```
* DESIGNED BY ENG.AHMED MOHAMED ABD ELMAWGOUD
* A4ATMEL88@GMAIL.COM
* USE THE IO NAMING ACOORDIG TO drawing
ANALOG BUFFER 2AI 2AQ-LAYOUT.PDF
* FOR ANALOG OUTPUT ,IT PRODUCE OUTPUT VOLTAGE FORM 0 TO 8 VOLT
BY PASSING VALUES FROM 0 TO 80
* FOR ANALOG INPUT , IT READ THE INPUT VOLTAGE FROM O TO 8 VOLT ,
AND RETRURN VALUES FROM 0 TO 80
#include <ESP32 BOARD.h>
#include <ANALOG BUFFER 2AI 2AQ.h>
int ai 0 value ,ai 1 value;
void setup() {
 Serial.begin(115200);
void loop() {
 // AIO
 ai 0 value = read ai0();
                                         //return value from 0 to
80 [voltage from 0 to 8 v , values from 0 to 80]
 Serial.print("AI 0 value = ");
 Serial.println( ai 0 value );
 // AI1
 ai 1 value = read ai1();
                                         //return value from 0 to
80 [voltage from 0 to 8 v , values from 0 to 80]
 Serial.print("AI 1 value = ");
 Serial.println( ai 1 value );
 // AQ0
 write aq0(13); // produce 1.3 v on AQ_0_pin [voltage from 0
to 8 v , values from 0 to 801
 // AQ1
 write aq0(54); // produce 5.4 v on AQ 0 pin [voltage from 0
```

```
// ANALOG BUFFER 2AI 2AQ IS CONNECTED TO CONTROLLER J5 SOCKET
* DESIGNED BY ENG.AHMED MOHAMED ABD ELMAWGOUD
* A4ATMEL88@GMAIL.COM
* functions
                        -- read from 0 to 100 [ 0v to 10v]
* int read ai0()
* int read ail()
* write aq0(int value) -- place from 0v to 10v [numbers from
0 to 100]
* write aq1(int value)
*/
//////// ONLY REPLACE J5 TO
THE CONNECTED SOCKET
#define AI 0 pin j5 pin3
#define AI 1 pin j5 pin4
#define AQ 1 pin j5 pin5
#define AQ 0 pin j5 pin6
//////// int map analog(int
adc value)
int map analog(int adc value)
 int int volt;
                                    ( adc value * 10 ) / 140;
 if (adc value < 140) int volt =
// int volt <= 1 volt</pre>
 else if (adc value < 425) int volt = (adc value * 20) / 425;
// int volt <= 2 volt</pre>
 else if (adc_value < 720) int_volt = (adc_value * 30) / 720;
// int volt <= 3 volt</pre>
 else if (adc value < 990) int volt = (adc value * 40) / 990;
```

to 8 v , values from 0 to 80]

```
// int volt <= 4 volt
 else if (adc value < 1285) int_volt = ( adc_value * 50 ) / 1285;
// int volt <= 5 volt</pre>
 else if (adc value < 1580) int volt = (adc value * 60) / 1580;
// int volt <= 6 volt</pre>
 else if (adc value < 1855) int volt = (adc value * 70) / 1855;
// int volt <= 7 volt
 else if (adc value < 2145) int volt = (adc value * 80) / 2145;
// int volt <= 8 volt</pre>
 else if (adc value < 2300) int_volt = ( adc_value * 85 ) / 2300;
// int volt <= 8.5 volt
 else if (adc value > 2300) int volt = 86; // int volt 0= 8.5 volt
 return int volt;
int read ai1()
int read ai0()
            // return value from 0 to 100 as innput ranges from
0 to 10v
 int adc value = analogRead(AI 0 pin);
 int voltage = map analog(adc value);
 return voltage;
int read ai1()
int read ail() // return value from 0 to 100 as innput ranges from
0 to 10v
 int adc value = analogRead(AI 1 pin);
 int voltage = map analog(adc value);
 return voltage;
void write ao1(int value)
void write aq0(int value) // value from 0 to 100 make output from
0 to 10 volts
```