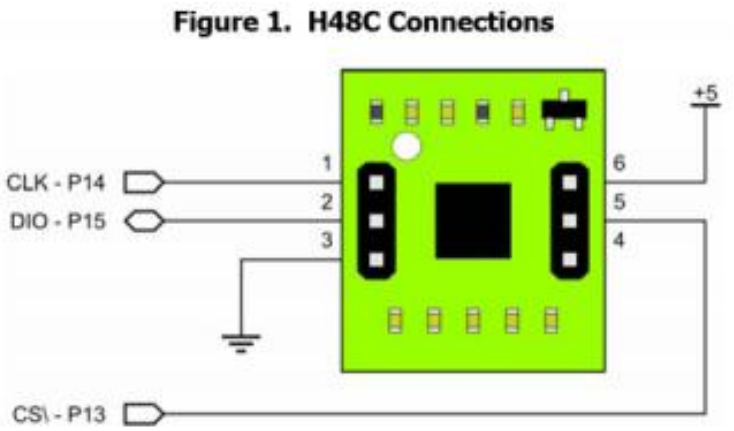
sage.medienkunst.ch

sage hgk fhnw : HowTo_Arduino_Parallax_H48C_Accelerometer

HowTo_Arduino_Parallax_H48C_Accelerometer

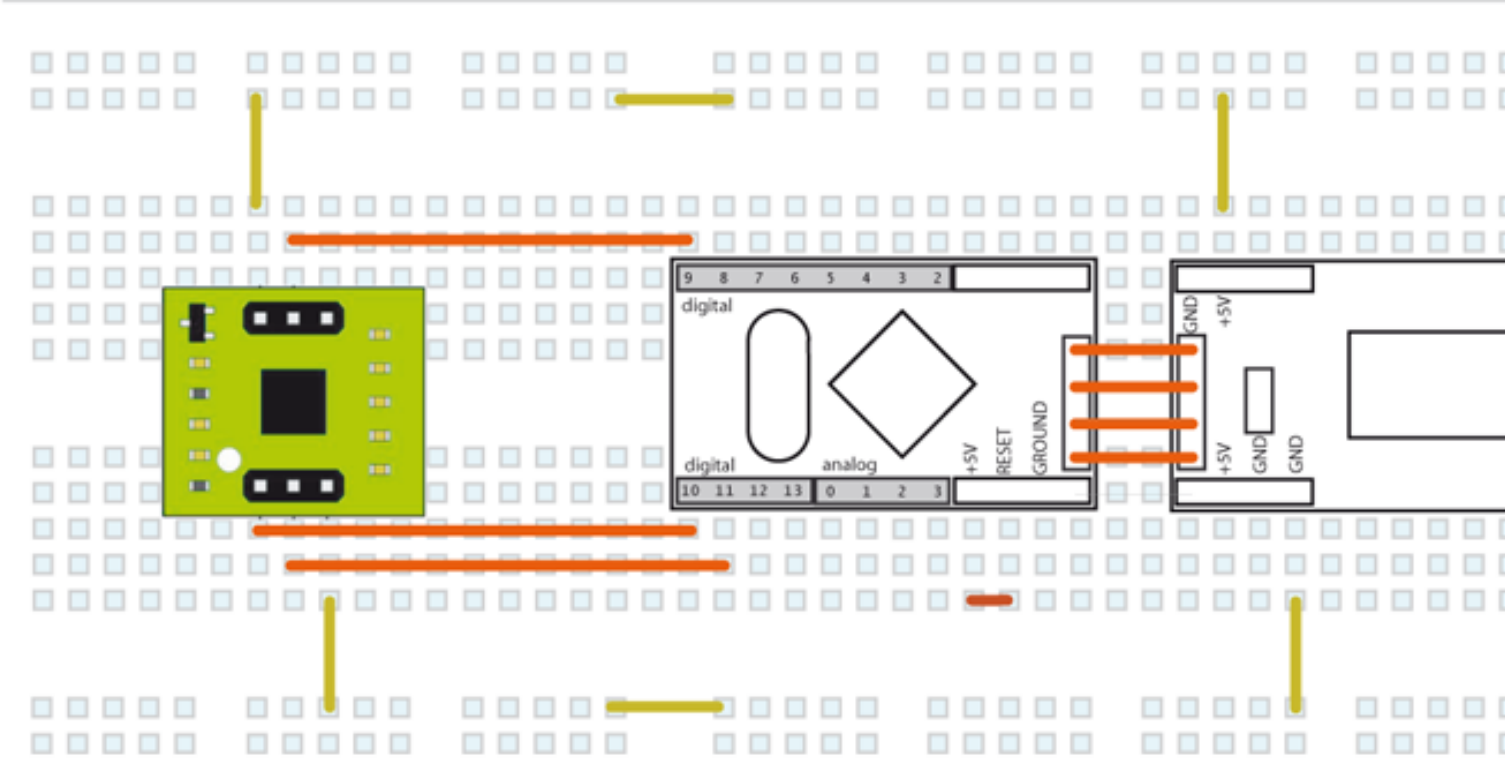
Inhaltsverzeichnis

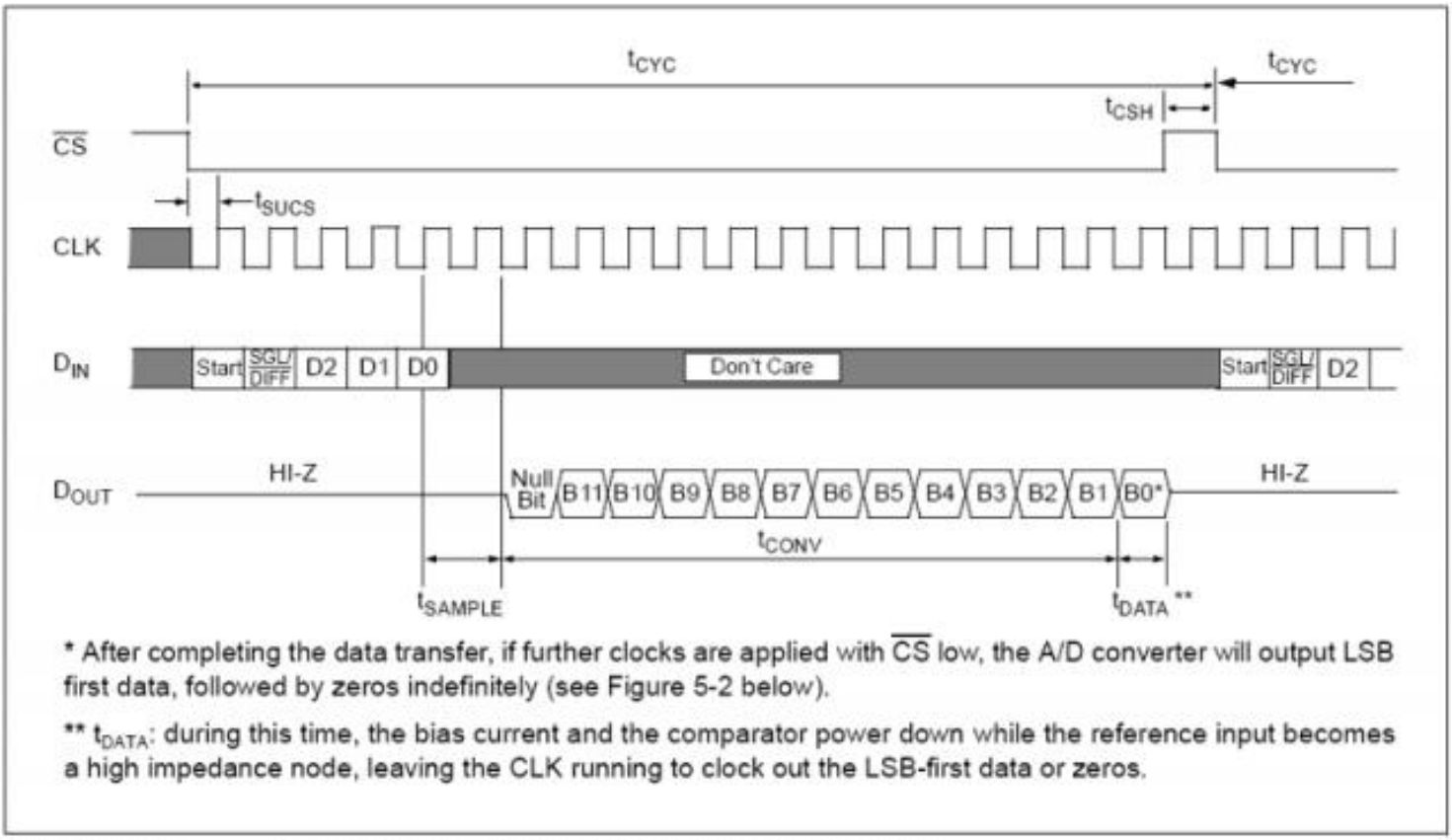


Keywords

Arduino Parallax H48C 3-Achsen Accelerometer

Kurzbeschreibung





* After completing the data transfer, if further clocks are applied with $\overline{CS}$ low, the A/D converter will output LSB first data, followed by zeros indefinitely (see Figure 5-2 below).

** t_{DATA}: during this time, the bias current and the comparator power down while the reference input becomes a high impedance node, leaving the CLK running to clock out the LSB-first data or zeros.

FIGURE 5-1: *Communication with the MCP3204 or MCP3208.*

TABLE 5-1: CONFIGURATION BITS FOR THE MCP3204

Control Bit Selections				Input Configuration	Channel Selection
Single/Diff	D2*	D1	D0		
1	X	0	0	single-ended	CH0
1	X	0	1	single-ended	CH1
1	X	1	0	single-ended	CH2
1	X	1	1	single-ended	CH3
0	X	0	0	differential	CH0 = IN+ CH1 = IN-
0	X	0	1	differential	CH0 = IN- CH1 = IN+
0	X	1	0	differential	CH2 = IN+ CH3 = IN-
0	X	1	1	differential	CH2 = IN- CH3 = IN+

* D2 is a "don't care" for MCP3204

Der Sensor wird per mit einem seriellen Protokoll angesteuert.

1. Ein Start Bit wird gesendet (Funktion **StartBit()**)
2. Eine Art Befehl wird gesendet welche Achse gemessen werden soll (Funktion **ShiftOutNibble(Comma**
3. Der chip braucht einen CLK Impuls Zeit zum Daten messen und der Eingang wird auf Ausgang umgesc
4. Die 12Bit Daten werden drei mal nacheinander in drei 4Bit Blöcken empfangen (Funktion **ShiftInNibb**
5. Die Messung wird beendet indem der Chip deselektiert wird (Funktion **EndBit()**)

Die Funktion **GetValue(byte Command)** fasst die Einzelaufrufe zusamen:

```
aX = GetValue(B1000); // Beschleunigung in X Richung
aY = GetValue(B1001); // Beschleunigung in Y Richung
aZ = GetValue(B1010); // Beschleunigung in Z Richung
```

Stückliste

- arduino mini
- breadboard
- Drahtbrücken
- Parallax H48C Sensor

Programmiercode

```
/*
```

```
////////////////////////////////
Htachi H48C3 Axis Accelerometer
parallax (#28026)

AUTHOR:   kiilo kiilo@kiilo.org
License:  http://creativecommons.org/licenses/by-nc-sa/2.5/ch/

http://parallax.com/Store/Microcontrollers/BASICStampModules/tabid/134/txtSearch/28026/1
http://sage.medienkunst.ch/tiki-index.php?page=HowTo_Arduino_Parallax_H48C_Accelerometer
http://arduino.cc

////////////////////////////////
*/
//// VARS
int CS_pin = 9;
int CLK_pin = 10;
int DIO_pin = 11;

int aX  = 0;
int aY  = 0;
int aZ  = 0;

//// FUNCTIONS
void StartBit() {
  pinMode(DIO_pin, OUTPUT);
  digitalWrite(CS_pin, LOW);
  digitalWrite(CLK_pin, LOW);
  delayMicroseconds(1);
  digitalWrite(DIO_pin, HIGH);
  digitalWrite(CLK_pin, HIGH);
  delayMicroseconds(1);

  }

void ShiftOutNibble(byte DataOutNibble) {
  for(int i = 3; i &gt;= 0; i--) { // i = 3 ... 2 ... 1 ... 0
    digitalWrite(CLK_pin, LOW);
    // set DIO first
    if ((DataOutNibble &amp; (1 &lt;&lt; i)) == (1 &lt;&lt; i)) {  // DataOutNibble AND
      digitalWrite(DIO_pin, HIGH);
      }
    else {
      digitalWrite(DIO_pin, LOW);
      }
    // with CLK rising edge the chip reads the DIO from arduino in
    digitalWrite(CLK_pin, HIGH);
    // data rate is f_clk 2.0 Mhz --&gt; 0,5 micro seeconds
```

```
    delayMicroseconds(1); // :-) just nothing
  }
}

void SampleIt() {
  digitalWrite(CLK_pin, LOW);
  delayMicroseconds(1);
  digitalWrite(CLK_pin, HIGH);
  delayMicroseconds(1);

  pinMode(DIO_pin, INPUT);
  digitalWrite(CLK_pin, LOW);
  delayMicroseconds(1);
  digitalWrite(CLK_pin, HIGH);
  if (digitalRead(DIO_pin)== LOW) {
    // Blink LED because ok
    }
}

byte ShiftInNibble() {
  byte resultNibble;
  resultNibble = 0;

    for(int i = 3 ; i &gt;= 0; i--) { // from bit 3 to 0
      // The chip Shift out results on falling CLK
      digitalWrite(CLK_pin, LOW);
      delayMicroseconds(1); // :-) just nothing
      if( digitalRead(DIO_pin) == HIGH) { // BIT set or not?
        resultNibble += 1 &lt;&lt; i; // Store 1 x 2^i in our ResultNibble
      }
      else {
        resultNibble += 0 &lt;&lt; i; // YES this is alway 0, just for symetry ;-)
      }
      digitalWrite(CLK_pin, HIGH);
      //delayMicroseconds(1); // :-) just nothing
    }
return resultNibble;
}

void EndBit() {
  digitalWrite(CS_pin, HIGH);
  digitalWrite(CLK_pin, HIGH);
}

int GetValue(byte Command) { // x = B1000, y = B1001, z = B1010
  int Result = 0;
  StartBit();
```

```
  ShiftOutNibble(Command);
  SampleIt();
  Result =  2048 - ((ShiftInNibble() &lt;&lt; 8) + (ShiftInNibble() &lt;&lt; 4) + Shift
  EndBit();

  return Result;
  }

//// SETUP
void setup() {
  Serial.begin(115200);
  pinMode(CS_pin, OUTPUT);
  pinMode(CLK_pin, OUTPUT);
  pinMode(DIO_pin, OUTPUT);
  // initialize device &amp; reset
  digitalWrite(CS_pin,LOW);
  digitalWrite(CLK_pin,LOW);
  delayMicroseconds(1);
  digitalWrite(CS_pin, HIGH);
  digitalWrite(CLK_pin,HIGH);
}


//// LOOP
void loop() {

  aX = GetValue(B1000);
  aY = GetValue(B1010);
  aZ = GetValue(B1011);

  Serial.print(aX);
  Serial.print(&quot; &quot;);
  Serial.print(aY);
  Serial.print(&quot; &quot;);
  Serial.print(aZ);
  Serial.println(&quot;&quot;);
  delay(100);  // loop every 10 times per sec.
}
```

Dateien

Links

- Parallax Product info H48C
- PDF Datasheet insbesondere Chapt.5. serial comunication protocoll S.15/16

Autor

- tobias DOT hoffmann AT fhnw DOT ch

Zu dieser Seite haben beigesteuert: kiilo .
Page last modified on 30.04.2008 11:57 by kiilo.

Historie ähnlich