24EC504 – Unit II – Part 2: Sensor Interfacing and Actuator Control

1. Sensor Interfacing Techniques

1.1 I/O Interfacing Basics

I/O interfacing enables communication between the processor and peripheral devices (memory, keyboard, sensors and so on), bridging gaps in data transfer speed, format and control.

Key functions of an I/O interfacing unit:

- Speed synchronization between the fast processor and slower peripherals.
- Device selection with the right input/output.
- Control and timing signals for the peripheral's operating sequence.
- Data buffering.
- Error identification and handling.
- Format conversion (parallel ↔ serial, digital ↔ analog).

The interfacing unit has a data bus buffer, RD/WR control logic, Port A, Port B and control and status registers. Two address lines (A0, A1) with RD and IO/M select one of four registers: Port A, Port B, Control register or Status register. It is used in peripheral connectivity, industrial control, multimedia, data acquisition and networking.

1.2 Serial vs Parallel Communication

Parameter	Serial	Parallel
Data transfer	Bit by bit, sequential	Multiple bits simultaneously
Lines required	Single line	Multiple lines
Speed	Slower	Faster over short distances
Best distance	Long distances	Short distances
Duplex	Full duplex (bidirectional)	Half duplex (unidirectional)
Reliability	More reliable, minimal interference	Less reliable, prone to interference
Typical use	External devices, USB, networking	Internal CPU / memory interfaces

Serial Communication Interface (SCI) sends bits over a single channel using separate transmit and receive lines in NRZ (Non-Return-to-Zero) format. Three types: simplex (one way), half-duplex (both ways, not simultaneously) and full-duplex (simultaneous two-way).

- *Advantages:* easy to design and use, cost-effective, versatile, good for long distances, provides error identification.

- *Disadvantages:* slower, limited to about 8 bits at a time, needs precise timing to avoid bit-sync problems, only detects errors and cannot correct them.

Parallel Communication Interface (PCI) sends several bits at once, bridging the microcontroller and peripheral or memory devices at high speed. Features: higher throughput, auto-detection and plug-and-play, distinct clock synchronization on all interfaces, support for concurrent processors, memories and I/O. Evolution: conventional PCI → PCI-X → PCIe (point-to-point serial lanes that scale bandwidth while staying backward-compatible at software level).

- *Advantages:* faster transfer, lower delay, scalable, wide compatibility.
- *Limitations:* sensitive to noise, limited cable length, higher complexity and cost.

1.3 Direct (Sensor) Interfacing

General scheme: Sensor → Signal conditioning and processing circuit (amplified, filtered) → ADC → Embedded processor.

A **direct interface circuit** connects the sensor straight to the digital system with no signal-conditioning circuit.

Advantages: simple, fewer components, low power, low cost, good accuracy and resolution.

Operating principle:

- The digital system excites the sensor to get a time-modulated signal and measures it with its digital timer.
- For resistive sensors, it measures the charging or discharging time of an RC circuit through the sensor resistance to a threshold voltage.

Microcontroller blocks used: CPU, memory and peripherals (I/O ports with embedded Schmitt trigger, and a digital timer). The timer increments on each rising edge of a high-frequency reference oscillator (period Ts, tens of MHz) and stops when V_TL is crossed. Example families: PIC16 (Microchip), MSP430 (Texas Instruments), AVR (Atmel).

Single resistive sensor with three-signal auto-calibration

Extra resistor Rs keeps the discharge current within the port pin's sinking limit; Ri improves rejection of supply noise at the cost of longer charging time. Three sequential measurements make the result insensitive to circuit errors:

1. **Sensor measurement:** discharge time through Rx (Tx).
2. **Reference measurement:** discharge time through a known Rref (Tref).
3. **Offset measurement:** discharge time through the port pin's internal resistance Rp (Toff).

Differential and bridge configurations

- **Differential:** measures Rx1, Rx2 and offset.

$$x_R = \frac{T_1 - T_2}{T_1 + T_2 - 2T_{off}}, \quad R_{x1} = R_0(1 \pm x_R), \quad R_{x2} = R_0(1 \mp x_R)$$

Estimating the measurand from x_R is better than from Rx1 or Rx2 alone because they can be altered by temperature, and x_R needs no reference resistor.

- **Bridge topology:** four discharge times (T1, T2, T3 and Toff, from series/parallel combinations of four sensors).

$$x_R = \frac{T_1 - T_3}{2(T_2 - T_{off})}$$

Worked examples

Example 1. R = 1 kΩ, C = 5 μF, capacitor charged to 5 V. Time to discharge to 1 V:

$T_d = RC \cdot \ln(5/1) = (1\times10^{3})(5\times10^{-6})(1.609) \approx 8.05$ ms

Example 2. Tx = 10 μs, Tref = 4 μs, Toff = 2 μs, Rref = 1 kΩ:

Rx = 1 kΩ × (10 − 2)/(4 − 2) = 4 kΩ

Uncertainty sources

- **Mismatch of internal resistances:** Rp is assumed equal for all port pins; mismatch causes error, negligible if small compared with the sensor's nominal resistance.
- **Quantization:** the stop point has an error of 0 to Ts; reduce with a high crystal frequency or high capacitance (at the cost of longer measurement).
- **Noise:** from the supply or microcontroller activity; reduce by averaging if the noise is random.

Resistive sensor arrays and FPGA interface

Arrays give spatial patterns (pressure maps from tactile sensors, thermal imaging, electronic nose; uses include electronic skin, touch screens and prostheses). A microcontroller may lack resources for direct interfacing, so an **FPGA** is a good alternative.

FPGA features: many I/O pins, reconfigurable hardware, parallel processing, high real-time performance, and low-cost rapid prototyping. An FPGA is made of logic cells with local memory, joined through switches to routing channels (switch matrix, configurable logic blocks, I/O banks).

Array read-out: capacitors are first charged (CL pins = 1, I/O pins = HZ). A row is selected by setting its I/O pins to 0; capacitors discharge through that row's sensing resistances and the column pins (CL set to HZ) monitor the voltages. A set of timers starts at the beginning of discharge and stops when V_TL is reached, so a whole row is read in parallel; the process repeats for other rows.

2. Actuator Control

2.1 What is an Actuator?

An actuator converts an electrical control signal into physical motion or action (mechanical, thermal or otherwise). It is the output of a control system, as a sensor is the input.

Microcontrollers only generate low-power control signals, so actuators bridge information processing and physical action. Examples: a motor turning a wheel, a valve opening, an LED, a buzzer, a heater.

Basic control loop: Sensor → Microcontroller → Actuator → Physical action.

Classification by energy source	Principle	Examples
Electrical	Current produces motion through magnetic fields	DC motor, servo, stepper, solenoid
Pneumatic	Compressed air drives pistons	Air cylinders, valves
Hydraulic	Pressurized fluid gives high-force motion	Hydraulic pistons, lifts
Thermal	Heat causes expansion/contraction	Bimetallic strips, shape-memory alloys

- **Linear actuators** give straight-line motion: solenoids, linear motors, pneumatic/hydraulic cylinders.
- **Rotary actuators** give rotation: DC, servo and stepper motors.

2.2 Common Electrical Actuators

Actuator	Key behaviour
DC motor	Continuous rotation; speed by voltage/PWM, direction by polarity
Servo motor	Precise angular position (typically 0–180°) using PWM and feedback
Stepper motor	Rotates in fixed steps; precise position without feedback
Solenoid	Linear push/pull; used in locks, valves, relays
Relay	Electromechanical switch for high-power loads

2.3 How a Microcontroller Controls Actuators

GPIO pins output only low voltage and current (3.3–5 V, mA range), not enough for most actuators, so a driver or interface circuit sits in between.

Control type	Behaviour	Example
Digital	ON/OFF switching	Relay, solenoid
PWM	Variable speed/position	DC motor speed, servo angle
Sequenced signals	Ordered step pulses	Stepper stepping

Driver	Role
Transistor (BJT/MOSFET)	Switch for low/medium current loads
Relay module	Isolates and switches high-power AC/DC loads
Motor driver IC (L293D, L298N)	Controls direction and speed
Optocoupler	Electrical isolation between control and power sides

Interfacing examples:

- **DC motor:** Microcontroller → motor driver IC → motor. PWM sets speed, IN1/IN2 set direction, and a flyback diode protects against back-EMF.
- **Servo motor:** three wires (VCC, GND, control PWM). A pulse of 1–2 ms within a 20 ms period sets the angle. Use an external supply for larger servos.
- **Relay module:** Microcontroller → transistor switch → relay → load (lamps, pumps, heaters). Optocoupler-based modules give safe isolation.
- **Stepper motor:** Microcontroller → driver IC (for example A4988) → stepper. A sequence of pulses controls coil order for precise positioning (CNC machines, 3D printers, robotic arms).

Design considerations: match the driver to the actuator's voltage and current; isolate with optocouplers or relays; use flyback diodes on inductive loads; keep actuator power separate from the logic supply; use the right signal type (digital, PWM or pulse sequence).

Real-world applications: robotics (arms, mobile robots), industrial automation (conveyors, pick-and-place), automotive (power windows, throttle, ABS valves), biomedical (infusion pumps, prosthetic limbs, ventilator valves), home automation (smart locks, curtains, HVAC dampers) and consumer electronics (camera autofocus, vibration motors).

2.4 Smart Material Actuators

Material	Working	Materials	Applications
Piezoelectric	AC voltage across the crystal makes it expand and vibrate; the reverse effect turns vibration into AC voltage (sensors, energy harvesters)	Quartz, $BaTiO_3$, PZT, AlN	Vibration/acceleration sensors; inkjet printers, micro-scanning, image stabilization, autofocus, biomedical devices
Magnetostrictive	A magnetic field changes the material's dimensions (up to about 2000 microstrain); a rod inside a coil elongates in proportion to field intensity	Cobalt, Nickel; Terfenol-D, Fe-Al, cobalt	Vibration control of machine tools, ships, helicopter blades, aircraft; micropumps; acoustics

Material	Working	Materials	Applications
		ferrite	
Shape memory alloy (SMA)	Phase change with temperature: deforms easily when cool, regains its programmed shape when heated (often by current); Austenite/Martensite states with hysteresis	Nitinol (Ni-Ti), Ag-Cd, Cu-Al-Ni	Prosthetic hand, endoscope, active bending catheter, automotive, aerospace
Metamaterial	Artificially structured; the unit-cell structure sets its properties, and permittivity and/or permeability can be negative; typically gold or silver on a dielectric	–	Cloaking, microwave antennas and filters, ultrathin absorbers, THz sensors

Other actuators

- **Thermal:** dimension changes with temperature, for example a bi-metallic strip in a thermostat.
- **Optical:** optical-fibre data transmission, IR remote controls, laser eye surgery, magneto-optical drives.
- **Pneumatic:** compressed air from a pump drives a cylinder for linear motion and pressure.
- **Hydraulic:** pumped incompressible fluid (mostly oil) drives a cylinder, giving large forces.
- *Applications:* industrial control valves, automation and motion control; automotive power steering and hydraulic brakes; aerospace flight, steering and brake control.