1. J’ai essayé de mettre un timer pour mesurer xTZ8PointDiamondSearch ou xTZ8PointSquareSearch



Et le problème que le fichier TEncSearch.cpp n’est pas appelé dans le encmain()…

2.Voici le encmain() …. je n’ai ajouté là que la déclaration de tempsdiam (en jaune)

/\* The copyright in this software is being made available under the BSD

 \* License, included below. This software may be subject to other third party

 \* and contributor rights, including patent rights, and no such rights are

 \* granted under this license.

 \*

 \* Copyright (c) 2010-2014, ITU/ISO/IEC

 \* All rights reserved.

 \*

 \* Redistribution and use in source and binary forms, with or without

 \* modification, are permitted provided that the following conditions are met:

 \*

 \* \* Redistributions of source code must retain the above copyright notice,

 \* this list of conditions and the following disclaimer.

 \* \* Redistributions in binary form must reproduce the above copyright notice,

 \* this list of conditions and the following disclaimer in the documentation

 \* and/or other materials provided with the distribution.

 \* \* Neither the name of the ITU/ISO/IEC nor the names of its contributors may

 \* be used to endorse or promote products derived from this software without

 \* specific prior written permission.

 \*

 \* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"

 \* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

 \* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

 \* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS

 \* BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR

 \* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF

 \* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

 \* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN

 \* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

 \* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF

 \* THE POSSIBILITY OF SUCH DAMAGE.

 \*/

/\*\* \file encmain.cpp

 \brief Encoder application main

\*/

#include <time.h>

#include <iostream>

#include "TAppEncTop.h"

#include "TAppCommon/program\_options\_lite.h"

//! \ingroup TAppEncoder

//! \{

#include "../Lib/TLibCommon/Debug.h"

// ====================================================================================================================

// Main function

// ====================================================================================================================

Double tempsdiam;

int main(int argc, char\* argv[])

{

 TAppEncTop cTAppEncTop;

 // print information

 fprintf( stdout, "\n" );

 fprintf( stdout, "HM software: Encoder Version [%s] (including RExt)", NV\_VERSION );

 fprintf( stdout, NVM\_ONOS );

 fprintf( stdout, NVM\_COMPILEDBY );

 fprintf( stdout, NVM\_BITS );

 fprintf( stdout, "\n\n" );

 // create application encoder class

 cTAppEncTop.create();

 // parse configuration

 try

 {

 if(!cTAppEncTop.parseCfg( argc, argv ))

 {

 cTAppEncTop.destroy();

#if ENVIRONMENT\_VARIABLE\_DEBUG\_AND\_TEST

 EnvVar::printEnvVar();

#endif

 return 1;

 }

 }

 catch (df::program\_options\_lite::ParseFailure &e)

 {

 std::cerr << "Error parsing option \""<< e.arg <<"\" with argument \""<< e.val <<"\"." << std::endl;

 return 1;

 }

#if PRINT\_MACRO\_VALUES

 printMacroSettings();

#endif

#if ENVIRONMENT\_VARIABLE\_DEBUG\_AND\_TEST

 EnvVar::printEnvVarInUse();

#endif

 // starting time

 Double dResult;

 clock\_t lBefore = clock();

 // call encoding function

 cTAppEncTop.encode();

 // ending time

 dResult = (Double)(clock()-lBefore) / CLOCKS\_PER\_SEC;

 printf("\n Total Time: %12.3f sec.\n", dResult);

 cTAppEncTop.destroy();

 return 0;

}

//! \}