Delta Timing Review

CSE 1302

Today's Outline

- Delta timing

- Using SerialComm

Review of Delta Timing

while (true)

→ now = millis()

    if (now >= loopEndTime) then

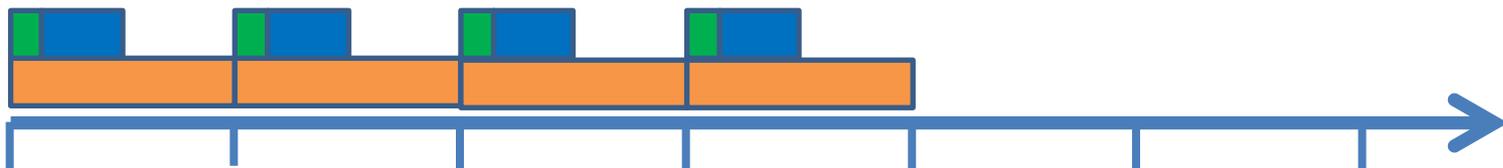   loopEndTime += deltaTime

        do some work

        output results

    end if

end while

What if work takes a while?

```
while (true)
    now = millis()
    if (now >= loopEndTime) then
            loopEndTime += deltaTime
            do some work
    end if
end while
```

Especially if work sometimes takes longer than deltaTime!

What if work takes a while?

while (true)

    now = millis()

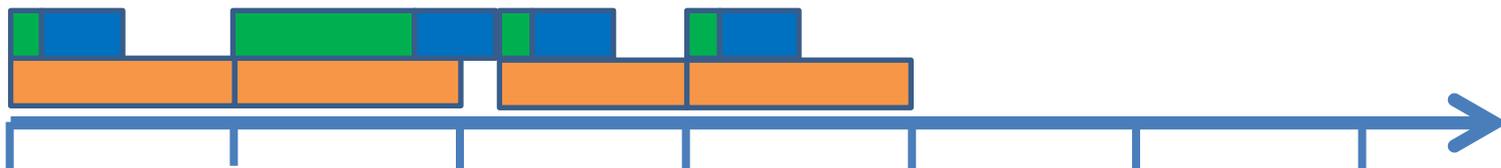   if (now >= loopEndTime) then

        loopEndTime += deltaTime

        do some work

    end if

end while

Alternative approach

```
while (true)
    now = millis()
    if (now >= loopEndTime) then
        ⟹  loopEndTime = loopEndTime + deltaTime
                            OR
        ⟹  loopEndTime = now + deltaTime
        do some work
    end if
end while
```
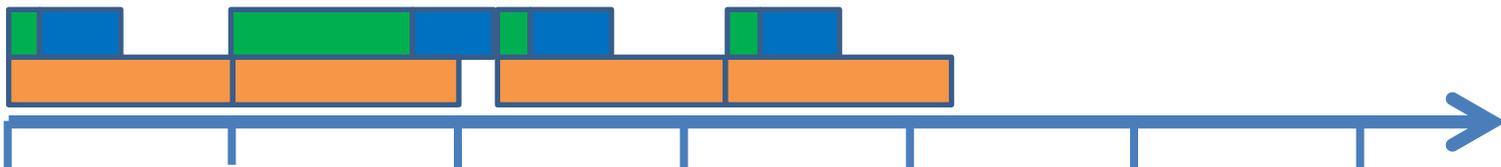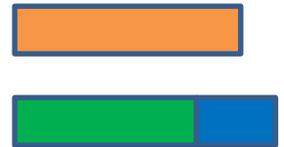
Alternative approach

while (true)

    now = millis()

    if (now >= loopEndTime) then

        loopEndTime = now + deltaTime

        do some work

    end if

end while

Think Like a Finite-State Machine

```
while (true)
    if (millis() > loopEndTime) then
        loopEndTime += deltaTime
        do some work
    end if
end while
```

Do some (but not all) of the work

Remember "state" information (in one or more variables)

Inside delta time conditional if, add switch statement

Back to Pushbutton Debouncing

- Switch between input pin and GND
  - Input goes LOW when switch is pressed
  - Input goes HIGH when switch is not pressed

```
setup() {
    pinMode(pin, INPUT);
}

loop() {
    inputVal = digitalRead(pin);
}
```

Let's See What Happens
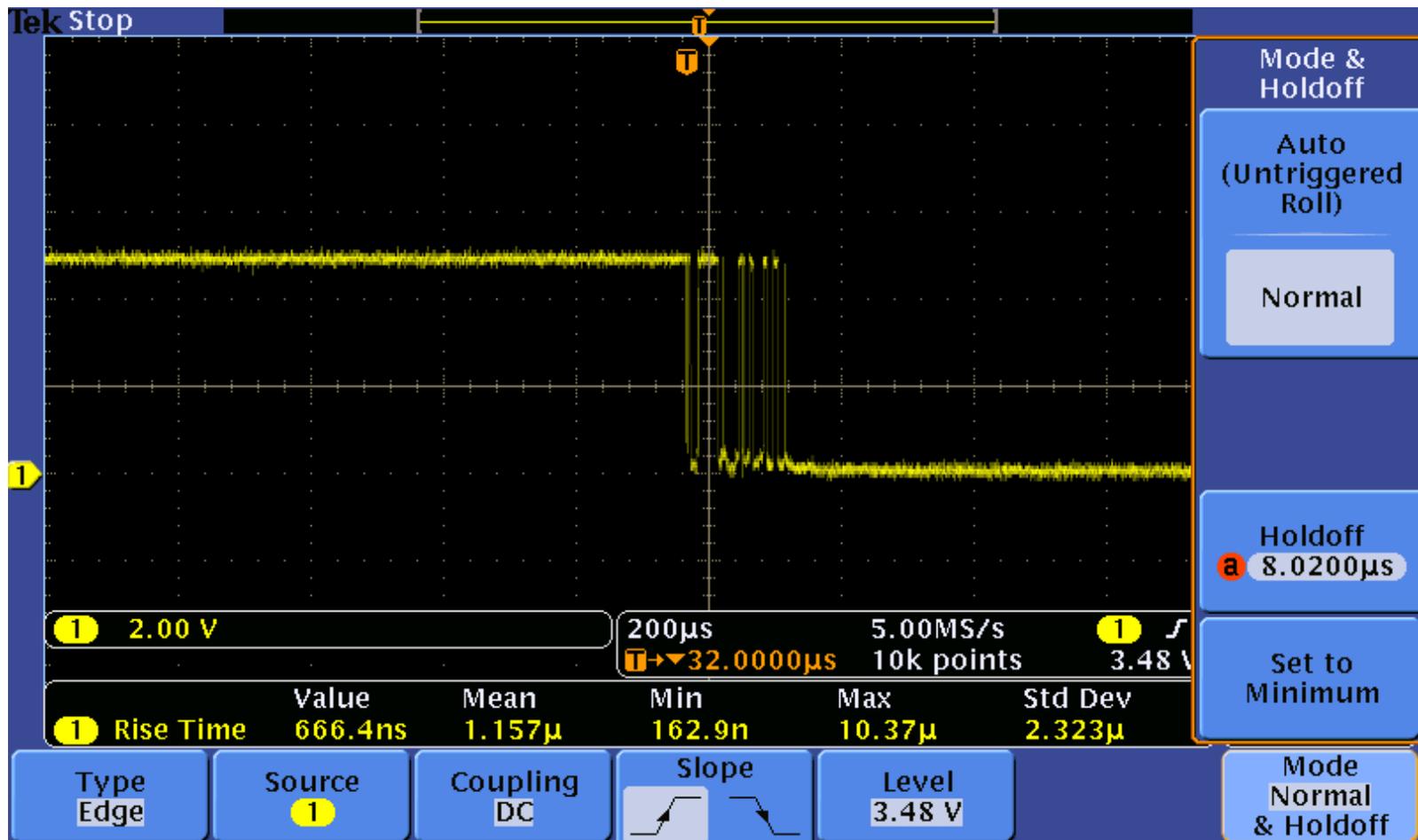
Switch "Debouncing"

Read switch state

Wait enough time for switch to quit bouncing ⬅

Read switch state again

if two switch states agree

    Done

else

    Start over

Delta-time Debouncing

```
while (true)
    now = millis()
    if (now >= dloopEndTime) then
            dloopEndTime = dloopEndTime + debounceTime
            switch (whichRead)
                    case 0: read1 = digitalRead(pin)
                            break
                    case 1: read2 = digitalRead(pin)
                            if (read1 == read2)
                                    inputVal = read1
                            endif
                            break
            whichRead = !whichRead
    end if
    if (now >= otherLoopEndTime ) then …
```

Quiz Time!

True or False?

Delta Timing and delay() accomplish the same thing.

Exam 1 Review

Exam is next Week!

Exam Logistics and Style

- Date and Time
  - Feb 18, here, Brauer 12
  - 1pm to 2:20pm, starting right at 1!
  - (or 2:30pm to 3:50pm, starting right at 2:30!)
- Questions
  - Question 1 will be a collection of short answer things (e.g., true/false, fill in the blank, quick definition)
  - Questions 2 through N will be longer (going more in depth on a particular subject)
- One-page "crib sheet" is allowed
  - 8.5 x 11 sheet, front and back, whatever you want to include (content-wise)

Practice and Help

- B quiz questions are mostly from old exams
- Extra credit opportunity
  - "Extra Credit Quiz 1E," on Canvas, due before exam
  - Practice exam questions (from old exams)

Information Representation

- Number systems
  - Binary, two's complement, hex – conversions
  - Other negative representations
  - Fixed point – Q notation
  - Floating point – definitions

- Text representations
  - ASCII (if you need ASCII table, I will provide it)
  - UTF (-8, -16, -32)

Finite State Machines

- What are they? What are they good for?
- Bubble diagrams
  - Reading (i.e., what does this FSM do?)
  - Writing (e.g., author an FSM to do X)
- Implementation
  - Authoring in Arduino C
  - What constitutes/triggers a state transition?

Programming When Time Matters

- Simple delays
  - Advantages and disadvantages
  - How to program

- Delta time
  - When it really matters
  - Advantages and disadvantages
  - How to program

Input and Output

- Analog Input
  - Linear calibration, scaling, units, ranges
  - Simple filtering
  - Programming interface
- Digital Input/Output
  - Meaning, polarity
  - Programming interface
- Physical construction
  - Wiring
  - Components – resistors, LEDs, potentiometer, temperature sensor

Practicalities

- How to use development environment(s)
- Commonly used library functionality
  - Controlling pins (in and out)
  - Printing to attached PC
  - Timing
- Details of Arduino C language
  - Standard data types
  - Similarities and differences relative to Java
  - Bit-level and logical manipulation

Not Covered in Exam 1

- Communications topics

- This material will be on exam 2