APPLICATION NOTES

4D SYSTEMS
TURNING TECHNOLOGY INTO ART

Visi-Genie Magic How to Read a File Arduino

DOCUMENT DATE:	**8th MAY 2020**
DOCUMENT REVISION:	**1.1**

Description

This application note shows how to write a sketch for an Arduino program that sends a **WRITE_MAGIC_BYTES** message to and handles **REPORT_MAGIC_EVENT_BYTES** messages coming from the display module.
Note: Workshop Pro is needed for this application.

Before getting started, the following are required:

- Any of the following 4D Picaso **touch** display modules:

uLCD-24PTU	uLCD-32PTU	uLCD-43(PT/PCT)
uLCD-28PTU	uLCD-32WPTU	

  and other superseded modules which support the ViSi-Genie environment

- The target module can also be a Diablo16 **touch** display

uLCD-35DT	uLCD-70DT
uLCD-43DT	uLCD-43DCT

  Visit www.4dsystems.com.au/products to see the latest display module products that use the Diablo16 processor.

- 4D Programming Cable or µUSB-PA5
- micro-SD (µSD) memory card
- Workshop 4 IDE with Pro License (installed according to the installation document)
- An Arduino board with a UART serial port
- 4D Arduino Adaptor Shield (optional) or connecting wires
- Arduino IDE
- When downloading an application note, a list of recommended application notes is shown. It is assumed that the user has read or has a working knowledge of the topics presented in these recommended application notes.

Content

Application Overview

To understand this application note more quickly, the reader is advised to read and understand first the application note below.

ViSi-Genie Magic How to Read a File

ViSi-Genie Magic How to Read a File shows how the Magic Object is used to implement a ViSi-Genie project that allows the host to access files on the uSD card of the display. The Magic Object is one of the objects under the Magic pane. It is actually a 4DGL code that allows the user to customise the behaviour of a Genie program. The Magic Object handles the WRITE_MAGIC_BYTES messages coming from the host. A WRITE_MAGIC_BYTES message may contain a file operation command byte and a filename. The Magic Object then uses 4DGL string class and FAT16 file functions to perform the requested file access operation. The Magic Object then sends REPORT_MAGIC_EVENT_BYTES messages back to the host. A REPORT_MAGIC_EVENT_BYTES message may contain data read from the file, list of filenames, file size information, etc., and/or acknowledgment.

In this application note, a sketch for an Arduino program that sends a WRITE_MAGIC_BYTES message to and receives and handles REPORT_MAGIC_EVENT_BYTES messages from the display module will be discussed. The WRITE_MAGIC_BYTES message will contain the name of the file to be read and the REPORT_MAGIC_EVENT_BYTES messages will contain the contents of the file.

Setup Procedure

At this point, it is assumed that the reader has a working setup of the project described in the application note **ViSi-Genie Magic How to Read a File.** The necessary files should also be present on the uSD card. The remaining task now is to write the sketch for the Arduino program and upload the program to the Arduino host.

Note 1: The attached sketch was tested on an Arduino Uno. A software serial port was used for communicating with the display. The hardware serial port, Serial0, was used for communicating with the Serial Monitor of the Arduino IDE.

Note 2: The attached ViSi-Genie project is a slightly modified version of the example project "**FileAccess.4DGenie**" in Workshop. Delays were inserted to the ViSi-Genie project code to give the Arduino host time to "catch-up". See the attached image "**insertedDelayGenie.png**" for more details. Without the inserted delays, the Arduino host would hang up or miss several bytes since it is performing other processes besides the handling of bytes coming from the display module.

Program the Arduino Host

A thorough understanding of the application note **ViSi-Genie Connecting a 4D Display to an Arduino Host** is required before attempting to proceed further beyond this point. **ViSi-Genie Connecting a 4D Display to an Arduino Host** provides all the basic information that a user needs to be able to get started with ViSi-Genie and Arduino. The following is a list of the topics discussed in **ViSi-Genie Connecting a 4D Display to an Arduino Host**.

- How to download and install the ViSi-Genie-Arduino library
- How to open a serial port for communicating with the display and how to set the baud rate
- **The genieAttachEventHandler() function – for Genie Magic, the equivalent of this is AttachMagicByteReader( )**
- How to reset the host and the display
- How to set the screen contrast
- How to send a text string
- The main loop
- Receiving data from the display
- The use of a non-blocking delay in the main loop
- How to change the status of an object
- How to know the status of an object
- **The user's event handler – for Genie Magic, the equivalent of this is myGenieMagicHandler()**

Discussion of any of these topics is avoided in other ViSi-Genie-Arduino application notes unless necessary. Users are encouraged to read **ViSi-Genie Connecting a 4D Display to an Arduino Host** first.

Understanding the Demo Sketch

Open the Arduino sketch "**FileReadSS.ino**" and the PDF file "**programFlow.pdf**" attached to this document. The PDF file explains the flow of the program.

Attach the Magic Event Handler

```
genie.AttachMagicByteReader(myGenieMagicHandler);
```

This simply tells the ViSi-Genie Arduino library which function to call to process the magic events received from the display. The function to be called is a user-defined function.

Set Up the Project

Refer to the section "**Connect the Display Module to the Arduino Host"** of the application note "**ViSi-Genie Connecting a 4D Display to an Arduino Host**" for the following topics:

- Using the New 4D Arduino Adaptor Shield (Rev 2.00)
    - Definition of Jumpers and Headers
    - Default Jumper Settings
    - Change the Arduino Host Serial Port
    - Power the Arduino Host and the Display Separately
- Using the Old 4D Arduino Adaptor Shield (Rev 1)
- Connection Using Jumper Wires
- Changing the Serial port of the Genie Program
- Changing the Maximum String Length

Note 1: The attached Arduino sketch uses a software serial port to communicate with the display module and a hardware serial port to communicate with Serial Monitor running on the PC.

Note 2: There are three filenames hardcoded in the sketch. Choose one at a time by commenting out the remaining two filenames before compiling the sketch and uploading the program to the target Arduino board.

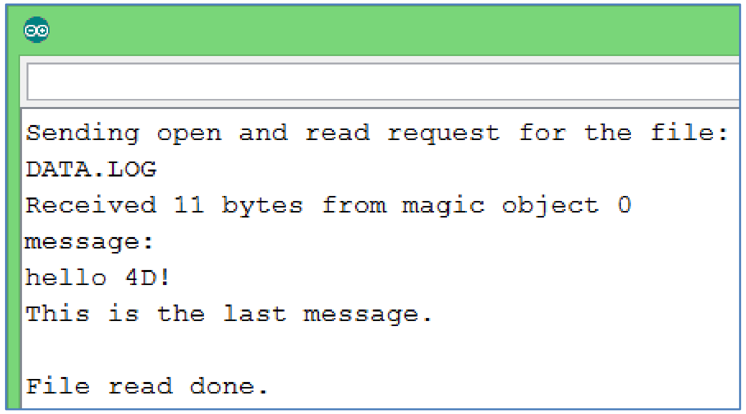
```
byte filename[] = {MFILE_READ, 'D', 'A', 'T', 'A','.',
        'L', 'O', 'G', NULL};     // Example 1

//byte filename[] = {MFILE_READ, 'D','U','M','M','Y',
        '.','T','X','T', NULL};        // Example 2

//byte filename[] = {MFILE_READ, 'A', 'S', 'C', 'I','I',
             'T','X','T', NULL};   // Example 3
```

Expected Serial Terminal Output

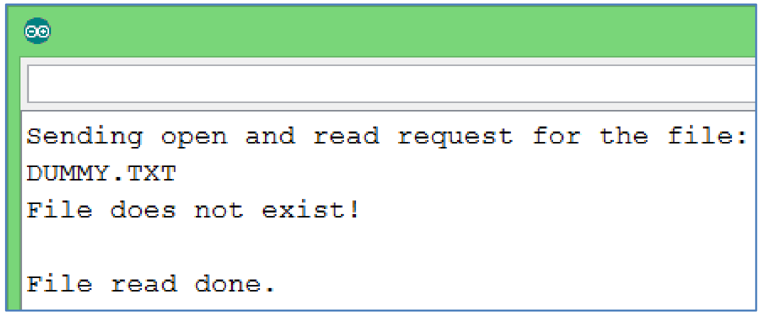
Example 1



```
Sending open and read request for the file:
DATA.LOG
Received 11 bytes from magic object 0
message:
hello 4D!
This is the last message.

File read done.
```

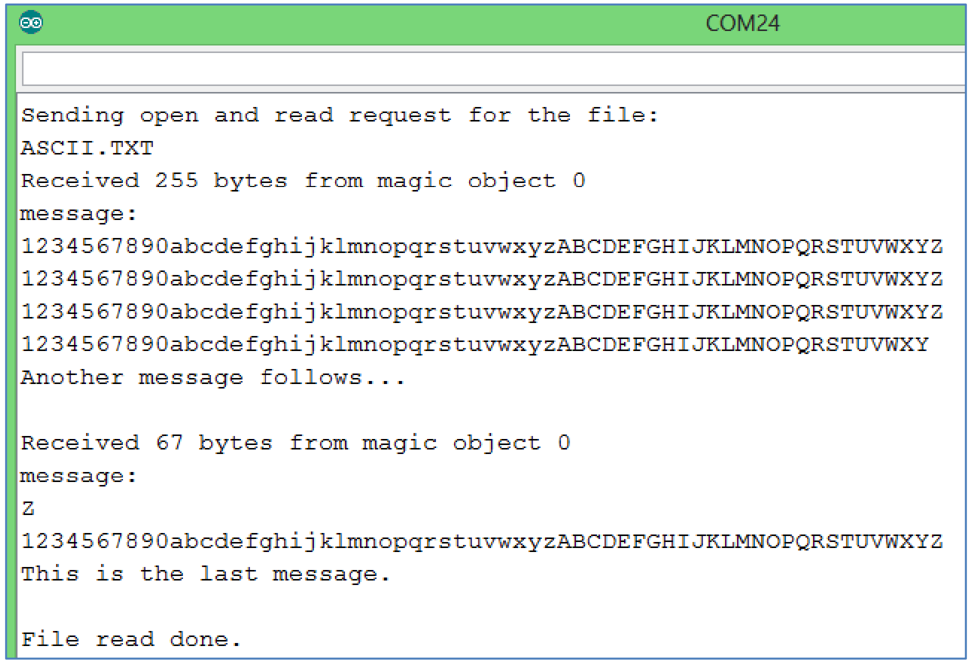
Example 2



```
Sending open and read request for the file:
DUMMY.TXT
File does not exist!

File read done.
```

Example 3



```
Sending open and read request for the file:
ASCII.TXT
Received 255 bytes from magic object 0
message:
1234567890abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXY
Another message follows...

Received 67 bytes from magic object 0
message:
Z
1234567890abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
This is the last message.

File read done.
```

Proprietary Information

Disclaimer of Warranties & Limitation of Liability