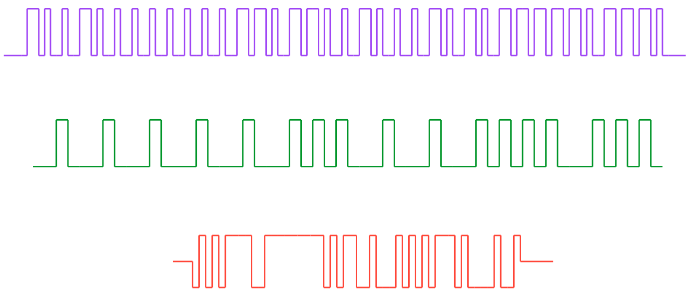


Data Streams

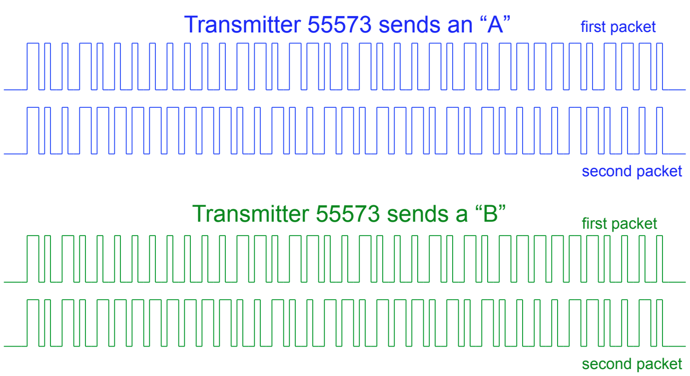
Scope Use; IR remotes
Flip-Flops, and other logic for lab

The H-ITT Transmitter Signal

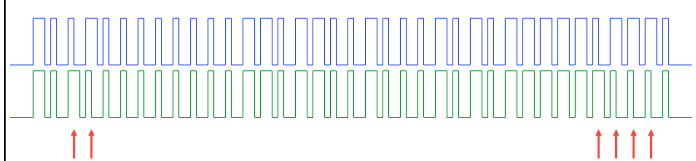
- When you click the H-ITT transmitter button:
  - A–E: LED indicator comes on, and at same time, TWO bursts of infrared light come out: LED stays on even after transmission stops, until button is released
  - * button: on *release* of button, LED flashes and two infrared bursts are sent

  - bursts last 53 milliseconds, are 9 ms apart, and have a bit-period of about 0.5 ms (about 2000 bits per second)
- Let's look at it on scope…

H-ITT Transmitter Protocol



Comparison of A & B first packets



Differences are minor, showing up only near beginning & end

We will represent "high" states (light on) as 1's, and lows (off) as 0's

Notice standard widths: choices are single- or double-width
(both for the zeros and the ones)

Decoding the A signal

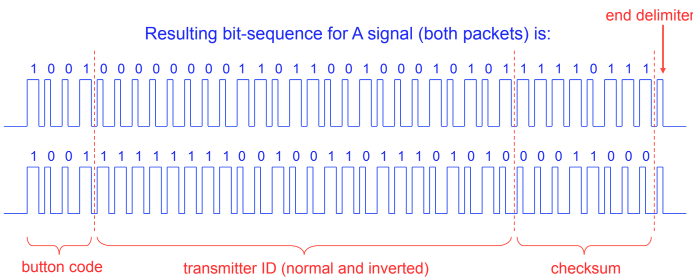
Sequence starts out: 01101001001101001001001001…

Notice the 01 delimiters: 01101001001101001001001001…

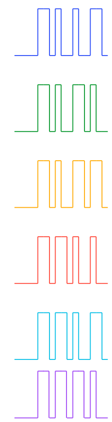
This gives the signal its choppy appearance (never see 3 1's or 0's in a row)

Actual data appears between delimiters: 1's look fat, 0's look skinny

Resulting bit-sequence for A signal (both packets) is:



The different buttons: first four bits



A 1001 → 001 → 1 (first bit always 1)

B 1010 → 010 → 2

C 1011 → 011 → 3

D 1100 → 100 → 4

E 1101 → 101 → 5

<< 1110 → 110 → 6

The Transmitter ID bytes

- Transmitter number is binary-coded in the usual sense:

32768 16384 8192 4096 2048 1024 512 256 128 64 32 16 8 4 2 1

0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 1 0 0 0 1 0 1 0 1

- Sum is:
  - 32768 + 16384 + 4096 + 2048 + 256 + 16 + 4 + 1 = 55573
  - this exactly the number pasted behind the battery
- Second packet inverts all the bits to ensure data integrity

What's with the Checksum?

1 0 0 1 | 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 1 0 0 0 1 0 1 0 1 | 1 1 1 1 0 1 1 1

button code — transmitter ID (normal first-packet version) — checksum

Break data into chunks of 8 bits (bytes) and add up:

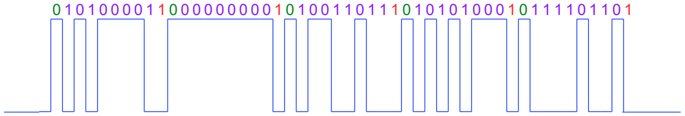
```
    1001
00000000
11011001
00010101
--------
11110111
```

Checksums provide a "sanity check" on the data integrity

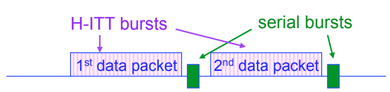
UCSD: Physics 121; 2012

A look at the H-ITT Serial Datastream

E-button on H-ITT (first of two packets):



- Serial datastream looks a lot different
  - this one allows many zeros or ones in a row
  - delimiters (called start bit and stop bit) bracket 8-bit data (1 byte)
  - in this case, 0's are positive voltage, 1's are negative (backwards!)
  - happens much faster than IR: in this case 19,200 bits/sec (baud)
- Packet breakdown:
  - first packet: button number (5 → E), with LSB first: 101000
  - next three packets are ID, also LSB first within each
  - last packet is checksum type of verification



Winter 2012 9

UCSD: Physics 121; 2012

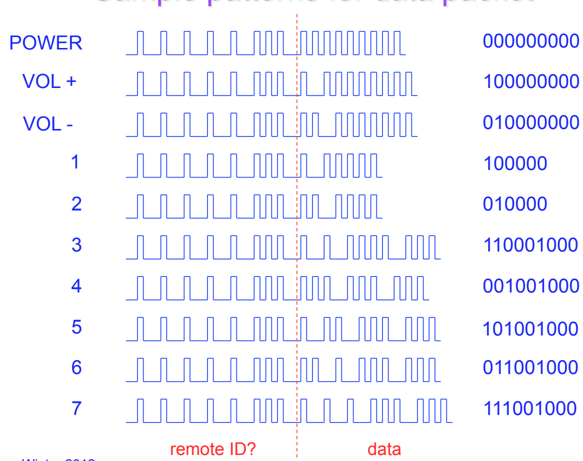
Stereo Remote Control

- Similar to H-ITT transmitters in principle:
  - bursts of infrared light carrying digital information
  - punctuated by delimiters so no long sequences of 1's or 0's
- Key differences:
  - signal initiated by a WAKE UP! constant-on burst
  - pattern that follows is repeated indefinitely until button is released
    - I can never get fewer than three packets…
  - packet is variable in length depending on button

Winter 2012 10

UCSD: Physics 121; 2012

Sample patterns for data packet

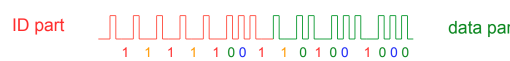


Winter 2012 11

UCSD: Physics 121; 2012

A Different Code…

- The radio remote uses a different scheme:
  - does not use the 01 delimiters like H-ITT did
  - instead, uses 10 to represent zero, and 1000 to represent 1
  - sequence for the 5 button is:
    - 100010001000100010001010100010001010001010…

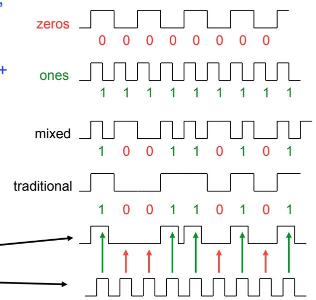


- in data part, least significant bit (LSB) is first
- so the number part of "5" is 101001000 → 1010
- least significant digit is first, so reverse order for more familiar form: 0101 = 5

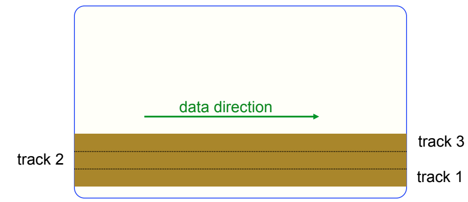
Winter 2012 12

Magstripe Idea

- On magnetic stripe, N-S junctions eat their own magnetic flux lines, but N-N or S-S present external flux lines of opposite direction
  - pattern of N-N and S-S creates + and - transitions
  - zero represented by long period
  - one represented by short period
  - zeros look fat; ones thin (sign irrelevant)
  - two streams are produced from this:
    - a data stream
    - a clock
  - data valid when clock high



Magstripe Geometry

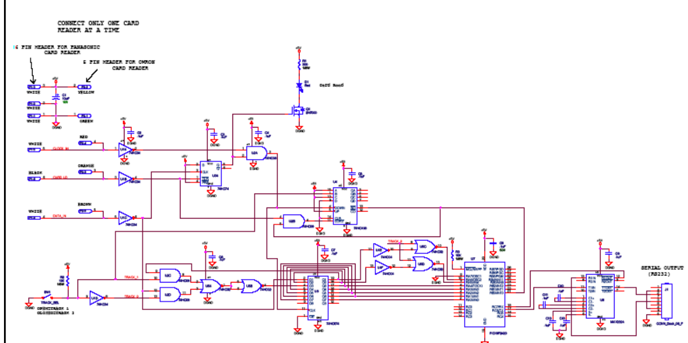


- There are up to three tracks of data
  - Tracks 1 and 3 typically higher-density (7-bit) alpha-numeric data
  - Track 2 typically lower-density (5-bit) numeric data
  - Track 2 used on almost every card; track 1 often, track 3 seldom

Track 2 Character Code

--Data Bits--				Parity		
b1	b2	b3	b4	b5	Character	Function
0	0	0	0	1	0 (0H)	Data
1	0	0	0	0	1 (1H)	"
0	1	0	0	0	2 (2H)	"
1	1	0	0	1	3 (3H)	"
0	0	1	0	0	4 (4H)	"
1	0	1	0	1	5 (5H)	"
0	1	1	0	1	6 (6H)	"
1	1	1	0	0	7 (7H)	"
0	0	0	1	0	8 (8H)	"
1	0	0	1	1	9 (9H)	"
0	1	0	1	1	: (AH)	Control
1	1	0	1	0	; (BH)	Start Sentinel
0	0	1	1	1	< (CH)	Control
1	0	1	1	0	= (DH)	Field Separator
0	1	1	1	0	> (EH)	Control
1	1	1	1	1	? (FH)	End Sentinel

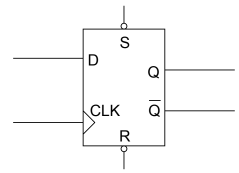
Mag-stripe Circuit Schematic



Unfamiliar bits

- You've seen many of the elements in the circuit by now
  - AND, NOR, NOT logic
- New pieces are down-counter, flip-flops (and latch version thereof)
- Also have a PIC microcontroller and an RS-232 level converter

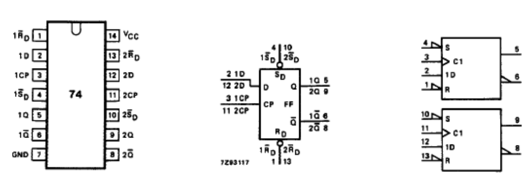
The D-type flip-flop



- Rules:
  - whatever logic level sits at D input, it gets transferred to Q (and inverse logic to Q-bar) when the clock sees a positive edge
  - if R (reset) goes low, Q goes to zero (Q-bar to logic high)
  - if S (set) goes low, Q goes to logic high (Q-bar to zero)
  - holds Q, Q-bar values as long as S, R, high, no matter what happens to D input, provided no clock edge

74HC74: Dual D-type flop

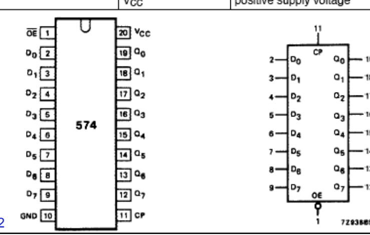
PIN NO.	SYMBOL	NAME AND FUNCTION
1, 13	$1\overline{R}_D$, $2\overline{R}_D$	asynchronous reset-direct input (active LOW)
2, 12	1D, 2D	data inputs
3, 11	1CP, 2CP	clock input (LOW-to-HIGH, edge-triggered)
4, 10	$1\overline{S}_D$, $2\overline{S}_D$	asynchronous set-direct input (active LOW)
5, 9	1Q, 2Q	true flip-flop outputs
6, 8	$1\overline{Q}$, $2\overline{Q}$	complement flip-flop outputs
7	GND	ground (0 V)
14	V_{CC}	positive supply voltage



74HC574 Octal D-flop

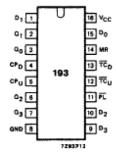
- Just 8 D-type flops with same clock, no set/reset
- Often called a "latch" preserves data input at clock

PIN NO.	SYMBOL	NAME AND FUNCTION
1	$\overline{OE}$	3-state output enable input (active LOW)
2, 3, 4, 5, 6, 7, 8, 9	D_0 to D_7	data inputs
10	GND	ground (0 V)
11	CP	clock input (LOW-to-HIGH, edge-triggered)
19, 18, 17, 16, 15, 14, 13, 12	Q_0 to Q_7	3-state flip-flop outputs
20	V_{CC}	positive supply voltage



74HC193 Counter

PIN NO.	SYMBOL	NAME AND FUNCTION
3, 2, 6, 7	Q_0 to Q_3	flip-flop outputs
4	CP_D	count down clock input[1]
5	CP_U	count up clock input[1]
8	GND	ground (0 V)
11	$\overline{PL}$	asynchronous parallel load input (active LOW)
12	$\overline{TC}_U$	terminal count up (carry) output (active LOW)
13	$\overline{TC}_D$	terminal count down (borrow) output (active LOW)
14	MR	asynchronous master reset input (active HIGH)
15, 1, 10, 9	D_0 to D_3	data inputs
16	V_{CC}	positive supply voltage

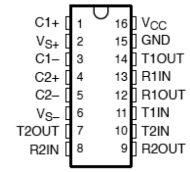


- Load input; count up or down; alert when at end

OPERATING MODE	INPUTS								OUTPUTS					
	MR	$\overline{PL}$	CP_U	CP_D	D_0	D_1	D_2	D_3	Q_0	Q_1	Q_2	Q_3	$\overline{TC}_U$	$\overline{TC}_D$
reset (clear)	H	X	X	L	X	X	X	X	L	L	L	L	H	L
	H	X	X	H	X	X	X	X	L	L	L	L	H	H
parallel load	L	L	X	L	L	L	L	L	L	L	L	L	H	L
	L	L	X	H	L	L	L	L	L	L	L	L	H	H
	L	L	L	X	H	H	H	H	H	H	H	H	L	H
	L	L	H	X	H	H	H	H	H	H	H	H	H	H
count up	L	H	↑	H	X	X	X	X	count up				H[2]	H
count down	L	H	H	↑	X	X	X	X	count down				H	H[3]

MAX232 RS-232 Level-shifter

- Powered by 5V
- uses charge-pump capacitors to generate ±12V for RS-232 levels
- Provides 2 channels of communication
  - transmit and receive for each



Reading

- For magnetic stripe stuff, see:
  - http://en.wikipedia.org/wiki/Magnetic_stripe_card
  - http://money.howstuffworks.com/question503.htm
  - http://stripesnoop.sourceforge.net/faq.html
  - http://stripesnoop.sourceforge.net/devel/phrack37.txt
- See data sheets linked via the Lab 9 Parts List