What’s Inside the Buffer?

This device always “drives” either high or low.

Current is a function of pin voltage

Never High Impedence ‘Z’

Note: this one inverts the signal, but its just an example…
I/O Port?

Write Reg

Read Reg

Pin

bus
Output driver can be disconnected from the pin so that input buffer can sense only the input signal.

This kind of bi-directional port requires a direction control register (SFR) for each bit of output (like StrongArm...).
The 8051 (always has to be different)

Eliminate the need for configuration bits by making outputs that can only drive strongly low (sink). There are three kinds of pins on the 8051 (of course)
- No pull up
- Weak pull up
- Weak pull up with momentary strong pullup

To use a input pin, set output value to 1 (weak or no pullup). External signals just have to overpower the weak pull up (low resistance to ground).
As output, will go from 0 to 1 slowly unless you add an external pullup

Data sheet doesn’t spec the resistance of the pull up, but it specs the Amount of current that will result in a given voltage at the pin. For Example, in Ports 1,2,3 Ioh = -25uA at .75Vcc.
I/O Ports (see 2-40 of AT89 Hardware Desc.)

It can drive a 0, but not a 1 (sinks current).

No direction bit required, just write a 1 to the port register to use as input pin.
Application: External I/O bus

Q1) How can a processor detect a collision?

Communication bus:
Each processor tries to send data, but detects collision. If collision, then stop transmitting.
Summary

- **Port 0:**
  used as address bus for external address/data bus. Uses active pullup in this mode. Fast
  Can use as GPIO. Must use external pullup. Pullup size is power/speed tradeoff, can sink up to 3.2mA while maintaining a logic zero output.

- **Port 1 and 3:**
  GPIO only. External pullups are optional. Power/speed tradeoff, can sink up to 1.6mA while maintaining a logic zero output

- **Port 2:**
  Also used for external address bus. Has active and passive internal pullups. External pullups are optional in GPIO mode, up to 1.6mA.
Example Problem (See elec. specs in AT89C55)

Open = 0
Closed = 1

1) As big as possible!

According to Data sheet:
Processor reads a zero if Vpin < .2Vcc - .3 = 0.7V
I_{low} (port 1) is .45V at 50uA. So what is max R?
\((.45/50e-6) = 9\text{Kohms}\)

So the switch resistor better be smaller than 9Kohms. 4.7K is a good choice. 2.7 is okay but higher power!
Careful w/ Coils (motors, valves, etc)

Steady state on current: $V_{cc}/R$
$V_{ds} \sim 0$ ($R_{ds} \sim 4\text{mOhm}$)
But, when we try to turn off the Mosfet quickly, what happens?
- $R_{ds}$ goes up quickly, but $I_{ds}$ drops slowly
- If $R_{ds}$ becomes 1K, then $V_{ds}$ becomes 100V
- And instantaneous power becomes 10W

Diagram:
- Current limiter
  - $R = 50\text{Ohms}$
  - $I = 0.1\text{A}$
- Coil (L)
- MOSFET Switch
- Graph showing $V_{ds}$ waveform
Absolute Maximum Current Ratings

- Is there a limit to how much current we can sink if we don’t care about what happens to our logic levels?
- How much current did you sink in Lab 1?
- What was the voltage at the pin?
- What is the maximum legal speaker power on port 0?
  
  \[10\text{mA} \times 8\text{ohms} = 0.8\text{mW} \ldots \text{we want } 200\text{mW}!\]
Design Meeting

How are we going to make louder more interesting sounds??

Sample rate v. Frequency

How can we generate a complex tone (multiple notes simultaneously)

How should we use timers/interrupts to do siren?

What is midi?

Spreadsheet for tone generation
Using single bit tones

- Two tones generated by two single bit outputs.
- How do we add tone1 and tone2.
Our Version

- Objective: convert number of bits to current

Diagram:
- 8051
- AMP (B to I)
- 5V
Does this sum? \( I = Bx \)

What must \( R \) be to protect the processor?

What is the worst case (1, 2, or 3 bits)?

How much power are we putting through the speaker in the worst case?

3 bits is worse (Higher \( R \) constraint)

\[
(26\text{mA})^2 \times 8 = 5.4\text{mW}
\]
Another Idea

1. Use a current amplifier (PNP Transistor)
   \[ I_{ce} = \beta I_b \text{ (assume } \beta = 100) \]
   Assume \( V_{be} = 0.7 \text{V} \) when “on”
   Assume \( V_{ce} = 1 \text{V} \) when “on”
   Assume \( \text{t1} = 0 \text{V} \)

   Pick \( R_c \) to protect the speaker
   Pick \( R_b \) to protect the processor

2. While turning on the transistor

   \[ I_s = \left(\frac{0.2}{8}\right)^{0.5}/3 = \sim 50 \text{mA} \]

   \[ R_c: 5 - (50mA \times 8) - V_{ce} - (50mA \times R_c) = 0 \]
   so: \( R_c = (5 - 1 - 0.4)/0.05 = 72 \text{ohms} \)

   \[ R_b: V_b/1 \text{mA} = [5 - (8 \times 0.05) - .7]/1 \text{mA} = 3.9K! \]
Final Circuit Design

Size R to match your speaker and to
Stay within the current limitations of the Processor.
Ideal Solution

Digital to Analog Converter

Speaker cares about current, not voltage
What is algorithm to superimpose 1KHz tone with 500Hz tone
With a sampling rate of 10KHz
Software Summation

- Add waveforms to get multiple tones (think current through speaker, not voltage)

Note that lower frequency is smoother for a given sample rate
What should we do in the next lab?
**Synthesizer Algorithm**

1. Let \( \sin[] \) be a look up table with 256 entries (1 complete cycle).
2. Every .1ms (10KHz)
   - \( P2 = \sin[t1] + \sin[t2] + \sin[t3] \ldots \)
   - For 1KHz, \( t1 += 256/10 \) is this hard? how do we implement this?
   - For 500Hz, \( t2 += 256/20 \)
   - At 8-bit resolution we can vary output from 0 to 255. Hi frequencies are smoother.

3. Can Compute arbitrary waveforms (not just tone summations)
Summary

- Review of basic Electronics
  - Capacitors
  - Inductors
  - Bipolar Transistors
  - MOS transistors
- Review of 8051 I/O configuration