Hardware Research

Thomas Schenk

Contents

[Introduction 3](#_Toc153821153)

[Problem 3](#_Toc153821154)

[Research Questions 4](#_Toc153821155)

[Requirements for a fitting hardware setup 5](#_Toc153821156)

[1. Which wireless communication method is best for audio transfer in our usecase? 6](#_Toc153821157)

[2. How do commercial wireless speakers work? 7](#_Toc153821158)

[2.1 DAC/DSP 7](#_Toc153821159)

[2.2 Amplifier 8](#_Toc153821160)

[3. Which signal parameters are commonly found in the music industry? 8](#_Toc153821161)

[Sample rate 8](#_Toc153821162)

[Bit Depth 9](#_Toc153821163)

[3.1 How do technical parameters relate to hardware choices? 9](#_Toc153821164)

[Conclusion 10](#_Toc153821165)

[DAC 10](#_Toc153821166)

[Wireless receiver 10](#_Toc153821167)

[Sources 11](#_Toc153821168)

## Introduction

To aid in choosing and considering multiple pieces of hardware, research is conducted to help choose the best. The “best” is determined alongside a set of given questions/needs to which each additional piece will be tested.

## Problem

Currently, there is no to little insight into which hardware components we need to use to successfully achieve a working wireless audio system.

## Research Questions

Main Research Question: What Hardware do we need to build a wireless audio system?

1. Which wireless communication method is best for audio transfer in our usecase?
2. How do commercial wireless speakers work?
3. Which signal parameters are commonly found in the music industry?

## Requirements for a fitting hardware setup

Naturally, with any project, there are a few parameters which have significant importance in deciding our final verdict.

One important notice is that we are omitting the research for an actuator; in our case: a speaker. This research is only for the processing and receiving of the wireless information.

#### Form Factor

The form factor of the complete setup must be small, so small it can be easily embedded or even integrated later. This means that if there are any receivers which function on a single chip; these are preferred since they can easily be transferred to an already existing system.

#### Simplicity

The goal of this project is to make a simple addon to an already existing system, this entails that the end product and its hardware; shouldn’t be immensely complex. The fewer parts, the better.

#### Audio Quality

A direct correlation exists between hardware and audio quality; the system needs to withstand itself compared to “real” wireless speakers. Of course, audio quality is subjective, but there are clear lines to be drawn as to what is bad; and what is good. And a choice in processing hardware can play a major role in that.

By answering our research questions, we can pull a fitting conclusion on a group of available hardware to pinpoint which is the best for the usecase.

# Which wireless communication method is best for audio transfer in our usecase?

There exist many wireless communication methods, both oneway and twoway. But which is the best for our application?

To answer this, we must first look again at our usecase for this project; we can use this to narrow it down significantly. To provide a quick reminder, our usecase can be shortly summarized as follows:

“The user turns on the audio system. When the system is booting up, they grab their favourite type of device (Phone, Tablet, Pc), they use this to quickly connect with the audio system and transfer the audio wirelessly.”

One of the great things we can take away from this usecase, is the device which the user will use to connect to our system. In this case, these are devices which are either already on the user; or are near them. Usually in a home-like situation. To limit ourselves, we are going to assume these are either:

* A phone
* A tablet
* A laptop/pc

Knowing this, we can look at which stock wireless functionalities each device usually has and make an estimate list of this.

The protocol below is the most supported and used wireless communication protocols for the abovementioned devices. To quickly show how effective they would be for our project, they are put in a nice spreadsheet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Range | Cost | Transfer speed | Power consumption |
| Bluetooth | Approx. 10M (33ft) | Low | 2 Mbps | Low |
| Cellular | Approx. 3 to 6.5Km (2 to 4 miles)\*1 | High | Approx. 100 KBs to 1Mbps | Moderate |
| Wi-Fi | Approx. 20 to 110M (66ft to 40ft) | Moderate | Varying from 100Mbps to 3500 Mbps\*2 | Moderate |

1. This is assumed the cellular network that’s already in place, is used.
2. Lab technical transfer speeds: normally in Embedded applications speeds of 10Mbps are common.

From this chart, we can quickly deduce that Bluetooth is the far superior option. Not only is it cost effective, with a low power consumption. More importantly, it is not overkill. Wi-Fi and Cellular, while superior when it comes to peak performance, are way overkill for the 9000kbps needed to stream high quality audio. Add the higher cost and power consumption, and it is apparent why Bluetooth is preferred.

# How do commercial wireless speakers work?

Wireless speakers are all around us nowadays, though how do they work? What drives them? And thus, what might we need which we don’t have already?

A wireless speaker usually consists of three main components:

* Actuator
* DAC/DSP (digital to analogue converter/Digital signal processor)
* Amplifier

While the actuator is very straightforward, this is “just” a speaker; the other two need some explanation.

### 2.1 DAC/DSP

In the digital world, things “exist” only in the digital domain. This digital domain is eventually built up out of 1’s and 0’s.

For some things, we need analogue values, like for our speakers! We cannot tell these speakers a bunch of 1’s and 0’s, they wouldn’t understand at all.

This is where the DAC shines. A DAC is a device which acts as a translator between the digital and analogue domain and makes 0’s and 1’s comprehensible to analogue devices.

#### How does the DAC convert binary to analogue?

Digital audio is stored/viewed differently than analogue music as has been proven. Digital music is often stored as a series of peaks, or amplitudes of the analogue music signal at regular intervals. These intervals are referred to as the sampling rate.

When recording a digital audio file, if we check the analogue source material 20 times per second, then our sample rate is 20 times per second. At the precise point a measurement is requested, the exact amplitude of the read value is stored as a binary number. This cycle continues until the source material has ended.

Naturally, we can deduct from this that the higher the sample rate, the ‘truer’ our measurement is. Since it has been checked against our source more times. While this is true, there is another factor at play which determines how ‘true’ a digital copy of an analogue source is. The format in which the previously mentioned measurements are stored.

These measurements can be stored in all sorts of ways, with all sorts of precision. 8, 16, 24 bits; and so on. These factors determine how precise the analogue value is copied into the digital memory. And with 24 bits, a machine can be a lot more precise than with 8.

Now that we know how a digital audio file is stored, how does our DAC convert this back to an analogue system? Easy! We just revert the steps taken to save one. Our DAC is built in such a way, it can understand the complex file full of 1’s and 0’s and can reconstruct that analogue signal the way it has been saved. This analogue signal can then be used by other devices.

### 2.2 Amplifier

An amplifier is nothing more than the name suggests, a well; amplifier! Though how does it work? And what exactly does it amplify?

From the previous section, we can take away that we will end up with an analogue signal. This signal will be, let’s say, 5V. Good enough for a little buzzer, but by far not enough for a big guitar amplifier for an ACDC concert. So how do we combat this?

We can combat this by “enhancing” our 5V signal, by injecting it with more power (volts); though how? Getting an analogue signal from a digital file was already very hard, why don’t we amplify it immediately? this has a great reason: we can detect amplitude and frequency using hardware.

Amplifiers all have a main ‘valve’, a transistor. This transistor is continuously powered by our amplifier, from the wall socket for example. But when amplifying a signal, it is not only powered by the wall socket, but also by our analogue signal. Since this signal has a specific amplitude and frequency, it will influence the transistor in a variable way.

This signal causes the transistor to rise or lower its resistive value, allowing a variable amount of current past the ‘valve’. In other words, this transistor copies our characteristics from the 5V signal. Influenced by how we set our amplifier, the volume knob, the amount of voltage added to the end signal will vary, though the way it is added will never change.

# Which signal parameters are commonly found in the music industry?

Music consists of many different characteristics, not only rhythms, basslines and melodies are musical characteristics; its technical parameters are also widely considered to be of extreme importance. Though not every piece of hardware has the capability to support this. Below are the most common parameters and what they mean.

### Sample rate

Sample rate refers to the amount of times in a single second a sample is taken. In our case, this is an audio sample. That means that the more samples there are, the more detailed a piece of audio is.

This does not mean however that indefinitely increasing sample rate means higher quality music, there comes a point were a high sample rate is too high; and the samples cannot represent anything new anymore.

This limit, for digital audio, usually sits somewhere around 44.1KHz (KiloHertz). It is somewhat of an industry standard. Increasing this sample rate is only smart to do if the source analogue audio is known to be recorded in a higher sample rate.

### Bit Depth

Bit depth refers to the number of bits saved to each sample, a bit depth of 16 means a 16 bit number is saved for every sample taken. With a sample rate of 44.1KHz, that makes the bitrate: 44100 \* 16 = 705600.

A difference portraited here, which is very important, is the distinction between bitrate; and bit depth. The bit depth is **only** for one sample; while the bitrate refers to the amount of bits over a set period of time, usually in seconds.

The most common bitrates vary from around 96 to 160 Kbps. While CD recorded music sits a lot higher, at 1411Kbps.

The ideal bitrate is hard to determine, since it relies on different parameters of a piece of audio. However, the human ear can notice drop-offs in the quality of audio. Bitrate tends to set our ears off if it falls below approximately 90Kbsp, meaning many services try and stay above this number.

### 3.1 How do technical parameters relate to hardware choices?

In theory, we can indefinitely scale our technical parameters until it fits all our needs. Though, there sadly is a limiting factor; which is hardware.

Many of the parameters rely on how fast a piece of hardware can read, write; or do both at the same time. Inferior hardware would naturally induce a slower processing time; or an inability to even try and output higher sample rate audio.

Not only speed can be a limiting factor, size allocation on the chips is also a common problem when choosing hardware. The number of bits in a DAC can drastically increase the spectrum of music it can output, solely due to the few added bits.

We found earlier what the standard values are for consumer grade digital music, this means we need to find hardware capable of atleast 44.1KHz samplerate; and up to 141100Kbps.

## Conclusion

According to our research, there are more factors at play in a wireless speaker system than just a Bluetooth receiver. It seems like apart from a means of receiving Bluetooth, we additionally need a DAC; the amplifier is optional, since it heavily relies on which actuator will be used.

### DAC

Like beforementioned, the DAC world is complicated. Though for little projects like these, it isn’t. The only real factor at play here is the number of bits desired. Many microcontroller come with built in DAC’s; though little come with great ones. The teensy 4.2 is a microcontroller not immediately used for audio, but ships with an audio board. This audio board is very sophisticated and has a small form factor. It allows a line in and out for audio but also microphone.

The DAC on this audio board is capable of doing 44.2KHz and had a 16 bit register per sample; essentially perfect for our use case.

The downside is that this board doesn’t have native Bluetooth, but we can get around this with the line-in option.

### Wireless receiver

We are going to use a small cheap Bluetooth receiver module for music, it is a no name brand which I purchased many years ago for my own radio. But it works extremely well. We can connect this to the teensy with the line-in feature.

## Sources

<https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/#:~:text=one%20by%20one.-,ESP32%20Active%20Mode,about%20240%20mA%20of%20power>.

<https://medium.com/@iamtecksay/how-fast-can-bluetooth-transfer-data-c9611fd67a6d>

<https://dem.nv.gov/uploadedfiles/demnvgov/content/ncsc/lte-broadbandandpublicsafetyprimer.pdf>

<https://www.quora.com/How-can-Bluetooth-deliver-high-resolution-audio-Bluetooths-maximum-file-transfer-speed-is-2-Mbps-and-the-high-resolution-audio-plays-at-around-9000-Kbps>

<https://www.whathifi.com/advice/dacs-what-is-a-dac-and-do-you-need-one>

<https://www.lifewire.com/stereo-amps-3135080>

<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/dac.html>

<https://www.microchip.com/en-us/product/mcp4725>

<https://www.techtarget.com/iotagenda/tip/Top-12-most-commonly-used-IoT->[protocols-and-standards](https://www.techtarget.com/iotagenda/tip/Top-12-most-commonly-used-IoT-protocols-and-standards)

<https://www.adobe.com/creativecloud/video/discover/audio-bitrate.html>