Application Note

Akademie věd České republiky
Ústav teorie informace a automatizace AV ČR, v.v.i.

Remote Terminal for Dense Optical Flow Demonstrator Using Python 1300 Camera Module

Lukáš Kohout
Kohoutl@utia.cas.cz

Revision history

Rev.	Date	Author	Description
0	17.10.2017	L. Kohout	Document creation
1			
2			

Contents

Acknowledgement

This work has been supported from project PANACHE, project number ENIAC JU 621217 and MSMT 7H14006.

1 Introduction

A remote terminal for dense optical flow demonstrator using Python 1300 camera module [4] is a simple application of STM32H743I-EVAL board [1]. It uses its touch screen providing a GUI (Graphics User Interface) to control remote device; optical flow demonstrator in this case. The application is based on the STM32CubeH7 package [2] compiled for STM32H743I-EVAL board with the IAR EWARM compiler [3]. The remote device is controlled via RS232 interface.

2 Description

The application uses the STM32H743I-EVAL LCD with the touch layer. It shows how to create GUI on STM32H743I-EVAL board enabling control dense optical flow demonstrator using Python 1300 camera module (remote device based on different platform) in a short time. An example of running application is shown in Figure 1. The GUI provides two vertical sliders to control analog and digital gain of the camera, and one slider to set the magnitude limit parameter of the algorithm computing in the remote device. The GUI provides also the switch to select the output of the remote device. There are two choices. The first, it selects a direct view from the camera (input frame of the filter). The second, it sets the resultant image of the filter as the output. The last control of the GUI is a button called *Preview.* It invokes a reading of the current image from the remote device. The received image is displayed on the screen, there can be seen a direct camera view or a filtered image; it depends on the selected output (see Figure 1). The image has smaller resolution than the original camera resolution is. The camera resolution is 1280x1024 whereas the transferring and displaying image has resolution 256x204 (approximately five-times smaller). All events from touchable controls are converted to commands using RS232 interface. The application is based on the Hello World example of the STemWin middleware of the STM32CubeH7 package.

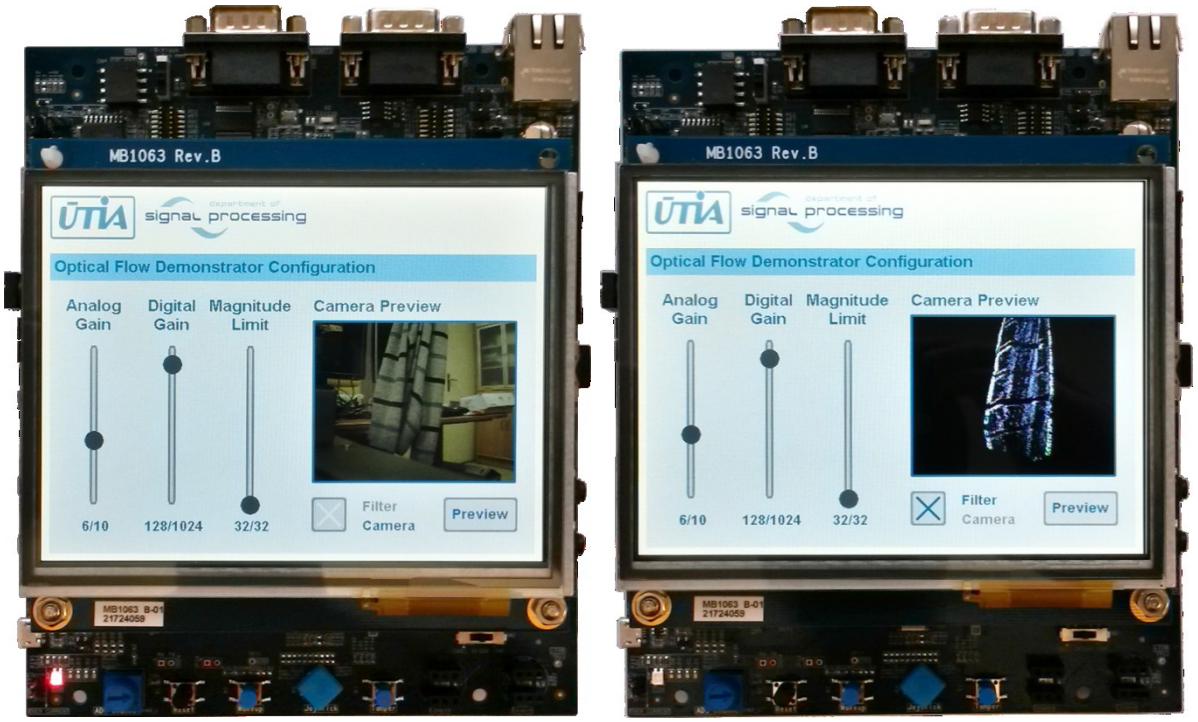

Figure 1: Remote terminal for Dense Optical Flow demonstrator.

2.1 Commands and Messages

To control the remote device, the terminal uses RS232 interface. The unit transforms the screen touches to the commands for the remote device and it reads messages form the remote device. Command always starts with **'!'** character and ends with **'*'** character (**!command*** for instance). All used commands are summarized in Table 1. Messages are remote device responds on the commands. Message starts with **'#'** character and ends with **'*'** character (for example **#message***). Table 2 presents implemented messages. Both, commands and messages, can be optionally delimited by **\CR\LF** sequence.

Table 1: Commands

Command	Description
!SAGxx*	Set Analog Gain, *xx* is decadic value of the analog gain in range 0 to 10
!SDGxxxx*	Set Digital Gain, *xxxx* is decadic value of the digital gain in range 0 to 1024
!SMGxx*	Set Magnitude Limit, xx is decadic value of the magnitude limit in range 1 to 32
!SOC*	Set Output to Camera
!SOF*	Set Output to Filter
!GIMG*	Get current Image of the camera

Table 2: Messages

Message	Description
#IMGdata…*	Response on !GIMG* command, where "data…" means bytes of the image. Concretely it is (256 * 204 * 3) Bytes (RGB – 3 Bytes per pixel)

3 Used Tools and Resources

- STM32H743I-EVAL board [1].
- STM32CubeH7 package is a firmware package for STM32H7 Series. It is downloadable from the STMicroelectronics web page [2]. The package gathers together all the generic embedded software components required to develop an application on STM32H7 microcontrollers. The version of the package used in this application is 1.1.0.
- Commercial edition of the IAR Embedded Workbench, alternatively a free edition with restrictions to the 30-day time-limited evaluation without code size limitation [3]. The version of the tool chain used to develop this application was 8.11.3.13984.
- STM32H743I-EVAL_Cam_GUI_OPTF package provided with this application note, the content of the package is described in Section 5.

4 Implementation

To implement the remote terminal demo, follow the steps bellow.

1. Download STM32CubeH7 package from its web page [2]. Version of the used package is 1.1.0. Decompress the package zip file. The package top folder is called *STM32Cube_FW_H7_V1.1.0*. In the text, the package path will be referenced as *$STM32Cube_path$*.
2. Go to the *$STM32Cube_path$\Projects\STM32H743I_EVAL\Applications\STemWin* folder and make a copy of the *STemWin_HelloWorld* project folder. The name of the copy will be *cam_gui_optf*.

3. Copy the content of the STM32H743I-EVAL_Cam_GUI_OPTF\sw\src folder to the *$STM32Cube_path$\Projects\STM32H743I_EVAL\Applications\STemWin\cam_gui_optf* folder. Rewrite all already existing files.
4. Start the IAR Embedded Workbench. In the text, this tool will be referenced as EWARM.
5. In the EWARM tool chain, open the *cam_gui_optf* workspace, go to the menu *File→Open→Workspace…* Select the path to the *cam_gui_optf* EWARM workspace description file *$STM32Cube_path$\Projects\STM32H743I_EVAL\Applications\STemWin\cam_gui_opt\EWARM\Project.eww*.
6. Build the project, menu *Project→Make*.
7. Connect the ST-LINK/V2-1 programming and debugging tool on the STM32H743I-EVAL board. Plug the USB cable to the CN23 ST-LINK/V2-1 USB connector of the board.
8. Let the STM32H743I-EVAL board configuration jumpers in their default settings, check them according to board manual [1].
9. Power the board up.
10. Download and debug the application. Menu *Project→Download and Debug*. This action writes the program code to the eFLASH of the STM32H743XI microcontroller on the STM32H743I-EVAL board and starts the debugger.
11. Debug the application, examine menu *Debug*. Snapshot of running debugger can be seen in Figure 2.

    NOTE: The program remains in the eFLASH after the debug session is finished. If the board is powered up, the latest application will start automatically.

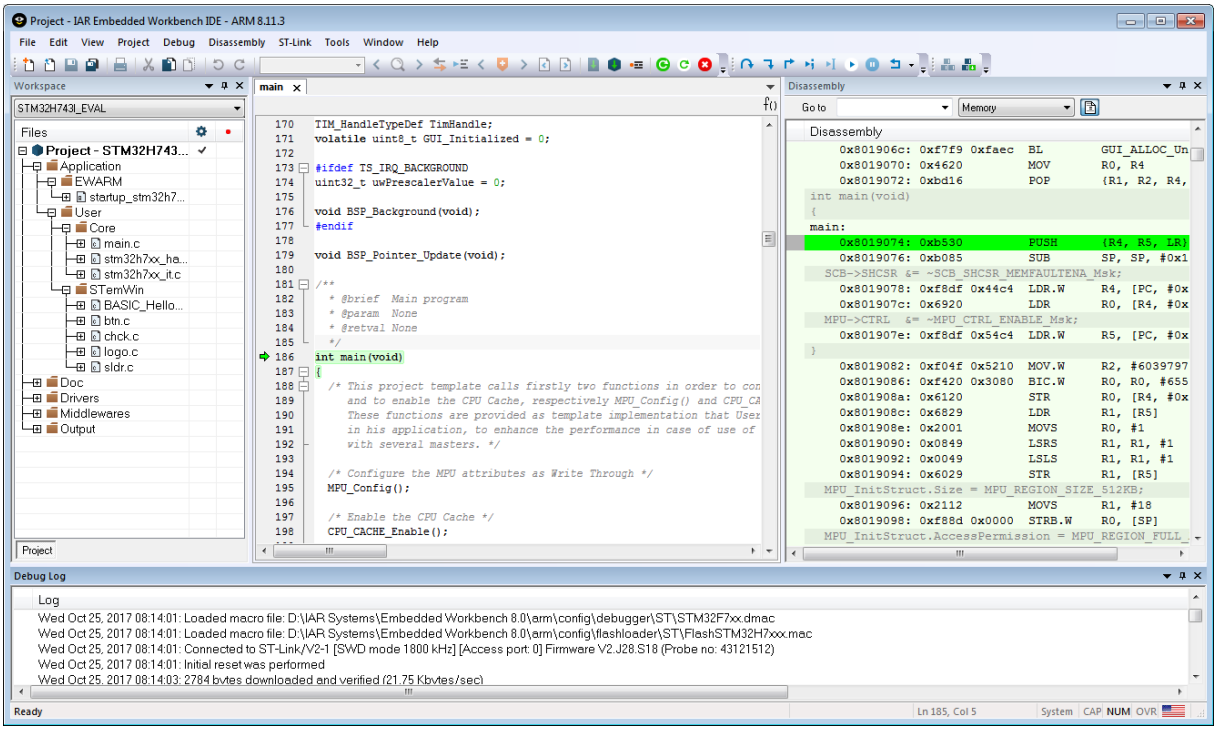

Figure 2: EWARM debugger.

The application can be tested without the remote device, commands can be observed via serial terminal (putty for instance). Settings of the terminal are presented in Table 3.

Table 3: Serial terminal settings

Parameter	Value
Baud rate	576000
Data bits	8
Stop bits	2
Parity	None
Flow control	None

5 Package contents

```
.
├── doc
│   └── cam_gui_optf_appnote.pdf
└── sw
    └── src
        ├── Core
        │   ├── Inc
        │   │   ├── btn.h
        │   │   ├── chck.h
        │   │   ├── colors.h
        │   │   ├── main.h
        │   │   ├── sldr.h
        │   │   ├── stm32h7xx_hal_conf.h
        │   │   └── stm32h7xx_it.h
        │   └── Src
        │       ├── main.c
        │       ├── stm32h7xx_hal_msp.c
        │       ├── stm32h7xx_it.c
        │       └── system_stm32h7xx.c
        ├── EWARM
        │   └── Project.ewp
        └── STemWin
            ├── App
            │   ├── BASIC_HelloWorld.c
            │   ├── btn.c
            │   ├── chck.c
            │   ├── logo.c
            │   └── sldr.c
            └── Target
                ├── GUIConf.h
                ├── GUIConf_stm32h743i_eval.c
                ├── LCDConf.h
                └── LCDConf_stm32h743i_eval.c
```

6 References

[1] STMicroelectronics, „UM2198 User manual Evaluation board with STM32H743XI MCU,“ 06 2017. [Online]. Available: http://www.st.com/content/ccc/resource/technical/document/user_manual/group0/48/5f/49/8e/5d/44/47/e3/DM00385071/files/DM00385071.pdf/jcr:content/translations/en.DM00385071.pdf

[2] STMicroelectronics, „STM32CubeH7,“ 10 2017. [Online]. Available: http://www.st.com/content/st_com/en/products/embedded-software/mcus-embedded-software/stm32-embedded-software/stm32cube-embedded-software/stm32cubeh7.html

[3] IAR Systems, „IAR Embedded Workbench,“ [Online]. Available: https://www.iar.com/iar-embedded-workbench/

[4] ÚTIA AV ČR, v.v.i., „Dense Optical Flow Demo, poster,“ 2017. [Online]. Available: http://sp.utia.cz/presents/posters/b_2017-000.jpg

7 Disclaimer

This disclaimer is not a license and does not grant any rights to the materials distributed herewith. Except as otherwise provided in a valid license issued to you by UTIA AV CR v.v.i., and to the maximum extent permitted by applicable law:

(1) THIS APPLICATION NOTE AND RELATED MATERIALS LISTED IN THIS PACKAGE CONTENT ARE MADE AVAILABLE "AS IS" AND WITH ALL FAULTS, AND UTIA AV CR V.V.I. HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and

(2) UTIA AV CR v.v.i. shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under or in connection with these materials, including for any direct, or any indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or UTIA AV CR v.v.i. had been advised of the possibility of the same.

Critical Applications:
UTIA AV CR v.v.i. products are not designed or intended to be fail-safe, or for use in any application requiring fail-safe performance, such as life-support or safety devices or systems, Class III medical devices, nuclear facilities, applications related to the deployment of airbags, or any other applications that could lead to death, personal injury, or severe property or environmental damage (individually and collectively, "Critical Applications"). Customer assumes the sole risk and liability of any use of UTIA AV CR v.v.i. products in Critical Applications, subject only to applicable laws and regulations governing limitations on product liability.