ARM® ARM926EJ-S

32-bit Microprocessor

NuMaker NuEZCam
User Guide

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

Table of Contents

1 INTRODUCTION

In the beginning we define the NuEZCam solution is that the NuEZCam board connects NuWicam debug board as shown in Figure 2–1.

NuMaker NuEZCam is the application to run AVI encoder, UVC+UAC and USB MSC of NuEZCam solution, AVI encoder could capture video and audio streams into one AVI file, and save the AVI file into SD card. It is based on Nuvoton's N3290X video MPU and provides a powerful JPEG codec for video encoder. The format of video stream is Motion-JPEG with VGA/HD(720P) resolution. The format of audio stream is PCM, 16 KHz, mono mode. For the application of UVC+UAC, user could run Windows tool AMCAP.exe to see and capture the result of UVC+UAC. As to USB MSC, user could run Windows explorer to see the content of the SD card for AVI encoder, and access the SD card. User could see UART log for NuEZCam solution. NuMaker_NuEZCam_Arduino_UNO.ino is Arduino sample, and could run under NuEdu UNO board to control NuEZCam solution by pressing the button and the specified flash times of LED is come and release the button. The flash times of LED is the input of UART log.

We also run ARM mbed OS by connecting NUC472 board and NuEZCam solution. The process is the same as Arduino, by using the SW1 button and specified flash time of LED.

In this document, we will describe the chapters as below:

- Board Interface
- Firmware Programming
- Demonstration
- Source code

2 BOARD INTERFACE

2.1 NuEZCam Solution

The NuEZCam solution includes NuEZCam board and NuWiCam debug board, the both boards connect together as shown in Figure 2–1,

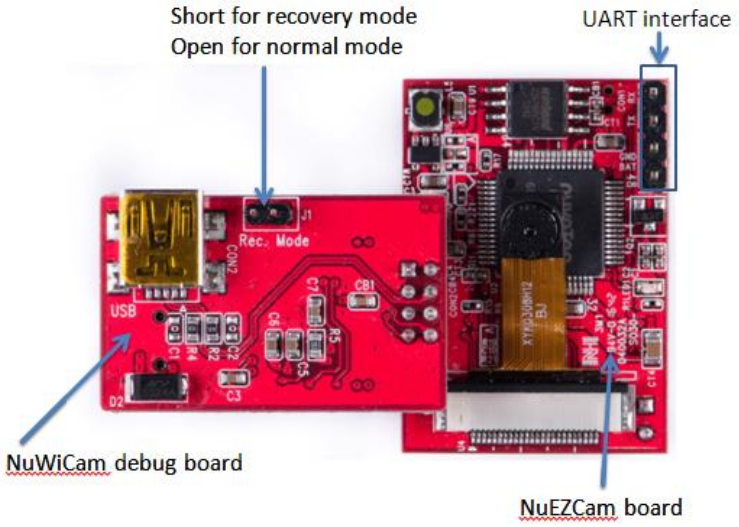
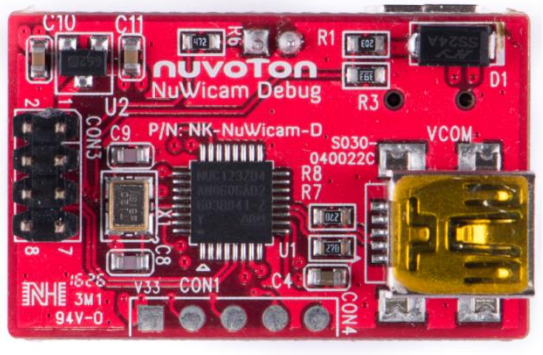


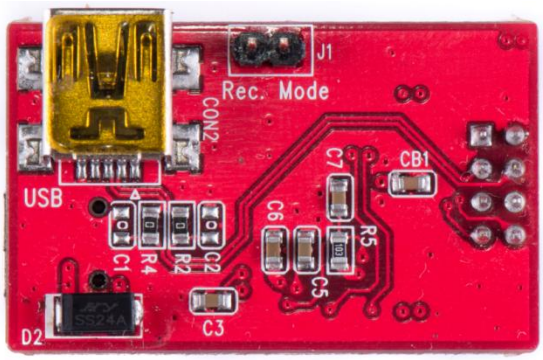
Figure 2–1 View of NuEZCam solution.

NuWicam debug board has two USB ports. The front view of NuWicam debug board is shown as follows.



The USB port is for USB VCOM. Before doing USB VCOM, user must install Windows driver VCOM_B002.exe within the subfolder utility of the path https://github.com/OpenNuvoton/NuMaker_NuEZCam_Samples. User should plug USB cable in between NuWicam debug board and Windows. User must use the pins of CON3 to connect NuEZCam board to become NuEZCam solution as shown in Figure 2–1. The power of USB cable could work the NuEZCam solution. The status of USB VCOM could be shown in UART log, so the execution result of NuEZCam solution could be shown in UART log by using USB VCOM.

The back view of NuWiCam debug board is shown as follows.



User could short J1 to set recovery mode, run Windows tool AutoWriter within the subfolder tool\AutoWriter_NuEZCam of the path https://github.com/OpenNuvoton/NuMaker_NuEZCam_Samples, and plug USB cable in between USB port and Windows. Then AutoWriter will burn the binary file AVIEncoder.bin into the SPI flash of NuEZCam solution automatically. About the processing of AutoWriter, please refer to the following chapter Firmware Programming for more details. Open J1 to set normal mode, and plug USB cable into between the USB port and Windows. The NuEZCam solution could execute UVC+UAC or USB MSC for USB applications.

The back view of NuEZCam board is shown as follows.

Before running the solution, user must plug microSD card in MicroSD slot. The audio of AVI encoder and UVC+UAC come from the microphone.

About the hardware information of NuEZCam board and NuWicam board, user could visit the subfolder hardware_design of the path https://github.com/OpenNuvoton/NuMaker_NuEZCam_RDK for more details.

2.2 NuEdu UNO board

In order to do the solution of NuMaker NuEZCam with Arduino, the setting of NuEdu UNO board should be shown in Figure 2–2. The sample code is located at the subfolder NuMaker_NuEZCam_Arduino_UNO of the the path https://github.com/OpenNuvoton/NuMaker_NuEZCam_Samples. If SW2 is VCOM mode, then the commands of NuEdu UNO board could not communicate with NuEZCam solution. So we must adjust the jumper 2, 3 and 4 to be off, SW2 is UART0 mode, then NuEdu UNO board could work with NuEZCam solution.

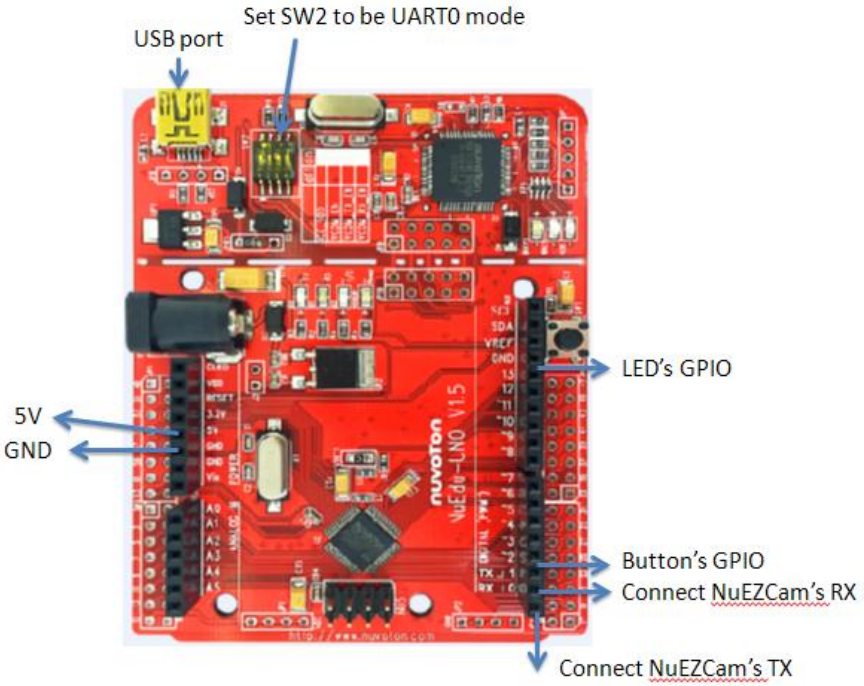


Figure 2–2 The Setting of NuEdu UNO board.

User could connect NuEdu UNO with NuEZCam solution in the following Figure 2–3.

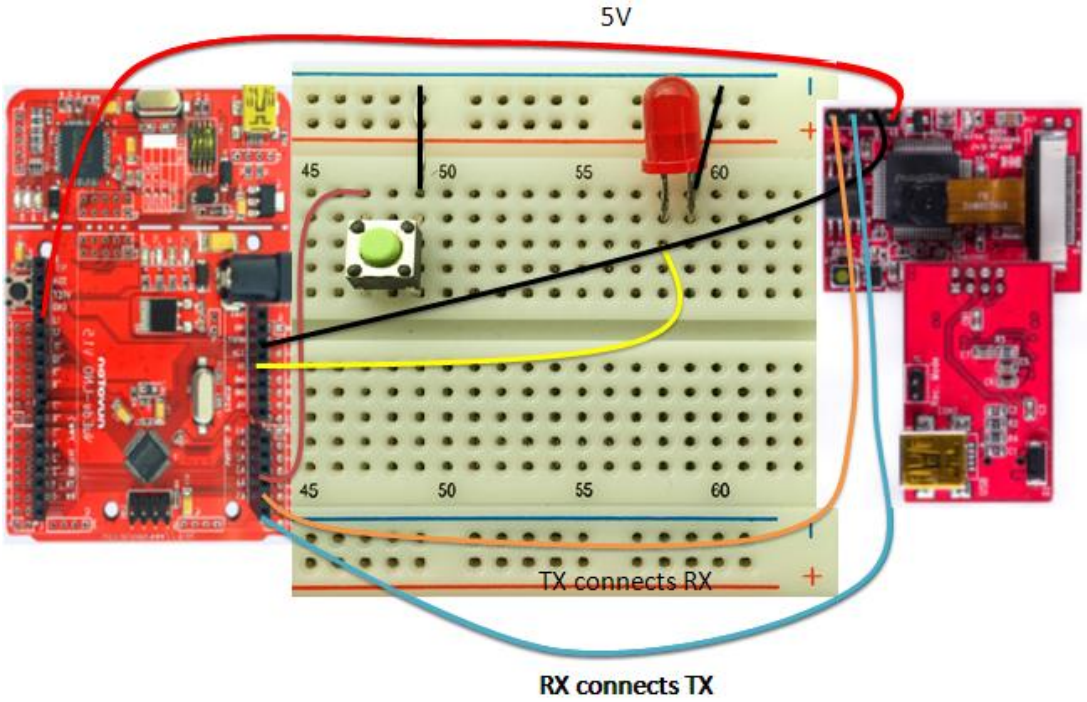


Figure 2–3 The connection between NuEdu UNO board and NuEZCam solution.

Then Arduino's code could control NuEzCam solution by using the button and specified flash times of LED. The specified flash times of LED is the input value for the options. User

presses down the button and see the flash times of LED. As the specified flash times of LED is come and user releases the button to set the input value. User could use NuEZCam solution to see the status of UART log. Then the UART log of NuEZCam solution is message display only. User cannot type in any data to input within the UART log, otherwise maybe NuEdu UNO board fails to connect NuEZCam solution.

2.3 NUC472 board

NUC472 board could connect NuEZCam solution in Figure 2–4. The code of ARM mbed OS (https://developer.mbed.org/compiler/) uses the SW1 button and specified flash times of LED to work. Currently the sample code of ARM mbed OS is located at the path (https://developer.mbed.org/users/shliu1/code/mbed-os-example-NuEZCam/). User could use on-line ARM mbed OS to run Keil to build the project, and obtain the built binary file. Later on user must adjust NUC472 board to be USB mass disk by shorting ICE_VBUS, and plug USB cable in. Windows explorer will display one MBED device, copy the binary file into the device and restart NUC472 board to run. User also opens ICE_VBUS to run debug mode for Keil. Same as NuEdu-Uno board, The specified flash times of LED is the input value for the options. User presses down the SW1 button and see the flash time of LED. As the specified flash times of LED is come and user releases the button to set the input value. Then the UART log of NuEZCam solution is message display only. User cannot key in to input within the UART log, otherwise maybe NuEdu UNO board fails to connect NuEZCam solution.

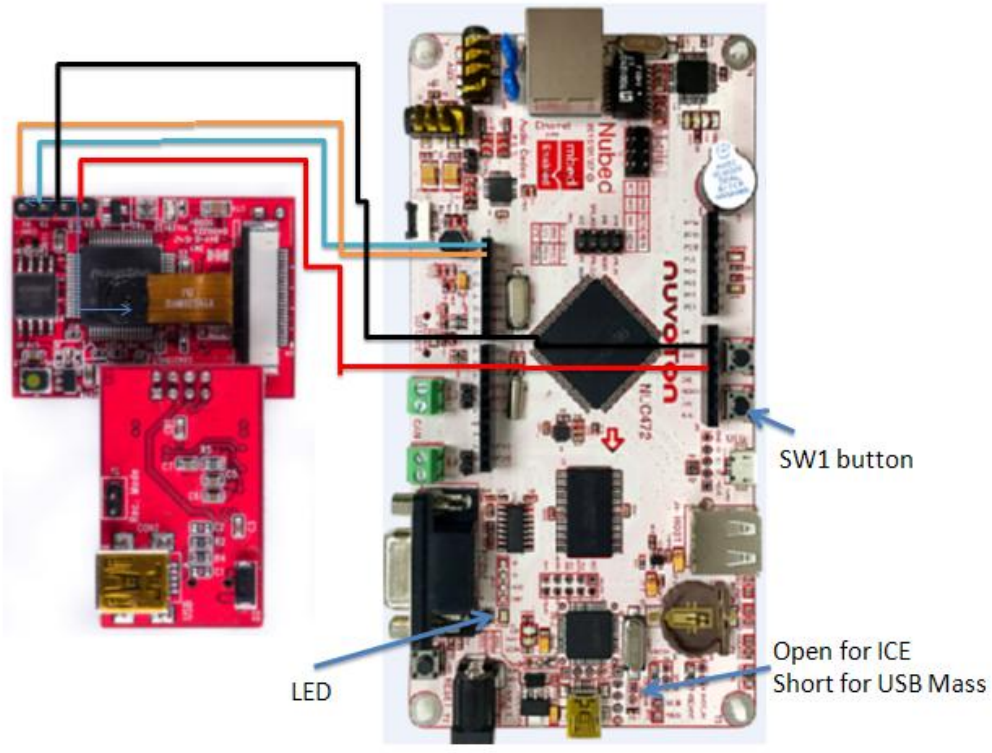


Figure 2–4 The connection between NUC472 board and NuEZCam solution

3 FIRMWARE PROGRAMMING

In this chapter, we will step by step to guide you program the firmware of AVI encoder for NuEZCam solution by using AutoWriter. We release the firmware for AVI encoder as shown the following Figure 3–1.

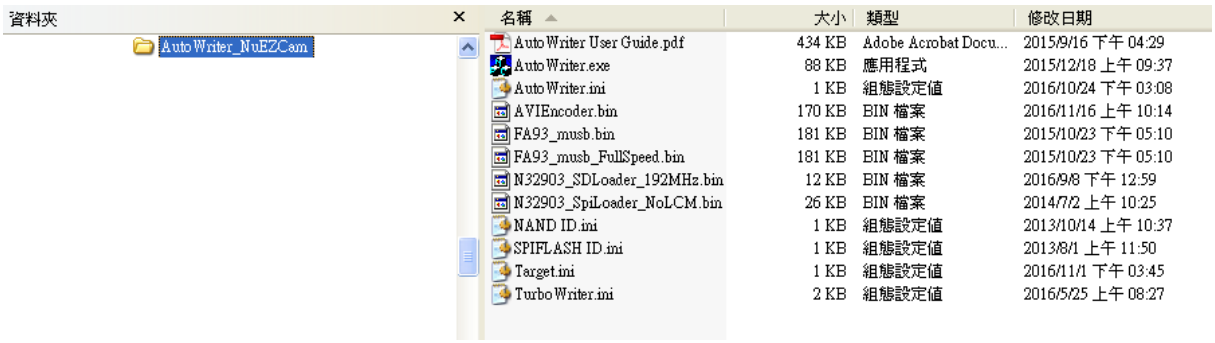


Figure 3–1 Firmware for NuEZCam solution

(1) running AutoWriter.exe execution, the UI of tool is shown in Figure 3–2. **The 'Current Target' is SPI by default**. Please keep the setting and follow the below steps:

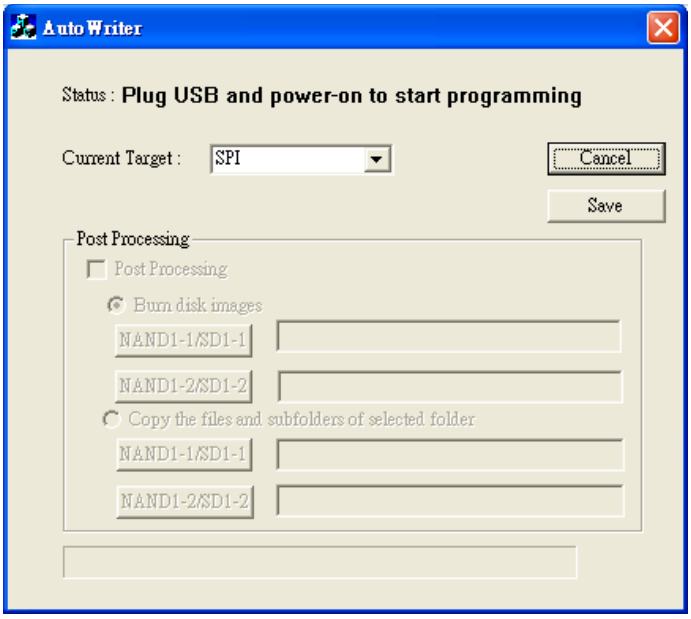


Figure 3–2 UI of AutoWriter

(2) sets the NuEZCam solution to be recovery mode, it means to short J1 pins of NuEZCam solution shown in Figure 3–3, and plug USB cable into PC/NB.

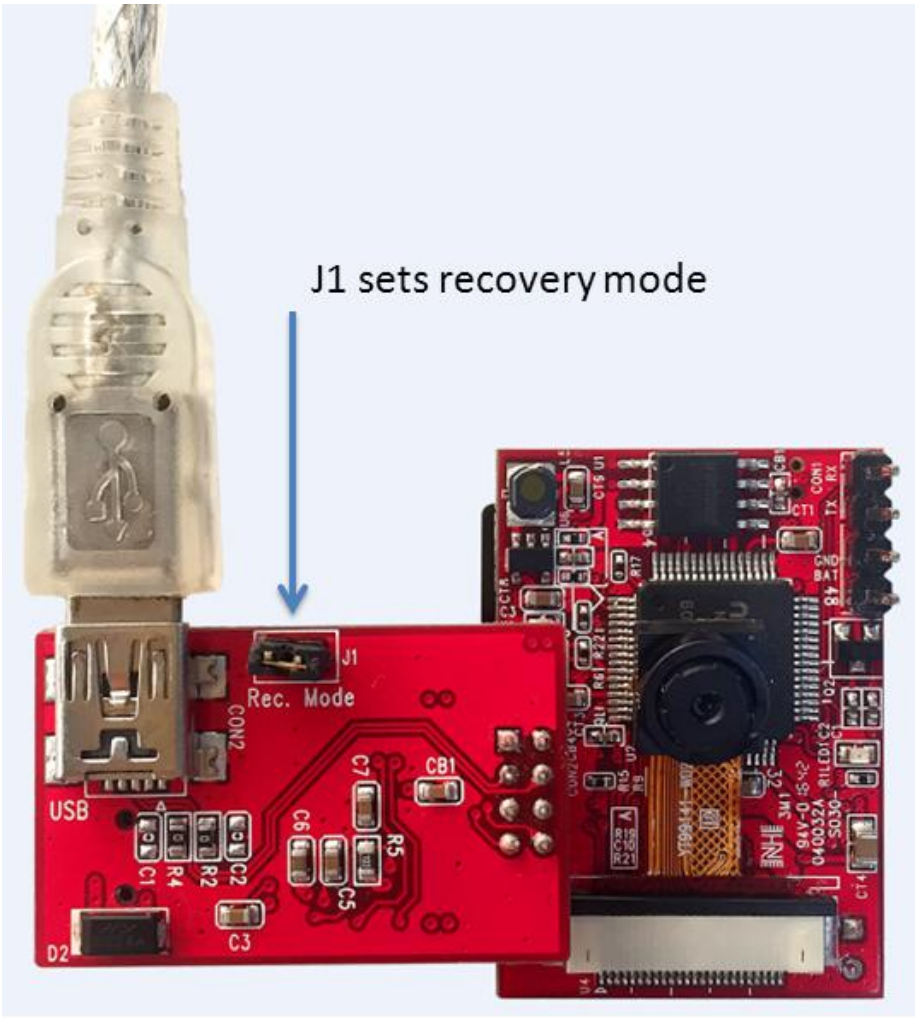
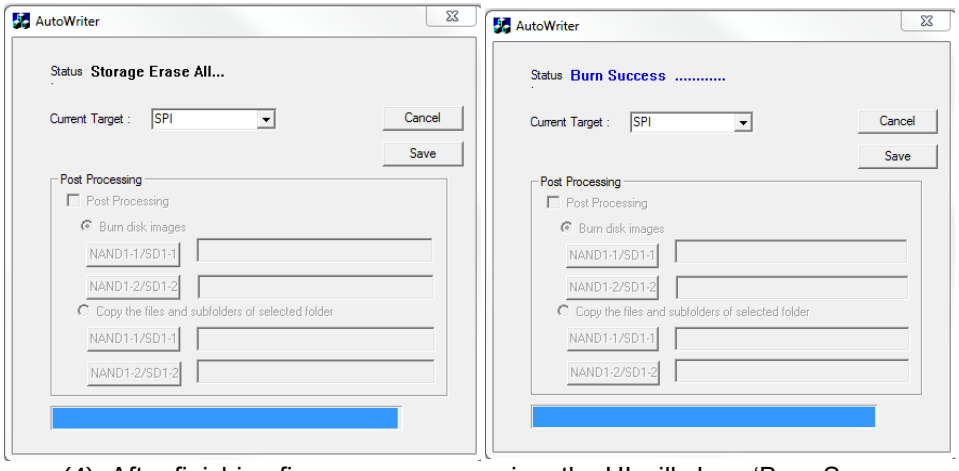


Figure 3–3 Booting setting in recovery mode

(3) Start the NuEZCam solution, the tool AutoWriter will burn the firmware automatically as follows



(4) After finishing firmware programming, the UI will show ‘Burn Success ….. ’.
(5) To Un-plug USB Line from PC.
(6) To leave ‘Recovery Mode’ and set ‘Normal mode’ by adjusting J1 to be open.

Then the operations of AutoWriter is finished.

4 DEMOSTRATION

In this chapter, we will demonstrate the solution of NuMaker NuEZCam by running NuEZCam solution alone mainly.

For NuEZCam solution alone, User could connect USB VCOM of NuEZCam solution to power on and boot the normal mode, you will see UART log as follows.

```
Init RTC....OK
DDR size:  8MB
SD Port0 Booting Fail - No Check ID
NAND Booting (Not support/No Device) Fail - No Check ID
SPI Booting Success
Clock Skew
 DQSODS 0x1010
 CKDQSDS 0x888800
Code Executes at 0x00700000
SPI Loader start (20131220).
DAC On
Load Image Load file length 0x400, execute address 0x80706764
Load file length 0x29C04, execute address 0x0
Jump to kernelDisable USB Transceiver
Disable ADC and LVR
Disable SPU and ADO
Disable USB phy
The code is for N32903
================================================================
Please select the target
================================================================
================================================================
 [1] AVI Encoder
 [2] USB UVC+UAC
 [3] USB MSC
================================================================
```

User types in ‘1’ to input, The solution sets AVI Encoder to execute, then the following UART log displays.

```
============================================================
 [1] NT99050 demo - 640X480
 [2] GC0308 demo - 640X480
 [3] NT99141 demo - 1280X720
 [4] Exit
============================================================
```

Now the solution supports three sensors NT99050, GC0308 and NT99141. User must select the correct sensor to do, otherwise it fails to work. ‘1’ for NT99050, ‘2’ for GC0308, ‘3’ for NT99141, ‘4’ stops AVI encoder and exit. After setting the sensor, UART log will be shown as follows.

```
Camera status
 [1] Normal photography
 [2] Time lapse photography
 [3] Exit
```

Normal photography is 30 FPS video including audio for sensors NT99050 and GC0308, 20 FPS video including audio for sensor NT99141. Time lapse photography is 1 FPS video only. Type in ‘1’ to run normal photography, The UART log is shown as follows.

```
Initial SD NonOS Driver (20160602 SDO_CD) for SD port 0
Sensor ID = 7
Device Slave Addr = 0x42
Sensor fail !
Please check the sensor !
Sensor ID = 8
Device Slave Addr = 0x54
Detectd sensor id0=14 id1=10
Sensor NT99141 is OK
encoded file = C:\smpl0001.avi
fsSetFileSize as 103 MB
fsSetFileSize take time ticks: 9
PLL clock = 184,363 KHz
Total divider = 9
DIV_N1, DIV_N0  = 3, 3
g_u32RecorderByte 32
```

```
T=1.00  (Vid #30 - 30) (Audio #200)
T=2.03  (Vid #61 - 31) (Audio #28)
T=3.06  (Vid #92 - 31) (Audio #96)
T=4.10  (Vid #123 - 31) (Audio #164)
T=5.13  (Vid #154 - 31) (Audio #32)
T=6.16  (Vid #185 - 31) (Audio #300)
T=7.20  (Vid #216 - 31) (Audio #168)
T=8.20  (Vid #246 - 30) (Audio #368)
T=9.23  (Vid #277 - 31) (Audio #36)
T=10.23  (Vid #307 - 30) (Audio #236)
T=11.26  (Vid #338 - 31) (Audio #104)
T=12.30  (Vid #369 - 31) (Audio #372)
T=13.33  (Vid #400 - 31) (Audio #40)
T=14.33  (Vid #430 - 30) (Audio #240)
T=15.36  (Vid #461 - 31) (Audio #308)
T=16.40  (Vid #492 - 31) (Audio #176)
Stop record
AVI record done.

Camera status
 [1] Normal photography
 [2] Time lapse photography
 [3] Exit
```

User could type in '1' to stop AVI encoder, otherwise AVI encoder will run for 120 seconds. User also snapshots one image to save JPEG file by typing in '2' as follows.

```
T=1.03  (Vid #31 - 31) (Audio #228)
T=2.03  (Vid #61 - 30) (Audio #428)
T=3.06  (Vid #92 - 31) (Audio #1,496)
T=4.16  (Vid #125 - 33) (Audio #300)
T=5.20  (Vid #156 - 31) (Audio #368)
T=6.23  (Vid #187 - 31) (Audio #236)
T=7.26  (Vid #218 - 31) (Audio #304)
T=8.30  (Vid #249 - 31) (Audio #172)
T=9.33  (Vid #280 - 31) (Audio #40)
T=10.33  (Vid #310 - 30) (Audio #240)
T=11.36  (Vid #341 - 31) (Audio #108)
T=12.40  (Vid #372 - 31) (Audio #176)
SnapShot one file C:\image001.jpg
T=13.20  (Vid #396 - 24) (Audio #4,768)
T=14.50  (Vid #435 - 39) (Audio #180)
T=15.53  (Vid #466 - 31) (Audio #248)
T=16.56  (Vid #497 - 31) (Audio #316)
T=17.60  (Vid #528 - 31) (Audio #184)
SnapShot one file C:\image002.jpg
T=18.60  (Vid #558 - 30) (Audio #384)
T=19.63  (Vid #589 - 31) (Audio #252)
T=20.66  (Vid #620 - 31) (Audio #120)
T=21.70  (Vid #651 - 31) (Audio #188)
T=22.70  (Vid #681 - 30) (Audio #388)
Stop record
AVI record done.

Camera status
 [1] Normal photography
 [2] Time lapse photography
 [3] Exit
```

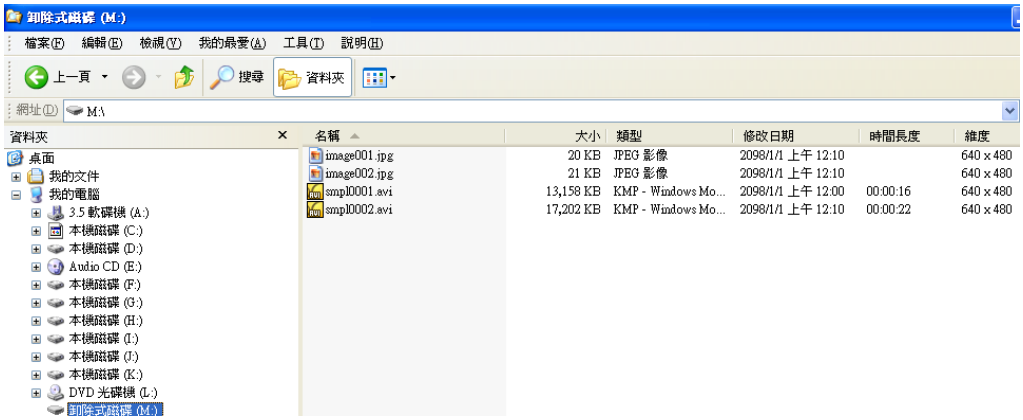
Type in '3' to exit AVI encoder and UART log is shown as follows.

```
 [1] AVI Encoder
 [2] USB UVC+UAC
 [3] USB MSC
======================
```

Back to the options of three applications. User should set USB MSC to see the result of AVI encoder. Before doing the option 2 and 3, user should connect USB cable between NuEZCam solution and PC/NB, then type in '3' to work USB MSC, UART log is shown as follows.

```
N3290 UDC Library (20150820)
Initial SD NonOS Driver (20160602 SDO_CD) for SD port 0
N3290 MSC Library (20151208)
MSC - SD Card detect pin is in use
```

User will find one USB disk under Windows explorer as follows, and checks the result of AVI encoder.



If user would like to end the process, plug USB cable out and UART log is shown as follows.

```
================================================================
 [1] AVI Encoder
 [2] USB UVC+UAC
 [3] USB MSC
================================================================
```

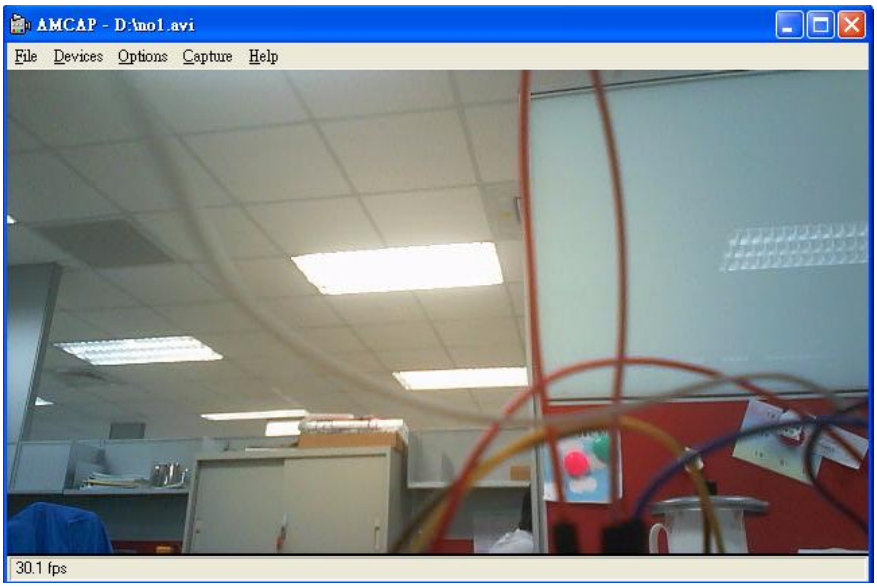
Before doing the option 2 USB UVC+UAC, user must connect USB cable between NuEZCam solution and PC/NB. Type in '2', the UART log is shown as follows,

```
================================================================
 [1] NT99050 demo - 640X480
 [2] GC0308 demo - 640X480
 [3] NT99141 demo - 1280X720
 [4] Exit
================================================================
```

Now the application supports three sensors NT99050, GC0308 and NT99141. User must set the correct sensor to do, otherwise it fails to work. After setting the sensor, UART log will be shown as follows.

```
Sensor ID = 7
Device Slave Addr = 0x42
Sensor fail !
Please check the sensor !
Sensor ID = 8
Device Slave Addr = 0x54
Detectd sensor id0=14 id1=10
N3290 UDC Library (20150820)
N3290 UAVC Library (20141217)
PLL clock = 184,363 KHz
Total divider = 9
DIV_N1, DIV_N0  = 3, 3
g_u32RecorderByte 32
```

and user could run the Windows tool AMCAP.exe to see the status.



If user would like to end the processing, close the tool AMCAP.exe and plug out the USB cable. UART log will be shown as follows.

```
UVC end
============================================================
 [1] AVI Encoder
 [2] USB UVC+UAC
 [3] USB MSC
============================================================
```

As to NuEZCam solution + NuEdu UNO board, user connects between NuEZCam solution and NuEdu UNO board like Figure 2–3.

Same as NuEZCam solution alone, User could connect USB VCOM of NuEZCam solution to power on and boot the normal mode, you will see the message of UART log. But user must NOT input any data within UART log, otherwise maybe NuEdu UNO board fails to work. Therefore UART log is only for display message.

The processing between NuEZCam solution alone and NuEZCam solution + NuEdu UNO board are the same, The input method is different. For example, type in '2' to input '2' into UART log for NuEZCam solution alone. For NuEZCam solution + NuEdu UNO board, user runs Arduino code NuMaker_NuEZCam_Arduino_UNO.ino to press down the button and see the flash of LED for two times and release the button, then Arduino code sends '2' to UART log. Laterly LED will flash for three time automatically to acknowledge. If LED could not flash automatically, it means that the connection fails, user must restart NuEZCam solution + NuEdu UNO board to work. Don't input any data within UART log by running NuEZCam solution + NuEdu UNO board.

As to NuEZCam solution + NUC472 board, user connects between NuEZCam solution and NUC472 board like Figure 2–4.

Same as NuEZCam solution alone, User could connect USB VCOM of NuEZCam solution to power on and boot the normal mode, you will see the message of UART log. But user must NOT input any data within UART log, otherwise NUC472 board fails to work. Therefore UART log is only for display message.

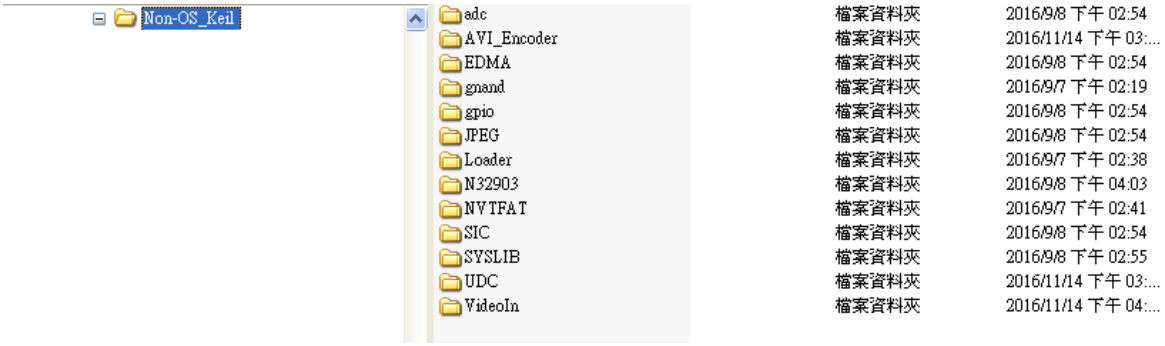
The processing between NuEZCam solution alone and NuEZCam solution + NUC472 board are the same, The input method is different. For example, type in '3' to input '3' into UART log for NuEZCam solution alone. For NuEZCam solution + NUC472 board, user runs the code of mbed OS main.cpp to press down the button and see the flash of LED for three times and release the button, then code of mbed OS sends '3' to UART log. Laterly LED will flash for three time automatically to acknowledge. If LED could not flash automatically, it means that the

connection fails, user must restart NuEZCam solution + NUC472 board to work. Don’t input any data within UART log by running NuEZCam solution + NUC472 board.

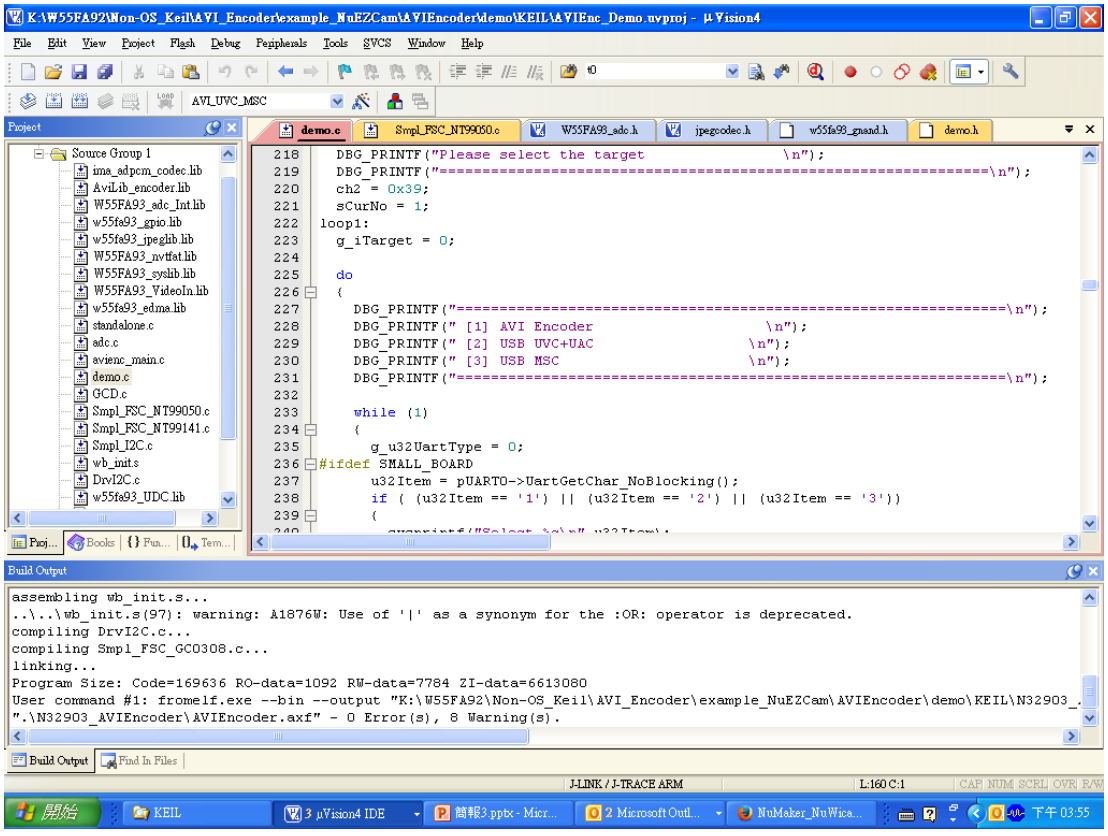
5 SOURCE CODE

The source code of AVI encoder is built by Keil. Currently the application supports the three sensors NT99141 (HD, 1280x720, 20 FPS), GC0308 (VGA, 30 FPS) and NT99050 (VGA, 30 FPS). Therefore the resolution supports HD and VGA.

The source code of Non-OS Keil is shown as follows.



User could open the project AVIEnc_Demo.uvproj within the subfolder AVI_Encoder\example_NuEZCam\AVIEncoder\demo\KEIL to build and export one binary file AVIEncoder.bin.



If user would like to use the other sensor, please contact Nuvoton for more information.

6 Q&A

Q: How does user build the source code of AVI encoder ?

A: User could unzip the file Non-OS_KeilBSP.7z, and build the project AVIEnc_Demo.uvproj within the folder AVI_Encoder by using Keil 4.54 later. Keil is not a free software.

Q: How many boards could do with NuEZCam under mbed OS ?

A: Nuvoton's NUC472 and M453, STM32F401, NXP K64F.

7 REVISION HISTORY

Date	Revision	Description
2016.12.01	1.02	1. Use NuEZCam solution.
2016.09.13	1.01	1. Initially issued.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.