











Sensore	Descrizione	Classe (Libreria GrovePi)	A/D	I/O
Sound Sensor 	E' un semplice microfono e può essere utilizzato per rilevare la forza sonora dell'ambiente. https://www.seeedstudio.com/Grove-Sound-Sensor-p-752.html	GroveSoundSensor • Double get(): recupera l'intensità sonora.	A	I
Light Sensor 	Utilizza una fotoresistenza per rilevare l'intensità della luce dell'ambiente. http://wiki.seeed.cc/Grove-Light-Sensor/	GroveLightSensor • Double get(): recupera l'intensità della luce.	A	I
Rotary Angle Sensor 	Produce un'uscita analogica rappresentante l'angolo di rotazione (compreso tra 0° e 300°) con un cambiamento di valore lineare. http://wiki.seeed.cc/Grove-Rotary-Angle-Sensor/	GroveRotarySensor • GroveRotaryValue get(): recupera un oggetto di tipo GroveRotaryValue che contiene l'angolo di rotazione espresso in gradi, il voltaggio misurato e il valore misurato dal sensore.	A	I
Button 	Bottone. http://wiki.seeed.cc/Grove-Button/	GroveButton • Boolean get(): true/false. • void setListener (GroveButtonListener listener): impostare un "ascoltatore" del click del bottone.	D	I
Led Sensor 	Fornisce una sorgente di luce LED (rosso, blu e verde). http://wiki.seeed.cc/Grove-Red-LED/	GroveLed • void set(boolean value): acceso/spento. • void set(int value): è una sorgente di luce dimmerabile con valori compresi tra 0 e 255.	D	O
Relay 	Questo sensore è un interruttore digitale. Grazie al suo funzionamento è possibile controllare il circuito di altatensione con bassa tensione (ad es. 5V sul controller). http://wiki.seeed.cc/Grove-Relay/	GroveRelay • void set(boolean value): aperto/chiuso.	D	O

Buzzer 	<p>Questo sensore emette un tono. In alternativa può emettere vari toni ed effetti.</p> <p>https://www.seeedstudio.com/Grove-Buzzer-p-768.html</p>	<p>GroveBuzzer</p> <ul style="list-style-type: none"> • void set(boolean value): acceso/spento. • void set(int value): è una sorgente audio dimmerabile con valori compresi tra 0 e 255. <p>D O</p>
Temperature humidity sensor 	<p>Sensore di temperatura e umidità. Il sensore non funziona per temperature inferiori a 0°.</p> <p>http://wiki.seeed.cc/Grove-TemperatureAndHumidity_Sensor/</p>	<p>GroveTemperatureAndHumiditySensor</p> <ul style="list-style-type: none"> • GroveTemperatureAndHumidityValue get(): recupera un oggetto di tipo GroveTemperatureAndHumidityValue che contiene la temperatura e l'umidità misurata. <p>D I</p>
Ultrasonic ranger sensor 	<p>Sensore ad ultrasuoni in grado di misurare la distanza (da 2cm a 350cm).</p> <p>http://wiki.seeed.cc/Grove-Ultrasonic_Ranger/</p>	<p>GroveUltrasonicRanger:</p> <ul style="list-style-type: none"> • Double get(): recupera la distanza degli oggetti circostanti. <p>D I</p>
LCD RGB 	<p>Display LCD testuale (16 x 2) che consente di cambiare il testo e il colore di sfondo. Utilizza le porte I2C per gestire la comunicazione con la scheda GrovePI.</p> <p>http://wiki.seeed.cc/Grove-LCD_RGB_Backlight/</p>	<p>GroveRgbLcd:</p> <ul style="list-style-type: none"> • void setRGB(int r, int g, int b): impostare il colore di sfondo del display specificando le tre componenti colore. • void setText (String text): impostare il testo da visualizzare fino ad un massimo di 32 caratteri. <p>D O</p>

Classe SensorMonitor

Metodo	Descrizione
SensorMonitor (SensorValueSupplier sensor , long readInterval)	<p>Questo metodo rappresenta il costruttore della classe SensorMonitor e crea un monitor sul sensore specificato come parametro di input. Il parametro readInterval ci indica ogni quanto tempo, il sensore specifico, deve acquisire (frequenza del processo di acquisizione del sensore).</p> <pre>GroveRotarySensor rotarySensor = new GroveRotarySensor(grovePi, 1); SensorMonitor<GroveRotaryValue> monitorRotarySensor = new SensorMonitor(rotarySensor, 100);</pre>
boolean start()	Questo metodo è responsabile di far partire il processo di acquisizione del sensore specifico.
R getValue()	Questo metodo ha la funzione di recuperare l'ultimo valore misurato dal sensore specifico. R è un parametro generico che indica il fatto che ogni sensore avrà il suo valore di ritorno (ad es. il rotary sensor ci darà un oggetto GroveRotaryValue, il light sensor ci darà un double , ecc.).
boolean isValid()	Nelle fasi iniziali di acquisizione, il sensore potrebbe fornirci un valore non valido, grazie a questo metodo, ci assicuriamo che il dato recuperato sia valido o meno.
void stop()	Questo metodo ha la funzione di fermare il processo di acquisizione del sensore specifico.

Classe OnOffCommuter

Metodo	Descrizione
<code>OnOffCommuter (GroveDigitalOut actuator, long commutingInterval, long resetInterval)</code>	<p>Questo metodo rappresenta il costruttore della classe <code>OnOffCommuter</code> e crea un commuter sul attuatore specificato come parametro di input.</p> <p>Il parametro <code>commutingInterval</code> ci indica il tempo di commutazione dallo stato on (true) allo stato off (false). Il parametro <code>resetInterval</code> ci indica il tempo dopo il quale l'attuatore ritorna allo stato iniziale (quindi allo stato on).</p> <pre>GroveBuzzer buzzer = new GroveBuzzer(grovePi, 3); OnOffCommuter buzzerCommuter = new OnOffCommuter(buzzer, 100, 200);</pre>
<code>boolean start()</code>	Questo metodo è responsabile di far partire il processo di commutazione dell'attuatore specifico.
<code>boolean getCurrentState()</code>	Questo metodo ha la funzione di recuperare lo stato corrente del commutatore.
<code>void setCommutingInterval (long interval)</code>	Questo metodo permette di modificare il tempo di commutazione specificato dal parametro <code>commutingInterval</code> .
<code>void setResetInterval(long resetInterval)</code>	Questo metodo permette di modificare il tempo di reset specificato dal parametro <code>resetInterval</code> .
<code>void stop()</code>	Questo metodo ha la funzione di fermare il processo di commutazione dell'attuatore specifico.