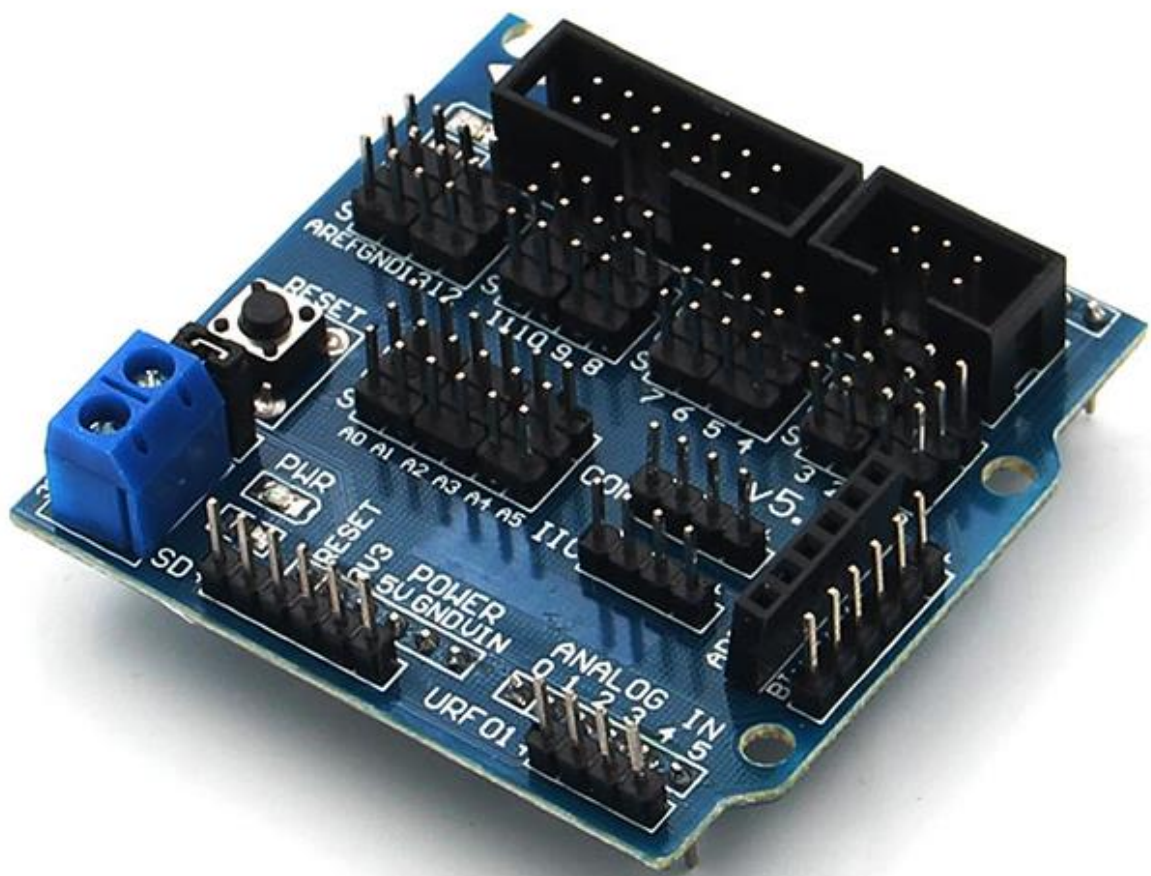




ARDUINO SENSOR SHIELD UNO

User Manual



Arduino Sensor Shield

Introduction

The Arduino Uno sensor shield is very useful as a connection point for the many interfaces you can plug into the Arduino. Using just the Arduino, you very quickly run out 0V and +5V connections for your sensors. Using a sensor shield gives you one +5V (Vcc) and one 0V (Gnd) for every Arduino signal pin. As these can be obtained for under £2.00 I think they are a good investment.

There are 2 versions of the Sensor Shield commonly available, the earlier V4 & the newer V5. While they look different, the important connectors are the same on both versions.

Sensor Shield V5

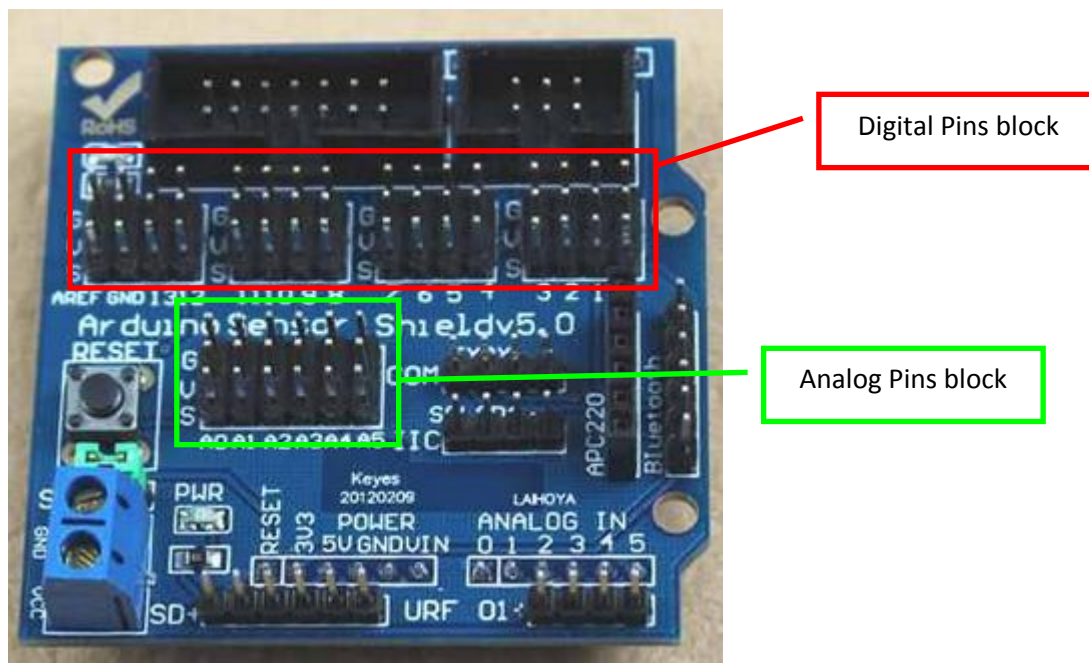


Figure 1: Sensor Shield V5.0

Digital Pins

The pins are arranged in stacks of 3:

- Top = Gnd (0V)
- Middle = Vcc(+5V)
- Bottom = Signal (Arduino Digital Signal Pin No.)

The pins are sequenced from right to left clearly marked on the board:

G			Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd	
V			Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc	
S	Aref	Gnd	13	12	11	10	9	8	7	6	5	4	3	2	1	0

These pins are driven by the Arduino language instruction:

digitalWrite (Pin4,1);

and read by the Arduino language instruction:

digitalRead (Pin4);

Analog Pins

The pins are arranged in stacks of 3: Top = Gnd (0V)
 Middle = Vcc(+5V)
 Bottom = Signal (Arduino Analog Signal Pin No.)

The pins are sequenced from left to right clearly marked on the board:

G	Gnd	Gnd	Gnd	Gnd	Gnd	Gnd
V	Vcc	Vcc	Vcc	Vcc	Vcc	Vcc
S	A0	A1	A2	A3	A4	A5

Sensor Shield V4

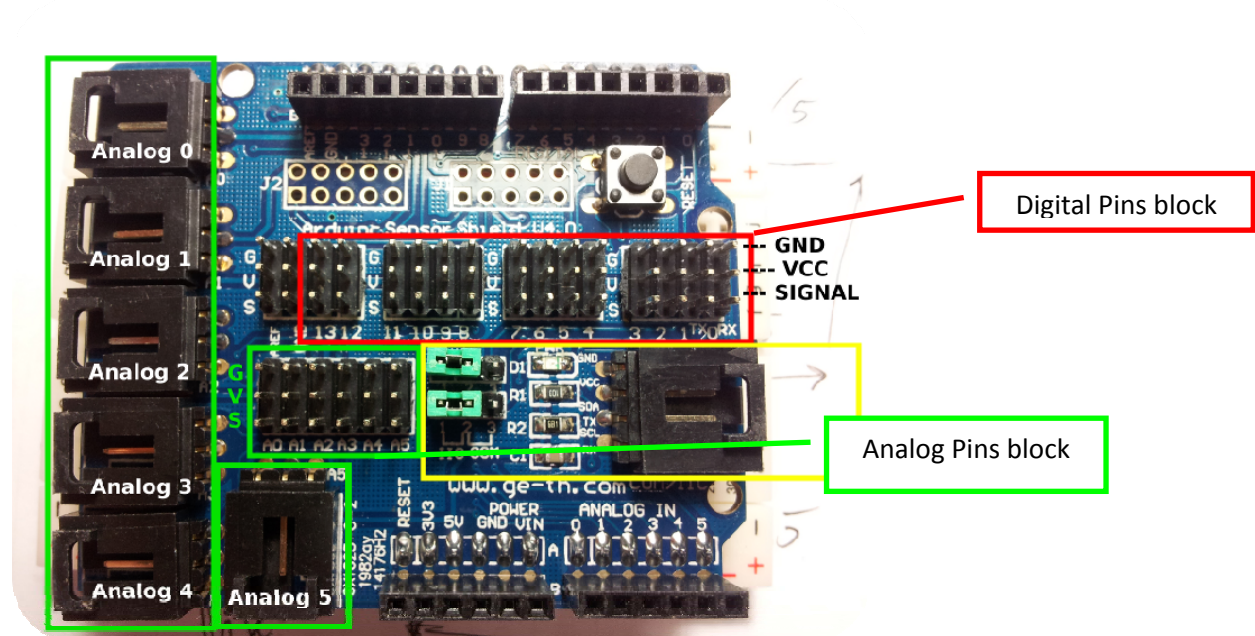


Figure 2: Sensor Shield V4.0

The blocks of pins are the same as the V5.0 Sensor Shield. The other connectors are not as useful as the ones on the V5.0.

Connecting Sensors & Output-Devices

When connecting sensors & output-devices to the Sensor Shield you must make sure to get the power pins the correct way round:

G goes to G or Gnd or GND or 0V on the sensor

V goes to V or Vcc or VCC or +5V on the sensor

S goes to the signal pin - OUT or IN etc..

Some sensors & output-devices will have 2 signal pins (or more) as well as 0V & +5V. For these you just choose one of the signal pins to connect the Signal, 0V and +5V to (on the S, G and V pins) and use just the S pins of another port for the other signal connections.

Some simple sensors e.g. "Photo-resistor Sensor (4-wire)" use 2 wires for power, as above, but have two signal pins, one marked "A0" one marked "D0". These are two versions of the same signal:

- The D0 signal is a digital representation of the light level, but it can only be two different states, logic-high (+5V) or logic-low (0V). The switchover level is set by the variable resistor on the sensor module. This can be adjusted to set the light to dark switchover point. This signal can be connected to a Digital Input on the Sensor Shield/Arduino. This can be read by a `digitalRead` instruction. The signal is 0 for light and 1 for dark. The LED monitor on the module is on for light and off for dark.
- The A0 signal is an analog representation of the light level, this is a voltage anywhere between 0V - maximum light, and 5V - dark. This signal can be connected to an Analog Input on the Sensor Shield/Arduino. This can be read by an `analogRead` instruction. The A0 signal will be read as a value of 0 for maximum light and 1023 for absolute dark.

```
int value = 0;
boolean level = 0;

void setup()
{
  Serial.begin(9600);
  Serial.println("Starting");
}

void loop()
{
  value = analogRead(A3);
  Serial.print(" Analog value = ");
  Serial.print(value);
  Serial.print(" Digital level = ");
  level = digitalRead(3);
  Serial.println(level);
}
```

NOTE the program above uses the `Arduino IDE Serial Monitor`. This can transmit information from inside you program back to a monitor window on your PC:

In the IDE: Tools menu>

Serial Monitor>

At the bottom right of the Serial Monitor window select the correct baud rate – in this case 9600. Once you have finished with the Serial Monitor window close it before you disconnect your Arduino from the PC as the IDE can get muddled up and lose the Arduino port connection.

Connecting Servos

Servos come with a 3 way socket that plug straight onto the Sensor Shield

G goes to the Brown or Black wire

V goes to the (middle) Red wire

S goes to the Orange wire

The good news is as the +5V is in the middle, things will not blow up if you get it the wrong way round, the servo just will not work until you plug it in correctly.