RawData unpacker and packer

Satoru Yamada

1, Unpacker

1-1, Example : how to get information of RawCOPPER header

You can get event # info from RawCOPPER object like this;

```
StoreArray<RawCOPPER> raw_cprarray; // When StoreArray is used
 for (int i = 0; i < raw_cprarray.getEntries(); i++) { //When StoreArray is used
   for ( int j = 0; j < raw_cprarray[ i ]->GetNumEntries(); j++) {
     //      Get Event number
     unsigned int event_no = raw_cprarray[ i ]->GetEveNo( j );

     //      Get RawCOPPER data block
     int* buf = raw_cprarray[ i ]->GetBuffer( j );
     //      See contents of a data block (from RawCOPPER header to RawCOPPER trailer)
     for( int k = 0; k < raw_cprarray[ i ]->GetBlockNwords( j ); k++ ) printf("%.8x¥n", buf[ k ] );

     //      Get Detector Buffer (raw data from detector electronics)
     for( int finesse_num =0; finesse_num < 4; finesse_num++) {
        int* buf_slot = raw_cprarray[ i ]->GetDetectorBuffer( j, finesse_num );
        //          See contents of raw data from detector
        for( int k = 0; k < raw_cprarray[ i ]->GetDetectorNwords( j, finesse_num ); k++ ){
          printf("%.8x¥n", buf_slot[ k ] );          }
     }
   }
 }
```

1-2, Test program to read RawCOPPER(RawCDC) data

1, Get dummy data file (data from two CDC FEE boards connected to FINESSE A and C.)
login.cc.kek.jp : ~yamadas/rawdata/root_output_RawCDC_rev7133.root

2, See contents of the data
% cd ${BELLE2_LOCAL_DIR}/daq/; svn update
% cd ${BELLE2_LOCAL_DIR}/daq/rawdata/examples/
% basf2 **ReadStoreTemplate.py** -i ./root_output_RawCDC_rev7133.root | less

```
[INFO] Steering file: ReadStoreTemplate.py
>>> basf2 Python environment set
>>> Framework object created: fw


===== DataBlock(RawCDC) : Block # 0 : Event # 0 : node ID 0x00000000 : block size 224 bytes
== Detector Buffer(FINESSE A)
0x0094c13a 0x91000001
== Detector Buffer(FINESSE C)
0x0094c13a 0x91000001


===== DataBlock(RawCDC) : Block # 1 : Event # 1 : node ID 0x00000000 : block size 224 bytes
== Detector Buffer(FINESSE A)
0x0094c23f 0xf1000001
== Detector Buffer(FINESSE C)
0x0094c23f 0xf1000001


===== DataBlock(RawCDC) : Block # 2 : Event # 2 : node ID 0x00000000 : block size 224 bytes
== Detector Buffer(FINESSE A)
0x0094c30d 0x69000001
== Detector Buffer(FINESSE C)
0x0094c30d 0x69000001

….
```

In this data,
Detector buffer contains only 2words(=8bytes) per/FINESSE/event.

Note that block # is a number used by DAQ software for handling data and not related with Event #.

1-3, How to assign a buffer to RawDataBlock, RawCOPPER, RawSVD…

```
int* buffer = new int[nwords];  // data

RawCOPPER raw_copper;

Int delete_flag = 1;  // if 1, raw_copper's destructor will call "delete buffer;"

Int num_event = 1, num_nodes = 1;  // If the buffer contains only 1 data block (usually so).

raw_copper.SetBuffer(buffer, nwords, delete_flag, num_event, num_nodes);


// When you want to convert a Raw*** type

{
          RawSVD raw_svd;

          delete_flag = 0;  // in this case, raw_copper will delete buffer. So delete_flag=1 may cause double-free.

          raw_svd.SetBuffer( raw_copper.GetWholeBuffer(), nwords, delete_flag, num_event, num_nodes);

          RawDataBlock raw_datablock;

          delete_flag = 0; // in this case, raw_copper will delete buffer. So delete_flag=1 may cause double-free.

          raw_datablock.SetBuffer( raw_copper.GetWholeBuffer(), nwords, delete_flag, num_event, num_nodes);

}
```

2, Packer

2-1, Function to store data in RawCOPPER object

void RawCOPPER::PackDetectorBuf(int* detector_buf_1st, int nwords_1st, int* detector_buf_2nd, int nwords_2nd, int* detector_buf_3rd, int nwords_3rd, int* detector_buf_4th, int nwords_4th, RawCOPPERPackerInfo rawcprpacker_info){}

Input variables :

int* detector_buf_*** : pointer to the detector buffer that you want to store as ***th FINESSE data.

int nwords_*** : length of the detector_buf_*** (unit -> word = 4bytes)

RawCOPPERPackerInfo rawcprpacker_info : Information to
 fill RawHeader

```
struct RawCOPPERPackerInfo {
    unsigned int exp_num; // 10bit
    unsigned int run_subrun_num; // 22bit
    unsigned int eve_num; // 32bit
    unsigned int node_id; // 32bit
    unsigned int tt_ctime; // 27bit clock ticks at trigger timing distributed by FTSW.
For details, see Nakao-san's belle2link user guide
    unsigned int tt_utime; // 32bit unitx time at trigger timing distributed by FTSW.
For details, see Nakao-san's belle2link user guide
    unsigned int b2l_ctime; // 27bit clock ticks at trigger timing measured by HSLB
on COPPER. For details, see Nakao-san's belle2link user guide
    unsigned int hslb_crc16_error_bit; // 4bit errorflag for CRC errors in data
transfer via b2link. ( bit0,1,2,3 -> finesse slot a,b,c,d)
    unsigned int truncation_mask; // Not defined yet
    unsigned int type_of_data; // Not defined yet
};
```

(#include <rawdata/include/RawCOPPERPackerInfo.h>)

2-2, test program to store data in RawCOPPER object

➢ Module to fill dummy data in RawCOPPER

- rawdata/modules/src/DummyDataPacker.cc

➢ Script to run the above module

- $ rawdata/scripts/DummyDataPacker.py

end

Revision history

- July 15, 2014 (rev.11616) : ver.1
- Aug. 8, 2014 (rev. 12158)
  - Add instruction about setting a buffer to Raw*** object.