

# The Voice Recognition Music Player

CS122A: Fall 2018

Karen Escareno

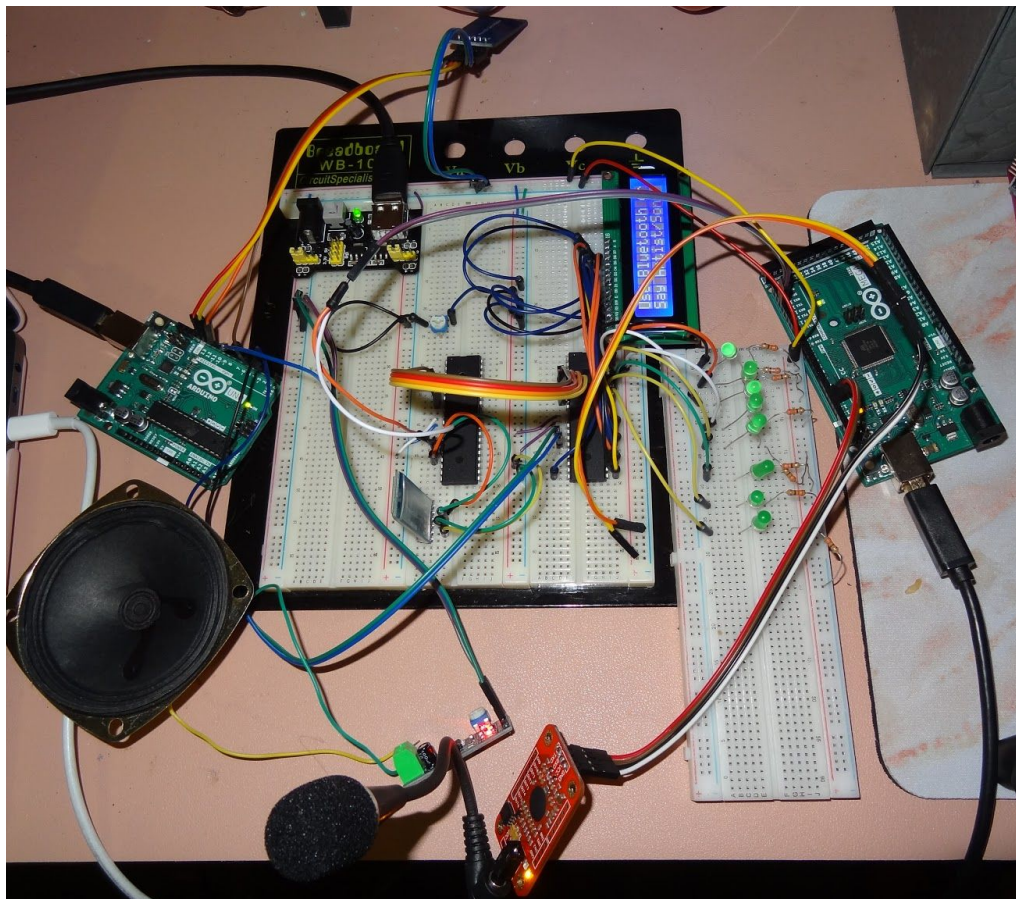
# Table of Contents

<b>Introduction</b>	<b>2</b>
<b>Hardware</b>	<b>3</b>
Parts List	3
Block Diagram	4
Pinout (For each microcontroller/processor)	5
<b>Software</b>	<b>6</b>
<b>Implementation Reflection</b>	<b>7</b>
Milestone	7
Completed components	7
Incomplete components	8
<b>Youtube Links</b>	<b>8</b>
<b>Testing</b>	<b>8</b>
<b>Known Bugs</b>	<b>9</b>
<b>Resume/Curriculum Vitae (CV) Blurb</b>	<b>10</b>
<b>Future work</b>	<b>10</b>
<b>Extra Credit</b>	<b>11</b>
<b>References</b>	<b>11</b>
<b>Appendix</b>	<b>11</b>

# Introduction

Maria the Voice Recognition Music Player is would take in data from the Voice Recognition Module and from the Bluetooth module. It would take in binary values from the Bluetooth module which would then send it to an ATmega and to another ATmega. It would then output the corresponding song to the user. The ATmega would have a specific LED sequence and output the song title and artist on the LCD.

The VRM would listen for artists or song titles and depending on the song or artist would send to an ATmega. The ATmega would send to another ATmega using SPI and then output to the user. The Arduino Uno would be connected to an SD Card Reader that would hold all the songs. Depending on what binary value it gets it will read from the SD card and play the wav file to the corresponding song.



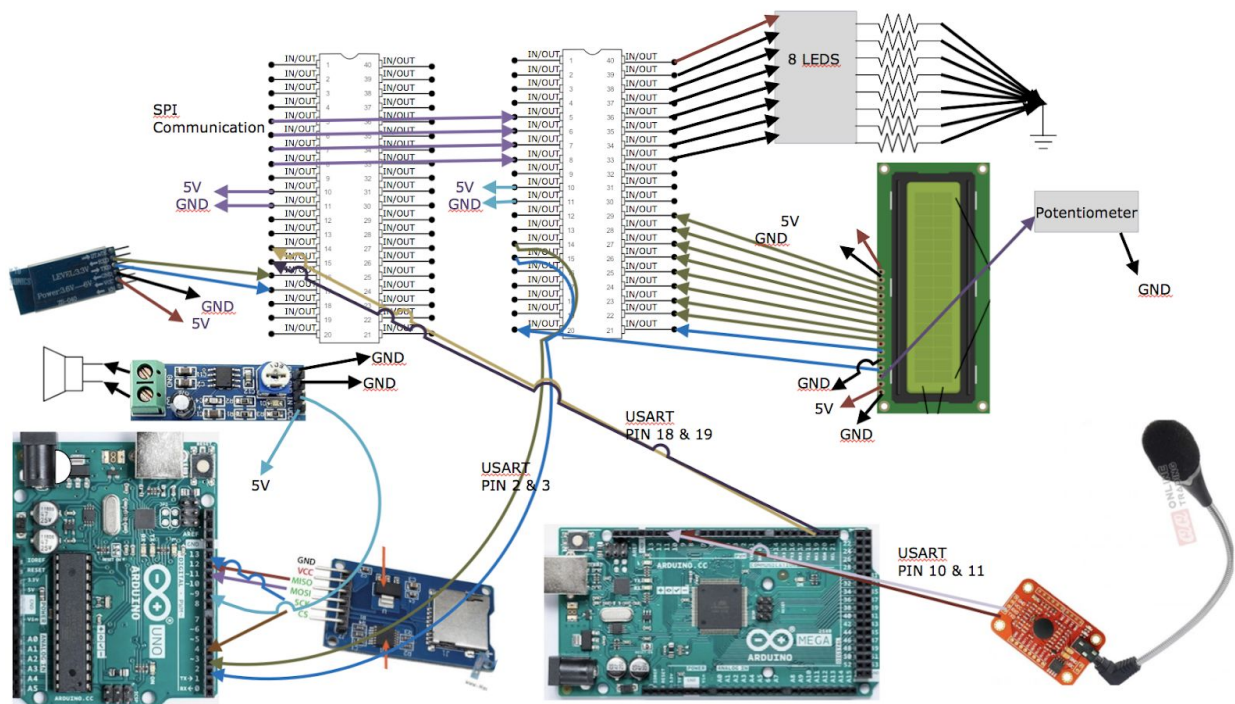
# Hardware

## Parts List

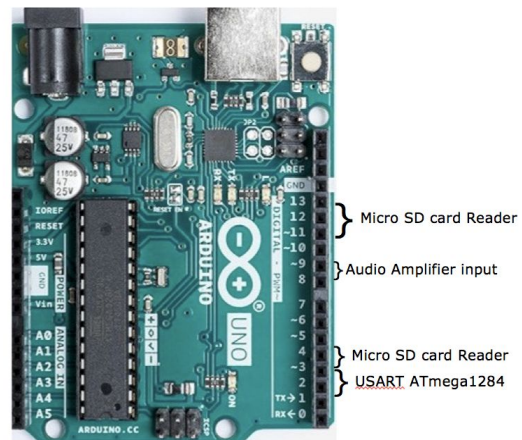
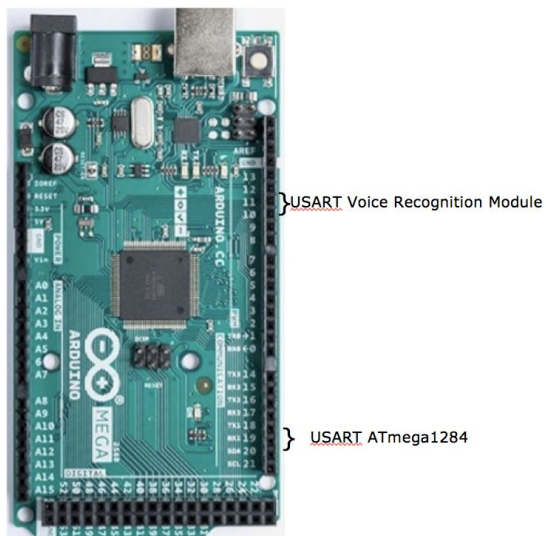
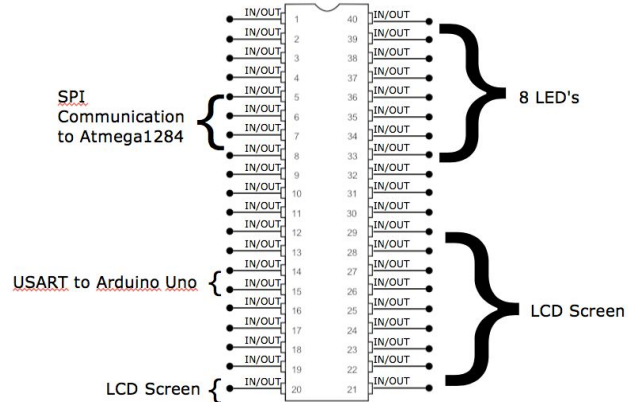
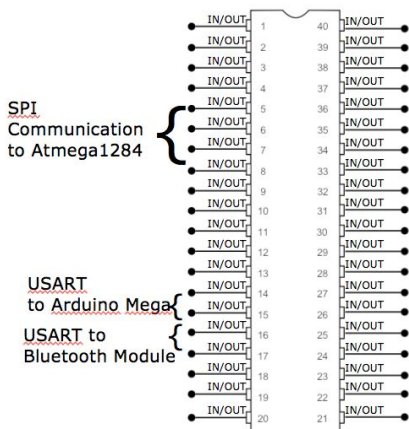
The hardware that was **used** in this project is listed below. The equipment that was not taught in this course has been bolded. *Include part numbers when available.*

Part	Part #	Quantity	Price (optional)
ATMega1284	ATMega1284	2	
LCD screen		1	
<b>Arduino Uno</b>	A000066	1	...
<b>Arduino Mega 2560</b>	A000067	1	
<b>Smakn Speak Recognition Module</b>	EVB746	1	
<b>HC-08 Bluetooth Module</b>		1	
<b>MicroSD card Adapter Reader</b>	JT0777	1	
<b>200x Gain LM386 Audio Amplifier Module</b>		1	
<b>4Ω General Speaker</b>	1314	1	
		Total	

## Block Diagram



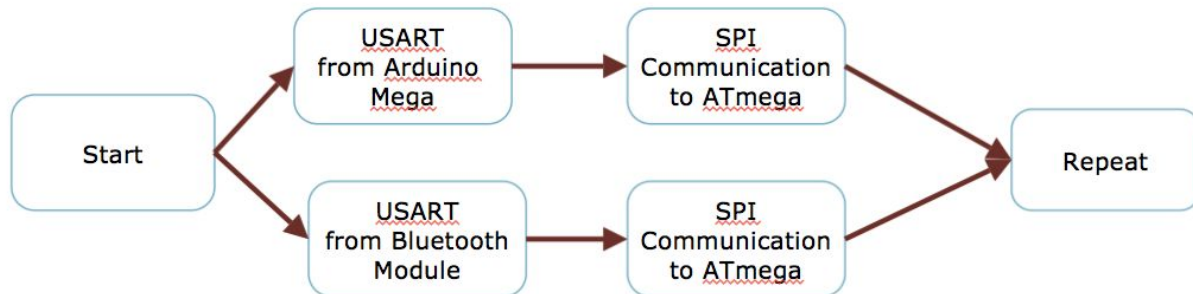
## Pinout (For each microcontroller/processor)





# Software

## First ATmega1284



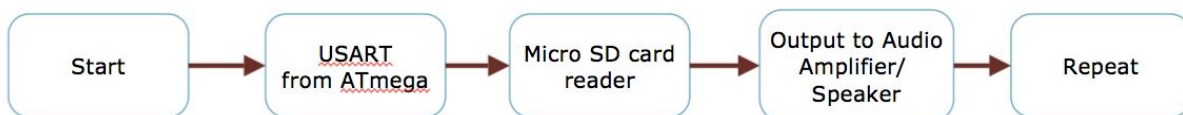
- The ATmega would receive values using USART from the Arduino and the Bluetooth module
- It will then send the values through SPI to another ATmega and keep repeating over and over when it receives a value from the Arduino or Bluetooth module.

## Second ATmega1284



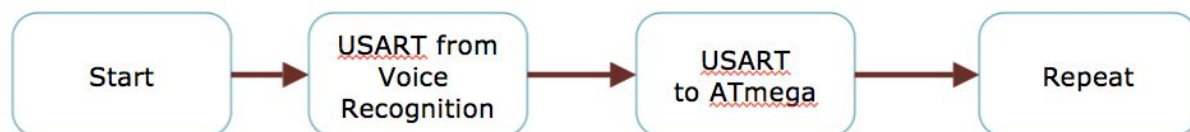
- The ATmega receives info through SPI from another ATmega
- Depending on the binary value it will output a specific Artist and Song Title to the LCD screen and specific 8 LED sequence.
- It will then send the binary value through USART to an Arduino

## Arduino Uno



- The Arduino Uno will receive a binary value from the ATmega and depending on the binary value will play a specific song from the MicroSD card reader.
- It will then output the audio from the Arduino to a speaker and will repeat in case the binary values received changes.

## Arduino Mega



- The Arduino Mega would be connected to the Voice Recognition Module using USART and would receive a binary value.

- That binary value would be sent through USART to an ATmega and would repeat in case it detects a different word from the voice recognition module.

## Implementation Reflection

The project I completed used many of the communications that we used during lab like USART and also SPI. I am very proud of the project I completed because it was hard to establish communication between the Arduino and the ATmegs but it was done. What I would do differently is I would of bought my voice recognition module on amazon a lot sooner. It was a slight setback because it only came a week before the deadline because my other package from Amazon got lost in the mail.

The best part of my project is that it was able to recognize voice commands that would then output the song depending on what title and artist it heard. Another great part was that there was communication between the bluetooth module and an IOS device. It was able to take in binary inputs and output the song it corresponded to.

## Milestone

Milestones:

- Have a button that when pressed it would display to user to say an artist or song title. Have all the 25 songs in SD card.
- Take info from SD card and output to speaker. Bluetooth would turn on led when there's a bluetooth connection.
  - My milestones were not as difficult as they should have been and were easy to finish but I had to change one a few weeks before because the speech recognition module got lost in the mail and never arrived.
  - My milestone goal was not ambitious enough and should have been harder because it could have been done in a day and in a way set me behind because they weren't challenging milestones.
  - I feel like I could have gone further in implementing the button to the voice recognition module if my milestones would of been more complex.

## Completed components

I was able to have a voice recognition module that would detect song titles or artist names and output to the user. I was also able to have each song have an LED sequence and display the song title and artist on an LCD. The bluetooth module was made more complex in that it would take binary values from the user and then output the corresponding song to the user. Before it was just suppose to show when it was connected to a phone via bluetooth.



## Incomplete components

### Button to activate Microphone

I was not able to implement a button that would turn on the mic so the user would speak into and listen for a song title or artist. I ran into the difficulty that it would have the voice recognition module stop working altogether.

- I was in the process of testing it but since it made the voice recognition module stop working it was something that wasn't really needed.
- It would probably take a week to complete it because the usual way it read a button in arduino did not work so I would have to find another way that would work with the voice recognition module.
- The issue was the voice recognition module not liking when i added the code for the button and just stopped working. The issues were with another component that this was dependent on.

## Youtube Links

### Make sure they are publicly viewable!

- Short video:
  - <https://youtu.be/uzPAZZUN-vY>
- Longer video:
  - <https://youtu.be/0ux4GrDUFBo>

## Testing

- ATmega1284
  - To test this part of my project I would use LED's to see what binary output it would get from either the USART or the SPI, so the values wouldn't change when going from one to another.
  - The test cases I used were that I would output the binary values going from one ATmega1284 to another to make sure that the values did not change. I would put 8 LED's on the first ATmega to see the input and the same on the second ATmega to see what it received.
- Arduino Uno/ Micro SD card Reader
  - To test the Arduino I tested to make sure that it was able to read from the SD card. It would output to the user if it was able to read from the SD card or not.
  - The test case I used is that I implemented code in the Arduino to see if it could successfully read from the Micro SD card reader and the serial monitor would either output "SD Success or SD fail". It would help see if everything was connected right and if it was able to read from the SD card.

- **Arduino Mega**

- To test that the arduino was getting the right values from the Voice recognition module, I sent the value through USART to an ATmega and then outputted that values to 8 LED's to see that value was correct.
- The test cases I used was using USART and then outputting binary value to 8 LED's. If the binary value from the VCM was 0x01 then the LED's would output 0x01.

- **Voice Recognition Module**

- To test that it would take in different people's voices I used different tones of voice and other people with different accents to see if it would output anything.
- The test cases I used was that I had different subjects and myself with different types of voices test that it would still detect the same either song title or artist. It was able to detect a few but with people with accents it was more difficult because the way the pronounced the words was different than when they were trained.

- **Bluetooth Module**

- To test the bluetooth module I was able to output what binary value it received from the bluetooth connection to the 8 LED's which would show if the values the ATmega received were correct.
- The test cases I used was that if i sent the bluetooth module 0x03 then the bluetooth module would send the ATmega through USART 0x03 and the LED's would display the 0x03 to show that it was it sending the right data.

- **Audio Amplifier/Speaker**

- To test this part of the project I would send in different binary values to the arduino and see what it would output to see if the speaker was working.
- The test cases I did was I would send in binary values of 0x01. If it was working correctly it would play Bad by Michael Jackson which would mean the communication between the SD card and arduino to audio amplifier is correct.

## Known Bugs

### **Power Source still output when turned off.**

- What was causing it was the cable from the Power source to the computer was not working so it would still turn on the breadboard even the power source was turned off.

- What I did to help this power issue was change the wires to the power source which helped turn off the power source when turned off.

#### **ATmega would read the wrong values from SPI.**

- What was causing this is still unknown because it just randomly started happening during demo day, but it may have been due to the power source not turning off completely or having a communication error.
- To debug this issue I reuploaded the code from the ATmegs over again and it solved the problem.

#### **Voice Recognition module would turn off when moved or when connected to voltage on Arduino.**

- What was causing this was that the arduinos voltage was not very stable because when the VRM would run the voltage would vary and not stay at 5V.
- What I did to debug this was that I connected it to power source on the breadboard because it was always constant and would not change like the arduino.

## Resume/Curriculum Vitae (CV) Blurb

The skills that I learned from this project is how to implement the communication between an Arduino Uno/Mega with an ATmega1284. I was able to use SPI between an SD card reader and an arduino. I was also able to use USART and SPI communication between two ATmegs. Also, USART communication between a bluetooth module/ ATmega and VRM/Arduino. I was able to use my debugging skills by using output LED's to see if the values received were correct.

## Future work

- The next feature I would add would be for the voice recognition module to look for more than 7 words at a time but instead all the 80 words.
- I would proceed by looking at the library of the voice control module look through the code to be able to see where it only load 7 words at a time and extend it to 80 words at a time.
- To design a case I would have a black matte box exterior which would have everything within the box. The only things that would be seen are the LCD and the LED sequences.
- I could try to create a PCB but due to the Arduino it would be pretty tough to do but it could also be replaced with other parts that do the same job.

I will be using it to apply to Amazon because it is very similar to the Amazon echo in that it takes artist and song titles and plays those songs.

## Extra Credit

Include a link to a public DIY for your project:

<https://www.instructables.com/id/CS122A-Voice-Recognition-Music-Player/>

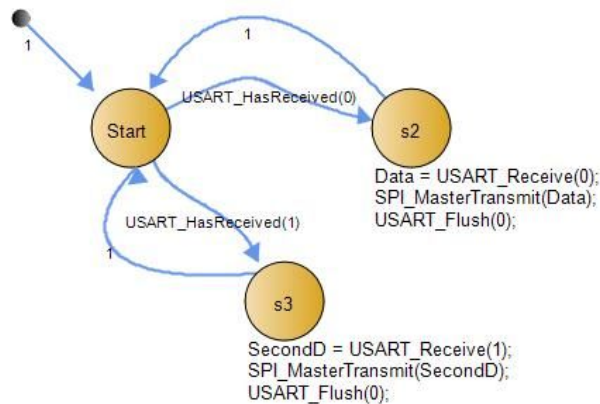
## References

If you used code or ideas from another source (including inspiration from a video or another product) please list those sources here. Include the license for the code that you used or referenced.

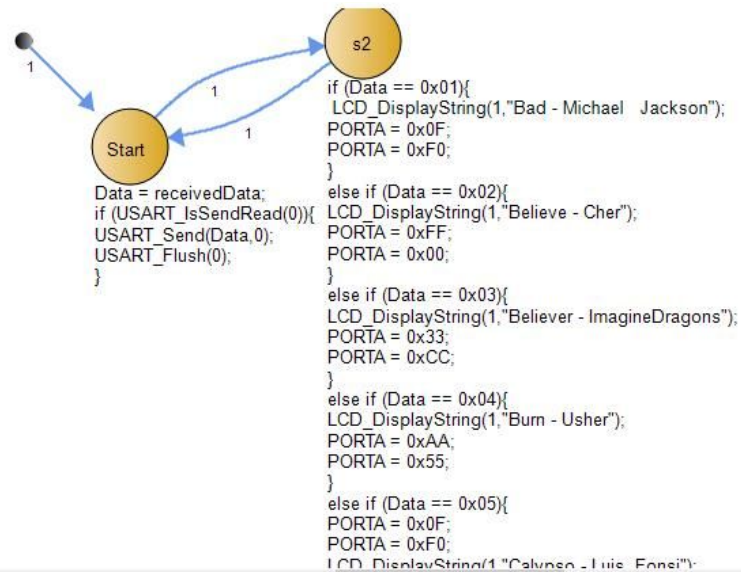
- <https://maxoffsky.com/maxoffsky-blog/how-to-play-wav-audio-files-with-arduino-uno-and-microsd-card/>
- <https://vimeo.com/94925137>
- <https://www.instructables.com/id/Audio-Player-Using-Arduino-With-Micro-SD-Card/>

## Appendix

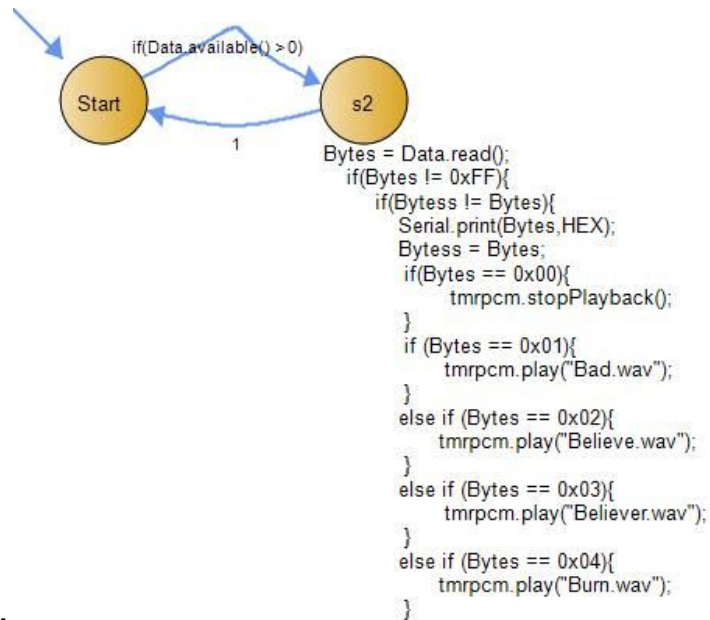
Include images of all of the SM's that you have designed and any other work that you think is relevant to this project.



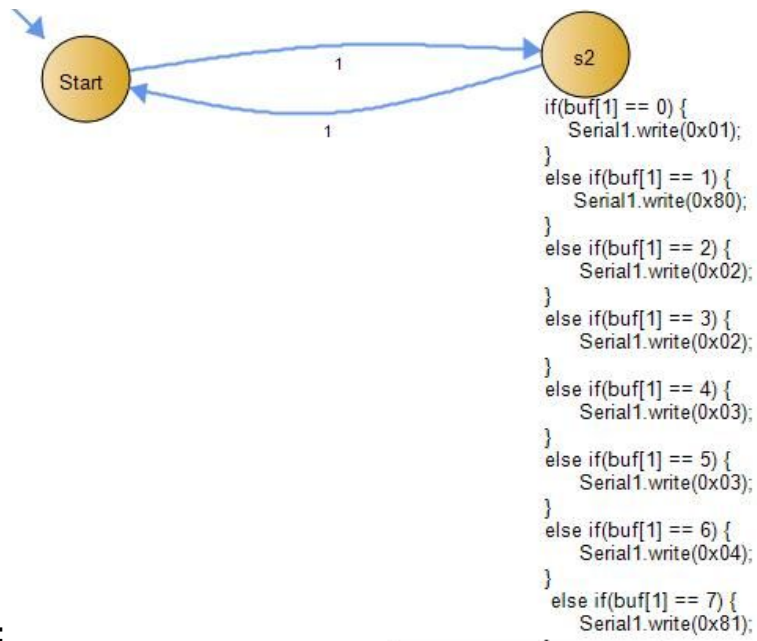
**First ATmega1284:**



**Second ATmega1284:**



**Arduino UNO:**



**Arduino MEGA:**