

ROHM Sensor Shieldの BD1020HFV用ソフトウェア資料

Feb 1, 2016
Sensor Application G

■ スケッチファイルの動作(BD1020.ino)

- setup関数
 - Arduino IDEのSerial Monitorにログを出力するためのSerial関数設定(9600bps)
 - BD1020初期化関数(引数にアナログ端子番号)
- loop関数
 - 温度センサの出力電圧を取得し、ADC後の値と温度値に変換して出力
 - 500msごとに実行

■ ライブラリファイル(BD1020.h, BD1020.cpp)

- コンストラクタ
 - 何もしない
- init関数
 - 引数に指定されたアナログ端子番号を保持
- get_rawval関数
 - init関数で指定したアナログ端子のADC後の値を返す
- get_val関数
 1. get_rawval関数の実行
 2. convert_degree関数の実行
 3. 温度値を返す
- convert_degree関数
 - 温度センサの出力電圧から温度値へ変換

■ 温度センサの出力電圧から温度値変換

- Arduino UnoのADCの基準電圧はデフォルト5V
- Arduino UnoのADCは10bitだから5V=1023 [counts]が最大
- 温度センサの電源電圧を3Vにした場合、ADCの最大は $1024 * 3/5 = 615$ [counts]になる
- 温度センサの出力電圧は以下の式で導出

$$\text{Temp_vout[V]} = (\text{ADC_Value} * 5[\text{V}]) / 1024 \text{ [counts]} - \textcircled{1}$$

• 温度 vs 電圧

- 温度感度と出力電圧値から以下の式が成り立つ(標準値)

$$\text{Vout [mV]} = -8.2[\text{mV}/\text{°C}] * (\text{Temp}[\text{°C}] - 30) + 1300 [\text{mV}]$$

- これを温度出力に整理すると

$$\text{Temp}[\text{°C}] = (-\text{Vout}[\text{mV}] + 1546[\text{mV}]) / 8.2 [\text{mV}/\text{°C}]$$

- さらにVoutに①式を代入して

$$\text{Temp}[\text{°C}] = (-\text{Temp_vout[V]} * 1000 + 1546[\text{mV}]) / 8.2[\text{mV}/\text{°C}] \text{ となる}$$

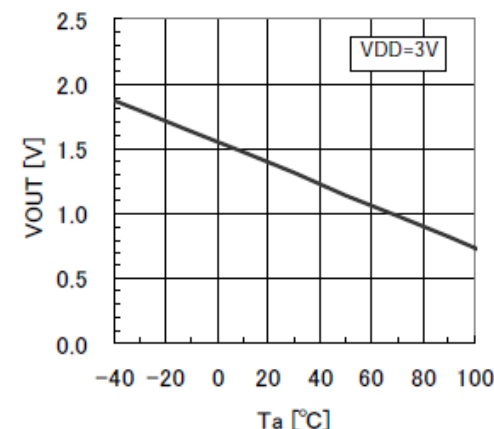


図.1 出力電圧 (VOUT) vs. 温度

温度感度	V _{SE}	-8.4	-8.2	-8.0	mV/°C	
消費電流	I _S	-	4.0	7.0	μA	
出力電圧	V _{OUT}	1.288	1.300	1.312	V	Ta = 30°C
出力電圧電源電圧変動	ΔV _{OUT} V _{DD}	-	-	4	mV	V _{DD} = 2.4~5.5V
出力電圧負荷変動	ΔV _{OUT} R _L	-	-	1	mV	I _{OUT} : 0μA / 0.7μA 差

