Specification of a Simple Cycle Computer

Overview

A cycle computer is required which provides the following information to the user:

- Distance travelled
- Duration of the ride
- Current Speed
- Pedal cadence

The computer is based around a single CMOS chip (to be designed by you). The chip receives information from two hall effect sensors one is mounted on the front forks with a small magnet on the front wheel, while the other is mounted on the bottom bracket with a small magnet on the pedal crank. Two buttons are provided for the user. These are labelled **Mode** and **Trip**. The chip must drive a multiplexed seven-segment L.E.D. display for the indication of distance, time, speed or cadence. The chip is clocked synchronously at 12.8 kHz.

Modes of operation

A fully configured cycle computer will include an odometer unit which records distance travelled, a trip timer unit which records ride duration, a speedometer which records the current speed and a cadence meter which records the rate at which the pedals turn.

After system reset or power on, the computer will show distance travelled on the L.E.D. display. Pressing the **Mode** button causes the computer to cycle through the following modes:

- A) Distance travelled
- B) Duration of the ride
- C) Current Speed
- D) Pedal cadence

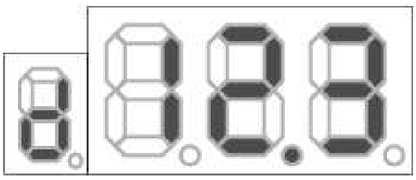
Odometer

The odometer is required to display the cumulative distance travelled by the bicycle as indicated by the hall effect sensor on the front forks.

The odometer will be zeroed at power on by an RC circuit which will operate the system reset. The odometer will also be zeroed whenever the **Trip** button is pressed. (N.B. This causes a SYNCHRONOUS resetting of this unit and is totally separate from the nReset input that causes an ASYNCHRONOUS reset of all registers).

The units of measurement for distance should be kilometres (km).

The following figure shows the expected output on the seven-segment display for a distance travelled of 12.3km



*Note that the display has four 'digits' with the first being smaller than the others. This left-most 'digit' tells us which mode we are in (in this case it displays the character is a '**d**' to indicate distance).*

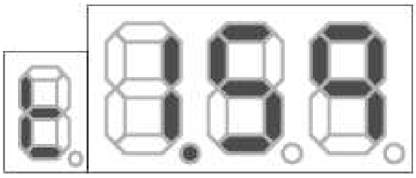
Trip Timer

The trip timer is required to display the cumulative time for which the bicycle has been in motion.

The trip timer will be zeroed at power on by an RC circuit which will operate the system reset. The trip timer will also be zeroed whenever the **Trip** button is pressed. The trip timer will not increment while the wheel is not turning, although to allow for downhill sections it will increment while the pedals are not turning.

The units of measurement for distance should be hours and minutes.

The following figure shows the expected output on the seven-segment display for a trip time of 1 hour and 59 minutes.



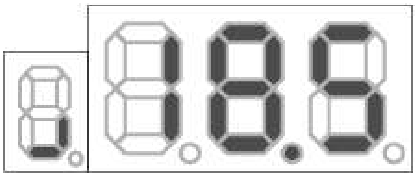
*(In this case the left-most digit displays the character is a '**t**' to indicate time).*

Speedometer

The speedometer displays the current speed of the bicycle. The algorithm chosen for the calculation of speed can make a big difference to the user experience. While some algorithms give values that oscillate significantly about the correct speed, others give obviously incorrect values. A sophisticated algorithm will give relatively stable results both at high speed and at low speed.

The units of measurement for speed should be kilometres per hour (kmh).

The following figure shows the expected output on the seven-segment display for a speed of 18.5kmh



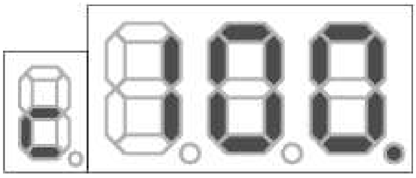
*(In this case the left-most digit is supposed to represent a '**v**' to indicate velocity).*

Cadence Meter

The cadence meter indicates the rhythm of the cyclist's pedalling, which should remain constant for the most efficient transfer of power to the wheels. It is required that this reading is frequently updated.

The units of measurement for cadence should be revolutions per minute (rpm).

The following figure shows the expected output on the seven-segment display for a cadence of 100rpm



*(In this case the left-most digit displays the character is a '**c**' to indicate cadence).*

Wheel Size

At power on, the cycle computer should assume that it is connected to a standard 700x28c wheel with a circumference of 2136mm.

A fully configured cycle computer will include the ability to set a custom wheel size. The wheel size alteration is achieved by pressing the **Trip** and **Mode** buttons simultaneously, there are three sub-states within this mode:

- Set Digit 2
- Set Digit 1
- Set Digit 0

In each state the **Trip** button is used to change the value of the chosen digit until it is correct, pressing the **Mode** button will advance to the next state. After three presses of the **Mode** button the computer should return to distance display mode.

The user guide for your computer should give a formula and/or table for the conversion of wheel size to the value input at this initialization stage. The user guide should also give an indication of the accuracy of the readout.

Range, Resolution and Accuracy

For each of the cycle computer modes, you should choose suitable values for range, resolution and accuracy.

- Range

  The range tells us the minimum and maximum values for the measurement. For the distance you might choose 00.0km - 99.9km for the range.

- Resolution

  The resolution tells us the increment used between adjacent values. For the cadence, you might choose 5rpm as the resolution.

- Accuracy

  The accuracy of a measurement will be affected by a number of factors such as the accuracy of the clock input (you can assume that this is within 0.01% of the specified value), the accuracy with which you store the wheel size and the operation of your cycle computer.

  For low values, the maximum error of a well made measurement should be comparable to the resolution (for this reason it is not appropriate to set an overly small resolution that cannot be supported by the system).

  For high values, percentage errors are likely to dominate over fixed errors. For this reason the error might be stated as "*no more than X units or Y% (whichever is the larger)*".

In particular, a proper estimation of the accuracy of your computer demonstrates that you have a good understanding of its limitations and is essential if you are to get a good mark for the design exercise.

Further Technical Information

Buttons

Normally open, wired between ground and chip input. The input protection pad must have integral pull up resistor.

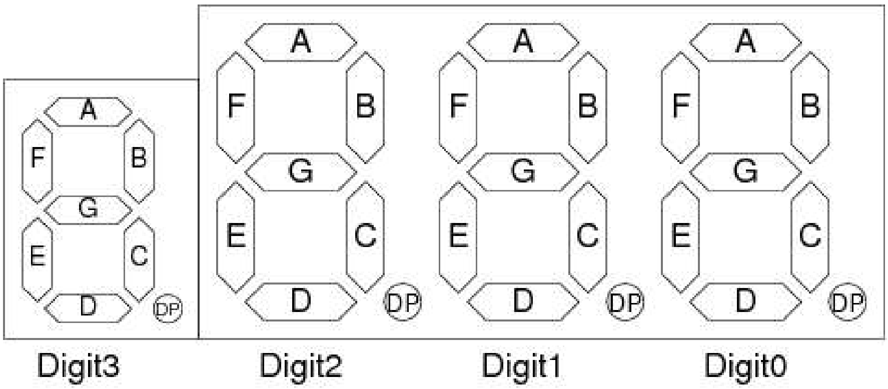
When designing a system to detect the buttons being pressed, you should be aware that the time that a button will be held down is unpredicable and that the switch may bounce when it is pressed and/or when it is released. You may assume that the switch bounce does not last for more than 25ms.

Hall effect sensors

Open collector, active low output (with hysteresis to reduce input noise). The input protection pad must have integral pull up resistor.

LED display

The arrangement of the digits and segments of the L.E.D. display is shown below. Each digit has seven segments marked A-G and a decimal point to the right marked DP. The left-most digit is Digit 3.



The display is a common cathode multiplexed display. Thus a segment is lit if and only if the relevant segment line is taken high and the relevant digit line is taken low.

I/O information

For the sake of common test vectors you must use the signal names given below in your design.

Core Inputs

```
Clock       (12.8 kHz with active rising edge)
nReset      (Active Low)
Test        (Active High)
SDI         (Active High)
nMode       (Active Low)
nTrip       (Active Low)
nFork       (Active Low)
nCrank      (Active Low)
```

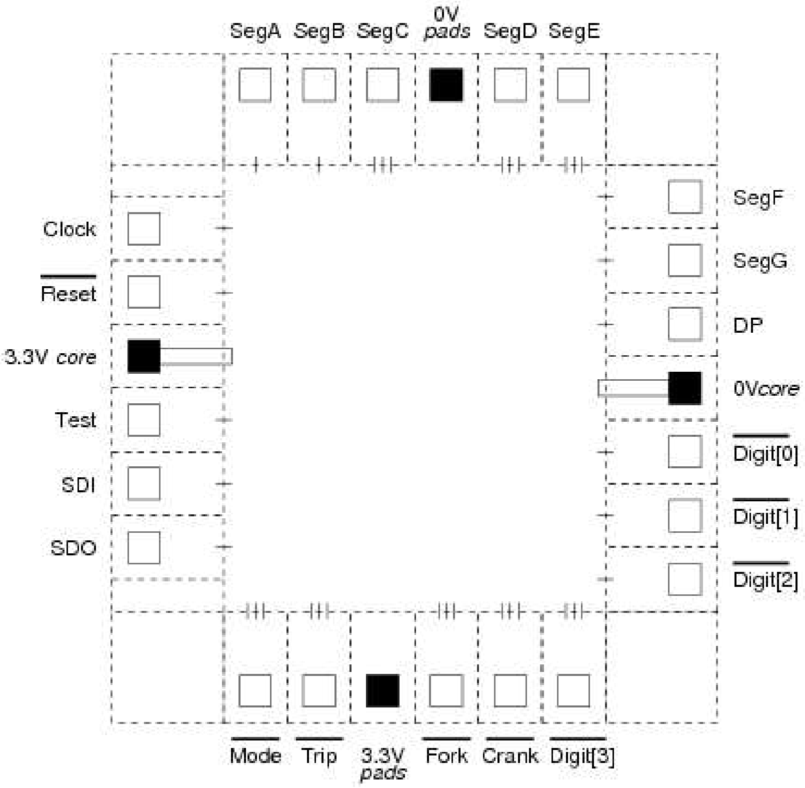
Core Outputs

```
SDO         (Active High)
SegA        (Active High)
SegB        (Active High)
SegC        (Active High)
SegD        (Active High)
SegE        (Active High)
SegF        (Active High)
SegG        (Active High)
DP          (Active High)
nDigit[3]   (Active Low)
nDigit[2]   (Active Low)
nDigit[1]   (Active Low)
nDigit[0]   (Active Low)
```

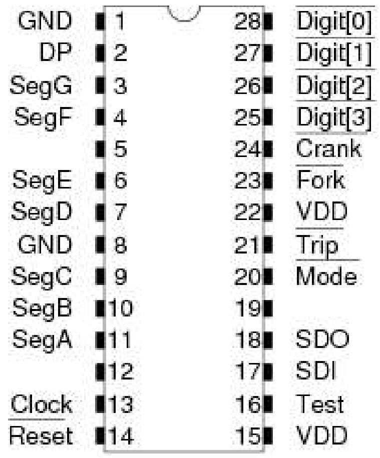
Pad Placement

The default arrangement of pads is provided below.

Arrangement of pads



DIL28 Pinout



Iain McNally

27-1-2020 (v0.1)