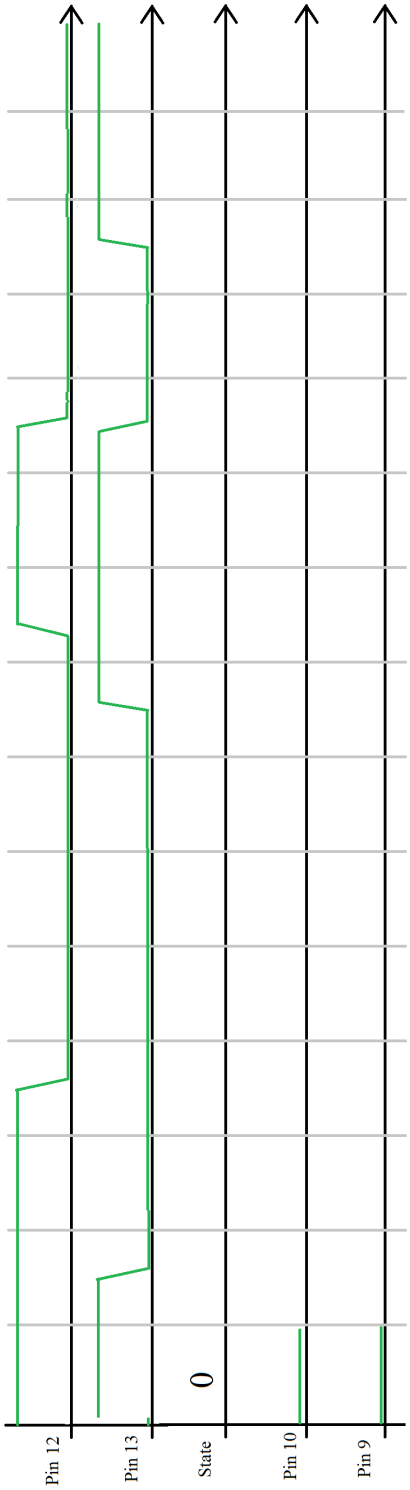
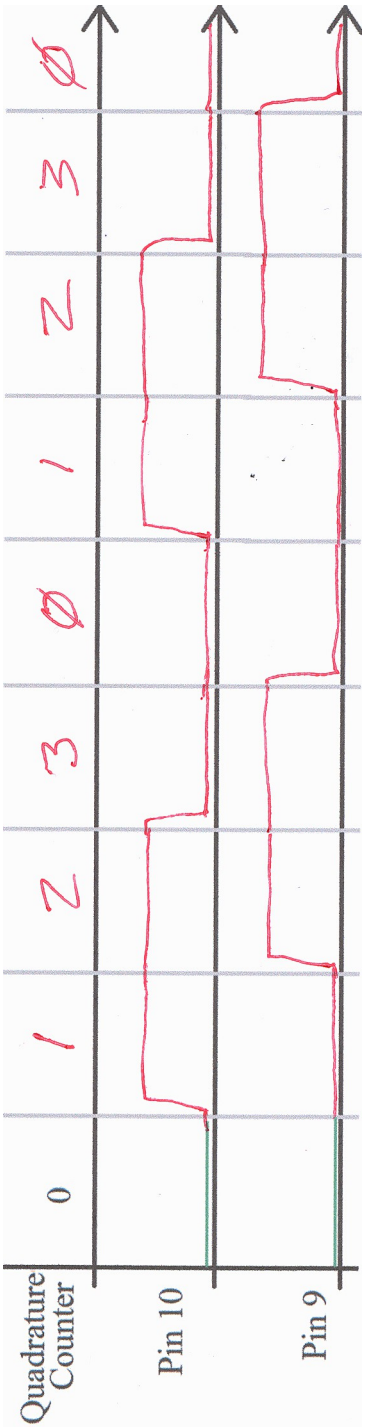
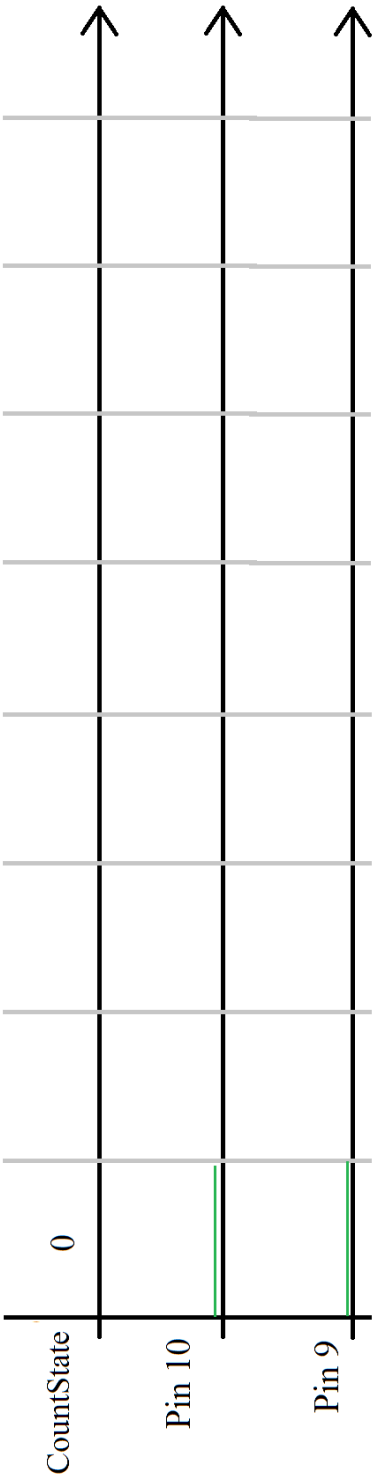
1) Fill in the Timing Diagram for the code in Appendix A. Note there are two copies of the Timing Diagram, in case you need to start over.



2) Fill in the Timing Diagram for the code in Appendix B. Note there are two copies of the Timing Diagram, in case you need to start over.





Appendix A: Code for Part 1.

```
#include <TimerOne.h>

int State = 0;
void Timer1ISR()
{
	switch (State)
	{
	case 0:
		if (bitRead(PINB, 5)) // Pin 13
			State = 1;
		break;
	case 1:
		if (bitRead(PORTB, 1)) // Pin 9
			bitClear(PORTB, 1);
		else
			bitSet(PORTB, 1);
		if (bitRead(PINB, 4)) // Pin 12
			State = 2;
		break;
	case 2:
		if (bitRead(PORTB, 2)) // Pin 10
			bitClear(PORTB, 2);
		else
			bitSet(PORTB, 2);
		if (!bitRead(PINB, 5)   // Pin 13
			&& !bitRead(PINB, 4)) // Pin 12
			State = 0;
		break;
	} // End of state switch

} // End of SerialTimerISR

// put your setup code here, to run once:
void setup()
{
	Timer1.initialize(500);
	Timer1.attachInterrupt(Timer1ISR);
	pinMode(13, INPUT);
	pinMode(12, INPUT);
	pinMode(10, OUTPUT);
	pinMode(9, OUTPUT);
} // End of setup.

// put your main code here, to run repeatedly:
void loop()
{

} // End of loop
```

Appendix B: Code for Part 2.

```
#include <TimerOne.h>

int QuadratureCounter = 0;
void QuadraturePulseTrain()
{
	switch (QuadratureCounter)
	{
	case 0:
		bitSet(PORTB, 2);    // Pin 10
		QuadratureCounter = 1;
		break;
	case 1:
		bitSet(PORTB, 1);    // Pin 9
		QuadratureCounter = 2;
		break;
	case 2:
		bitClear(PORTB, 2);
		QuadratureCounter = 3;
		break;
	case 3:
		bitClear(PORTB, 1);
		QuadratureCounter = 0;
		break;
	}
} // End of QuadraturePulseTrain

// put your setup code here, to run once:
void setup()
{
	Timer1.initialize(500);
	Timer1.attachInterrupt(QuadraturePulseTrain);
	pinMode(10, OUTPUT);
	pinMode(9, OUTPUT);
} // End of setup.

// put your main code here, to run repeatedly:
void loop()
{

} // End of loop
```