



GameCube Controller Communication Protocol

Version 1.0
2018/08/04

About

This document contains the specification of the interface between the official Nintendo GameCube controller and its master. Its master being a Nintendo GameCube console or a Nintendo Wii console. The purpose of this document is to maintain this information in a clean format that can be referenced later. All information in this document was generated from the GC+ replacement controller board repository.
<https://github.com/Aurelio92/GCPlus>

Overview

The GameCube Console (now referred to as master) and GameCube Controller (now referred to as slave) communicate over a one-wire bus with a pull-up resistor to **3.3v**. The bit stream itself is duty cycle based where one bit is approximately 4-5µs wide and a duty cycle above 50% is a **1**, below 50% is **0**.



Both the master and slave communicate in a string of bytes that is terminated with a stop bit. So any communication will be **8*n + 1** bits where **n = number of bytes**.

A stop bit can be detected by timing how long the bus has been active if high. If it's longer than the standard bit width (>5µs) then communication has ended. The open-drain connections simply release control of the bus.

GameCube Master Command Set

The master communicates with the slave with a set of 5 one-byte commands with 0-2 optional 1 byte parameters. The command set is defined as follows.

*NOTE: The parameters refer to bytes that follow the initial command from the master. For example the 0x40 command looks like 0x40 0x03 0x00
Where 0x03 0x00 are the two parameters passed from the master*

Command	Parameters	Description
0x00	0	Get ID/Hand Shaking
0x40	2	Get Button States
0x41	0	Re-calibrate
0x42	2	Re-calibrate
0xFF	0	Reset

Command Description

- **0x00 & 0xFF - Get ID/Reset**
 - The Get ID and RESET commands are used for polling for the slave information and re-initializing if need be. An official Nintendo GameCube controller responds with

0x09 0x00 0x30

NOTE: The 3 at the end may not be necessary as some controllers have responded with 2. It's possible that this is merely a controller revision number.

Other controller definitions, like the Wavebird for example, can be found in the famous "Yet Another GameCube Documentation"

- **0x41 & 0x42 - Re-calibrate**
 - As far as we are concerned both of these commands are the same. They request that the slave re-calibrate it's analog positions to find their new "centers". The only difference is that 0x42 also has two parameters passed from the master. At this time it is undocumented what these parameters are for but they are not necessary for a controller replacement. It's easier to handle the calibration manually on the slave side and hand back a standard 10 byte answer to the master.

0x00 0x80 0x80 0x80 0x80 0x80 0x00 0x00 0x00 0x00

This is typically what's sent back to the master from a standard slave after receiving a re-calibrate command.

- **0x40 - Get Buttons**

- The get buttons command requests the button states and analog positions from the slave. The first parameter is used to specify the analog mode requested by the master, and the second parameter defines the rumble. A typical "get buttons" command is formatted like so:

0x40 0x03 0x00

Where 0x03 is the analog mode, which is the most common, and 0x00 is the rumble command.

Rumble Definition

The rumble command is parameter 2 of the get buttons command. It controls an H-Bridge on the slave that controls the rumble motor. This parameter is defined as:

Parameter	Description
0x00	Free Spin
0x01	Motor On
0x02	Hard Stop/Hold

Analog Mode Definition

The analog mode defines the second 4 bytes of the of the 8 byte response word. The first 4 bytes are defined in the following section. We'll refer to these last 4 bytes as **B4**, **B5**, **B6**, and **B7**. The analog mode can be 0-7, but 0, 5, 6, and 7 are all the same. Six out of the eight analog values are passed from the slave in these last four bytes. These values are

CX - C-Stick X-Axis

CY - C-Stick Y-Axis

LA - Left Analog Potentiometer

RA - Right Analog Potentiometer

AA - A Analog Value

AB - B Analog Value

*Note: **AA** and **AB** are separate analog values that are used in special controllers with extra analog values. These are not to be confused with the digital A and B value To avoid confusion I have relabeled them **AA** and **AB**. The official Nintendo Controller simply returns them as 0x00.*

The return logic for the analog modes is defined as follows:

Mode 0, 5, 6, or 7	
B4	CX
B5	CY
B6	(LA & 0xF0) (RA >> 4)
B7	(AA & 0xF0) (BA >> 4)

Mode 1	
B4	(CX & 0xF0) (CY >> 4)
B5	LA
B6	RA
B7	(AA & 0xF0) (BA >> 4)

Mode 2	
B4	(CX & 0xF0) (CY >> 4)
B5	(LA & 0xF0) (RA >> 4)
B6	AA
B7	AB

Mode 3	
B4	CX
B5	CY
B6	LA
B7	RA

Mode 4	
B4	CX
B5	CY
B6	AA
B7	AB

The first 4 Bytes of the Returned status word are as follows

First 4 bytes of Slave Status Word								
B0	0	0	0	Start	Y	X	B	A
B1	1	L	R	Z	D-U	D-D	D-R	D-L
B2	JoyStick X Value							
B3	JoyStick Y Value							

NOTE: The digital values for the buttons are active high, so a '1' means it is pushed
NOTE 2: Bits 0 and 1 of B0 are error bits that can be ignored, bits 2 and 0 of B0 and B1 respectively are unused.
NOTE 3: The Left and Right analog values are 0xFF when not pushed. They are in essence active low.

Change History

Document Creation 5/22/2017