

```
/*
Sketch generated by the Arduino IoT Cloud Thing "Untitled"
https://create.arduino.cc/cloud/things/8e650f96-8893-40a7-84ff-f1f006ca31ff

Arduino IoT Cloud Variables description

The following variables are automatically generated and updated when changes are made to the T

String boxStatus;
String mailStatus;
float distance;
bool mailboxClosed;
bool status;

Variables which are marked as READ/WRITE in the Cloud Thing will also have functions
which are called when their values are changed from the Dashboard.
These functions are generated with the Thing and added at the end of this sketch.
*/

#include "thingProperties.h"
#define MAX_RANG 520
#define ADC 1023.0

int ultrasonic = A5;
int magnet = 2;
int LED = 4;

float d1;
float threshold = 4.0;

void setup() {
  Serial.begin(9600);
  delay(1500);
  pinMode(magnet, INPUT);
  pinMode(LED, OUTPUT);

  d1 = 43;

  initProperties();
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
}

float sensity;
boolean inside;

void loop() {
  ArduinoCloud.update();
  sensity = analogRead(ultrasonic);
  mailboxClosed = digitalRead(magnet);

  distance = (sensity * MAX_RANG / ADC);
```

```
Serial.print("Reading... ");
Serial.print(distance);
Serial.println(" cm");

if (mailboxClosed) {
    digitalWrite(LED, LOW); // Mailbox closed
    status = true;
    d1 = distance;
    delay(500);
} else {
    digitalWrite(LED, HIGH); // Mailbox open
    status = false;
    delay(500);
}

if (status == true) { // Only check status when mailbox is open
    boxStatus = "Mailbox is close";
    distance = (sensity * MAX_RANG / ADC);
    if (d1 > 400) {
        Serial.println("Mailbox is here");
        mailStatus = "Mail is here";
        inside = true;
    } else {
        Serial.println("Someone took your mail");
        mailStatus = "Mailbox is empty";
        inside = false;
    }
} else {
    boxStatus = "Mailbox is open";
}

// Double security
if (status == false && distance < 400){
    mailStatus = "Mailbox is empty";
} else if (status == false && distance > 400){
    mailStatus = "Mail is here";
}

Serial.print("D1 | Current Distance: ");
Serial.print(d1);
Serial.println(" cm");
}

void onDistanceChange() {
    // Add your code here to act upon Distance change
}

void onStatusChange() {
    // Add your code here to act upon Status change
}

void onMailStatusChange() {
    //prevStatus = status;
}
```

```
94  /*
95     Since BoxStatus is READ_WRITE variable, onBoxStatusChange() is
96     executed every time a new value is received from IoT Cloud.
97  */
98  void onBoxStatusChange() {
99      // Add your code here to act upon BoxStatus change
100 }
101
102 /*
103     Since MailboxClosed is READ_WRITE variable, onMailboxClosedChange() is
104     executed every time a new value is received from IoT Cloud.
105  */
106 void onMailboxClosedChange() {
107     // Add your code here to act upon MailboxClosed change
108 }
109
110
```

PDF document made with CodePrint using [Prism](#)