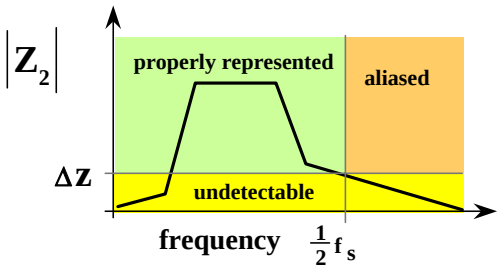
Lecture 17 objectives



- Real-time Data Acquisition System, Time jitter
- Nyquist Theorem
- Performance Measurements

To prevent aliasing => no measurable signal above 0.5f_S.

What: Definition of time-jitter

data acquisition system, every **Δt**

start the analog-to-digital converter, read the result, store in FIFO

Δt=1/f_s

control system, every **Δt**

read the sensors

perform the digital control equations

output to the actuators at a fixed rate

system that generates signals, every **Δt**

output to the digital-to-analog converter (DAC)

```
// background thread, reads samples from ADC
interrupt 12 void TC4handler(void){
  ATD0CTL5 = 0x81;           // start sequence, channel 1
  Time++;
  while((ATD0STAT1&0x01)==0){}; // wait for CCF0
  if(Fifo_Put(ATD0DR0) == 0){   // send to foreground
    DataLost++;
  }
  TC4 = TC4+PERIOD;
  TFLG1 = 0x10;    // acknowledge
}
```

time-jitter, **δt**,

nΔt be the time it is scheduled to be run

t_n the time the task is actually run, `ATD0CTL5 = 0x81;`

$\delta t_n = t_n - n\Delta t$

For a real time system with periodic tasks, we must be able to place an upper bound, **k**, on the time-jitter.

$-k \le \delta t_n \le +k$ for all **n**

For all three of the above examples it is more important to control the time difference between periodic events rather than the absolute time itself. Let Δt_n be the actual time difference between two executions of a software task (e.g., starting the ADC). The desired time difference is $1/f_s$. For this situation, we define the time-jitter at sample **n** to be

$\delta t_n = \Delta t_n - 1/f_s$

Again, we must be able to place an upper bound, **k**, on the time-jitter.

$-k \le \delta t_n \le +k$ for all **n**

V(t) rather at V(t+δt)

δV = δt * dV/dt

Why: Consequences of delayed service

data acquisition systems and control systems

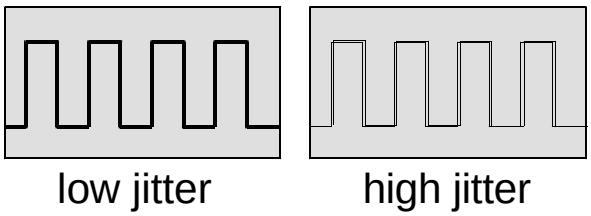
calculated math is incorrect. E.g., derivative

dx/dt = ((x(t)-x(t-Δt))/ Δt

system that generates signals

the sound is distorted

the picture is blurry



How: Measurements of time-jitter

A. **Using a scope**
B. **All the causes of jitter?**
   a. **I=1, other ISR, critical sections removed by I=1**
   b. **Currently executing instruction**

```
interrupt 12 void TC4handler(void){
#if DEBUG
  PTT ^= 0x80;
#endif
  ATD0CTL5 = 0x81;       // start sequence, channel 1
  Time++;
  while((ATD0STAT1&0x01)==0){}; // wait for CCF0
  if(Fifo_Put(ATD0DR0) == 0){   // send to foreground
    DataLost++;
  }
  TC4 = TC4+PERIOD;
  TFLG1 = 0x10;    // acknowledge
}
```

B. Using a TCNT

```
unsigned short MaxTimeDiff;  // max TCNT difference
unsigned short ThisTime;     // TCNT at current sample
unsigned short LastTime;     // TCNT at previous sample
interrupt 12 void TC4handler(void){
#if DEBUG
  ThisTime = TCNT;
#endif
  ATD0CTL5 = 0x81;           // start sequence, channel 1
  Time++;
  while((ATD0STAT1&0x01)==0){}; // wait for CCF0
  if(Fifo_Put(ATD0DR0) == 0){   // send to foreground
    DataLost++;
  }
#if DEBUG
  if(Time > 1){
    MaxTimeDiff = max(ThisTime-LastTime,MaxTimeDiff);
  }
  LastTime = ThisTime;  // setup for next measurement
#endif
  TC4 = TC4+PERIOD;
  TFLG1 = 0x10;    // acknowledge
}
```

Later, dump the results using debugger or output port

```
#if DEBUG
  SCI_OutString("Max Jitter (cycles) = ");
  SCI_OutUDec(MaxTimeDiff-PERIOD);
#endif
```

Add time jitter to Lab 6 solution, build and run on TExaS

What factors limit measurement resolution?

- **Transducer noise**
- **Electrical noise**
- **ADC precision**
- **Software errors**

What factors limit measurement accuracy?

- **Resolution**
- **Calibration**
- **Transducer stability**

$$\text{Average accuracy (with units of X)} = \frac{1}{n}\sum_{i=1}^{n}\left|X_{ti} - X_{mi}\right|$$