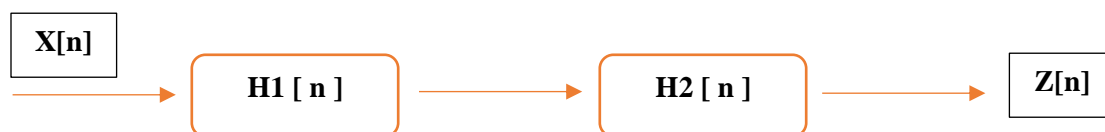


قسمت دوم

با توجه به تمرین ششم، فرمنت های اول، دوم و سوم را در اختیار داریم. در این قسمت سعی داریم با دادن ورودی سیگنال متناوب و سپس استفاده از سیستم اول (در نقش فیلتری)، فیلتر شده ی سیگنال ورودی که معادل فرمنت اول سیگنال است را استخراج کرده و سپس با دادن خروجی سیستم اول به سیستم دوم و استفاده از فرمنت دوم برای سیستم دوم، صداهای نظیر واکه های /e/ و /u/ را تولید نماییم. در ابتدا بر روی شکل توضیحات مربوط به پاسخ فرکانسی فیلتر IIR درجه دو را داده و در ادامه سعی میکنیم سیستمی طراحی کنیم تا خواسته ی سوال را برآورده سازد. در انتها نیز از معایب سیستم طراحی شده سخن خواهیم گفت و راه حل هایی جهت رفع این مشکلات را بیان خواهیم کرد.

شکل زیر سنتز گفتار با استفاده از فرمنت های اول و دوم را نشان می دهد.



در شکل فوق سیستم اول فرمنت اول را میسازد و فرمنت دوم به وسیله ی سیستم دوم ساخته می شود. خروجی حاصل ($Z[n]$)، سیگنالی دارای فرمنت اول و دوم میباشد. برای مدلسازی شکل فوق، همانطور که در صورت سوال ذکر شده است کافی است ورودی سیستم ($X[n]$) را سیگنال متناوب در نظر بگیریم.

با استفاده از توابع آماده ی موجود در برنامه نویسی متلب می توانیم پاسخ فرکانسی سیستم اول و دوم را بدست آوریم و در نهایت با دستور Filter ورودی هر سیستم را با استفاده از پاسخ فرکانسی هر سیستم فیلتر نماییم.

دستور IIRpeak در متلب، کار شبیه سازی فرمنت ها (پاسخ فرکانسی فیلتر IIR درجه دو) را برای ما انجام می دهد. به این صورت که مولفه ی اول آن یک فرکانس معرف فرکانس پیک سیگنال را دریافت می کند که مقدار تیزی پیک سیگنال وابسته به مقدار پهنای باندی است که به آن می دهیم. همانطور که می دانیم پاسخ فرکانسی فیلتر IIR درجه ی دوم دارای یک پیک به اندازه ی فرکانس مشخص و پهنای باند مشخصی است.

همانطور که گفتیم، سیستم اول باید فرمنت اول را بر روی سیگنال ورودی فیلتر نماید. به همین منظور فرکانس F1 بدست آمده از تمرین ششم برای واکه های مدنظر را به جای مولفه ی اول دستور IIRpeak قرار می دهیم و سپس پاسخ بدست آمده را بر روی سیگنال ورودی فیلتر می کنیم.

برای سیستم دوم نیز همین کار را انجام خواهیم داد با این تفاوت که چون سیستم دوم معرف فرمنت دوم میباشد باید فرکانس F2 را به عنوان مولفه ی اول به دستور **IIRpeak** بدهیم. و خروجی این دستور را بر روی خروجی سیستم اول فیلتر کنیم. و درنهایت سیگنال نهایی بدست آمده از سیستم دارای دو پیک منصوب به فرمنت اول و دوم خواهد بود. (کد نوشته شده در انتها آورده شده است)

سوالی که ممکن است پیش بیاید این است که آیا سیگنال نهایی بدست آمده به وضوح واکه های مورد نظر را نشان می دهد یا خیر؟!، برای پاسخ به این سوال باید فاکتورهای مختلفی را درنظر بگیریم. اولین فاکتور و مهم ترین فاکتور انتخاب نوع سیگنال متناوب ورودی است که برای این تمرین سیگنال های متناوب Sin، Cos و مثلثی امتحان شده اند. دومین فاکتور انتخاب نوع پارامترهایی است که توسط کاربر تعیین و مشخص می شوند مانند دوره تناوب سیگنال ورودی و غیره. و اما آخرین فاکتور که بسیار بدیهی است، مشخص کردن تعداد فورمنت های مدنظر برای تولید گفتار است. در این تمرین فرمنت های اول و دوم خواسته شده اند ولی به تجربه می توان این را متوجه شد که هرچه تعداد فورمنت های مورد استفاده را افزایش دهیم کیفیت و دقت صدای نهایی شباهت بیشتری با صداها ی واقعی دارد. در این تمرین ما از سه فورمنت F0, F1, F2 استفاده کرده ایم که به وضوح دقت و کیفیت بالاتری نسبت به تعداد کمتر فورمنت ها دارد. (لازم به ذکر است که به تعداد فورمنت ها سیستم داریم در نتیجه اگر از سه فورمنت استفاده می کنیم باید در شکل سه سیستم را درنظر داشته باشیم که به صورت سری، خروجی هر سیستم ورودی سیستم بعد خواهد بود).

کد نوشته شده

```
% Hello my name is Hossein Simchi, 98443119
```

```
Fs = 16000;  
F0 = 122.8;  
F1 = 431;  
F2 = 1624.8;
```

```
bw = 0.05; >>> پهنای باند
```

```
T = 20*(1/15);  
fs = 16000;  
t = 0:1/fs:T-1/fs;
```

```
x = sawtooth(2*pi*250*t); >>> سیگنال متناوب ورودی
```

```
[a0,b0] = iirpeak(F0/8000,bw); >>> سیستم اول  
[a,b] = iirpeak(F1/8000,bw); >>> سیستم دوم
```

```
[c,d] = iirpeak(F2/8000,bw); >>> سیستم سوم  
x1 = filter(a0,b0,x);  
x2 = filter(a,b,x1);  
x3 = filter(c,d,x2);  
  
sound(x3,Fs);  
audiowrite('C:\Users\Lenovo\Desktop\e_Simchi_98443119.wav',  
x3,Fs);
```

از زحمات شما در طول ترم گذشته متشکرم.

حسین سیم‌چی، ۲۹ دی ۱۳۹۹