

Development of 3-Axis Acceleration Data Logger for Non-Wired Sensing and Recording of Wellbore Paths

Tongjun Ruan

Reservoir Evaluation and Advanced
Computational Technologies Group
PRRC/New Mexico Tech
Socorro, NM 87801 USA
e-mail: ruan@prrc.nmt.edu

Robert Balch

Reservoir Evaluation and Advanced
Computational Technologies Group
PRRC/New Mexico Tech
Socorro, NM 87801 USA
e-mail: balch@prrc.nmt.edu

Abstract—A 3-axis acceleration data logger was designed for non-wired sensing and recording of small diameter laterals where wire or wireless communication with ground/surface is impractical or not feasible. This is accomplished by measuring 3-axis acceleration, storing the data on the data logger, offloading data to an onsite laptop computer, and then back-calculating the path with developed software.

Keywords-Data logger; MEMS 3-axis accelerometer; Radial Drilling; SPI

I. INTRODUCTION

The path of small diameter laterals has heretofore been a mystery due to very limited space and harsh environments. Radial drilling is a versatile technology with applications ranging through well enhancement, increased injection rates in disposal wells, deep acid penetration, and directed fracturing, to name just a few.

Drilling is accomplished through a high pressure water jet and can extend up to 300 ft horizontally from a vertical well-bore. Since the water jet is deployed on the end of a 300 ft rubber hose attached to a coiled tubing unit there is no method to directly measure the path of the radially jetted lateral since no wire or other direct connection is possible. The small size of the hose and drilling tip (0.5 in. or 12.7mm diameter) and the shoe that turns the hose horizontally in the well-bore, having a short turning radius, limits the size of any measuring device to about 1.5 in. or 39mm length by 0.5 in. or 12.7 mm diameter.

In addition the measurements must be stored directly within the device, and survive a hostile environment with high temperatures, solvents, and high g impacts.

It is very difficult to directly measure the path (direction and distance) underground. Usually accelerations in three directions are measured; then velocity is calculated by integrating acceleration, distance is calculated by integrating velocity based on Newton's laws. The data logger is designed to measure the accelerations in X, Y and Z directions and store the data into EEPROM during its travels forth and back to and back through the radial well. The recorded data will be offloaded into an onsite laptop computer/PC, processed, and used to calculate the path.

II. HARDWARE DESIGN

Due to very limited space and harsh environments, the data logger must be small enough into a protective tube less than 39mm long and diameter less than 12.7 mm.

A. Limited Space

Considering the wall thickness of the protecting tube and the two end caps, the device PCB must be less than 25mm long and 11mm in diameter. Fig. 1 shows the protecting tube/shell of the data logger.

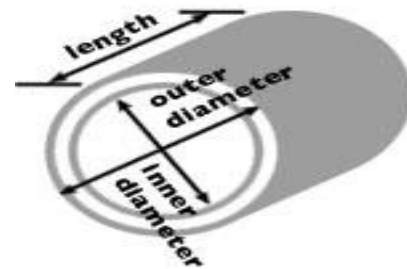


Figure 1. T-316/316L Stainless steel tube

T-316/316L Stainless steel tube size: 0.5 in. OD × 0.035 in. Wall × 0.43 in. ID. (10.922mm)

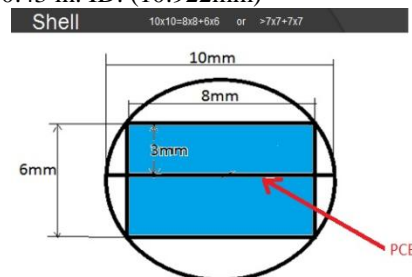


Figure 2. Usable space inside tube

Fig. 2 shows the inside usable space. The PCB spans the diameter with the remaining space (blue area for two sides mount) available for chips and other items. The PCB size is limited to 10.5mm × 25mm.

B. Chips and Battery

Based on the requirements, we need :

1. A 3-axis accelerometer to measure accelerations in X, Y, and Z directions;

2. A memory chip to store the data ;
3. A MCU to configure the accelerometer and flash memory, read data from accelerometer and write data into memory;
4. A battery to provide operating power;
5. Two capacitors needed to make the circuit work properly.

There are quite many three-axis accelerometer products available from different companies. One of them is ST’s LIS331hh which is an accurate accelerometer that utilizes the SPI bus for communication. This was chosen as the accelerometer for the data logger mainly due to following attributes:

- MEMS digital output
- Ultra low-power (Current consumption in normal mode is 250uA)
- High full-scale 3-Axis “nano” accelerometer(+/-6 g, +/-12 g, or +/-24 g)
- Small size

The SST25VF64C 64Mb SPI serial flash memory was chosen to store data in the data logger due to:

- SPI interface,
- Single read/write low operating voltage,
- 64Mb capacity (the largest on the market, it can store three hours data; discussed further in section III),
- Small size.

Microchip’s PIC12LF1840 was chosen as the MCU in the data logger, because of the following:

- Extra low power consumption,
- 4KW program flash memory
- Integrated MSSP with SPI and I2C.
- 8 pins
- Small size

All three chips are small enough to meet the space limitation requirements.

A lithium coin battery LR1025 is only choice for the data logger with 3V voltage and the smallest size among lithium coin batteries(diameter=10mm and length=2.5mm) and 30 mAH capacity.

Fig. 3 show the main features of components. The total device length: $3.9 + 3 + 5 + 2.5 = 14.4$ mm and width is ≤ 10 mm. So the space inside the shell is enough for the data logger with dimension W 10mm x L 25mm x H 3mm(PCB and chips).

Name	PIC12F1840	LIS331	SST25VF64C	Battery	Cap
Size	3.9x3.9mm	3x3mm	5x5mm	D 10 x L 2.5 mm	3x2 mm
Voltage	1.8-.6v	2.16-3.6v	2.7x3.6v	Li. 3v	
Current	75uA	250uA	12mA	30mAH	
Standby	500nA	10uA	5uA		

Figure 3. the main parameters of components

The data logger has been designed and implemented as simple and small as possible by choosing small chips and battery and other components. The final design version contains only three IC chips: MEMS 3-axis accelerometer LIS331, 64Mb serial flash memory SST25VF64C and 8-PIN microcontroller PIC12LF1840 plus a coin battery CR2510. All of three chips have SPI (Serial Peripheral Interface) available and are used to communicate among them. The 8-PIN of the microprocessor 2 pins for power supply and 3 pins for SPI , 1 pin for acceleration data ready interrupt and 2 pins for chip selection/enable as shown in Fig. 4.

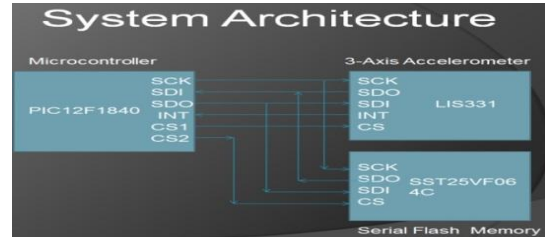


Figure 4. System Architecture of Designed data logger.

Fig. 5 shows the prototype built with LIS331HH breakout and PIC12F1840 and the SST25VF64C chips inside a piece of pen shell with inner diameter 10mm and length 25 mm.



Figure 5. The prototype, diameter =10mm , inside a piece of pen shell.

III. SOFTWARE DESIGN FOR MCU

The circuit design meeting the requirements is built on a prototype PCB board and is illustrated in last section. In order to make it work for our specific application. We need to write and program software into MCU → PIC12F1840 chip. The data logger software was developed by using Microchip’s MPLAB X IDE with XC8 compiler and programmed into PIC12F1840 by using Microchip’s Real ICE which is a develop device with programming PIC12F1840 and debug function.

A Software Functions

The Software functions of the PIC12LF1840 are:

1. Configure PIC12F1840’s SPI bus at pin 5,6 and 7, interrupt function at pin 4 , SPI slave selection pin: pin 1 for LIS331HH and pin2 for SST25VF64C and clock rate at 500kHz
2. Configure LIS331HH scale rang as {-6g, 6g}, sampling rate as 100/S and data ready signal output at “int1” pin;
3. Enable SST25VF64C page program.

4. Waiting for the data ready signal from LIS331HH
5. If the data is ready, PIC12LF1840 will response and start reading 6 bytes acceleration data from LIS331HH
6. Write them into SST25VF64C and return to 4.

Fig. 6 shows entire flow chart.

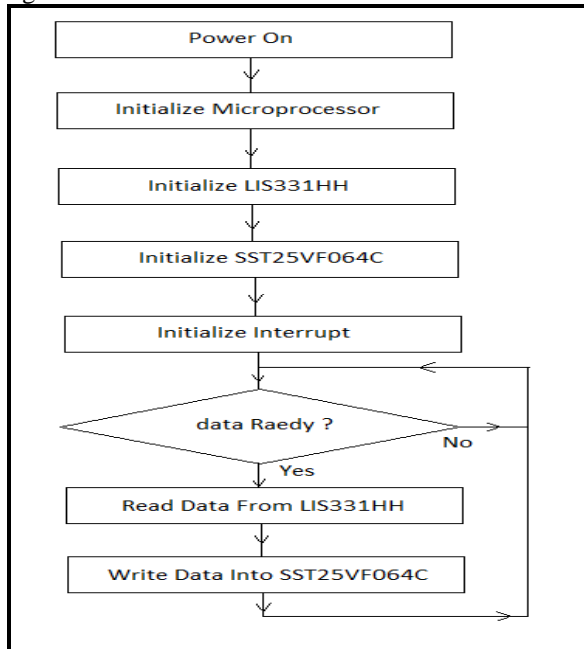


Figure 6. Flow chart.

The power consumption is an important issue since we use cell battery LR2510 to power the Data logger. PIC12LF1840 8-bit microcontrollers is with active currents of less than 50 $\mu\text{A}/\text{MHz}$. To save power, the clock rate of PIC12F1840 is set to lower 0.5MHz and SPI clock rate is set as 125KHz. So the SPI bandwidth is $125\text{K}/8 \rightarrow$ about 10.5K.

The sampling rate of LIS331HH is set as 100 times per second, each time there are 6 bytes data generated, that means there are less than 10ms for interrupt service routine to read, write data and return. Totally there are 600 bytes data needed to be read from LIS331HH and written into flash memory per second which means total 1200 bytes on SPI for reading and writing during one second. 1200 is still less than 10.5k even we consider the doubled overhead plus other delay caused by other operations, 10.5kB bandwidth is good enough. So 10ms is more than enough for MCU to read and write data.

Lab tests show that the interrupt service routine totally takes about 4.6ms to complete its task(See Fig. 8 at last page).

Recorded data will be read out from chip using serial flash memory programmer in binary format and write as a binary file on PC. Further data processing will be discussed in next section.

How many hours the SPI flash memory can store? Based above discussion, there are 600bytes per second $(64\text{Mb}/8)/600 \rightarrow 13333$ seconds $\rightarrow 3.7$ hour data can be stored in SST25VF64C.

B Configuration And SPI Switching

PIC12LF1840 has a MSSP model with configurable SPI/I2C bus. The MSSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I2C™)

First we need to configure the MSSP work as SPI. Then we need to configure the SPI modes for LIS331HH and for SST25VF64C. The LIS331HH works with SPI mode 3 and SST25 works with SPI mode 0, we first configure the SPI work in mode 3 and witching to mode 0 when we need to write data into SST25VF64C.

More details can be found in following C functions.

```

void pic1840_system_init(){
  ANSELA=0x00; //all pins set as Digital I/O pins
  SSP1IE=0; //forbid interrupt
  PIR1bits.SSP1IF=0; //PIR1 bit, SSP1IE=0;
  INTCON=0; //all 0 then setup one by one
  INTCONbits.GIE=0; // Global interrupt disabled first,
  will set after system init
  SSP1CON1bits.SSPEN=1; //Enable SPI
  SSP1CON1bits.SSPM=0b0000; //clock =Fosc/4
  SSP1STAT=0x00;
  //SST25 need SPI mode 0 and LIS331 needs SPI mode
  3,first set SPI mode 3 for LIS331HH
  //switching to SPI mode 0 and back in SST25 codes
  SSP1CON1bits.CKP=1;
  SSP1STATbits.CKE=0;
  SSP1STATbits.SMP=0; //sampling at middle
  //bit5--bit0 -->TRISA5---TRISA0
  TRISA=0b00001100; // SDO/RA0 and
  SPC/SCK/RA1,RA4 and RA5 are outputs all others as
  inputs
  WPUA = 1; //All RAX are weak pull up }
  void interrupt_init(void){
  //set RA3 as Lis331's interrupt input pin
  IOCAP3 = 1; //allow positive edge interrupt
  INTCONbits.IOCIE = 1; //enable the Interrupt On
  Change
  INTCONbits.PEIE = 1; //enable peripheral device
  interrupt
  INTCONbits.GIE = 1; // global interrupt enabled
  IOCAF3 = 0; //clear corresponding interrupt flag;
  RA3=0;}
  
```

C Band-Pass Filter

LIS331HH has internal low-pass filter and high-pass filters. The low-pass filter is fixed with the sampling rate configured on fly and Offset can be eliminated by enabling the built-in high pass filter. To avoid Gravity affection and the offset we configure the LIS331HH sampling rate as 100Hz with default low-pass filter cut-off frequency 74Hz (non-configurable) and configure high-pass filter cut-off frequency as 2Hz. So the bandwidth is 2Hz—74Hz. The final bandwidth will be decided based on site tests' data analysis.

IV. DATA PROCESSING

The data processing software was developed using Java which transfers the binary data file into a text file on PC and the data are filtered and integrated twice to get the position in X, Y and Z directions. The path of data logger traveled can be visualized on the screen of PC except the readable data file.

Recorded raw data in SPI serial memory are in Gravity unit, will be read out by using SF600 SPI memory programmer with clip and stored into a binary file on PC compatible computer. We first transform the data from Gravity unit to Meter/Second*Second unit and write them into a file in decimal format, then filter out the noise and computing velocity and distance in X, Y and Z direction. Fig. 7 show the Data process procedure.

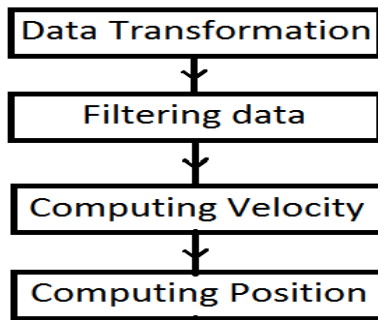


Figure 7. Data processing flow chart

The configured scale of LIS331HH in the data logger is {-6G, 6G}. Though LIS331HH provides 16-bit data output, i.e. the raw data is signed 16-bit integer in Gravity unit, the lowest 4 bits of the data are 0x0000. Raw data (signed integer in G unit) can be transferred to signed decimal float in M/S² unit with following formula:

$$\text{Meter_Second}^2 = \text{Gravity} * \frac{12}{(2^{16})} * \text{Raw}$$

Where “Raw” is raw acceleration data in 16 bit signed binary integer.

“Gravity” is 9.81M/S².

After the transformation, Kalman filter is applied to the transferred data, then we integrate filtered acceleration data once to get velocity and again integrate velocity to get the position/distance. Following is the codes:

```

for(i=0;i<Filtered_Data_Length-1;i++)
  { //First integration---->computing
    //Velocity: 0 1 2 indicate x y z axis
    //delta t=1/100 = 0.01 second
    V[i+1][0]=V[i][0]+(A[i][0]+A[i+1][0]) *0.5*Delta_T;
    V[i+1][1]=V[i][1]+(A[i][1]+A[i+1][1]) *0.5*Delta_T;
    V[i+1][2]=V[i][2]+(A[i][2]+A[i+1][2]) *0.5*Delta_T;
  }
  
```

```

//Second integration---->computing //position: 0 1 2
//indicate x y z axis
  
```

```

P[i+1][0]=P[i][0]+(V[i][0]+V[i+1][0]) *0.5*Delta_T;
P[i+1][1]=P[i][1]+(V[i][1]+V[i+1][1]) *0.5*Delta_T;
P[i+1][2]=P[i][2]+(V[i][2]+V[i+1][2])*0.5*Delta_T; }
  
```

V. TEST RESULTS

After the prototype was built and software was developed, we programmed the codes into PIC12F1840 and ran a Lab test (Site test has been scheduled in August, more test results will be added after site test).

From the logic analyzer we can see the data logger run properly (see Fig. 8).

From top to bottom: LIS331HH enable, LIS331HH interrupt, PIC12F1840 SPI data output, PIC12F1840 SPI clock, PIC12F1840 SPI data input and SST25VF64C SPI serial flash memory chip selection. Sampling rate is 100 times/S, i.e. every 0.01s the LIS331 generate a data ready signal and PIC12F1840 response the interrupt and enter the interrupt service subroutine: read data and write them into memory. The interrupt signal will reset after reading the 6 bytes acceleration data and set again once next data ready (in 0.01s).

Fig. 9 shows the raw data (blue) and Filtered data (red) in X-axis.

Calculation and test show that the cell battery (LR1025 30mAH) can power the data logger to work more than 3 hours and the flash memory (SST25VF64C) can store 3.7-hours data. In oil field the entire procedure of the drilling path test will take no more than 40 minutes from powering on data logger and installing the data logger to finishing the test and power off the data logger.

The evaluated position error based on the recorded data is less than 10% in a lab 30 feet test. To improve the precision, Better filters are being designed and more lab tests and site tests will be done soon.

REFERENCES

- [1] Micochip Technology Inc, “PIC12F1840 data sheetons,” 2011,pp.1-5
- [2] STMicroelectronics Group of companies,LIS331HH data sheet, 2009, pp.5–10.
- [3] Silicon Storage Technologies, Inc, “64Mbit SPI Serial Dual I/O Flash,” 2010, 2-10.
- [4] Micochip Technology Inc, “MPLAB X IDE”, 2012.
- [5] Micochip Technology Inc, “XC8 programming”, 2012.
- [6] ST Microelectronis, MEMS Inertial Sensor: 3-Axis Linear Accelerometer Specification Sheet,2005.
- [7] Andrew Blak etc, Deriving Displacement from a 3 axis Accelerometer,2012

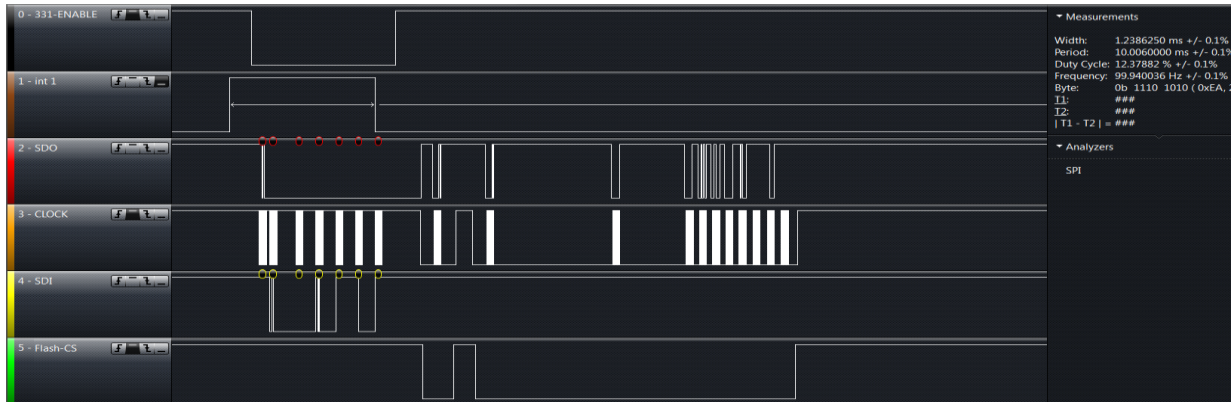


Figure 8. The data logger running wave

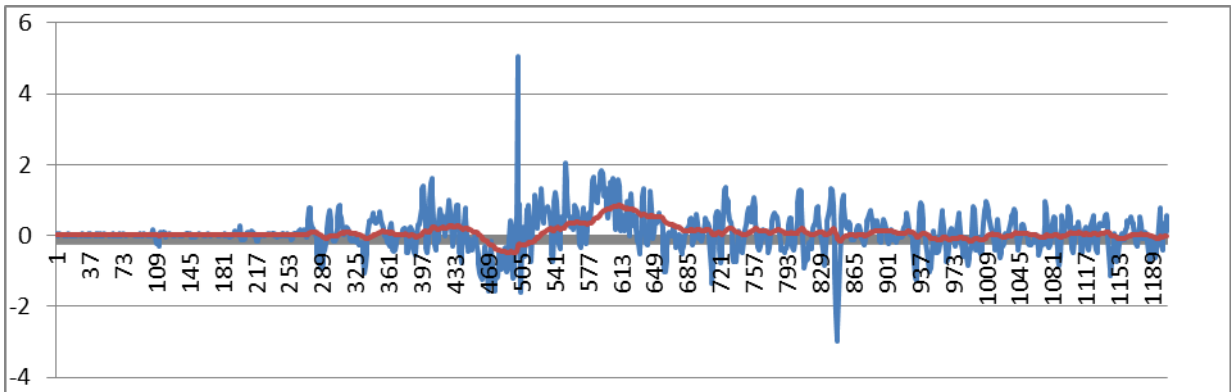


Figure 9. Raw data(blue) and Filtered data(red)