بسم الله الرحمن الرحیم

گزارش تکلیف دوم کامپیوتری

پردازش سیگنال های دیجیتال

محدثه غفوری ۹۶۳۲۱۳۳

مهیار عنصری ۹۶۳۲۰۹۳

سوال اول:

سیگنال chirp سیگنالی است که فرکانس آن متغیر با زمان است.به طور کلی سیگنال chirp را برحسب صعودی یا نزولی بودن فرکانس آن در گذر زمان به دو دسته up-chirp و down-chirp تقسیم بندی می کنند.از سیگنال chirp در رادارها و لیزرها استفاده می شود.

روابط کلی سیگنال chirp:

$$x(t) = \sin(\phi(t))$$

$$\omega(t) = \frac{d\phi(t)}{dt},\ f(t) = \frac{\omega(t)}{2\pi}$$

$$\gamma(t) = \frac{d^2\phi(t)}{dt^2} = \frac{d\omega(t)}{dt},\ c(t) = \frac{\gamma(t)}{2\pi} = \frac{df}{dt}$$

برای تولید دو نوع Linear Chirp و Exponential Chirp داریم که ما در اینجا به روابط Linear Chirp می پردازیم:

در Linear Chirp فرکانس رابطه خطی با زمان دارد و به صورت زیر تعریف می شود

$$f(t) = ct + f_0$$

$$c = \frac{f_1 - f_0}{T}$$

که در اینجا $f0$ فرکانس اولیه ما است و $f1$ فرکانس نهایی ما می باشد و c نرخ تغییرات فرکانس با زمان است و T زمان رسیدن از $f0$ به $f1$ می باشد.

برای به دست آوردن Φ در سیگنال Linear Chirp از روابط زیر استفاده می کنیم:

$$\begin{aligned}\phi(t) &= \phi_0 + 2\pi \int_0^t f(\tau)\, d\tau \\ &= \phi_0 + 2\pi \int_0^t (c\tau + f_0)\ d\tau \\ &= \phi_0 + 2\pi \left(\frac{c}{2}t^2 + f_0 t\right)\end{aligned}$$

و $\Phi 0$ فاز اولیه سیگنال در لحظه صفر است.پس سیگنال نهایی به فرم زیر می شود:

$$x(t) = \sin\left[\phi_0 + 2\pi \left(\frac{c}{2}t^2 + f_0 t\right)\right]$$

سوال دوم:

با توجه به خواسته های سوال که زمان برابر ۱۶ ثانیه باشد و فرکانس از ۰ تا ۴۰۰۰ برود با توجه به روابط فوق $f0=0$ و $c=250$ می شود و $\Phi 0$ هم که صفر است.پس روابط به صورت زیر می شود:

$f(t)=250t$

$x(t)=\sin(2\pi(125(t^{۲}))$

با توجه به رابطه اول یعنی $f=250t$ انتظار داریم فرکانس با گذشت زمان بیشتر شود و صدای شنیده شده زیرتر باشد.

سوال سوم:

به سیگنال *chirp* که تولید کردیم گوش کردیم و در طول ۱۶ ثانیه اجرای آن به صورت پیوسته صدا زیرتر می شد که با انتظار ما مطابقت داشت.در کد با قرار دادن یک حلقه *for* با گام های به اندازه دوره تناوب نمونه برداری(یعنی نمونه برداری را در مضارب دوره تناوب نمونه برداری انجام می دهیم) مقادیر سیگنال *chirp* را در یک بردار که طول آن را به اندازه زمان×فرکانس نمونه برداری(که اینجا برابر ۴۸۰۰۰۰ می شود) تعریف کرده ایم ذخیره کردیم و سپس با دستور *sound* که پارامتر دوم آن را همین فرکانس یعنی ۳۰۰۰۰ قرار داده ایم گوش کردیم.

سوال چهارم:

پیش از گوش کردن به سیگنال با نرخ های مختلف انتظار داریم چون بازه فرکانسی ما در این سوال و در این سیگنال از ۰ تا ۴۰۰۰ هرتز تغییر میکند و روی ۴۰۰۰ هرتز ما مولفه فرکانسی داریم با توجه به قضیه نایکوییست اگر فرکانس نمونه برداری ما از ۸۰۰۰ هرتز بیشتر باشد سیگنال نمونه برداری شده دچار پدیده الیاسینگ نمی شود و به عبارتی سیگنال اصلی قابل بازسازی است.

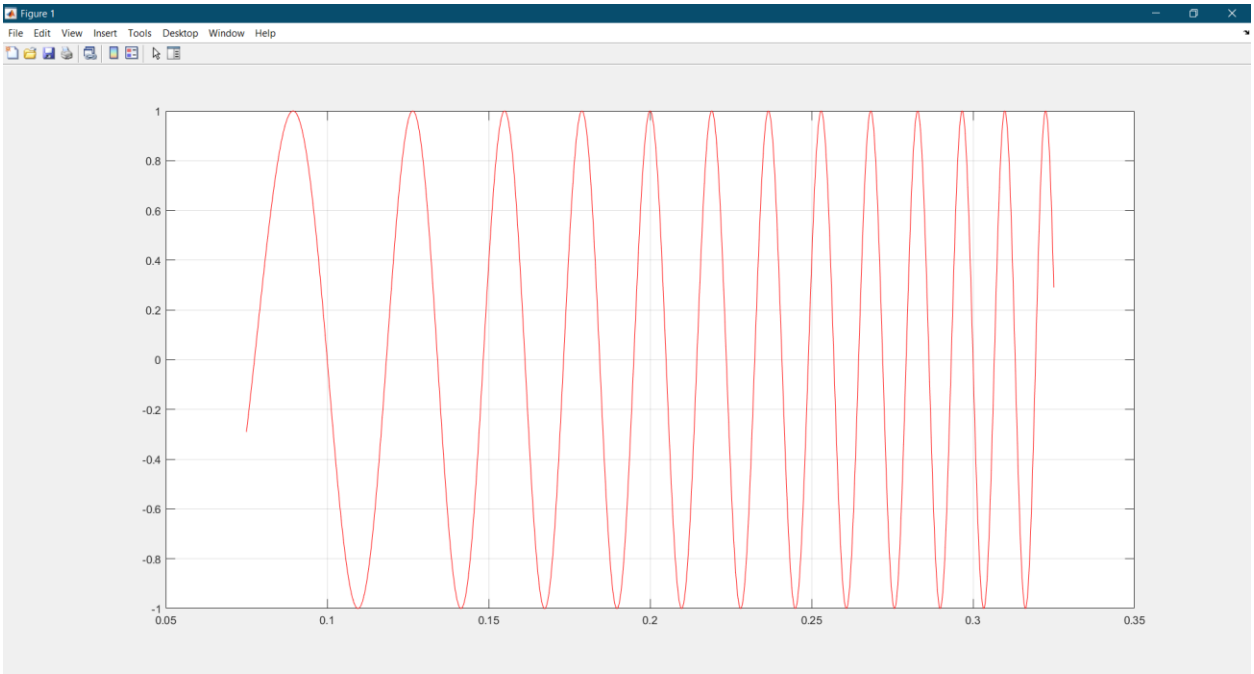
پس از پخش صدا با فرکانس های ذکر شده در صورت سوال به درستی فرض خود پی بردیم و تا فرکانس های بزرگ تر از ۸۰۰۰ هرتز شاهد پدیده الیاسینگ نبودیم و مطابق انتظار با توجه به رابطه خطی فرکانس و زمان با گذشت زمان و افزایش فرکانس صدا زیرتر شد.

اما هنگامی که فرکانس نمونه برداری را از ۸۰۰۰ هرتز کمتر کردیم در اثر مخلوط شدن مولفه های فرکانسی صدای پخش شده دچار تکرار شد که نشان دهنده پدیده الیاسینگ بود و به طور واضح درفرکانس نمونه برداری ۴۰۰۰ هرتز دو بار صوت شنیده شده تکرار می شود و در فرکانس ۲۰۰۰ هرتز چهاربار تکرار می شود و سیگنال به دلیل رخ دادن پدیده الیاسینگ و مخلوط شدن مولفه های فرکانسی انگار متناوب شده است.

******در کد ارسال شده روی سامانه برای جلوگیری از تکرار کد فرکانس نمونه برداری ۳۰۰۰۰ هرتز است و برای شنیدن بقیه نمونه ها کافی است پارامتر *Fs* تعریف شده در ابتدای کد را تغییر دهیم.**

سوال پنجم:

ابتدا بعد از تعریف کردن پارامترهای اولیه،یک پارامتر کمکی به نام *n* تعریف میکنیم که با آن در حلقه یک پنجره زمانی متحرک می سازیم.سپس یک حلقه *for* قرار می دهیم و پارامتر آن را *i* می نامیم.با ضرب کردن *i* و *n* نقطه شروع پنجره زمانی را تعیین می کنیم و مثل عکس ضمیمه تمرین که طول پنجره ۰.۲۵ ثانیه بود آن را ۰.۲۵ ثانیه قرار می دهیم.مقدار *i* و *n* را به گونه ای انتخاب کردیم که مانند فیلم ضمیمه تمرین از لحظه صفر تا لحظه ۱.۰۵ را ببینیم.

به تجربه دیدیم که با زیاد کردن گام حلقه *for* شکل نرمی خود را از دست می دهد و با خیلی کوچک قرار دادن آن نیز زمان اجرا طولانی می شود البته نسبت طول پنجره به طول گام نیز مهم است زیرا تعداد نمونه های *t* و در نتیجه تعداد نمونه های سیگنال *chirp* را تعیین می کند و زیاد شدن زمان اجرا به همین دلیل است زیرا برای یک طول پنجره ثابت با کاهش زمان هر گام تعداد نمونه ها زیادتر می شود و رسم آن ها طولانی تر خواهد شد.مقادیر انتخابی در کد ارسالی برای *i* و *n* به صورت تجربی و در راستای صحبت های فوق انتخاب شده است.

سپس سیگنال *chirp* را در بازه *t* که دیدیم متغیر است رسم می کنیم و آن را بر حسب زمان با کمک دستور *drawnow* نمایش می دهیم.برای سهولت دیدن از دستور *grid on* استفاده کردیم که صفحه را مدرج می کند.همجنین از دستور *ylim* برای محدود کردن بازه نمایش محور *y* کمک گرفتیم.در انتها یک *pause* قرار دادیم تا مکث کوچکی حین کشیدن سیگنال داشته باشیم و به تجربه دیدیم که ۰.۲۵ برای پارامتر مکث زمان مناسبی است.