GUÍA DEL ESTUDIANTE • SESIÓN 4 DE 4

El Arduino recibe la letra

Terminar el circuito de la práctica 3, entender cómo viaja una letra por un solo cable y agregar el Arduino como segundo receptor, con la pantalla LCD

SESIÓN	DURACIÓN	UNIDADES	PLATAFORMA
4 de 4	4 horas	3, 4, 7 y 8	Arduino Nano o Uno

La idea de hoy, en una frase. Ayer el 74LS151 mandó una letra por **un solo cable** y el 74LS138 la mostró en ocho LEDs. Hoy el Arduino escucha ese mismo cable, arma la letra y la escribe en la pantalla: **el circuito no se desarma, solo se le agregan cables**.

OBJETIVOS

- Terminar y probar el sistema de la práctica 3: contador, emisor y receptor.
- Explicar con sus palabras cómo viaja un byte por un solo cable.
- Leer con el Arduino la cuenta del 74LS193 y el bit del 74LS151, y reconstruir la letra.
- Mostrar la letra en la pantalla LCD por I^2C y en la computadora por el puerto serie.
- Usar el Arduino como emisor: mandar a los LEDs una letra escrita en el teclado.

MATERIALES Y EQUIPO

- El circuito de la práctica 3 armado: 555, 74LS193, 74LS151, 74LS138 y los ocho LEDs.
- Capacitores de 100 µF, 10 µF y 100 nF para el 555.
- Arduino Nano o Arduino Uno con su cable USB, y computadora con el Arduino IDE y la biblioteca `LiquidCrystal I2C` (Frank de Brabander).
- Pantalla LCD 16×2 con módulo I^2C y cuatro cables macho-hembra.
- Seis cables *jumper* macho-macho para unir el Arduino con el circuito.
- MO-MB102 en 5 V y multímetro UT33B+.

PLANIFICACIÓN DE LA SESIÓN

Parte	Actividad	Tiempo
—	Apertura: cómo funciona el circuito de ayer	30 min
A	Terminar y probar la práctica 3 (parte D)	55 min
—	Receso	15 min
B	Conectar el Arduino y la LCD; cargar el programa	35 min
C	El Arduino recibe: reloj lento, medio y rápido	30 min
D	Reto: el Arduino es el emisor	25 min
—	Evaluación en Socrative	25 min
—	Desarmado y cierre del curso	25 min
	Total	**240 min**

NOTA

Quien no trabaje con Arduino termina la parte A y sigue con los retos de la práctica 3: su inicial en ASCII y el reloj elegido con el 74LS157.

Cómo funciona el circuito de ayer

Cuatro bloques, cuatro trabajos

Piense en un salón donde el profesor **pasa lista**. Dice un número, el alumno de ese número contesta «sí» o «no», y alguien lo anota en la pizarra. El circuito hace lo mismo con los ocho bits de una letra:

Bloque	En el salón	Qué hace en el circuito
555	El reloj de la pared	Da un pulso cada cierto tiempo: marca el paso
74LS193	El profesor que dice el número	Suma 1 en cada pulso. Su cuenta $Q_C Q_B Q_A$ (0 a 7) es el **turno**
74LS151	El alumno que contesta	Mira el bit del turno (D_0 a D_7, puestos con puentes) y lo pone en **un solo cable**
74LS138	Quien anota en la pizarra	Con el mismo turno, enciende el LED de ese número, pero **solo si** el bit es 1

El secreto es que emisor y receptor reciben **el mismo turno**: los tres cables Q_A, Q_B y Q_C del contador van a los dos. Así, cuando el 74LS151 manda el bit 5, el 74LS138 sabe que es el bit 5.

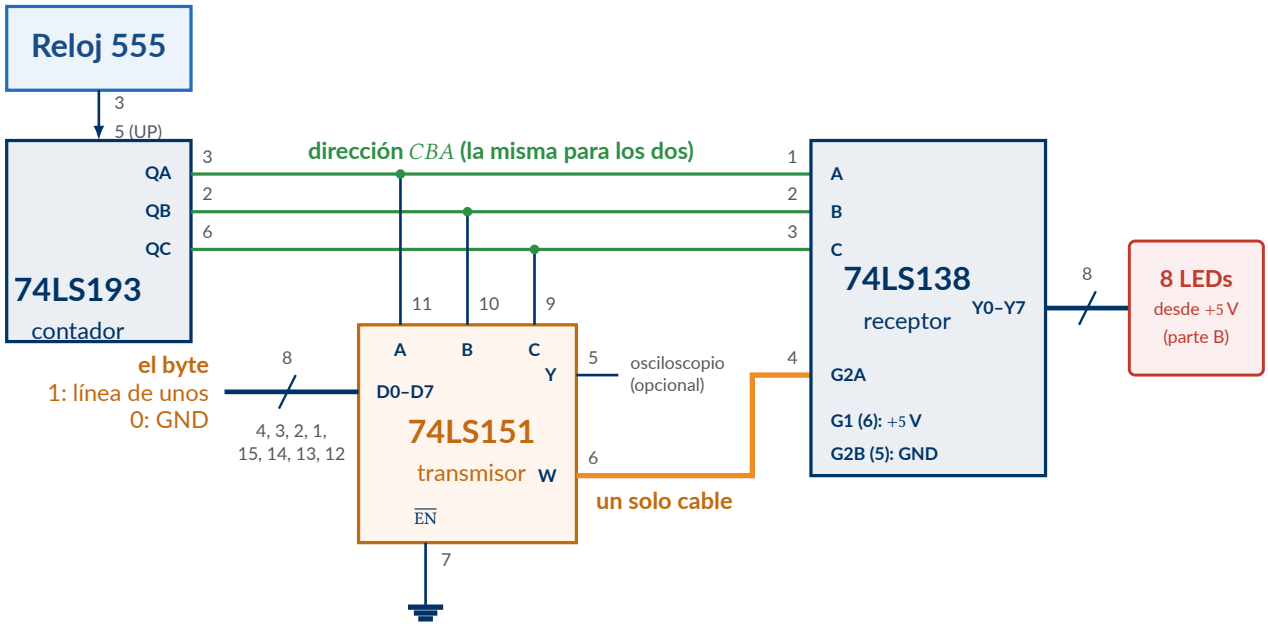


Figura 1. El sistema de la práctica 3. La dirección (verde) va a los dos integrados; el bit viaja por el cable naranja, de `W` a `G2A`.

Por qué el cable sale de W y llega a G2A

`G2A` es una habilitación **activa en bajo**: con 0 el `74LS138` trabaja y con 1 se apaga. `W` es el bit **negado**, así que:

Bit D_k	W = G2A	Qué pasa en el turno k
1	0	El `74LS138` trabaja: se enciende el LED k
0	1	El `74LS138` se apaga: ningún LED

La otra salida, `Y` (terminal 5), es el bit **sin negar** y ayer quedó libre. Hoy el Arduino la usa.

La letra U, turno por turno

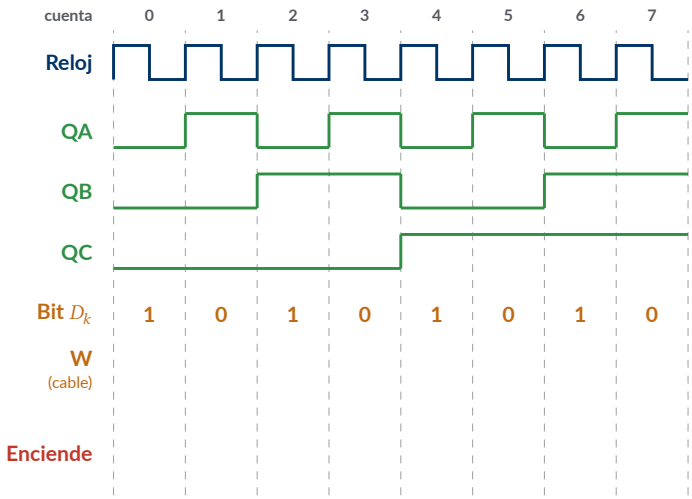


Figura 2. La letra U (0101 0101). En cada turno sale un bit; los LEDs Y0, Y2, Y4 y Y6 se encienden, uno a la vez.

Por qué con 100 nF se ven todos a la vez

C del 555	Frecuencia	Tiempo por turno	Lo que se ve
100 µF	0.69 Hz	1.46 s	La luz salta de un LED a otro; se puede seguir con el dedo
10 µF	6.9 Hz	0.15 s	Parpadeo rápido
100 nF	690 Hz	1.5 ms	Los LEDs de la letra parecen encendidos al mismo tiempo

Con 100 nF cada LED se enciende 86 veces por segundo y el ojo no alcanza a ver que se apaga. Es la misma idea de una pantalla de cine.

Dónde entra el Arduino

El Arduino es un **segundo receptor**. Lee el turno (los mismos Q_A, Q_B y Q_C) y el bit (`Y` del `74LS151`). En cada turno guarda el bit en su lugar; cuando llegan los ocho, en orden, ya tiene la letra y la escribe en la LCD.

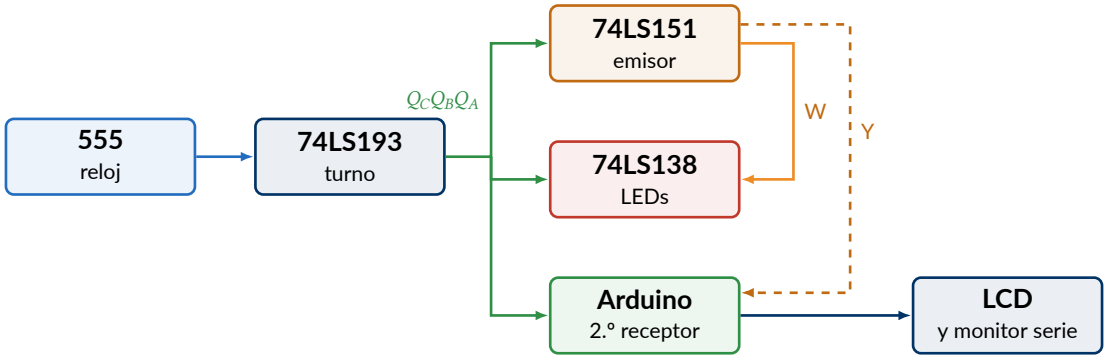


Figura 3. El Arduino recibe los mismos datos que el 74LS138: el turno por tres cables y el bit por uno.

NOTA

El Arduino **solo lee**: sus terminales D4 a D7 son entradas. Si se desconecta, el circuito de la práctica 3 sigue igual.

Procedimiento

NORMAS DE SEGURIDAD

- Apague el MO-MB102 y desconecte el USB antes de mover cables.
- El GND del Arduino va al riel GND del protoboard: sin tierra común, el Arduino no entiende los niveles de los integrados.
- El 5V del Arduino va **solo** a la LCD, nunca a los rieles: dos fuentes de 5 V unidas pelean entre sí.
- Una salida nunca va a otra salida: en el reto, quite el cable de W antes de poner el de D9.

Parte A. Terminar y probar la práctica 3

55 min

Revise el circuito por bloques, en este orden. No pase al siguiente si el anterior falla.

	Bloque	Cómo se prueba	¿Listo?
1	Alimentación	5 V entre 16 y 8 de cada 74LS, y entre 8 y 1 del 555	
2	Reloj (100 µF)	El LED del 555 destella cada 1.5 s	
3	Contador y receptor (parte C)	Con `G2A` a `GND`, la luz recorre Y0 a Y7	
4	Emisor (parte D)	`74LS151` con 16 a +5 V, 8 y 7 a `GND`; 11, 10 y 9 a Q_A, Q_B y Q_C	
5	La letra	Puentes de la U: D7 a D0 = `0101 0101`	
6	El cable del dato	`G2A` (4 del 138) sale de `GND` y va a `W` (6 del 151)	

Tabla 1. La práctica 3 funcionando

Prueba	Resultado
Con 100 µF: LEDs que se encienden en una vuelta, en orden	
Con 100 µF: turnos en que no se enciende ninguno	
Con 100 nF: LEDs que parecen encendidos a la vez	
Con 100 nF y los puentes de la R (`0101 0010`): LEDs encendidos	

RECOMENDACIÓN

Si no enciende ningún LED en la parte D, revise que `G2A` esté en `W` (6) y no en `Y` (5). La guía *Cómo van conectados los circuitos* de la sesión 3 tiene cada cable dibujado.

Parte B. Conectar el Arduino y la LCD

35 min

Ponga el 100 µF en el 555 y la letra U en los puentes. El Arduino puede quedar fuera del protoboard: los cables se meten en un hueco libre de la **misma columna** del terminal que se quiere leer.

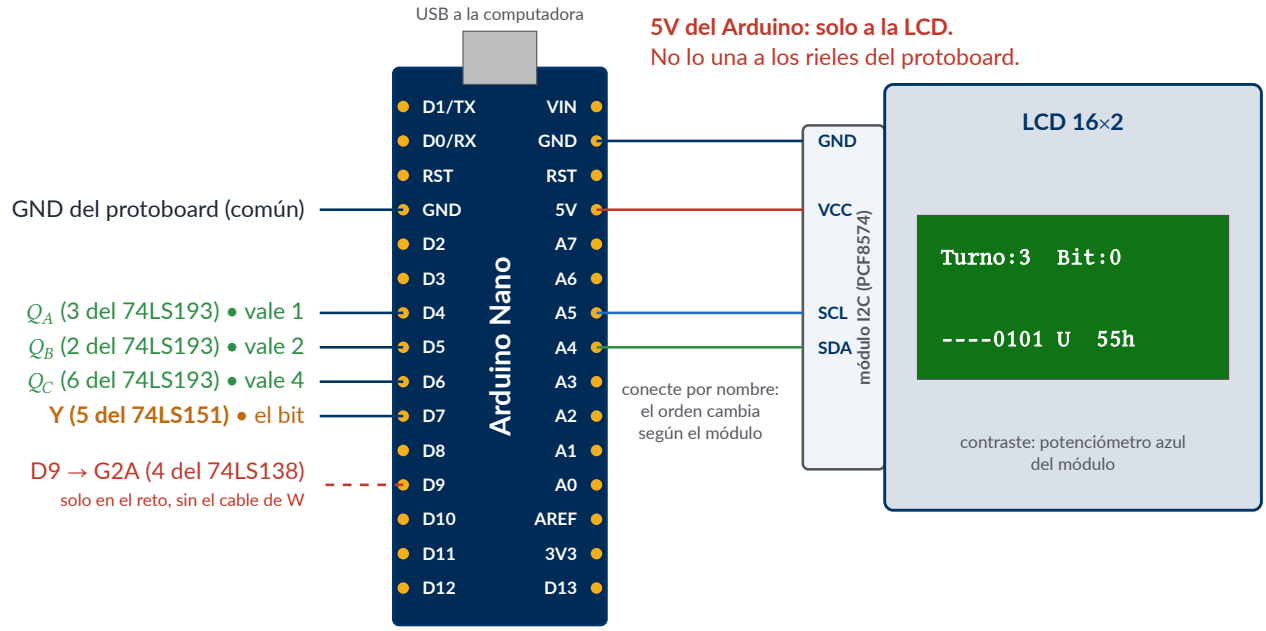


Figura 4. Conexiones del Arduino. El Uno tiene los mismos nombres: `D4` a `D9`, `GND`, `5V`, `A4` y `A5`.

Tabla 2. Cables del Arduino

Arduino	Va a	Qué lee
`GND`	Riel `GND` del protoboard	Tierra común
`D4`	Terminal 3 del `74LS193` (Q_A)	Turno, bit de peso 1
`D5`	Terminal 2 del `74LS193` (Q_B)	Turno, bit de peso 2
`D6`	Terminal 6 del `74LS193` (Q_C)	Turno, bit de peso 4
`D7`	Terminal 5 del `74LS151` (`Y`)	El bit que se transmite
`5V` y `GND`	`VCC` y `GND` de la LCD	—
`A4` y `A5`	`SDA` y `SCL` de la LCD	—

Cargar el programa

1. En el Arduino IDE abra `S04_Receptor_Letra.ino` (carpeta `Arduino` del curso). Si no está, cree un programa nuevo y copie el del final de esta guía.
2. Si falta, instale la biblioteca `LiquidCrystal I2C` (Frank de Brabander) desde el gestor de bibliotecas.
3. En *Herramientas* elija la placa: `Arduino Nano` con el procesador `ATmega328P (Old Bootloader)`, o `Arduino Uno`. Elija el puerto.

4. Pulse **Subir**. Encienda el MO-MB102 y abra el **Monitor serie** a **9600** baudios.

ADVERTENCIA

¿Falla al subir el Nano? Pruebe el procesador `ATmega328P` sin *Old Bootloader*. ¿No aparece el puerto? Instale el controlador `CH340` o cambie de cable USB. ¿La LCD no muestra letras? Gire el potenciómetro azul del módulo; si solo salen cuadros, cambie `0x27` por `0x3F` en el programa y súbalo otra vez.

Parte C. El Arduino recibe la letra

30 min

Reloj lento: 100 µF

La primera línea de la LCD dice el turno y el bit que acaba de llegar. La segunda va llenando la letra de derecha a izquierda (- es un bit que todavía no llega); al final aparecen la letra y su código.



Tabla 3. Turno por turno, con la letra U

Turno	$Q_C Q_B Q_A$	Bit (LCD)	Línea 2 de la LCD
0	000		
1	001		
2	010		
3	011		
7	111		

Pregunta	Respuesta
¿En los turnos en que el bit es 1, qué LED está encendido?	
¿Cuánto tarda en llegar la letra completa? (cronómetro)	
¿Coinciden la LCD y el monitor serie?	

Reloj medio y rápido: 10 µF y 100 nF

Apague, cambie el capacitor y encienda. Con 100 nF los turnos son demasiado rápidos para leerlos: la primera línea muestra la frecuencia del 555 y la segunda, la última letra completa.

Tabla 4. Tres velocidades

C del 555	Lo que muestra la LCD	Letras completas por segundo
10 µF		
100 nF		

1. Con 100 nF, cambie los puentes a la R (`0101 0010`), la A (`0100 0001`) y la L (`0100 1100`). La LCD, el monitor serie y los LEDs deben mostrar la misma letra.
2. Cargue su inicial. Busque su código en la tabla ASCII: la A es 41h, la B 42h, y así sucesivamente.

Mi inicial: _____ Código: _____ Bits D7 … D0: _____

Parte D. Reto: el Arduino es el emisor

25 min

Ahora el Arduino reemplaza a los puentes del **74LS151**: usted escribe una letra en la computadora y aparece en los LEDs. El programa ya lo hace; solo hay que cambiar un cable.

1. Apague el **MO-MB102**. **Quite** el cable de `W` (6 del **74LS151**) a `G2A` (4 del **74LS138**).
2. Ponga un cable de `D9` del Arduino a `G2A` (4 del **74LS138**).
3. Con 100 nF en el 555, encienda. Los LEDs muestran la U: es la letra que el Arduino manda al empezar.
4. En el monitor serie escriba una letra y pulse Enter.

Tabla 5. El Arduino manda la letra

Escribe	Bits D7 … D0	LEDs encendidos
A		
K		
Su inicial		

Pregunta	Respuesta
La LCD sigue mostrando la letra de los puentes. ¿Por qué?	
¿Por qué el Arduino manda el bit negado, igual que W?	

ADVERTENCIA

Nunca deje D9 y W unidos al mismo tiempo: son dos salidas, y si una pone 1 y la otra 0 se hace un cortocircuito. Al terminar, quite el cable de D9 y regrese el de W.

Si algo no funciona

Síntoma	Revise
LCD encendida, sin letras	El contraste (potenciómetro azul del módulo)
LCD con cuadros en la primera línea	La dirección: cambie 0x27 por 0x3F
El turno no cambia o siempre es 0	El reloj; el puente BORRAR del 74LS193 quitado; el GND común
El turno salta (0, 2, 4…) o va en desorden	Los cables de D4, D5 y D6: van a los terminales 3, 2 y 6, en ese orden
La línea 2 nunca se completa	Un cable del turno flojo: si falta un turno, la letra empieza de nuevo
Llega la letra negada: 1010 1010 y ? en lugar de la U	D7 está en W (6); va en Y (5)
La LCD y los LEDs muestran letras distintas	Puentes de D0 a D7 flojos; terminales del 151 cruzados (D0 es el 4 y D7 el 12)
En el reto, los LEDs no cambian	El cable de D9 en el terminal 4 del 138; pulsar Enter después de la letra
Símbolos raros en el monitor serie	La velocidad: 9600 baudios

Preguntas de cierre

1. ¿Por qué el Arduino necesita Q_A, Q_B y Q_C además del bit?

2. ¿Por qué el Arduino lee Y y el 74LS138 recibe W?

3. El puerto USB del Arduino también manda las letras bit por bit, por un solo cable, pero sin los tres cables del turno. ¿Cómo sabe el receptor qué bit es cada uno?

El programa completo

Parte del programa	Qué hace
`leerTurno()`	Suma los pesos: $Q_A \cdot 1 + Q_B \cdot 2 + Q_C \cdot 4$
`loop()`, paso 2	Espera a que cambie el turno: cada pulso del 555 es un turno nuevo
`loop()`, paso 3	Guarda el bit en su lugar con `bitWrite`; con el turno 7 y los ocho bits, la letra está completa
`mostrar()`	Escribe en la LCD el turno y los bits, o la frecuencia si el reloj es rápido
`loop()`, paso 1 y `EMISOR`	Reto: guarda la letra escrita en el monitor serie y pone en `D9` su bit negado

S04_RECEPTOR_LETRA.INO

```
// Universidad Rural de Guatemala - Electronica Digital FE731
// Sesion 4: el Arduino se integra al circuito de la practica 3
//
// En la practica 3 el 74LS151 (emisor) manda una letra bit por bit por un solo
// cable y el 74LS138 (receptor) la muestra en ocho LEDs. Los dos usan la cuenta
// del 74LS193 para saber en que turno van.
//
// Aqui el Arduino es un SEGUNDO RECEPTOR: lee el turno (QA, QB, QC del 74LS193)
// y el bit que sale por Y del 74LS151, arma la letra y la muestra en la LCD
// y en el monitor serie. Ademas mide el reloj del 555.
//
// Reto: el Arduino tambien puede ser el EMISOR. Escriba una letra en el monitor
// serie; el Arduino pone en D9 el bit de ese turno, negado, como W del 74LS151.
// Para usarlo, quite el cable de W (6 del 74LS151) a G2A (4 del 74LS138) y
// ponga uno de D9 a G2A. Nunca deje unidos D9 y W al mismo tiempo.
//
// Conexiones:
//   D4 <- QA del 74LS193 (3)   vale 1     D5 <- QB del 74LS193 (2)   vale 2
//   D6 <- QC del 74LS193 (6)   vale 4     D7 <- Y del 74LS151 (5): el bit
//   D9 -> G2A del 74LS138 (4), solo en el reto del emisor
//   A4 -> SDA de la LCD        A5 -> SCL de la LCD
//   GND del Arduino al GND del protoboard. El 5V del Arduino solo va a la LCD.
//
// Biblioteca: LiquidCrystal I2C (Frank de Brabander).

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

```
28
29  const int QA     = 4;
30  const int QB     = 5;
31  const int QC     = 6;
32  const int BIT_Y  = 7;
33  const int EMISOR = 9;
34
35  LiquidCrystal_I2C lcd(0x27, 16, 2);   // si solo salen cuadros, use 0x3F
36
37  int turnoAntes = -1;
38  byte recibido = 0;              // los bits que van llegando
39  byte llegaron = 0;              // que turnos ya llegaron (un bit por turno)
40  int letra = -1;                 // ultima letra completa (-1: ninguna todavia)
41  int letraAntes = -1;
42  char letraEnviar = 'U';         // la que manda el Arduino por D9 (reto)
43  unsigned long ultimoCambio = 0;  // en microsegundos
44  unsigned long periodo = 0;       // tiempo de un turno = periodo del 555
45  unsigned long ultimaPantalla = 0;
46
47  int leerTurno() {
48    return digitalRead(QA) * 1 + digitalRead(QB) * 2 + digitalRead(QC) * 4;
49  }
50
51  bool esLento() {
52    return periodo > 20000;        // mas de 20 ms por turno: se ve turno a turno
53  }
54
55  void escribirBits(byte valor, byte cuales) {
56    for (int i = 7; i >= 0; i--) {
57      if (bitRead(cuales, i)) {
58        lcd.print(bitRead(valor, i));
59      } else {
60        lcd.print("-");
61      }
62    }
63  }
64
65  void escribirLetra(int valor) {
66    if (valor < 0) {
67      lcd.print("       ");
68      return;
69    }
70    lcd.print(" ");
71    if (valor >= 32 && valor <= 126) {
72      lcd.print((char)valor);
73    } else {
74      lcd.print("?");
75    }
76    lcd.print("  ");
77    if (valor < 16) {
78      lcd.print("0");
79    }
80    lcd.print(valor, HEX);
81    lcd.print("h");
```

```
}

void serieLetra(int valor) {
  Serial.print("Letra recibida: ");
  for (int i = 7; i >= 0; i--) {
    Serial.print(bitRead(valor, i));
    if (i == 4) {
      Serial.print(" ");
    }
  }
  Serial.print(" = ");
  if (valor < 16) {
    Serial.print("0");
  }
  Serial.print(valor, HEX);
  Serial.print("h = '");
  Serial.print((char)valor);
  Serial.println("'");
}

// Linea 1, lento:  "Turno:3  Bit:1"     rapido:  "f = 686 Hz"
// Linea 2:          "0101---- U  55h"   bits D7..D0 (- = todavia no llega)
void mostrar(int turno, int bit) {
  lcd.setCursor(0, 0);
  if (esLento()) {
    lcd.print("Turno:");
    lcd.print(turno);
    lcd.print("  Bit:");
    lcd.print(bit);
    lcd.print("  ");
  } else {
    lcd.print("f = ");
    if (periodo > 0) {
      lcd.print(1000000UL / periodo);
    } else {
      lcd.print("---");
    }
    lcd.print(" Hz      ");
  }
  lcd.setCursor(0, 1);
  if (esLento()) {
    escribirBits(recibido, llegaron);
  } else if (letra >= 0) {
    escribirBits(letra, 0xFF);
  } else {
    lcd.print("--------");
  }
  escribirLetra(letra);
}

void setup() {
  // INPUT_PULLUP ayuda a que el 1 de una salida TTL llegue bien al Arduino
  pinMode(QA, INPUT_PULLUP);
  pinMode(QB, INPUT_PULLUP);
```

```
  pinMode(QC, INPUT_PULLUP);
  pinMode(BIT_Y, INPUT_PULLUP);
  pinMode(EMISOR, OUTPUT);
  digitalWrite(EMISOR, HIGH);    // W en 1: el 74LS138 no enciende nada

  Serial.begin(9600);
  Serial.println("URURAL FE731 - El Arduino recibe la letra");
  Serial.println("Escriba una letra y Enter para que el Arduino la envie por D9");

  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("URURAL  FE731");
  lcd.setCursor(0, 1);
  lcd.print("Receptor serie");
  delay(2000);
  lcd.clear();
}

void loop() {
  // 1. Reto: una letra escrita en el monitor serie es la que envia el Arduino
  if (Serial.available() > 0) {
    char c = Serial.read();
    if (c >= 32 && c <= 126) {
      letraEnviar = c;
      Serial.print("El Arduino envia por D9: '");
      Serial.print(letraEnviar);
      Serial.println("'");
    }
  }

  // 2. Cada pulso del 555 cambia la cuenta del 74LS193: empieza otro turno
  int turno = leerTurno();
  if (turno == turnoAntes) {
    return;
  }
  delayMicroseconds(20);         // espera a que el 74LS151 ponga el bit nuevo
  turno = leerTurno();
  int bit = digitalRead(BIT_Y);

  // Como emisor: el bit de este turno, negado, igual que W del 74LS151
  digitalWrite(EMISOR, bitRead(letraEnviar, turno) ? LOW : HIGH);

  unsigned long ahora = micros();
  if (ultimoCambio != 0) {
    periodo = ahora - ultimoCambio;
  }
  ultimoCambio = ahora;

  // 3. Guardar el bit. Si se salto un turno, la letra empieza de nuevo
  if (turno == 0 || turno != (turnoAntes + 1) % 8) {
    recibido = 0;
    llegaron = 0;
  }
```

```
  bitWrite(recibido, turno, bit);
  bitSet(llegaron, turno);
  turnoAntes = turno;

  if (turno == 7 && llegaron == 0xFF) {
    letra = recibido;            // llegaron los 8 bits, en orden
  }

  // 4. Mostrar. Con reloj lento, cada turno; con reloj rapido, 3 veces por segundo
  if (esLento()) {
    Serial.print("Turno ");
    Serial.print(turno);
    Serial.print("  (CBA = ");
    Serial.print(bitRead(turno, 2));
    Serial.print(bitRead(turno, 1));
    Serial.print(bitRead(turno, 0));
    Serial.print(")  bit = ");
    Serial.println(bit);
    if (turno == 7 && letra >= 0) {
      serieLetra(letra);
    }
    mostrar(turno, bit);
  } else {
    if (letra != letraAntes && letra >= 0) {
      serieLetra(letra);
      ultimoCambio = 0;
    }
    if (millis() - ultimaPantalla > 300) {
      mostrar(turno, bit);
      ultimaPantalla = millis();
      ultimoCambio = 0;          // la pantalla tarda mas que un turno: no medir
    }
  }
  letraAntes = letra;
}
```

EVALUACIÓN EN LÍNEA (SOCRATIVE)

Al terminar la práctica se aplicará una evaluación breve en **Socrative**. Las preguntas combinan la teoría de la clase magistral en línea con lo que se acaba de hacer en el laboratorio.
El instructor indicará el código de la sala. Anótelo aquí, junto con cualquier indicación del día.

Código de sala / enlace

Espacio para la evaluación

Reservado para Socrative: el instructor colocará aquí el acceso o las indicaciones