(1) **P542.ino** Write program to determine the execution times of various Arduino instructions as shown in the following Table.

: Arduino UNO R3 : Timer-1 Clock = 16 MHz :				
Sn	Instruction	What does the function do?	Execution Time	
			Cycles	µs
1	pinMode(2, INPUT_PULLUP)	Port Line 2 of Port-D is input with internal pull-up	78	4.875
2	pinMode(4, OUTPUT);	Port Line 4 of Port-D is output	70	4.375
3	DDRD = 0b00001000;		4	0.250
4	digitalWrite(4, HIGH);	Asserts LH at port line 4 of PORTD	59	3.687
5	bitSet(PORTD, 4);	Same as above	4	0.250
6	bitWrite(PORTD, 4, HIGH)	Same as above	4	0.250
7	digitalWrite(4, LOW);	Asserts LL at port line 4 of PORTD	61	3.812
8	bitClear(PORTD, 4);	Same as above	4	0.250
9	bitWrite(PORTD, 4, LOW);	Same as above	4	0.250
10	bitWrite(PORTD, 4, !bitRead(PORTD, 4));	Toggles the logic value of port line 4 of PORTD	8	0.500
11	bitRead(PORTD, 4);	Reads (from internal latch) the value previously written into port line 4 of PORTD	3	0.187
12	digitalRead(2);	Reads logic value from physical pin of port line 2 of PIND.	36	2.250
13	bitRead(PIND, 2);	Same as above	3	0.187
14	int x = analogRead(A0);	Reads ADC value of Ch-0 and assigns to x.	3328	208.000
15	analogReference(DEFAULT);	Sets the Reference value (5V) for ADC	4	0.250
16	analogWrite(3, 0x23);	Direct value for pulse width of PWM	94	5.875
	analogWrite(3, OCR2B);	Same as above, but value comes from a register	153	9.562
17	analogWrite(3, analogRead(A0));	Same as above, but value comes from ADC	3477	217.312

Methodology used: The procedure involves the following tricks: start T1 at 16 MHz clock before the execution of the 'target' instruction; execute the target instruction; stop the T1 after the execution of the target instruction; show the clock accumulated by TCN1 on LCD.