

Converting a hexadecimal string to a binary in order to extract specific bits from it

V-1.0.0

=====

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
}  
  
void loop() {  
  
  byte byte7[8];  
  byte variable = 0xAF;  
  Serial.print("\nVariable in HEX: ");  
  Serial.print(variable,HEX);  
  Serial.print("\nVariable in BIN: ");  
  Serial.print(variable,BIN);  
  Serial.println("");  
  
  // SEPARATING BIT FROM BYTE  
  Serial.print(" SEPARATING BIT FROM BYTE\n");  
  
  
  // Count from RSH to LSH  
  
  
  //1st bit of binary xxxxxxxx no.  
  byte mask = 1 << 0;  
  byte answer = (variable & mask) >> 0;  
  Serial.print("\n1st bit->");  
  Serial.print(answer);  
  byte7[0]=answer;  
  
  //2nd bit of binary xxxxxxxx no.  
  mask = 1 << 1;  
  answer = (variable & mask) >> 1;  
  Serial.print("\n2nd bit->");  
  Serial.print(answer);  
  byte7[1]=answer;  
  
  //3rd bit of binary xxxxxxxx no.  
  mask = 1 << 2;  
  answer = (variable & mask) >> 2;
```

```
Serial.print("\n3rd bit->");  
Serial.print(answer);  
byte7[2]=answer;
```

```
//4th bit of binary xxxxxxxx no.  
mask = 1 << 3;  
answer = (variable & mask) >> 3;  
Serial.print("\n4th bit->");  
Serial.print(answer);  
byte7[3]=answer;
```

```
//5th bit of binary xxxxxxxx no.  
mask = 1 << 4;  
answer = (variable & mask) >> 4;  
Serial.print("\n5th bit->");  
Serial.print(answer);  
byte7[4]=answer;
```

```
//6th bit of binary xxxxxxxx no.  
mask = 1 << 5;  
answer = (variable & mask) >> 5;  
Serial.print("\n6th bit->");  
Serial.print(answer);  
byte7[5]=answer;
```

```
//7th bit of binary xxxxxxxx no.  
mask = 1 << 6;  
answer = (variable & mask) >> 6;  
Serial.print("\n7th bit->");  
Serial.print(answer);  
byte7[6]=answer;
```

```
//8th bit of binary xxxxxxxx no.  
mask = 1 << 7;  
answer = (variable & mask) >> 7;  
Serial.print("\n8th bit->");  
Serial.print(answer);  
byte7[7]=answer;
```

```
// PRINT THE ARRAY
```

```
Serial.print("\n*****PRINT THE ARRAY*****\n");  
  
for(int i=0;i<8;i++)  
Serial.print(byte7[i]);
```

```
}
```

V-1.0.1

```
=====
```

```
void setup() {  
  // put your setup code here, to run once:  
  Serial.begin(9600);  
}  
  
void loop() {  
  
  byte byte7[8];  
  byte variable = 0xAF;  
  Serial.print("\nVariabke in HEX: ");  
  Serial.print(variable,HEX);  
  Serial.print("\nVariabke in BIN: ");  
  Serial.print(variable,BIN);  
  Serial.println("");  
  // SEPARATING BIT FROM BYTE  
  Serial.print(" SEPARATING BIT FROM BYTE\n");  
  
  // Count from RSH to LSH  
  
  for(int counter=0;counter<8;counter++){  
    byte mask = 1 << counter;  
    byte answer = (variable & mask) >> counter;  
    byte7[counter]=answer;  
  }  
  
  // PRINT THE ARRAY  
  Serial.print("Output: ");  
  for(int i=0;i<8;i++)  
    Serial.print(byte7[i]);  
  
}
```

V-1.0.2

=====

```
void hextobin(byte byteArr,byte variable); // custom function prototype
```

```
void setup() {  
  Serial.begin(115200);  
}
```

```
void loop() {  
  
  byte byte7[8],canMessage[8];
```

```
  
  byte variable = 0xAF;
```

```
  
  Serial.print("\n");  
  Serial.print(variable,HEX);  
  Serial.print("\n");  
  
  Serial.print(variable,BIN);  
  Serial.print("\n");
```

```
  
  // function calling for separating Bit from HEX  
  hextobin(byte7,variable);
```

```
  // PRINT THE ARRAY
```

```
  Serial.print("Output: ");  
  for(int i=0;i<8;i++)  
    Serial.print(byte7[i]);
```

```
}
```

```
void hextobin(byte byteArr[8],byte variable){
```

```
  // Count from RSH to LSH
```

```
  for(int counter=0;counter<8;counter++){  
    byte mask = 1 << counter;  
    byte answer = (variable & mask) >> counter;  
    byteArr[counter]=answer;  
  }
```

```
}
```