Please read,

or take a peek

BUZZ WIRE GAME DESCRIPTION

Scan here to read later:

Buzz Wire Game

Quick Overview

You get a timed lap of the Snetterton Racetrack with this game! The game will speak your player number and lap time, and you may get on the leaderboard or make a new lap record.

When the game first powers up it checks the battery capacity and tells you to place the car at the left or right start point, if it's not already there.

The game then says it's initialised and suggests you press the yellow Help Button, which speaks some instructions. To start racing, you press the green Race Button, wait for the green light, lift the car clear of the start point and place it at the other end of the track.

The race is bidirectional, so leave the car at the end and this becomes the start point for the next player. During the game you'll be told if you're going too fast or too slow based on your crashes and time. At the end of the game your lap time is spoken and you may get some advice if you crashed too often or went too slow! If you get on the leaderboard of top 20 places, you'll get some praise and if you get the lap record you'll get a fanfare too.

If you start the race too soon, you'll get a 10 second time penalty. You'll also get a 3 second time penalty if you skid off course by touching the buzz wire. Correct your course quickly because the crash penalty is added every 0.5 seconds. "When" you crash you'll hear a buzz tone and some car crashing sounds too. If your crash penalties add up to more than your driving time you'll also get an extra 1 minute penalty for "stock car racing".

If you're not crashing too often during a race, you'll hear some background music or sound effects played. During idle times different sounds are played, mostly with a car theme, which are interspersed with adverts every 90 seconds to encourage game play.

If you want a practice, just lift the handle and return it to an endpoint when finished. Unless you press the Race Button you won't get a player number or lap time though.

Pressing the Leaderboard Button will speak out the top 4 player numbers and their lap times. The leaderboard records were cleared for today and the results are not lost when the game is switched off. The fastest lap of the day will receive the Piston Cup Trophy.

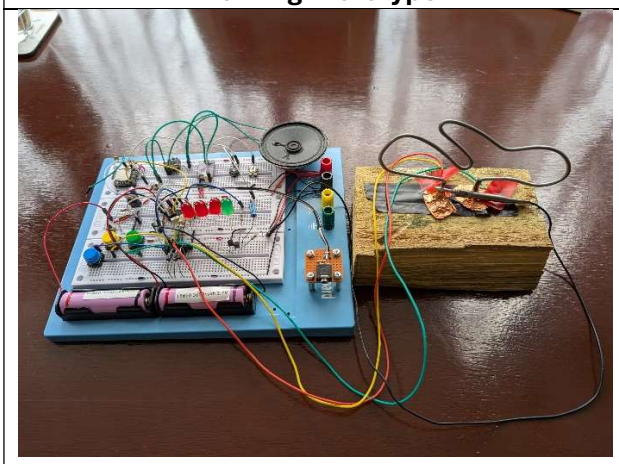
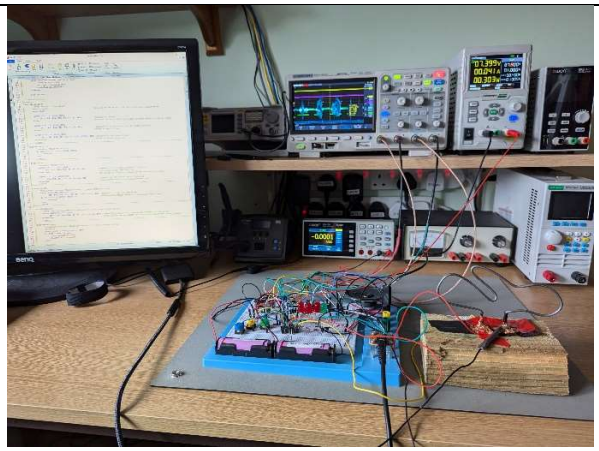
The battery lasts about 10 hours when the volume is high, or about a day if the volume is low.

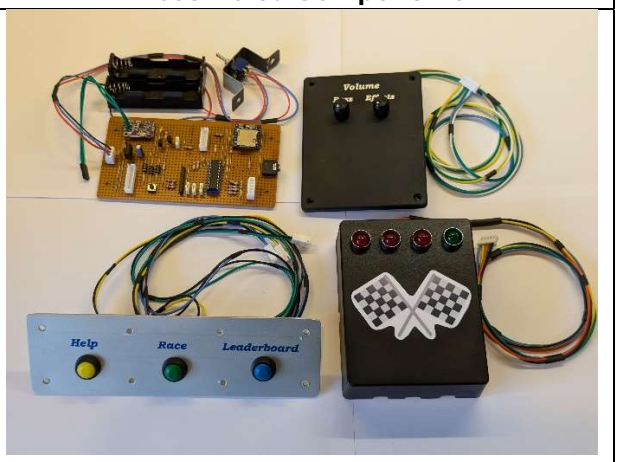
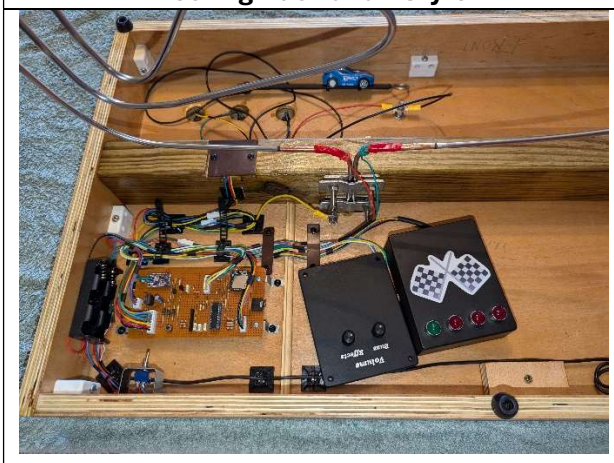
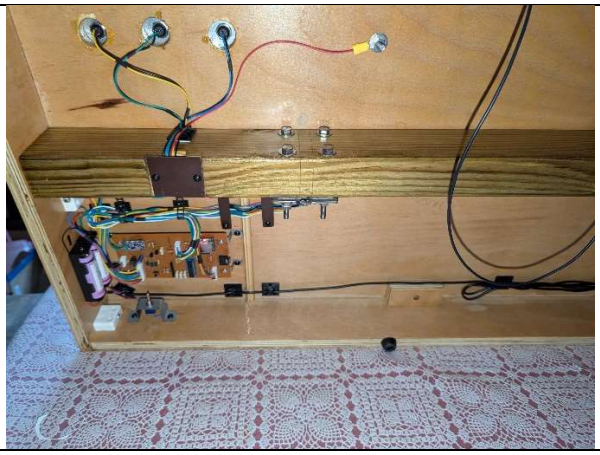
Technical Summary

The game uses an MCU (Microcontroller Unit), which is a low power computer chip as used in something like a washing machine. It has 4 Kbytes of program memory, 56 bytes of RAM for general purpose variables and 256 bytes of permanent storage where the leaderboard is placed. The game almost filled of all these memory locations. The program is about 1,000 lines of code and is written in BASIC because it's simple, fun and I'm old. The MCU is a PICAXE 20X2.

An audio module stores sound on an SD card, consisting of songs, effects and downloaded AI generated words or phrases. These sounds are strung together and controlled by a communication link to the microcontroller. The microcontroller has a timer and the electrical input and output pins for the buttons, buzz wire connections, buzz sound, LEDs (Light Emitting Diodes). The inputs use transient voltage suppression to protect the electronics against static electricity, up to 30 kV.

Photos

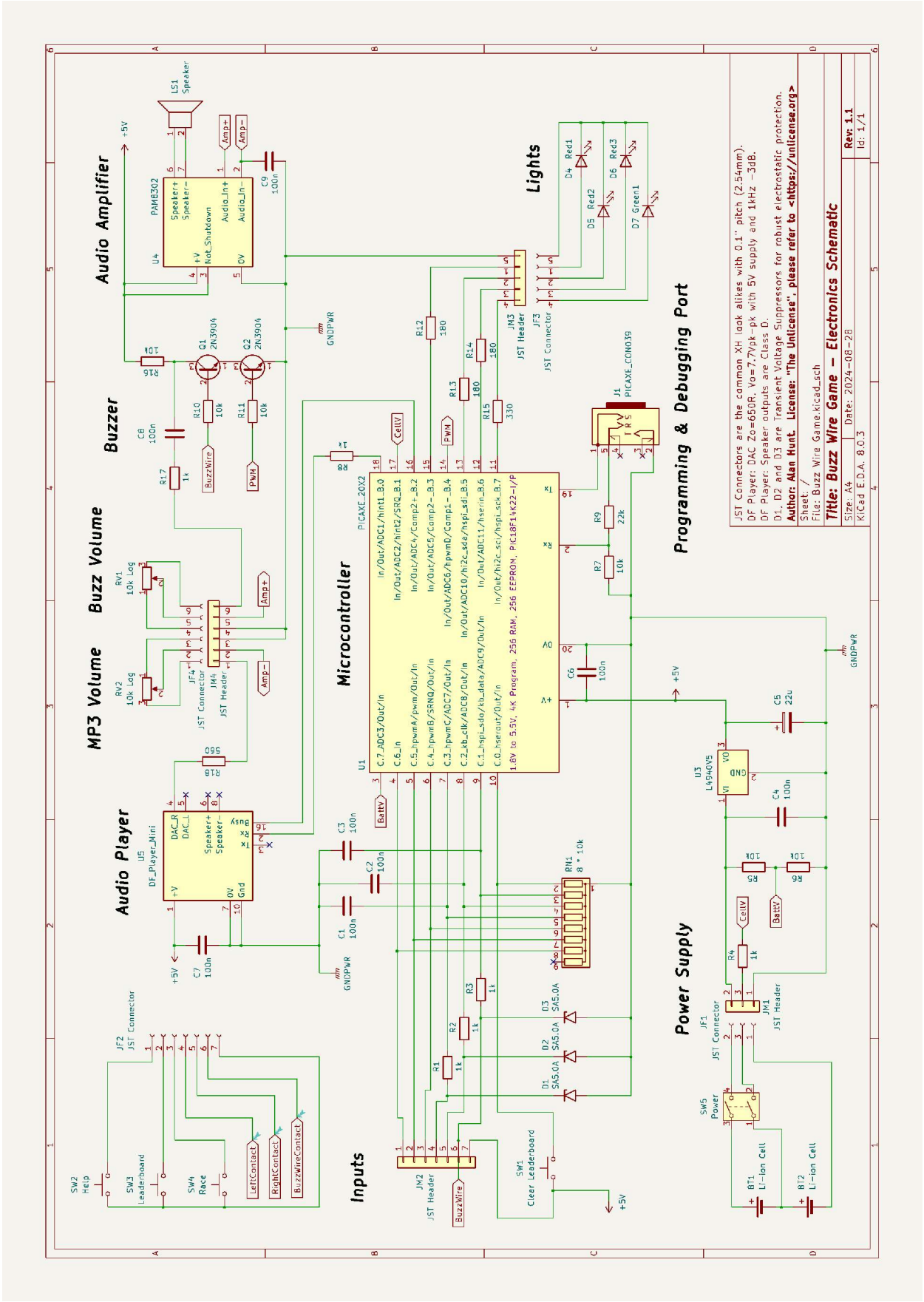
Working Prototype	The Workbench
	
Handle V1 (Final was brazed)	**1st Semblance**
	
Some 30 Year Old Varnish	**The Underneath**
	

Button Panel	Assembled Components
	
Testing Australian Style	**The Piston Cup Trophy**
	
A Look From The Top	**A Look Underneath**
	

It's Finished!
(Until another 22 tracks were added!)

The Circuit Diagram

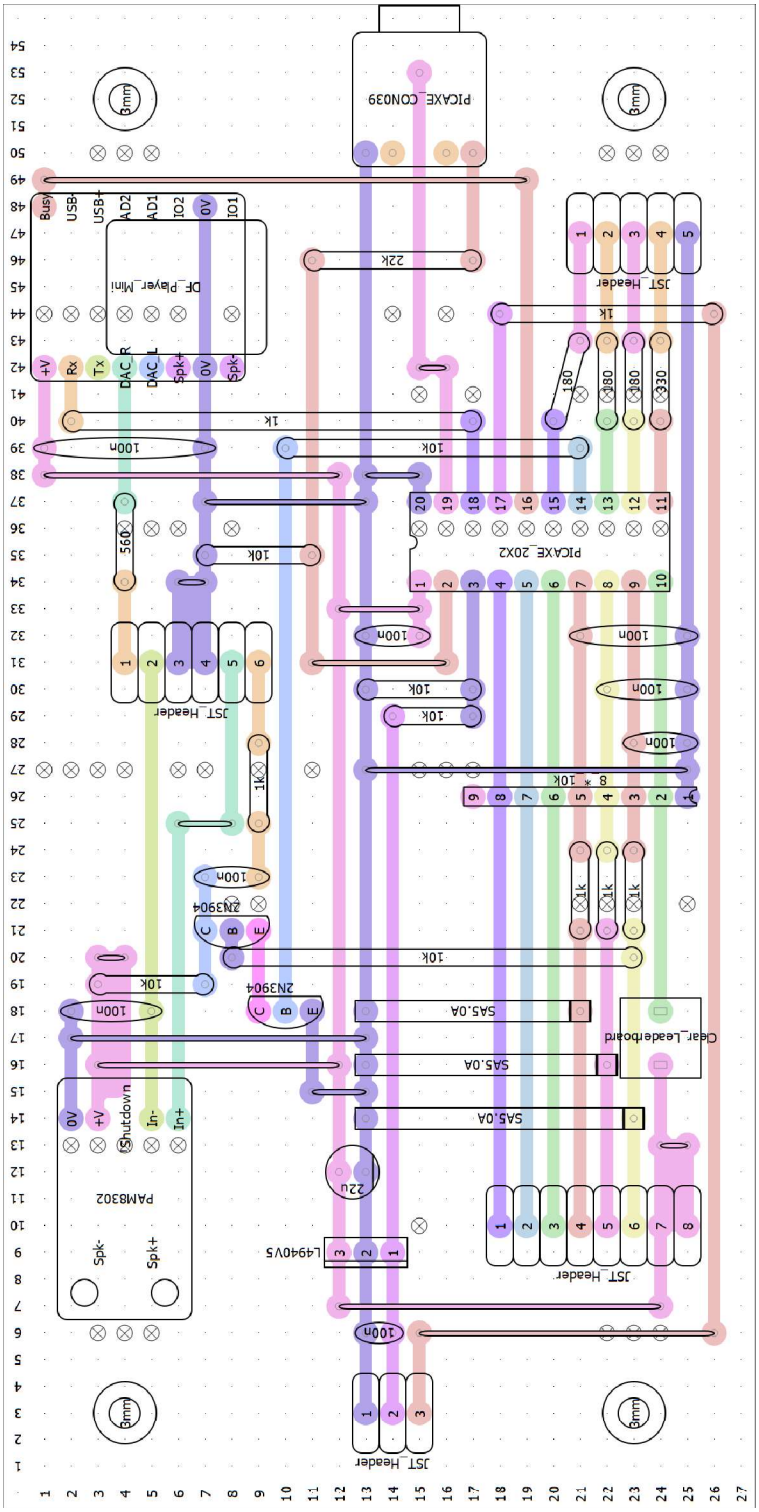
The Electronics Schematic was produced in KiCad and the connectivity of all components was exported as a “netlist”, so that the circuit board design could be produced in VeroRoute.



Circuit Board

The game uses a type of circuit board, which is referred to as “Stripboard” these days. It has holes 0.1” apart, which early electronic chips used and this size spacing is still common today. The board has copper strips, which are easy to visualise and connect components together. Components are passed through the holes and soldered to the copper strips. Where you want to isolate components a hole is cut in the copper strip.

Some of the components I used are over 40 years old, from when I first played with electronics. These days it’s increasingly hard to use this type of layout because components are miniaturised to tiny sizes and then surface mounted on printed circuit boards.



Detailed Features

The game was going to be used in a charity event with a prize attached. So, for the first time ever, I created a list of detailed features for testing. This helped to perform a final code review in addition to identifying test cases. It's challenging to test all the program states of the microcontroller, so both steps proved worthwhile. As a bonus the list helps to describe the game's features too:

1. Initialisation, Battery Checks and Program Updates:
    a. The game tells you when it's powered on or restarted.
    b. The battery voltage is checked at restart and between idle tracks.
    c. If either of the 2 battery cells are nearly exhausted (below 3,000mV) a warning will be spoken and the game will shut down into a low power state to prevent over discharge. If either cell is low (below 3,300mV) then a warning will be spoken between sound tracks when the game is idle, or directly after a game has finished, then the game continues as normal.
    d. The Reset Button only works during a restart and it clears the games stats.
    e. Program updates do not clear the game stats.
    f. After a restart the previous number of games is spoken and if games have been played the lap record is spoken.
    g. After a restart the handle must be placed at a starting position, if not it will speak a warning every 3 seconds before proceeding.
2. When the game is waiting to be played:
    a. Background music with a driving or fun theme is always played. There are several soundtracks played in a loop.
    b. Every 90 seconds an advert is played to encourage game play. These adverts play in a loop:
        i. "Roll-up, roll-up! It's only 50p a game. Have some fun and a chance to win the coveted Piston Cup Trophy!"
        ii. "Most racing car legends start their careers with BUZZ WIRE RACING! It's not too late to start now, so give it a try!"
        iii. "Here is an important announcement: It's championship race day here at Snetterton. Get the fastest lap of the day and get that darn tooting trophy!"
        iv. "Please, please, PLEASE have a game. I'm processing 4 million lines of code a second, so I'm kind of getting bored over here! "
        v. "Hey you! Yes you! Do you want some fun? It's a mere 50p for a chance to become a racing car legend!"
        vi. "Come on, have a game! I promise you won't EVEN kill yourself if you crash."
        vii. "This game has a special 2 for 1 offer! You have some FUN! And your money will go to restoring the Village Sign TOO!"
        viii. "Is there no one else! Is there no one else!", a Brad Pitt extract from Troy.
3. Starting a practise:
    a. If the handle is raised from either start point without pressing the Race Button first, then a practise session is assumed. "Having a practise hey! There's no lap time for you." is spoken, but other sound effects of a real game are still played.
    b. When the handle is returned to either endpoint, "Practise over and ready to race! Press the green button next time!" is played.
4. Starting a race:
    a. Pressing the Race Button only functions when the game is in idle mode, or while the help information is being spoken.

b. The game is bidirectional and can be started at either start point, even if there's been a practise and the previous race endpoint is changed.
c. After pressing the Race Button a coin drop effect is played, then the player number is incremented and spoken, then starting beeps and the LED sequence (Red 1, Red 1 and 2, Red 1, 2 and 3, and then Green only) are synced to run in parallel.
d. To allow for an accidental wiggle of the handle, the handle can lose contact with the start point for up to 100mS before a premature start is determined.
e. If the player starts before the green light a 10 second penalty is created and a message "Time penalty! Started too soon!" is played.

5. During a race:
    a. Race themed background music starts with the game. There are several soundtracks in a loop and the next track is played if a game is started, or if a track finishes during a game.
    b. The player gets some feedback:
        i. The audio clip "Taking it slow huh. I can respect that" is played after 10 seconds if crash count is zero, or after 30 seconds if crash count is less than 2.
        ii. The audio clip "Slow down!" is played after 10 seconds if crash count is more than 2, or after 30 seconds if crash count is more than 5.
        iii. "You've been playing a while now, if there's someone waiting let them have a go to" is spoken after 90 seconds.
    c. Whenever the handle touches the buzz wire:
        i. There's a buzz tone.
        ii. The background sound is interrupted with a crash sound effect, the sound effect changes each time and ultimately loops.
        iii. There's a 3 second penalty, which is repeated in 0.5 seconds unless the handle is free from the buzz wire.
6. Cancelling the race:
    a. If the handle is returned to the start point during a race, then the race is cancelled and the next race will use the same player number. To debounce any handle wobbles during the race start phase, this event won't be acknowledged until at least 2 seconds after the normal race start, or a premature race start.
    b. Then "Going back huh. I would have done the same" and "Press the Race Button to start again!" are spoken.
7. Finishing the race:
    a. If the handle touches the finish point:
        i. The green race light goes out and the log of player numbers is incremented and stored permanently.
        ii. "You've finished!" is spoken.
        iii. If the total crash penalty time is higher than the driving time an audio clip "Slow down stop the insanity!" is played and a cartoon voice states "You spent more time crashing than racing!" and a normal voice states "A 1-minute penalty is added for stock car racing!".
        iv. If the player's adjusted lap time is not quick enough for a leaderboard entry then "Aarh you just missed out on the leaderboard! Have another try!" is spoken.
        v. If the player's adjusted lap time achieves a leaderboard ranking their player number and lap time is inserted in the permanent memory of the leaderboard. All lower positions are shuffled downwards and eventually out of the leaderboard, which is limited to 20 places.

vi. If the player is ranked 1st the audio clip “A new record!” is played. If this isn’t the 1st player, then 1 of 4 random victory audio clips is played.
vii. If the player achieves a leaderboard ranking that is less than half the number of games played, the announcement “Here's a musical tribute just for you!” is spoken. A short fanfare is then played from a loop of tracks. Then a voice states their position on the leaderboard.
viii. If the crash count is zero then “No crashes. That's amazing for this crowd!” is spoken. If it’s not a lap record the audio clip “Wow all I can say is wow!” is played.
ix. If the crash count is greater than 9 “Unfortunately the car is a right off” is spoken.
x. If there were any crashes, a voice states the number of crashes and how much time each crash cost.
xi. Then the lap time is spoken in seconds to 1 decimal place.
xii. Then “Waiting for next player!” is spoken and the game returns to idle/waiting mode.

8. Help Button:
    a. The help button only functions when the game is in idle mode, waiting to be played.
    b. This soundtrack is spoken: “This is a souped-up buzz wire game! The idea of this game is to get your race car around a lap of the racetrack as quickly as possible. If the car touches the metal tube you'll hear the car crashing, your parents worrying about the repair bill and most importantly time will be added to your lap of the race! To start a race, press the green button and wait for the green light. All you have to do is lift your car up and place it down at the other end of the track. The game will tell you your lap time and whether you've set a record for the day. Take it in turns to play. Have fun and Good luck!”.
    c. As an admin function, if the Help Button is held down and the Leaderboard Button pressed then the battery cell voltages will be spoken.
9. Speaking the Leaderboard:
    a. The Leaderboard Button only functions when the game is in idle mode, or while the help information is being spoken.
    b. If any games have been played an announcement starts with “The top N positions are”, where N is the number of games played up to the maximum of 4.
    c. Then from 1st to 4th place: the place, player number and lap time in tenths of a second are spoken.
    d. If the number of games is more than 4 the number of games played is spoken.
    e. Then “Press the green race button to beat them!” is played.

The Program

Written in PICAXE BASIC, the program occupies 4,072 bytes of the 4,096 available on the 20X2 microcontroller. As more functionality, like the Leaderboard, was added some memory had to be saved by reducing the debug logging.

The program is organized into the following regions to improve quality:

1) Program Notes
2) Compiler Directives
3) Resources
    a. Pins
    b. Variables
    c. Constants
    d. Utilities
4) Initialisation
5) Main:
6) Interrupt:
7) Subroutines

Four software modules were written for the game, these are intended to be re-useable utilities for future projects:

1) "Debug_Terminal.basinc" helps to easily display status information to a computer screen while the program is running.
2) "Timing.basinc" sets the processor clock rate and provides timing functions regardless of the clock rate.
3) "DF_Player_Mini.basinc" provides functions to easily control sound output from a DF Player Mini audio module mounted on the circuit board.
4) "Voltage.basinc" calculates the chip supply voltage and the input voltage levels of the battery cells.