```
// FILE: stopWatch1_0a.nxc - Works!
// DATE: 03/13/23 11:46
// AUTH: P.Oh
// DESC: Display seconds elapsed
// VERS: 1_0a - release version for Spring 2023 ME 425/625
// REFS: mtrSpeed0_2a5.nxc; stopWatch0_1.nxc

task main() {

  // Declare variable -----------------------------------------------------

  // Button related variables
  bool orangeButtonPushed, rightArrowButtonPushed;

  // Timing related variables
  long ticPrev, ticCurr, ticDelta;    // previous, current and delta ticks
  float elapsedSeconds;               // seconds elapsed

  // Initialize variables -------------------------------------------------
  elapsedSeconds = 0.0; // set elapsed time to zero

  // Algorithm starts here ------------------------------------------------
  // (1) Prompt user to begin stopwatch
  TextOut (0, LCD_LINE1, "-> starts" );
  do { // wait until user hits right button
   rightArrowButtonPushed = ButtonPressed(BTNRIGHT, FALSE);
  } while(!rightArrowButtonPushed);
  ClearScreen();
  TextOut (0, LCD_LINE1, "Orange Btn quits" );
  TextOut (0, LCD_LINE2, "Time = " );

  // (2) User started stop watch
  ticPrev = CurrentTick(); // <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
  do {
    // (2A) Poll timer with second CurrentTick
    ticCurr = CurrentTick(); // read timer value <<<<<<<<<<<<<<<<<<<<<<<<<<<<<
    // (2B) Difference in CurrentTick values is elapsed milliseconds
    // Take sum to of elapsed milliseconds to calculate total time elapsed
    // Format as a string so value can be displayed on Brick
    ticDelta = ticCurr - ticPrev;  // difference in ticks [msec]
    elapsedSeconds = elapsedSeconds + (ticDelta/1000.0); // elapsed time [sec]
    TextOut(0, LCD_LINE6, FormatNum("%5.2f s" , elapsedSeconds));
    // (2C) make previous tick value now equal to last read value i.e. ticCurr
    ticPrev = ticCurr;
    // Check if user wants to quit
    orangeButtonPushed = ButtonPressed(BTNCENTER, FALSE);
  } while( !orangeButtonPushed );

  // Orange button pressed, so quit
        TextOut(0, LCD_LINE2, "Quitting", false);
  PlaySound(SOUND_LOW_BEEP);  // Beep to signal quitting
        Wait(SEC_2);
  StopAllTasks();
} // end main
```