

Converting digital pin input to byte

timkoers

mai '17 #1

I am working on a project where I have got a byte array (that should represent a 7 digit display output) and the 7 digit display pins are connected to my ESP8266 (with Arduino IDE).

What I'd like to do, is convert the digital pins 0~7 (8 in total) to a byte.

I've been using a boolean array to check whether everything is working, it isn't.

I am missing bits when I am checking what the output of the conversion is. This is my code:

```
//pinMode(0, OUTPUT); Lets the esp crash
//pinMode(1, OUTPUT); Lets the esp crash
pinMode(2, OUTPUT);
pinMode(3, OUTPUT);
pinMode(4, OUTPUT);
pinMode(5, OUTPUT);
pinMode(6, OUTPUT);
//pinMode(7, OUTPUT); Lets the esp crash

digitalWrite(0,HIGH);
digitalWrite(1,HIGH);
digitalWrite(2,HIGH);
digitalWrite(3,HIGH);
digitalWrite(4,LOW);
digitalWrite(5,LOW);
digitalWrite(6,HIGH);
digitalWrite(7,LOW);

for(int i = 0; i < 8; i++)
{
    if(digitalRead(i) == 1){
        bitSet(inputByte, i);
    }
    //bitWrite(inputByte, 9-i, inputBytes[i]);
}

Serial.println("Converting: ");
Serial.println(inputByte, BIN);
```

The Serial outputs:

Converting: 110101

What am I missing out here?

Stitch10925

mai '17 #2

Maybe this post can help you out:

FORUM set 4 pins values a single variable?

Afternoon all, I have been digging on this for a bit, not sure if I am not searching correctly, or if it is not possible. I would like to create a variable that can be called later in the script that sets the output values of four different...

UKHeliBob

mai '17 #3

This is my code:

or at least a very small part of it.

Are you by any chance using Serial input or output in your program ? The reason that I ask is that pins 0 and 1 are used by the Serial interface.

timkoers**mai '17 #4**

UKHeliBob: or at least a very small part of it.

Are you by any chance using Serial input or output in your program ? The reason that I ask is that pins 0 and 1 are used by the Serial interface.

Yeah sorry, the forum was doing very annoying with editing my message. Fixed now

UKHeliBob **mai '17 #5**

What am I missing out here?

As I suspected, you are using pins 0 and 1 which are also used by the Serial interface. Use pins other than 0 and 1

You still have not posted your complete program so it is not obvious that you have set the pinMode(s) of the pins correctly.

When using BIN with Serial.print() it does not print leading zeroes. Try printing 0 or 1 for each individual bit.

Whandall**mai '17 #6**

UKHeliBob: As I suspected, you are using pins 0 and 1 which are also used by the Serial interface. Use pins other than 0 and 1

On an unknown ESP8266 module (not ESP-01, that does not even have 8 io pins) ?

timkoers: I am working on a project where I have got a byte array (that should represent a 7 digit display output) and the 7 digit display pins are connected to **my ESP8266** (with Arduino IDE).

timkoers**mai '17 #7**

Whandall:

On an unknown ESP8266 module (not ESP-01, that does not even have 8 io pins) ?

That is correct, TX and RX are not digital pins 0 and 1. I am using the ESP-12F from Geekcreit.

UKHeliBob:

When using BIN with Serial.print() it does not print leading zeroes. Try printing 0 or 1 for each individual bit.

I didn't know that the Serial.print() function cuts the leading zeroes. Is it doing the same with the ending zeroes?

Whandall**mai '17 #8**

timkoers: I didn't know that the Serial.print() function cuts the leading zeroes. Is it doing the same with the ending zeroes?

Of course not, only insignificant leading zeros are suppressed, no matter what base is choosen.

How many zeros in front would you expect when you execute Serial.print(1) ?

timkoers**mai '17 #9**

Whandall: Of course not, only insignificant leading zeros are suppressed, no matter what base is chosen.

How many zeros in front would you expect when you execute `Serial.print(1)`?

Zero haha

UKHeliBob: As I suspected, you are using pins 0 and 1 which are also used by the Serial interface. Use pins other than 0 and 1

Reading back on the topic, I figured out you are right. Calling `digitalWrite(1, HIGH)`; is setting the TX line high.

I need to use D0 and D1, not 0 and 1.

The code that I currently have works perfectly!

```
byte pins[9] = {  
  D0,  
  D1,  
  D2,  
  D3,  
  D4,  
  D5,  
  D6,  
  D7,  
  D8  
};  
  
for (int i = 0; i < 8; i++)  
{  
  pinMode(pins[i], OUTPUT);  
}  
  
byte inputByte = B00000000;  
digitalWrite(D0, HIGH);  
digitalWrite(D1, HIGH);  
digitalWrite(D2, HIGH);  
digitalWrite(D3, HIGH);  
digitalWrite(D4, LOW);  
digitalWrite(D5, LOW);  
digitalWrite(D6, HIGH);  
digitalWrite(D7, LOW);  
  
for (int i = 0; i < 8; i++)  
{  
  if (digitalRead(pins[i]) == 1) {  
    inputByte |= (1<
```

Output is:

Converting: 1001111 (without the leading 0)

[Back to top](#)[Help Center](#)[Distributors](#)[FOLLOW US](#)[Contact Us](#)[Careers](#)[Trademark & Copyright](#)[Brand Guidelines](#)