram\_chat\_id, "text": message}

try:

response = requests.request("POST", url, params=data)

print("This is the Telegram URL")

print(url)

print("This is the Telegram response")

print(response.text)

telegram\_data = json.loads(response.text)

return telegram\_data["ok"]

except Exception as e:

print("An error occurred in sending the alert message via Telegram")

print(e)

return False

## 

## Compare Prices and Send Alert Messages

Append this code to "crypto\_alert.py" to check if the market price is greater than selling price and sends alert.

while True:

market price > price\_check()

profit=market price-SELLING PRICE

print("Current Market price is ", market price) ", SELLING PRICE)

print("Your Selling price is

print("It's the time to trade your Bitcoins!")

time.sleep(4)

if market price > SELLING PRICE:

bolt.digitalWrite("0", "HIGH")

message="Alert! Bitcoin price has gone up the selling price.\n"

str(market price)+" Your selling price is INR "str(SELLING PRICE)

Total profit is INR "str(profit). This is the right time to trade and become

rich."

telegran\_status=send\_telegran\_message (message)

print("This is the Telegram status: ", telegram\_status,"\n\n")

tine.sleep(30)

bolt.digitalWrite("0","LOW")

**NOTE:** Save and exit by pressing "CTRL+X" and then "Y". Hit "ENTER".

**Step 5:** Run the code.

sudo python3 crypto\_alert.py

## Output

