The Zumo 32U4 line sensor array is a separate board that attaches to the main underneath board, placed in front of the robot as shown in Figure 1.3. The board features five line sensors connected to the microcontroller. The five line sensors face downward and can help the Zumo distinguish a non-reflective (black) line on a reflective (white) surface. Each reflectance sensor consists of a down-facing analog infrared (IR) emitter LED paired with a phototransistor that can detect reflected IR light from the LED. Three of the five sensors are placed in the middle of the board at gaps of about 10 mm and the last two are situated at the far end of both sides such that the total sensor coverage is about 80 mm.

The line sensors are just an indication of surface reflectivity. The sensor value is lower when over a white surface and higher when over a dark surface. The actual values are heavily dependent on the external lighting conditions even though sensor array is shielded. Further, there is appreciable difference between the readings of various sensors due to various factors like placement, manufacturing differences, etc.
To get usable values from all the sensors, the sensor values are normalized for environmental conditions and sensor differences and the normalization values are stored in Data Flash of the microcontroller. This enables the normalization data to be used multiple times. To normalize the sensor values, the robot is placed on the track and rotated such that all the sensors pass over the black and white surfaces. The maximum and minimum readings for all the sensors are noted down and stored in non-volatile memory (Data Flash) of the microcontroller.
The `lineSensors.readLine(lineSensorValues)` function returns a value between 0 and (number_of_sensors) - 1)×1000. As the Zumo 32U4 has an array of 5 sensors, the reading will be 0-4000. 0 represents the left most sensor and each increment of 1000 after that represents another sensor. 0 = first sensor, 1000 = second sensor and there are ranges in between as well. 500 means the line is between the first and second sensors, 1200 means it's between the second and third sensors but closer to the second, etc. Note that if the reading is 2000, it means that that the robot is centered on the line.

This output of the IR line sensor array is fed to the microcontroller which calculates the error term between the sensor value and the line position setpoint. The most convenient and reliable way to compute the error term is by dividing the weighted average of the sensor readings by the average value and normalizing the values. The formula used to compute the error $\varepsilon(k)$ at time-instant k is given as follows:

$$\varepsilon(k) = 1000 \times \frac{\sum_{i=0}^{4} i \times L_i(k)}{\sum_{i=0}^{4} L_i(k)} - 2000$$

In the equation, $L_i(k)$ refers to the output of the i-th line sensor in the array. Here, 1000 is the normalizing factor which is used to ensure that the contribution of the error term from each sensor is normalized to a value of 1000. The subtraction by 2000 is to ensure that zero error occurs when the robot is centered on the line.
The sensors are numbered from the left to the right. An error value of zero refers to the robot being exactly on the center of the line. A positive error means that the robot has deviated to the left and a negative error value means that the robot has deviated to the right. The error can have a maximum value of ±2000 which corresponds to maximum deviation.
The main advantage of this approach is that the five sensor readings are replaced with a single error term which can be fed to the control algorithm to compute the motor speeds such that the error term becomes zero. Further, the error value becomes independent of the line width and so the robot can tackle lines of different thicknesses without needing any change in code.

1.3.1.2 The actuators

The robot is driven by two micro metal gearmotors with extended motor shafts, coupled to wheels with differential drive. It is battery powered, with the motors supplied with regulated power supply