



Line Following Robot in Verilog





Abstraction

This application note provides a learning opportunity for users to generate different control modes with various algorithms on an FPGA based portable platform named Terasic A-Cute car. The A-Cute car is equipped with sensors and motors. The goal is for users to be in full control of this platform and learn:

1. The design and application of peripheral circuitry for sensor and motor
2. FPGA-based logic design
3. The algorithm of control mode

The project introduced in the following sections is written in VerilogHDL code only and realized on Terasic DE0-Nano board. A reference design of controlling A-Cute car through push-button and InfraRed (IR) remote controller is provided.

Content



Abstraction.....	1
Content	2
Reference Design Overview ..	3
Demo Setup.....	9
Rebuild Project	11

Reference Design Overview



Terasic A-Cute car is designed to following the black lines with the implementation of Register Transfer Level (RTL).

The 7 sets of reflective IR sensors equipped are used to track the black lines. The left and right motors are controlled and turned by the DE0-Nano according to the data collected from the IR sensors. The control algorithm and data processing are implemented in Verilog, including the conversion of 8- channel ADC, where 7 of them handle the conversion of sensors from analog to digital and the last channel monitors the battery voltage. It also includes IR remote and manual controlling, as well as the dual motor PWM rotation speed and steering process. All the functions are implemented on DE0-Nano board

Block Diagram of A-Cute Car

The A-Cute car consists of the followings, as shown in Figure 1

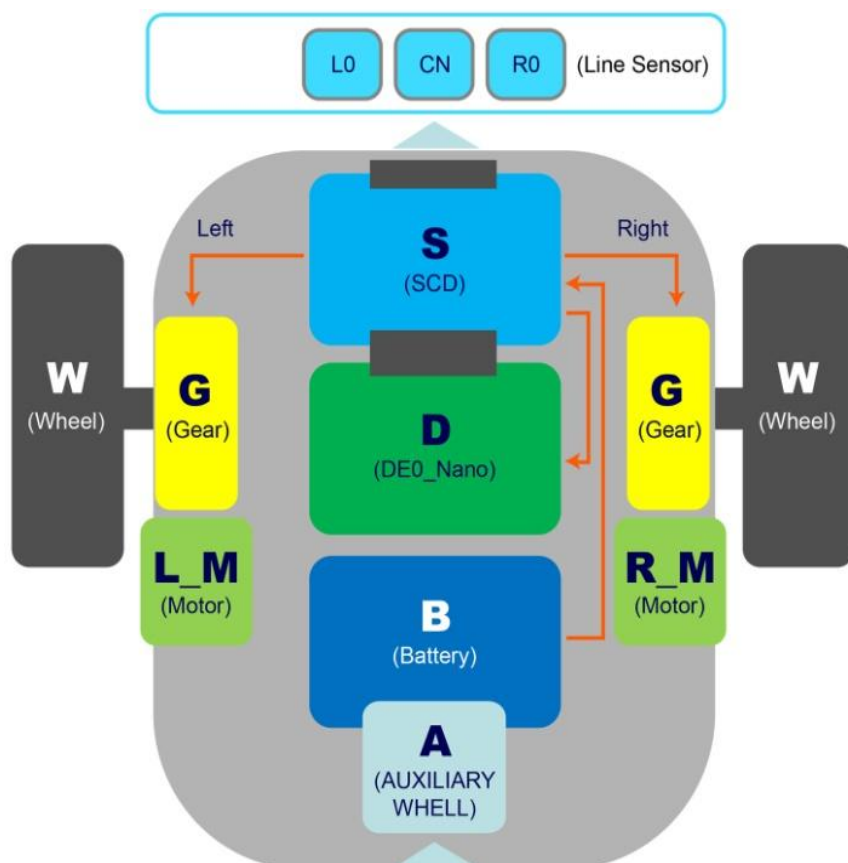


Figure 1 Block diagram of A-Cute car hardware



1. Line Sensor Board: consists of 7 sets of reflective IR sensor
2. SCD (S): includes buck-boost DC/DC IC, motor driver IC, 8-channel ADC, IrDA, and over-voltage protection components.
3. DE0-NANO (D): an FPGA-based mainboard.
4. Battery (B): a battery pack which includes 4 AA batteries.
5. Auxiliary Wheel (A): support the car body.
6. Motor (M) / Gear (G) / Wheel (W): one set each (left & right) to drive the car.

Setup the Track

Use anti-glare black tape with approximately 18mm width to setup a circle on a board or on the floor, where white background is most recommended, as shown in Figure 2

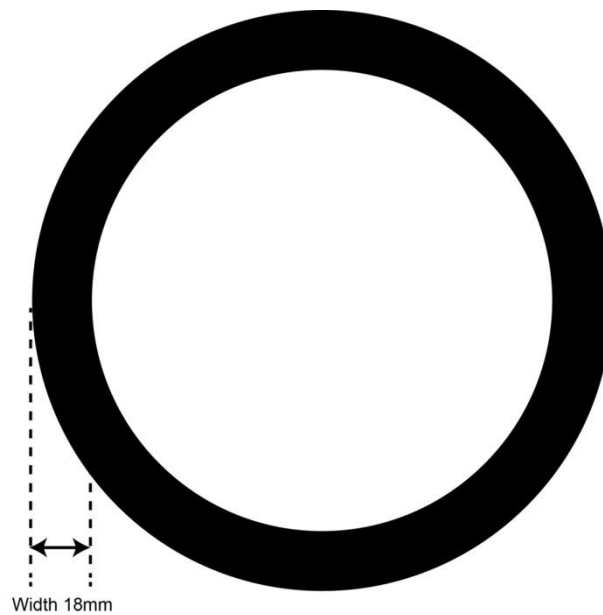


Figure 2 Track Setting

Mechanism of Track Control

This project implements a simple 3-level line following algorithm to control the A-Cute car and follow the track. The algorithm uses only the middle 3 sets of IR sensors and categorized in L0 (left), CN (middle), and R0 (right).

When the A-Cute car gets close to the black tape, the infrared rays emitted from the sensors will be absorbed by the black tape and will not be reflected back to the receiver of the sensors. This is represented by “1” in digital logic. If the A-Cute car is moving away from the black tape, the logic would be in “0”. The readings from 6 sensors will be discerned according to the orientation of the A-Cute car as impartial, small bias to the right, large bias to the right, small bias to the left, large bias to the left, and completely off the line.

IMPARTIAL : [L0 CN R0] =111, as shown in Figure 3.

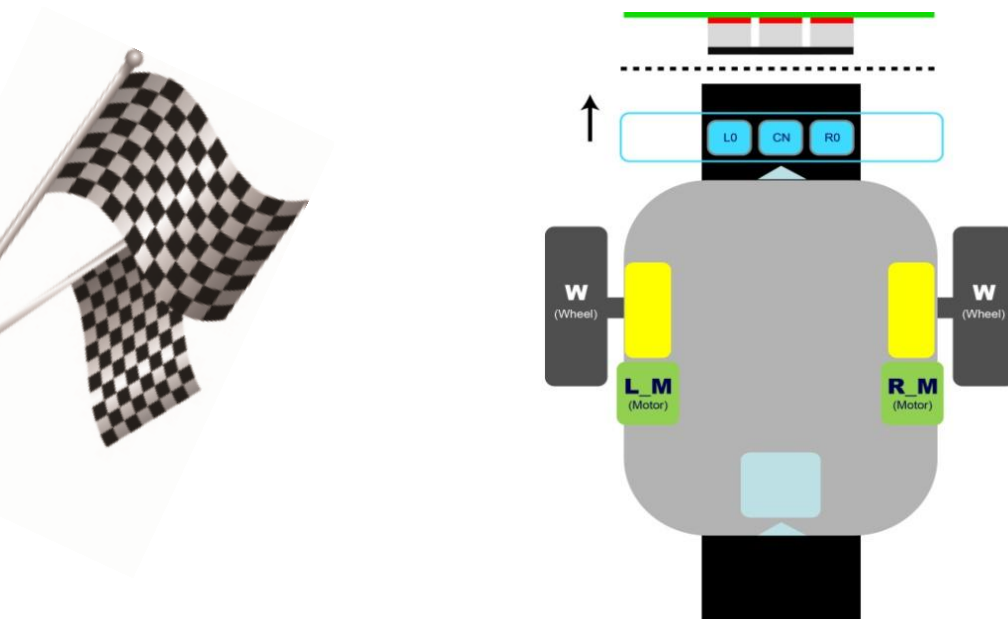


Figure 3 Impartial

Prediction to the left and right motors

L_M = same speed

R_M = same speed

Small bias to the right : [L0 CN R0] =110, as shown in Figure 4 .

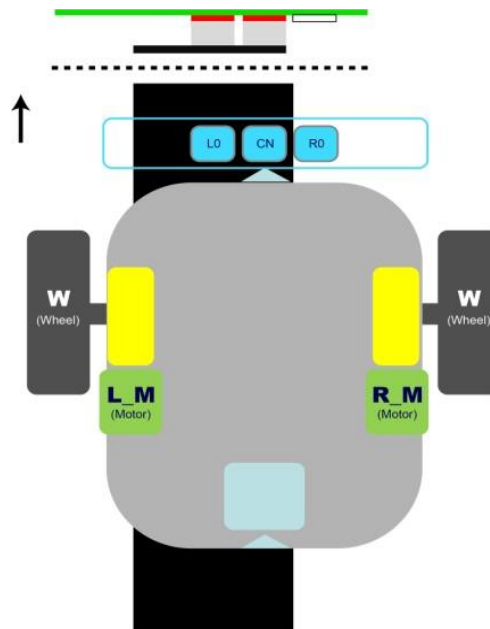


Figure 4 Small bias to the right

Prediction to the left and right motors

L_M = slow down the speed in half

R_M = same speed

Large bias to the right : [L0 CN R0] =100,as shown in Figure 5.

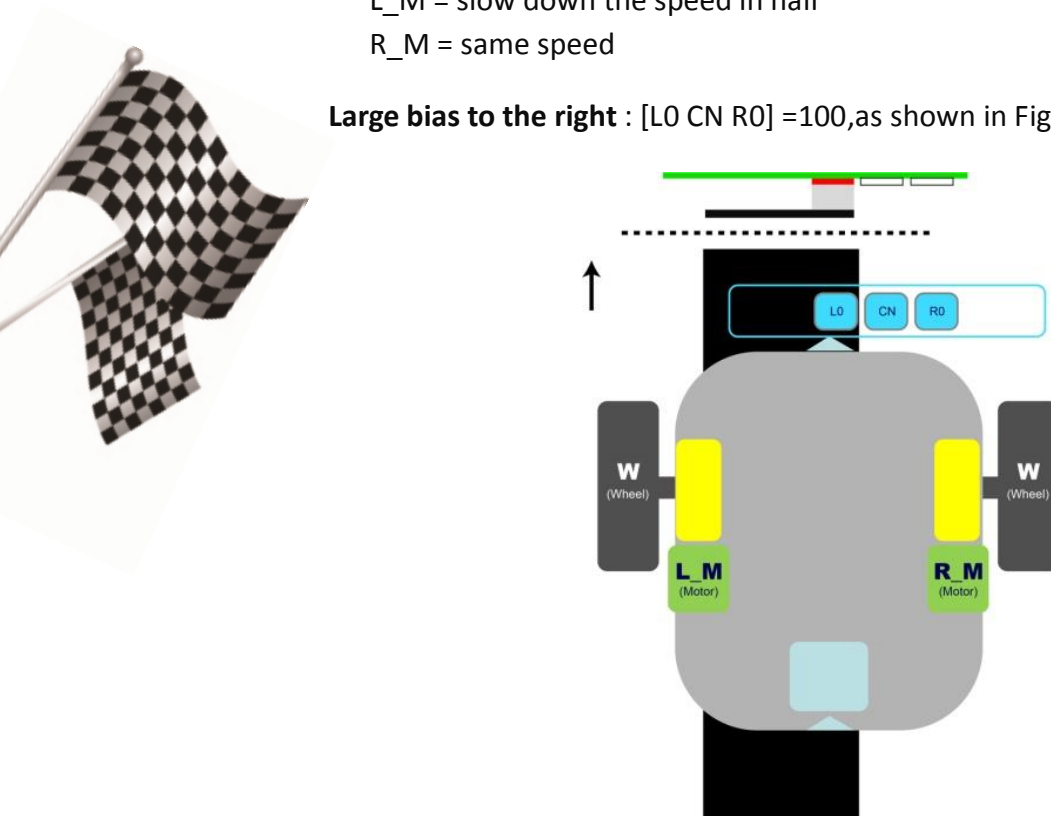


Figure 5 Large bias to the right

Prediction to the left and right motors

L_M = 0

R_M =speed down

Small bias to the left : [L0 CN R0] =011,as shown in Figure 6

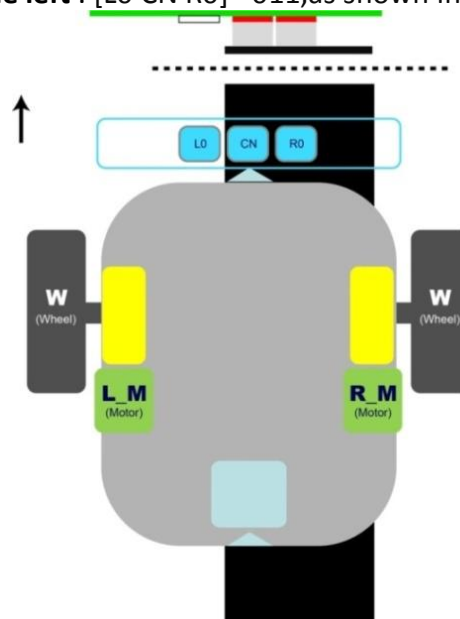


Figure 6 Small bias to the left

Prediction to the left and right motors

L_M = speed remain

R_M = slow down the speed in half

Large bias to the left : [L0 CN R0] =001,as shown in Figure 7

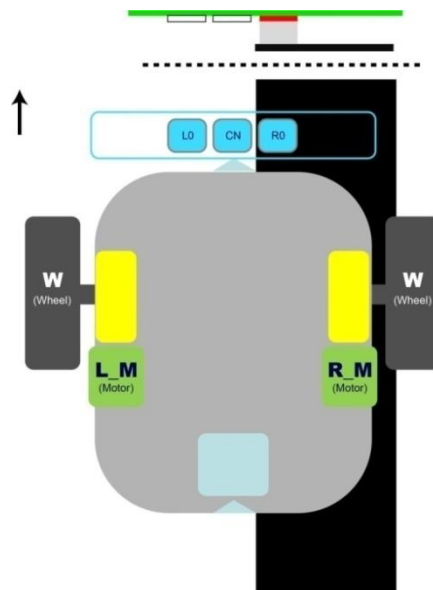


Figure 7 Large bias to the left

Prediction to the left and right motors

L_M = speed down

R_M = 0

Completely out of line : [L0 CN R0] =000,as shown in Figure 8.

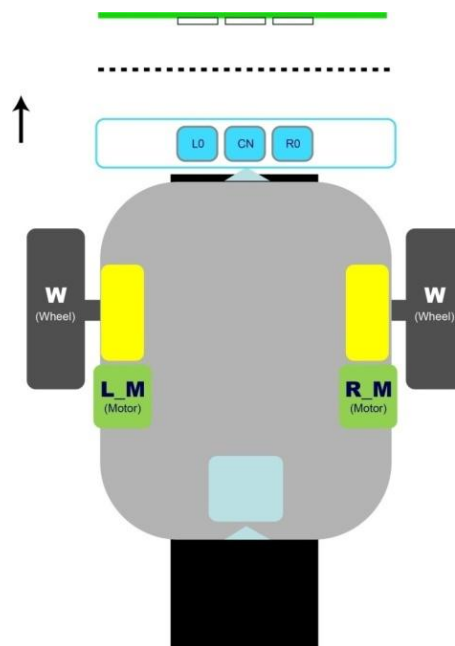


Figure 8 Completely out of line

Prediction to the left and right motors

L_M = speed down

R_M = speed down

The direction of both motors will be changed from forward to backward.

FPGA Function Block Diagram

Figure 9 shows the control block diagram of Terasic A-Cute car. It includes seven modules to achieve functions implemented on FPGA. The following sections will illustrate how each module works.

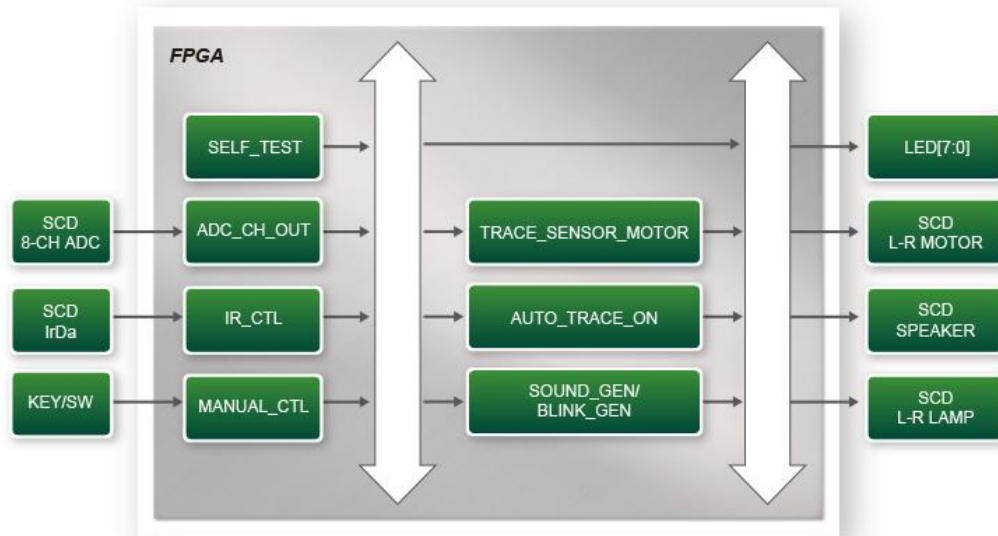


Figure 9 Block diagram of control system

ADC_CH_OUT

This module acquires ADC data from 7-channel IR sensor and 1-channel battery power detection. The working frequency is 50MHz. It reads data from 8-channel LTC2308 of SCD in 12-bit through 4-wire SPI interface. Each cycle takes approximately 33us to read the ADC data of each channel.

The channels 0~6 of ADC are connected to 7 sensors with non-attenuation 0 ~ 4.09V analog input. The channel 7 of ADC is for battery voltage detection with 4 times attenuation at 0 ~ 16V analog input.

IR_CTL

There is a component on SCD for receiving IrDa signal. When the IR remote controller sends out signal, the module will decode the signal received from the component and execute corresponding actions. The only buttons used from the IR remote control here are Up, Down, Left, Right,

Return, Manual, and Play.

MANUAL_CTL

It will generate control commands according to the KEY0/KEY1/SW0 manually pressed on DE0-Nano.

SELT_TEST

When the A-cute car is powered up, the module will light up the LED on DE0-Nano to indicate the FPGA is configured successfully.

SOUND_GEN/BLINK_GEN

SOUND_GEN is a sound generator, which generates 625 Hz square wave, as an acoustic source of the speaker. BLINK_GEN generates 0.25 Hz square wave as the blink source of the LED on DE0-Nano.

TRACE_SENSOR_MOTOR

This module includes three submodules:

- a. **SENS_RESPONS**: It categorizes the digitized values of sensors converted from ADC in black and white, which are represented as “1” and “0” in digital logic, when the value is over or below the threshold, respectively.
- b. **MOTOR_DIR_DET**: It determines the rotation speed of left and right motors to control the direction of A-Cute car based on the result of **SENS_RESPONS**.
- c. **MOTOR_PWM_OUT**: It's the PWM generator for rotation speed of both left and right motors. The speed and direction are set according to the input control mechanism of DRV8848PWP (bi-channel motor drive IC) based on its specifications.

AUTO_TRACE_ON

It generates buffer to delay self-propelling after the A-Cute car receives command for self-propelling .

Demo Setup

The steps to setup the demonstration are:

- Set the power switch of SCD card to OFF position.



- Insert four AA batteries .
- Set the power switch of SCD card to ON position.
- The demonstration will be executed upon power up. If the default code is erased, please execute test.bat from the folder below to configure the FPGA on DE0-Nano:

A-Cute system cd/DE0_NANO_CAR_RTL/demo_batch

■ **Key Control:**








1. Press “KEY0” button and the motors will move wheel forward or backward, depending on the status of SW0. When the SW0 is set to “0”, the wheel will move forward. The wheel will move backward when SW0 becomes “1”.
2. When “KEY1” button is pressed, the two lamps on the SCD will twinkle and there will be a “Beep” sound from the speaker at the same time.
3. Press and hold “KEY0” button for two seconds and the LED6~4 will be blinking. Three seconds later, the A-Cute car will start circling clockwise for five seconds before circling anti-clockwise for another five seconds. It’ll take total of ten seconds to complete this demonstration and it can be stopped at any time by pressing “KEY0” button.
4. Press and hold “KEY1” for 2 seconds and the LED6~4 will be blinking. Three seconds later, the A-Cute car will move toward the black line. The mode can be stopped by pressing “KEY1” button.

■ **Perform IR Remote Control**

Figure 10 shows the IR remote control used to interact with the A-Cute car wirelessly.



Figure 10 IR remote control

1. Press the  button to move the smart car forward.
2. Press the  button to move the smart car backward.
3. Press the  button to turn left.
4. Press the  button to turn right.
5. Press the  button to make the “Beep” sound and the lamp will blink at the same time.
6. Press the  button to enter the traction mode i.e. the A-Cute car will travel along with the black line.
7. Press the  button to exit the traction mode.

Rebuild Project

The project is built in Quartus II v15.1. Its source code is located under the folder : A-Cute system cd/DE0_NANO_CAR_RTL

Open the Quartus project file DE0_NANO_CAR_RTL.qpf and click the menu item “Processing → Start Compilation” to start the compilation. When the compilation is complete, an output file named DE0_NANO_CAR_RTL.sof will be generated under the output_files folder.