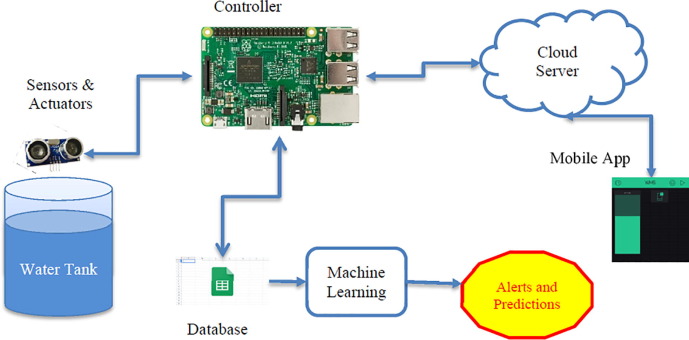
**IOT IMPLEMENTATION**

An IoT smart water meter keeps tabs on the quantity, quality, and pressure of water used in a building or industry. An IoT smart water sensor can be used to track the flow of water across the entire plant and over the distribution channels. Helping in leakage detection, to reduce water wastage.

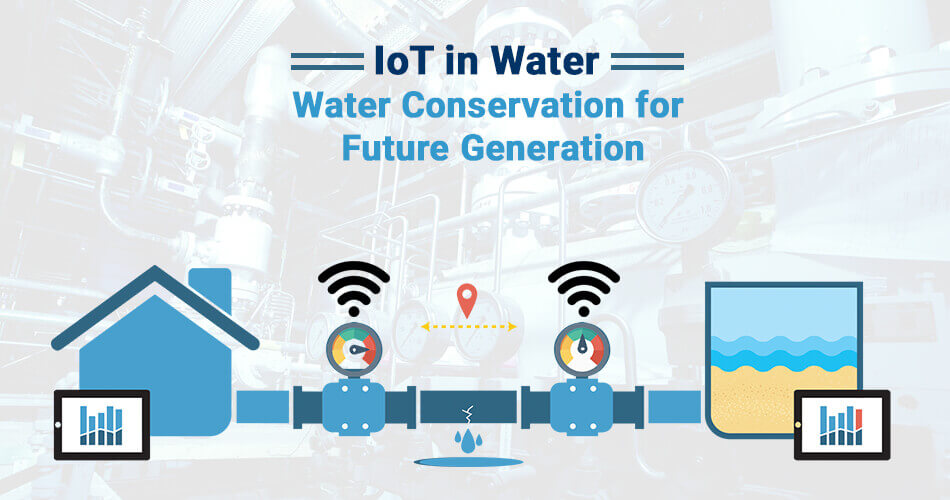


**Methodology**

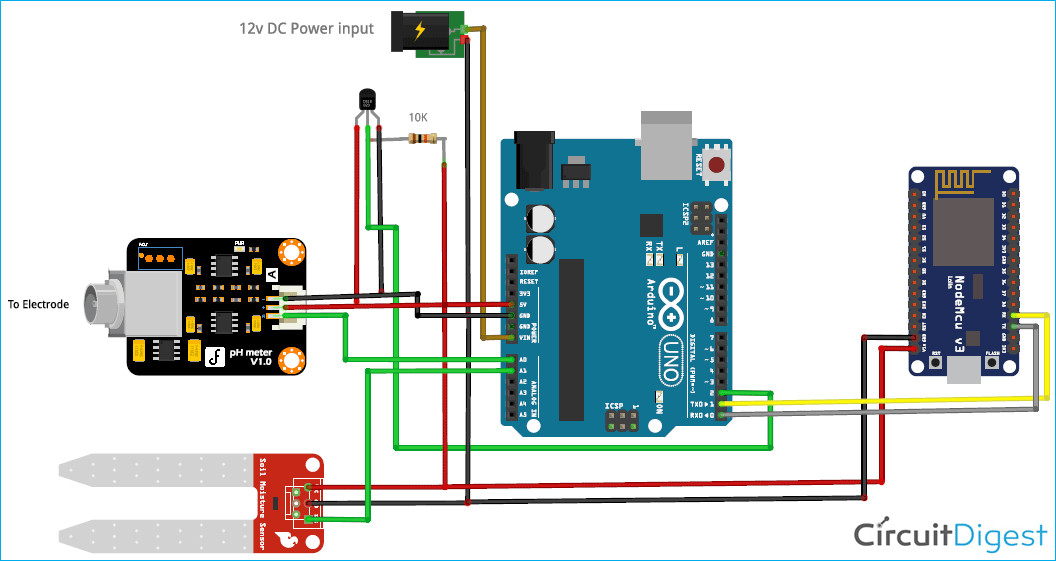
Water turbidity, PH, and temperature are continuously monitored by an autonomous system controlled by a single chip microprocessor. The data are collected, processed, and analyzed on a single chip. Using an IOT environment, the data will be transmitted to a monitoring center while simultaneously alerting the general public.

**Project Description**

Utilizing a variety of IoT technologies to promote transparency and ensure more prudent and sustainable use of these water resources, smart water management is the planning, development, distribution, and management of the use of water resources.



**Circuit Diagram**



**Working Principle of Prototype**

The intelligent controller, which connects to sensors, gathers signals, converts them to digital form, analyses the data, applies algorithms, and then displays the user's water quality data on an HMI or display before sending it to the cloud, is the brains of STREAM.

**Coding**

from flask import Flask, render\_template, request, jsonify

from flask\_sqlalchemy import SQLAlchemy

from datetime import datetime

app = Flask(\_\_name\_\_)

app.config['SQLALCHEMY\_DATABASE\_URI'] = 'sqlite:///water\_usage.db'

db = SQLAlchemy(app)

class WaterUsage(db.Model):

id = db.Column(db.Integer, primary\_key=True)

timestamp = db.Column(db.DateTime, default=datetime.utcnow)

amount = db.Column(db.Float)

db.create\_all()

def receive\_water\_data():

data = request.get\_json()

if 'amount' in data:

new\_water\_usage = WaterUsage(amount=data['amount'])

db.session.add(new\_water\_usage)

db.session.commit()

return jsonify({"status": "success"})

else:

return jsonify({"status": "error", "message": "Invalid data format"}), 400

def get\_water\_data():

water\_data = WaterUsage.query.all()

data = [{"timestamp": entry.timestamp, "amount": entry.amount} for entry in water\_data]

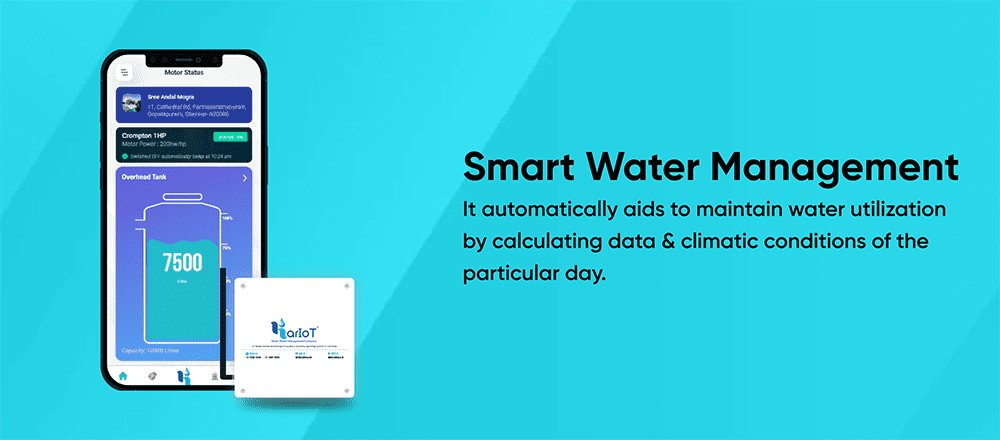
return jsonify(data)

def home():

return render\_template('index.html')

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**App Screenshot**