JOINT INSTITUTE
交大密西根学院

UM–SJTU JOINT INSTITUTE

BIOMEDICAL INSTRUMENTATION AND DESIGN LABORATORY (VE458)

POST-LABORATORY REPORT

FINAL PROJECT
ALTERNATIVE COMMUNICATION WITH SINGLE-MUSCLE EMG

group
7

Group 7

Lu Xiuneng	ID: 5140109100
Wang Yuxuan	ID: 5143709064
Lin Junwei	ID: 5143709177
Tommaso Cemmi	ID: 715370990063

Date: 9 December 2017

Instructors
Prof. Jiajia Luo

Contents

1 Introduction & Background

ALS, also known as amyotrophic lateral sclerosis, is a disease causes the death of neurons which control voluntary muscle. In the situation, about 30% of people will lose their ability to communicate normally, requiring the use of technology to compensate. The goal of this project is to develop an alternative communication system based on single-muscle EMG signal input. The system should be cheap, and should allow patients affected by severe forms of ALS to communicate, even when they can only control a single muscle in their body.

2 System design

The system flow chart is summarized in the following diagram. The EMG signal is recorded and conditioned through hardware and software part; the software logic is carried out with LabVIEW and it results in a message shown on the computer screen.

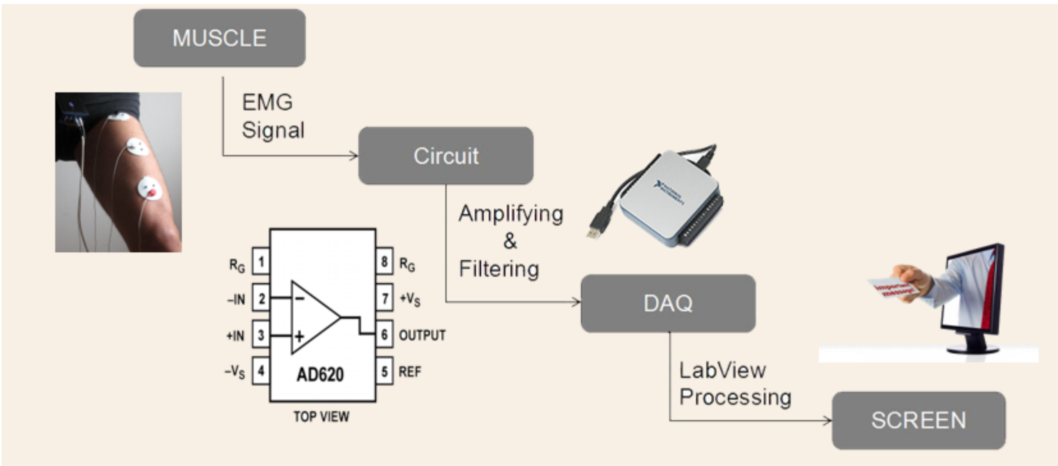


Figure 1: Flow Chart of the system

2.1 Hardware Part

The hardware is composed by a differential amplifier and by a band-pass filter, resulting in an overall gain of about 400 and cut-off frequencies of 50-450Hz.

2.1.1 Components

The hardware part makes use of several resistors and capacitors as well as of a AD620 differential amplifier for amplification and a LM741 operating amplifier for filtering. Details about the hardware components can be found in Appendix A.

2.1.2 Function

The function of the hardware part is both to amplify the EMG input signal, limiting common mode noise and to filter frequencies out of the desired range. Further details regarding amplifier and filter design can be found in Appendix A.

2.2 Software Part

The software part of the system takes care of post-acquisition signal processing and includes the software logic that allows communication via single-muscle EMG signal.

2.2.1 DAQ

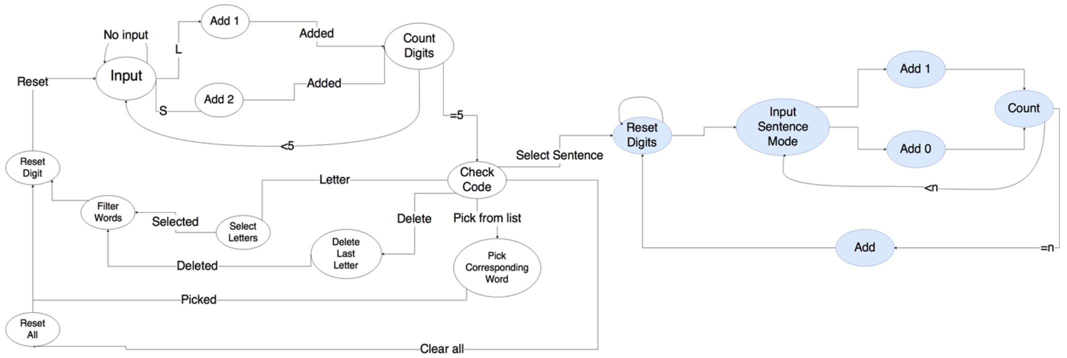
The input EMG signal is processed by the DAQ VI, which differentiates into short and long inputs based on the length of the muscle contraction. Detailed logic of this VI can be found in Appendix B.

2.2.2 Signal encoding

Short EMG input signals are encoded as zeros, while long signals are encoded as ones. By creating a sequence of 5 digits the patient is able to choose among 32 options. While most of these options represent letters of the English alphabet, some offer additional functionalities.

2.2.3 FSM

The Finite State Machine (FSM) is the core of the software logic, that allows to convert sequences of binary digits in commands and eventually in messages to be shown on the screen. The following diagram shows a breakdown of the FSM states. Detailed explanation of the FSM can be found in Appendix B.



3 Budget

The following table shows an overview of the system components and cost. The total cost of the system is estimated to be less than 5 USD (excluding NI DAQ board). The goal of creating a cheap system has been achieved. As a comparison, the cost for a customized communication system for patients affected by ALS is in the range of 1M USD. The single-muscle EMG system developed represents a valid and cheap alternative to expensive commercial systems, for situations where price and portability are the priority.

Components	Quantity	Cost per unit (CNY)	Cost (CNY)
AD620 Amp	1	7.8	7.80
LM741 Amp	1	2.5	2.50
Wires	1.5m	1.18/m	1.77
Capacitors	2	0.75	1.50
Resistors	4	0.44	1.76
DAQ board	1	3048	School Provided
Total			15.33

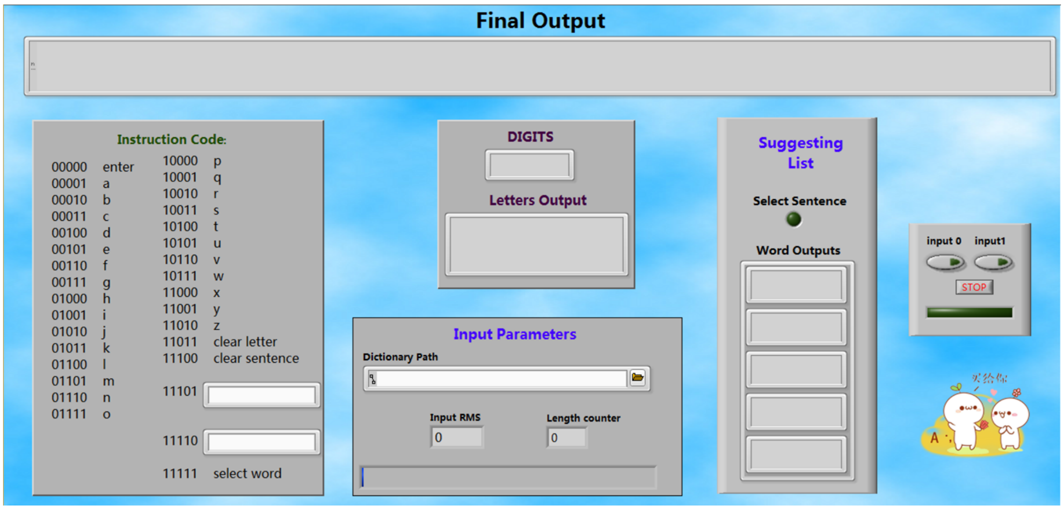
Figure 2: Components and Cost

4 Results

The system developed fulfills all the requirements set at the beginning of the project.

4.1 Interface of our design

The following figure shows an overview of the system front panel. Detailed explanation of the controls and indicators can be found in Appendix B.



4.2 System setup

The image below shows a picture of the system design setup, including hardware, DAQ and laptop.

Figure 3: System Setup

4.3 Functions

Besides allowing input of single letters, the system also allows to delete wrong inputs, shows sentences on the screen, suggests words that can be selected, and picks sentences from a user-defined list, all by using only a single-input EMG signal.

5 Suggestions & Conclusions

Several improvements are possible. Some of these are:

- Use more sensitive filters
- Enlarge the dictionary
- Optimize the word searching algorithm
- Add a new modes e.g. to enter sentences based on the initial letters of the sentence

6 Teamwork distribution

The teamwork has been distributed evenly among the members of the team:

- Lu Xiuneng: Hardware design
- Wang Yuxuan, Lin Junwei: Software Design and FSM
- Tommaso Cemmi: Systems and DAQ design

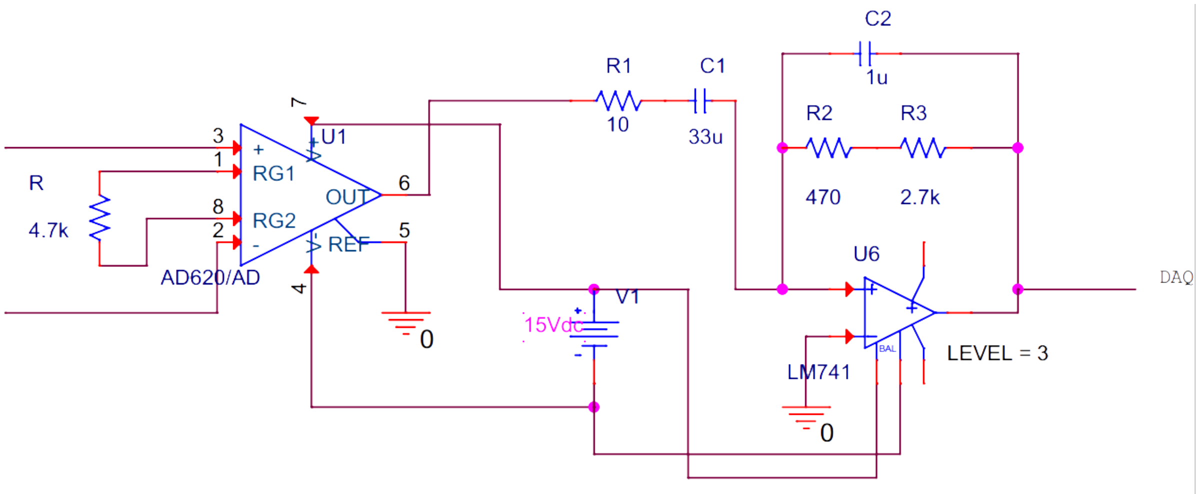
A Hardware Design

For the hardware part, we want the gain to be approximately 400 so that the RMS value of our EMG signal voltage would have appropriate value. In addition, we want the cutoff frequency of the filter to be 50Hz and 450Hz to filter as much noise as possible and at the same time reserve the most signal.

To satisfy the requirement, we carefully choose the appropriate resister size and capacitor size which is listed in the schematics part.

A.1 Schematics of our hardware Design

Here is the schematics of our design:



A.2 Testing Data & Result of our Hardware

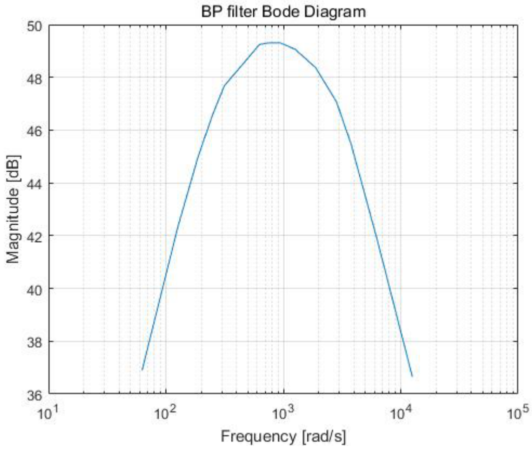
A.2.1 Testing Data

To get the actual gain and cutoff frequency of our hardware design, we have done a lot of testings and below are the testing data(we set the $V_{in} = 50mVpp$):

Frequency (Hz)	Vout (V)	Gain
10	3.50	70.0
20	6.51	130.2
30	8.9	178
40	10.7	214
50	12.1	242
100	14.5	290
120	14.6	292
150	14.6	292
200	14.2	284
300	13.1	262
450	11.3	226
600	9.4	188
1000	6.2	124
2000	3.4	68

A.2.2 Bode plot

By Using the data we measured above, we Matlab to draw the bode plot:



A.2.3 Gain & Cutoff frequency

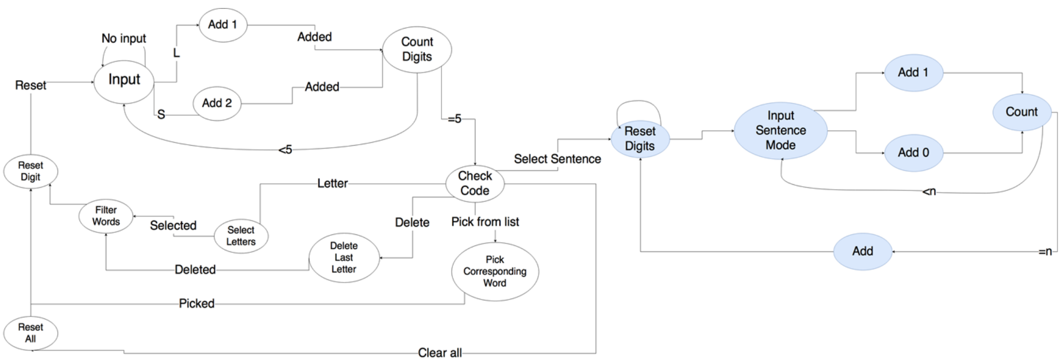
By referring to the Bode plot we draw, we can immediately get the actual gain of our design is 290 and the actual cutoff frequency is 35Hz and 480Hz, which is satisfying.

B Software Design

B.1 Data Acquisition

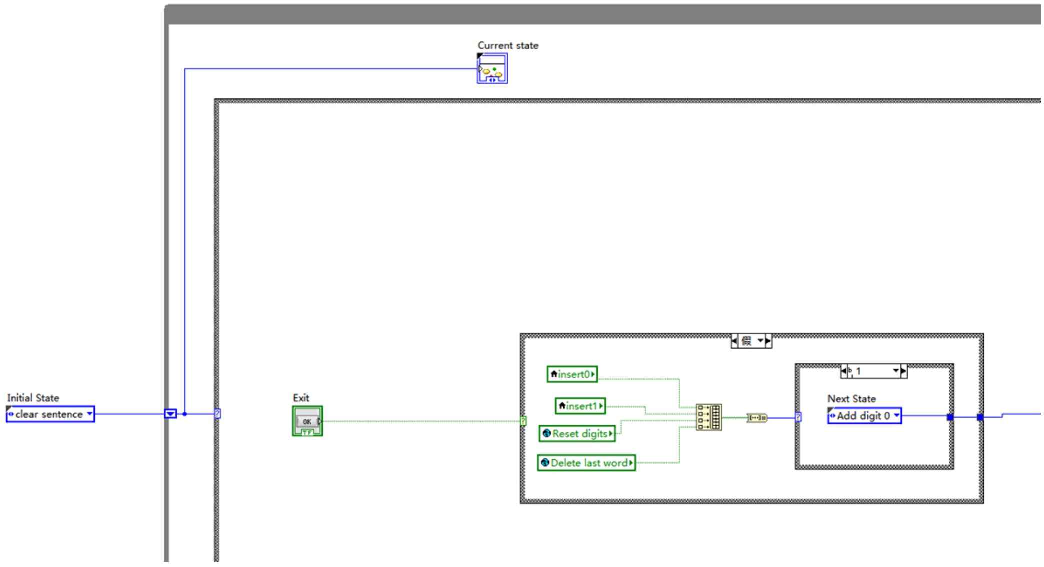
B.2 LabVIEW Code for FSM

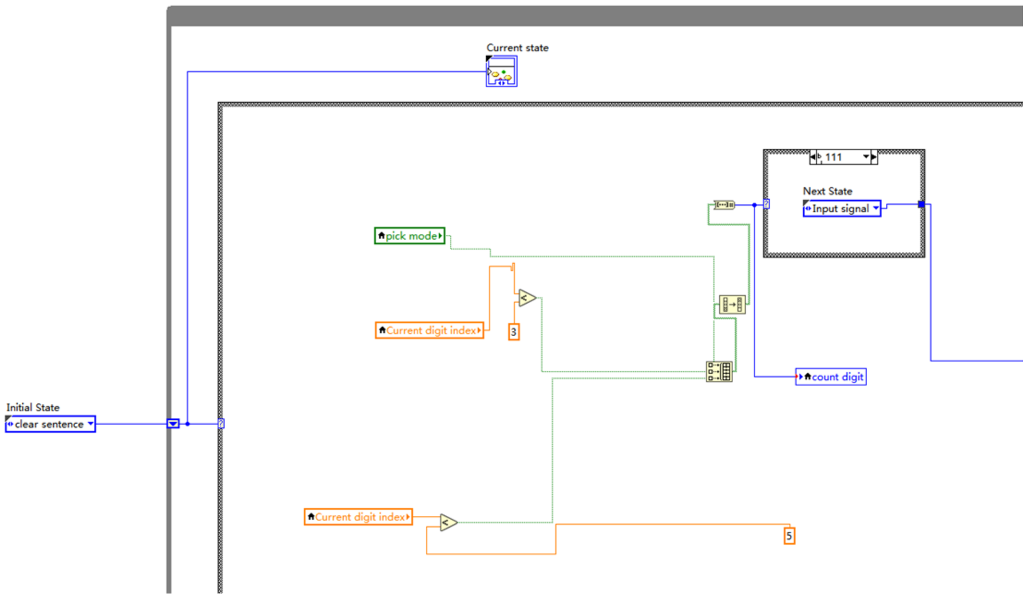
For the detailed core coding part, to realize the input method, we choose to use the state machine function in Labview. The FSM figure is shown below:



B.3 Input Signal

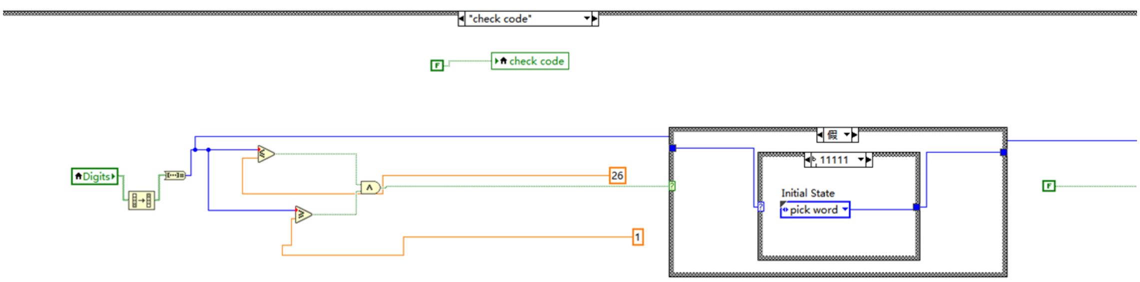
The main idea is that we translate the input boolean digits into binary number and use a state called ”count digits” to count digits until 5. There is one exception is that when the system enter pick mode, the count of digits becomes 3. So I use a truth table to tidy up the logic.





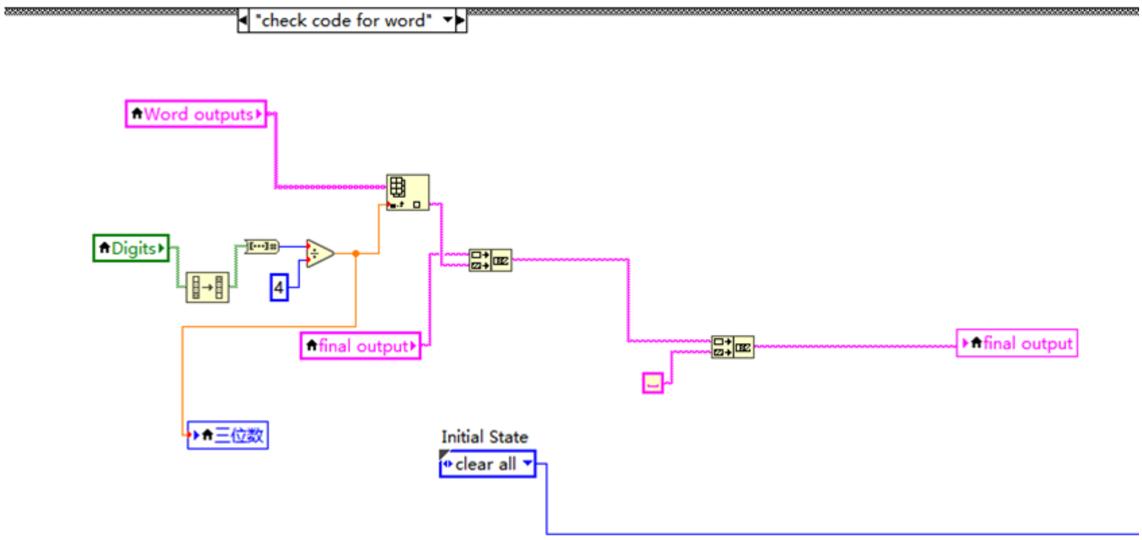
B.4 check code

Then, the 5-digit binary number is translated into 32 cases. Among them, for example: ”00000”: put contents in the letter output into final output, ”1 26”: output ”A Z” into the letter output.



B.4.1 check code for word output

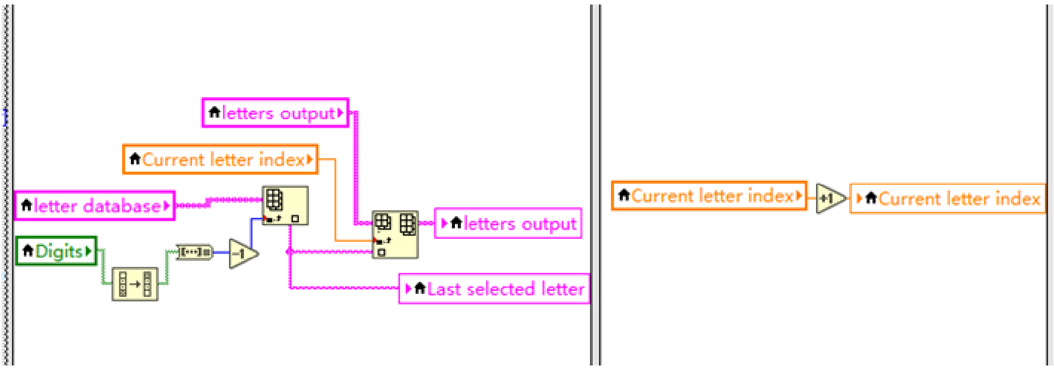
When use the three-fifth of digit as the selection signal to select words in the words output into final output, we need to shift left 2 digits (divide 4) for the digit.



B.5 select letter

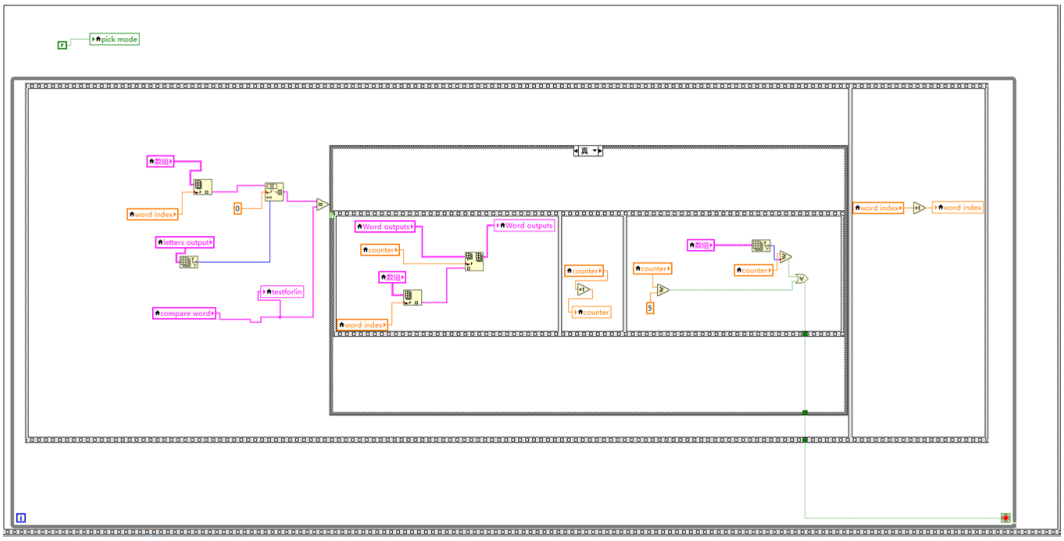
The idea is that we first build a constant string array from "A to Z", Then we use the function "index of array" to find to correct letter and output it in the array.

Also to show it in the string type for better user experience



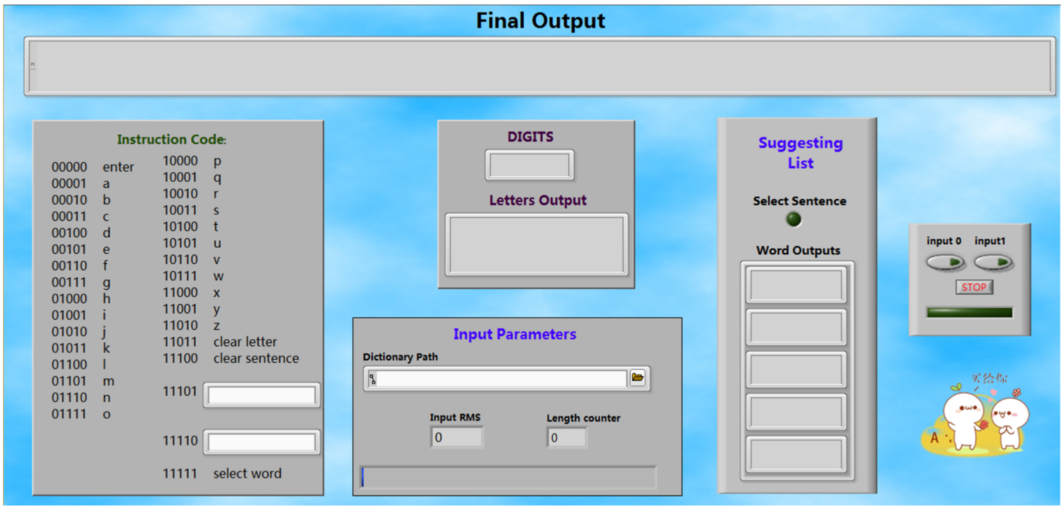
B.6 word filter

First, Labview reads an existing words dictionary and save it into arrays. The idea is that when we input a letter, we compare the letter with the first letter in the dictionary and show 5 choices in word output. Users can then enter "11111" into pick mode and enter "000 100" to select the 5 choices in the list. When the user enter another letter, the comparing letter string would also be updated so that the word output could be also updated.



B.7 Final user interface

We build two front panels. The previous one is for test purpose and the other for the real interface. And we can clear see the instructions helper on the left part.



C Mablab Code

1. Matlab code for calculate the magnitude of each component:

```
fL= 0.05; %low curoff frequency
fH= 125; %high cutoff frequency
c1= 33*10^(-6); %we have found this kind of capacitors in our lab
c2= 1*10^(-6);
TotalGain = 1000; %1000 would be a suitable gain in this lab

%Other component of filter:
r1= 1/(2*pi*c1*fH)
r2= 1/(2*pi*c2*fL)
gain_of_filter= c1*pi*(1/(2*pi*c2*fL))/(0.5*(fH^(-1)+fL^(-1))+
sqrt((fH*fL)^(-1)))

```

```
%For filter part, since in the lab, the kinds of capacitors are
%limited but the range of
%resisters can be very large, It will be easier if we just get the
%magnitude of capacitors first, and calculate the magnitude of
%resistors.Having known the gain of filter and the total gain, we
%are able to calculate the load resistor of AD620:

% load resistor of AD620
gain_of_diff_amp = TotalGain/(c1*pi*(1/(2*pi*c2*fL))/(0.5*(fH^(-1)+
fL^(-1))+sqrt((fH*fL)^(-1))))
LoadResistor= 49400/(TotalGain/(c1*pi*(1/(2*pi*c2*fL))/(0.5*(fH^(-1)+
fL^(-1))+sqrt((fH*fL)^(-1))))-1)
```

2. Matlab code for Theoritical value of the Bode Plot of bandpass filter

```
c1=33*10.^(-6); %c1 and c2 should be equal to get the unit gain
c2=1*10.^(-6);
r1=38; %r1 should be small enough to get unit gain
r2=3180000;
a1=tf([r2*c1 0],[r1*r2*c1*c2 (r1*c1+r2*c2) 1]);
bode(a1);
grid on;
fLPcutoff=1./(2*pi*r2*c2)
fHPcutoff=1./(2*pi*r1*c1)
Gain=pi*r2*c1/(0.5*(fLPcutoff^(-1)+fHPcutoff^(-1))+sqrt(1/
(fLPcutoff*fHPcutoff)))

%after process 1, we get the roughly magnitude of resistor, then
%we need to round off it to some integer so that we can use fewer
%component to get the approcimate value. And we plug in those value
%here, to get the theoritical bode plot to verity its correctness.
```

3. Matlab Code for actual figure of bode plot of bandpass filter

```
c1=0; %c1 and c2 should be equal to get the unit gain
c2=1*10.^(-6);
r1=2000; %r1 should be small enough to get unit gain
r2=318000;
a1=tf([r2],[0 c2*r2*r1 r1]);
bode(a1);
grid on;
fLPcutoff=1./(2*pi*r2*c2)
fHPcutoff=1./(2*pi*r1*c1)
Gain=pi*r2*c1/(0.5*(fLPcutoff^(-1)+fHPcutoff^(-1))+sqrt(1/(fLPcutoff*fHPcutoff)))
```

D Reference

1. http://vibrationtrainingdevice.com/VG-Evolution/emg-elettrodi-b.jpg
2. http://sine.ni.com/images/products/us/04231404_m.jpg
3. https://s3.amazonaws.com/siteninja/multitenant/images/6912/images/slide/Screen-hand-540x300_1_.jpg_1266619279?1340887223

4. http://baijiahao.baidu.com/s?id=1563218123737396&wfr=spider&for=pc