**Wispr RPi SD Card Interface**

The RPi mounts to the bottom of the Wispr main board using the 40 pin RPi bus connector. The SD cards on the Wispr main board are connected to the Rpi using a multiplexer that switches the cards between the Wispr and the RPi. When the Wispr is writing to one SD card, the other card is connected to the RPi bus. The two cards on Wispr are toggled between the Wispr processor and the RPi. So when Wispr is writing data to one card, the RPi can read data from the other card.

***Enabling the Rpi from Wispr***

Power to the RPi is provided by the Wispr main board through the bus connector. A separate switching regulator on the Wispr board provides the 5 Volts supply to the RPi. The WISPR processor control the ON/OFF state of this 5V regulator through software.

The commands to turn ON and OFF the RPi from Wispr are:

ioport\_set\_pin\_level(PIN\_ENABLE\_5V, 1); // Turn RPI ON

ioport\_set\_pin\_level(PIN\_ENABLE\_5V, 0); // Turn RPI OFF

***Setting up the Rpi to mount the Wispr SD card***

The standard RPi kernel doesn’t include automatic detection of a secondary SD card present on the RPi bus. The standard kernel also doesn’t include support to recognize exFAT file systems. So you will need to install updates and change the RPi kernel.

First install support for exFAT on your RPi using the following commands:

sudo apt-get update

sudo apt-get install exfat-fuse exfat-utils

Then modify the kernel configuration to detect a second SD card at boot by adding the following lines to the file */boot/config.txt* on your RPi.

# Enable second SD card

dtparam=sdio\_overclock=25 # This makes the Pi4 Rev 1.1 work for some reason

dtoverlay=sdio,poll\_once=off

NOTE: On the RPi 4, enabling “dtoverlay=sdio” excludes the use of wireless networking on the RPi board.  The Broadcom chip’s wireless functionality shares the same GPIO as the external SD card access pins and thus it is “either/or” in setting whether to see the SD card using SDIO or using the wireless network.

While you’re modifying */boot/config.txt* add the following line to enable the extra UART lines on the RPi bus. Or use `sudo raspi-config` on the desktop version, and edit the “Interfacing” options to disable “serial from USB” and enable “UART”.

enable\_uart=1

This extra UART on the Rpi can be used as a console or for communication.

Once you’ve done all this, when you boot the RPi connected to the Wispr and the SD card is enabled by the Wispr software, The SD card should show up as a device file. Now you can mount the SD card on the RPi as follows: (add this to “/etc/rc.local” or “~/.profile” to execute every time the RPi boots)

mkdir /media/wispr\_sd

sudo mount -t exfat /dev/mmcblk1p1 /media/wispr\_sd

At this point you should be able to see the file on the card and open them to access data.

**Using diskpart to format SD card from windows command prompt**

Open window command prompt as ***administrator***.

Microsoft Windows [Version 10.0.19043.1586]

(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>diskpart

Microsoft DiskPart version 10.0.19041.964

Copyright (C) Microsoft Corporation.

On computer: CHRIS-ZENBOOK

DISKPART> list disk

Disk ### Status Size Free Dyn Gpt

-------- ------------- ------- ------- --- ---

Disk 0 Online 476 GB 0 B \*

Disk 1 Online 476 GB 476 GB

DISKPART> select disk 1

Disk 1 is now the selected disk.

DISKPART> clean

DiskPart succeeded in cleaning the disk.

DISKPART> create partition primary

DiskPart succeeded in creating the specified partition.

DISKPART> format fs=exfat