**The Hong Kong Polytechnic University**

**Department of Electronic and Information Engineering**

**Demonstrations Report: EIE3105 Integrated Project**

**Group members:** CHAN Kai Ho (19057769D)

SZE Kin Sang (19062606D)

**Abstract:**

An integrated project that requires students to do hardware programming with ARM architecture to use different components and modules of the control board, in order to allow the cart to finish different tasks.

**Content:**

1. **The design and difficulties on Demonstration 1**

In Demonstration 1, students are required to implement 2 sets of code to get two products. Product 1 requires students to be able to use the remote control to move forward, backward, forward-towards-left and forward-towards-right through Bluetooth module. While Product 2 requires students to get r reading from the ultrasonic module on three different distances.

For product 1 remote part, row scanning method is applied to detect the key press, with also software debounce of 200ms applied to avoid sending excessive data through Bluetooth module.

At the same time, the cart used in product 1 will send different PWM signals to control two motors based on the character received from Bluetooth module. If the cart detects the button has been pressed is same as last button, then the cart will stop its movement.

For product 2, the cart will do send a 10us trigger pulse to ultrasonic module by using GPIO to set bit, TIM3 delay, and then GPIO reset bit. With the distance calculated based on the formula given in datasheet, .

In this Demonstration 1, both of our groupmates did not face any difficulties as code can be easily referenced from Laboratory exercise and teaching material.

1. **The design and difficulties on Demonstration 2**

At demonstration 2, students are asked to implement line tracking function into the cart, based on the sensors below the cart.

At the very beginning, we used the center four bits to indicate if the line is allocated with the center of the cart, if the four bits were shifted aside, different PWM signals would be sent out to turn the cart back to the center. However, this caused a severe problem that the remaining four bits should also be considered when doing case switching. Thus, it is exceedingly difficult to track the location of the cart when it reaches different points, and the inner circle part cannot be done properly.

After consultation with Dr Cheung, we have learnt that it is possible to do line tracking with only two center bits and position tracking with the two outermost bits. With the advice, it is much easier to develop a switch with four cases. For the position tracking part, we have used some counters to track how many crosses the cart has travelled. Therefore, it is necessary to protect the counter from overcounting. One way to do this is to use a Boolean and toggle off when it reaches any cross, and toggle on when it reaches point A or C.

This solution provides a good line tracking without shifting leftwards, while providing a consistent detection over point B using the leftmost bit, where the cart will swap into the inner loop in the second cycle. However, this provides a poor result in detecting the point Y, the middle of the “8” shaped inner loop, preventing the cart change from anti-clockwise to clockwise movement.

After another consultation session with Dr Cheung, we are suggested to check the actual activated bits in point Y, to see whether the leftmost bits given a consistent detection or need to change the state detecting condition. With more testing put on the cart, we found out that using the second outermost bits (i.e., bit 1 and 6) gives a satisfy results that consistently detect the point Y.

As a result, the cart can track the line of the outer part and know when to do the inner part. Since the inner part has a higher curve than the outer part, a higher PWM ratio is sent to the wheels and the cart can follow the inner part when exiting the cross. When the cart reaches the center of the inner part, it should be switched to clockwise mode and switched back when reaching the center again. Also, to protect the counter, a variable is used to delay the detection of the center. Videos of this demonstration are taken, with result of SZE is 30 seconds shape and CHAN is 57 seconds.

1. **The design and difficulties on Demonstration 3**

This part can be established on demonstration 1 product 2. While moving, the cart will detect the distance between the wall and itself and stop before hitting the wall. Since both the motor movement and the ultrasonic sensor reading will make use of timer 1, using the delay function or separating the initialization would be a good way to implement these actions. For example, only initialize the ultrasonic sensor after the cart starts moving.

For triggering the ultrasonic sensor, a 10 us worth of pulse should be sent to the trigger pin of the sensor. This can be done by setting a timer or delay function of 10us intervals to toggle the pin input. Then the reading can be retrieved from the echo pin by input capture and simple calculation of dividing capture value by 58.

There is a minor part that should be considered, which is since the cart starts at different positions, the stopping distance should be different due to the inertia of the cart. Therefore, when the cart starts moving, the stopping distance can be determined by retrieving the reading from the ultrasonic sensor. By tuning the stopping distance, the cart can stop in front of the wall as close as possible. As a result, both of us are able to meet the time and distance requirements of 4 seconds and 7cm.

1. **The design and difficulties on Demonstration 4A**

The code is mostly the same as Demonstration 1 product 1. However, to increase responsiveness, the debounce time decreased from 200ms to 100ms to give faster control and change the two of the buttons on right side of control pad to self-rotate clockwise and anti-clockwise to give faster turn.

While our group faced huge difficulties when recording the demonstration video. As we found out the ball will go out very easily when we hit the ball too hard and stuck at the green zone when we hit the ball too soft. Also, it is very hard to control the direction in straight using the very short edge on the head of the cart. Moreover, we misunderstand the requirement that we cannot touch the ball unless we need to hit the ball, previously we use the cart to stop the ball and then hit back.

After consultation with Dr. Cheung, we were told that we can try using the inclined edge of the cart to hit the ball in a diagonal way, and the cart should go backwards to give more buffer if the ball moves in a non-ideal direction. If those did not help, we were advised to create a soft-hit and hard-hit mode to create consistent ball hitting distance, reducing the randomness on the where the ball will stop.

As a result, we get a decent result on video recording, which uses 34 seconds to do all 6 ball hits and this result meets the full mark requirement.

1. **The design and difficulties on Demonstration 4B**

In this Demonstration, we were asked to do a relay race, where car 1 will go from point A until meeting car 2, then car 2 will follow the designated track until meeting car 1 again, then car 1 will move to point D.

The biggest challenges that we have faced so far are knowing how to leave the track and when to get back on the track. While the position tracking part can use the same code as in demonstration 2, the cart should go forward until reaching the second map's line. It is possible to switch the cart back to line tracking mode when the sensor detects the line, it is much more straightforward to simply just use a delay function to control how long the cart should go straight. Then the cart can detect the distance and stop before hitting the front cart. After the front cart departures, the cart can go forward and rotate for the next detection.

However, we failed to finish this demonstration. For some reason, the cart doesn't seem to work properly after entering the "wait and rotate" mode. It will stop after the limited distance but is unable to restart the wheels after waiting. After receiving some advice from Dr Cheung, the reason might be some previous mode logic is changed during the detection. Still, even rewriting the whole logic, the cart is still unable to do the part correctly. There may be some conflicts between the timers and cause such result.

**Conclusion:**

Even though we meet difficulties in our development on various tasks, especially the line tracking issue, we success to finish Demonstration 1, 2, 3 and 4A.

The development period on Demonstration 2 and 4B is far longer than expected due to the line tracking detection issue. While Demonstration 4B is unable to finished due to software logic bug even after rewriting the whole logic which suggested by Dr Cheung.

- END -