# Chapter 10

# Implementation

## 10.1 Introduction

So far we have focused on concepts and ideas. Since controllers are actual devices we will now briefly discuss what controllers look like physically. This is difficult for two reasons, control are implemented in many different technologies and the application areas are very wide. Technology has also changed significantly over the years. We will therefore concentrate on the principles and give a few practical illustrations.

Controllers have been described as differential equations or transfer functions. To implement a controller it is necessary to construct a device that solves differential equations in real time. In addition it is necessary to connect the computing device to the process which is done using sensors and actuators. A controller also has other functions apart from the control algorithm itself. Most controllers have facilities for man-machine interaction. Controllers can also operate in different modes. When discussing implementation we must consider issues such as mode, switches, safety, reliability and user interfaces.

Early controllers were implemented with devices that combined sensing, computing and actuation. The centrifugal governor is a typical example. Later as technology developed the different functions were separated. Computing can be done in many different ways using a wide range of technologies. Analog computing was used in the early controllers but today it is typically performed with digital computers. Analog control is still used in devices that require very high speed. It is also commonly used in MEMS devices because analog controllers require less silicon surface than digital controllers.

Feedback has had a central role in the development of sensing, actuation

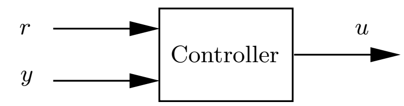


Figure 10.1: An abstract representation of a controller with two degrees of freedom.

and computing. Many advances in the technology have been associated with the introduction of feedback in the devices. There have thus been close synergies between the development of ideas and devices.

A significant standardization has also occured in specific application domains. The advantages of standardization was recognized early in process control where standards for signal transmission were introduced. This made it possible to have different suppliers of sensors, actuators and controllers. It was also possible to collect all controllers into central operating rooms so that operators could have a good overview over large manufacturing processes. Standardization of digital communication is also underway but progress has been slow because of special vendor interest. An interesting feature is that the internet protocol is beeing used increasingly.

Feedback has had an essential role both in analog computing and in implementation of controllers. The devices used to implement controllers are amplifiers and systems with dynamics. The amplifiers can be nonlinear because linear behavior can be created using feedback. The development of controllers is interesting, great ingenuity has often been demonstrated and a wide range of technologies have been used. Today two technologies are predominant. Analog controllers based on operational amplifiers and computer control. Since biological systems use pulse based computing extensively we also include a section on that.

## 10.2 Sensing Actuation Computing and Communication

For the most part of this book a controller has simply been viewed as a box with two inputs, the reference $r$, the measured process variable $y$, and one output, the control variable $u$ as illustrated in Figure 10.1. The linear behavior of a controller is typically described by a transfer function or a differential equation. A more detailed representation of a controller is given in Figure 10.2. In this figure the controller is decomposed into three blocks,

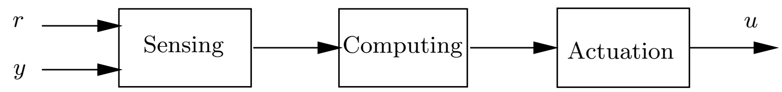


Figure 10.2: A more detailed representation of a controller.

representing sensing, computing and actuation. The sensor converts the physical process variable to a representation that can be handled by the computing device. The computing device performs the operation on the signal expressed by the controller transfer function. The actuator converts the result of the computations to a physical variable that can influence the process. Actuation requires energy or forces.

Many processes have large dimensions and the control system has to cover a large area. For large processes it is therefore common to centralize the computing in a control room or master computer. It is then necessary to connect sensors actuators and computing functions. If signals are represented electrically by currents and voltages the connections are done with wires. If the system have many sensors and actuators there will be a lot of wires and then is is highly advantageous to use communication networks. A wide variety of networks can be used, electrical, optical and wireless. The dimensions of the systems can vary widely. A CD player is about 0.1 m. Systems for a car have sizes of a few meters. Climate control of buildings have sizes of hundreds of meters. For large paper mills the distances can be of the order of kilometers. Power systems cover several nations and the Internet covers practically the whole planet.

Figure 10.3a shows some of the trends in sensing, actuation, and computation in automotive applications. As in many other application areas, the number of sensors, actuators, and microprocessors is increasing dramatically, as new features such as antilock brakes, adaptive cruise control, active restraint systems, and enhanced engine controls are brought to market. The cost/performance curves for these technologies, as illustrated in Figure 10.3b, is also insightful. The costs of electronics technologies, such as sensing, computation, and communications, is decreasing dramatically, enabling more information processing. Perhaps the most important is the role of communications, which is now inexpensive enough to offer many new possibilities.

Modern control engineering is also closely related to the integration of software into physical systems. Virtually all modern control systems are implemented using digital computers. Often they are just a small part of

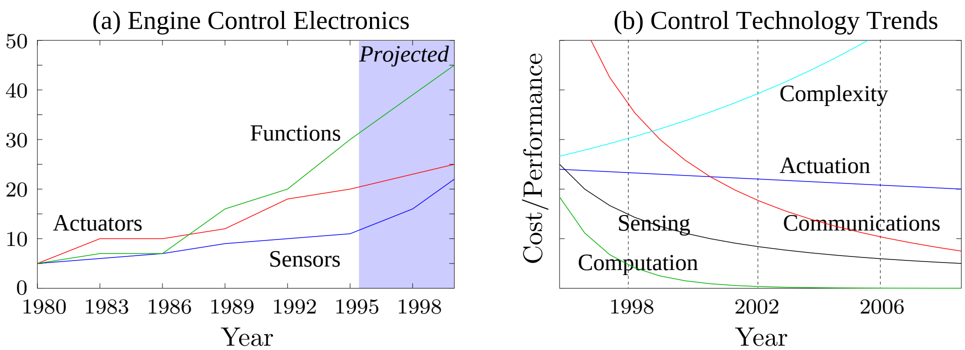


Figure 10.3: Trends in control technology: (a) the number of sensors, actuators and control functions in engine controls [4] and (b) illustration of cost/performance trends for component technologies.

much larger computing systems performing various other system management tasks. Because of this, control software becomes an integral part of the system design and is an enabler for many new features in products and processes. Online reconfiguration is a fundamental feature of computer controlled systems and this is, at its heart, a control issue.

This trend toward increased use of software in systems is both an opportunity and a challenge for control. As embedded systems become ubiquitous and communication between these systems becomes commonplace, it is possible to design systems that are not only reconfigurable, but also aware of their condition and environment, and interactive with owners, users, and maintainers. These "smart systems" provide improved performance, reduced downtime, and new functionality that was unimaginable before the advent of inexpensive computation, communications, and sensing. However, they also require increasingly sophisticated algorithms to guarantee performance in the face of uncertainty and component failures, and require new paradigms for verifying the software in a timely fashion. Our everyday experience with commercial word processors shows the difficulty involved in getting this right.

One of the emerging areas in control technology is the generation of such real-time embedded software [21]. While often considered within the domain of computer science, the role of dynamics, modeling, interconnection, and uncertainty is increasingly making embedded systems synonymous with control systems. Thus control must embrace software as a key element of control technology and integrate computer science principles and paradigms into the discipline. This has already started in many areas, such as hybrid systems and robotics, where the continuous mathematics of dynamics and control are intersecting with the discrete mathematics of logic and computer science.

### Sensors

Sensors are the devices that convert physical variables to signals that can be used for computation. A wide variety of sensors are available. There are standard devices for measuring variables such as temperature, pressure, flow, force, torque, acceleration, velocity and position. Sensors for measuring concentration and composition are also emerging. It is interesting to note that new sensors have often resulted in new control systems. Feedback is often an important ingredient which is used internally in sensors. Precision of sensors have often been improved drastically when feedback has been introduced.

### Actuators

The actuators are the devices that influences the process. This typically requires force and energy. There is a wide variety of actuators, typical examples are valves, and motors. Valves are one of the most common actuators. They are used to control flow rates and are particularly common in the process industries. A schematic picture of a pneumatic control valve is shown in Figure 10.4. The membrane can be pushed down by letting air into the upper chamber. The spring pushes the membrane up when there is no pressure. This gives a safety feature, if something goes wrong so that there is no pressure the valve stem will always be in the up position. A significant force may be required to move a control valve. Valves which are well sealed also have a substantial friction. Feedback is often used to improve the properties of a valve. This can be done by using a feedback system that controls the air pressure in the upper chamber by measuring the position of the valve stem, see Figure **??**. This feedback ensures that valve position is proportional to the reference of the valve positioner. The effects of forces on the valve stem caused by the flow and friction are also reduced by the feedback. Precision of actuation have often been improved drastically by introducing feedback.

Hydraulics are used when large forces are required, see Figure 10.4. Hydraulic valves often have several stages. The valves are often driven by electrical signals. The first stage can consist of a small valve that where the spool is positioned with an electromagnet. The flow from the small valve

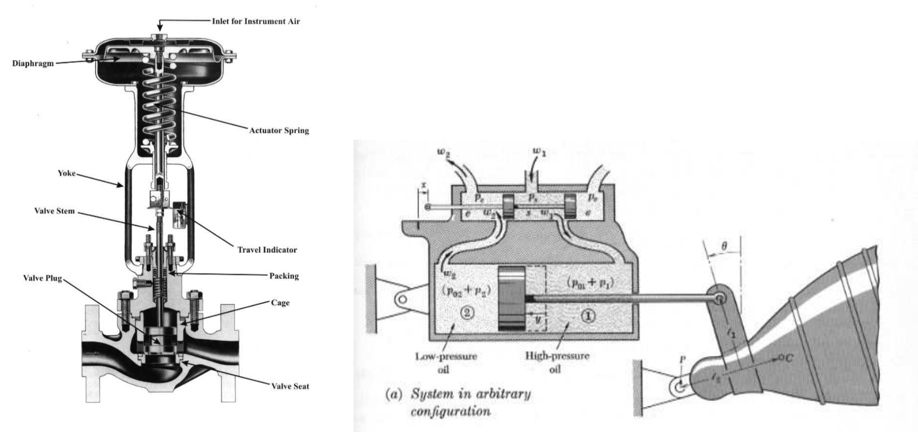


Figure 10.4: A schematic picture of a valve with pneumatic actuation (left) and an hydraulic actuator for a rocket engine.

is then used to position the spool of a larger valve. Feedback is also used extensively.

Actuators are typically driven by electric signals. The signals have to be amplified to provide the forces required by an actuator. The amplifiers can broadly speaking be divided into continuous and discrete devices. Thyristors and triacs are typical discrete devices. A triac is a device that can switch on and off an AC signal when is passes through zero. A thyristor can be switched on at any time but it can only be switched off when the current is zero. Transistors can be used both as switches and continuous amplifiers. Even when transistors are used it is advantageous to use them as switches because they can handle more power in switching mode.

It is possible to approximate continuous behavior by switching amplifiers if the switching is sufficiently fast and if the actuator itself can provide smoothing. A typical case is temperature control where a heater is switched on and off using pulse width modulation. Since thermal devices often have long time constants the behavior of an heater with pulse width modulation can be well approximated by a continuous system.

Actuation in information systems, such as the Internet, involves the manipulation of information flow. In congestion control, for example, we change the rate at which packets are sent out across the network. In financial systems, we can modulate the price we are willing to pay for a service.

### Computing

A general controller can be described by the equations

$$\begin{aligned}\frac{dx}{dt} &= f(x,y,r)\\ u &= g(x,y,r)\end{aligned} \tag{10.1}$$

where $r$ is the reference, $y$ the measured variable, $u$ the control variable and $x$ the state vector of the controller. The signals can be scalars or vectors. When the controller is linear linear the functions which $f$ and $g$ have the form

$$\begin{aligned}f(x,y,r) &= Fx + Gy + Hr\\ g(x,y,r) &= Cx + Dy + Er\end{aligned}$$

The computations required for control is thus integration of the differential equation (10.1).

Computing can be done in many different ways. Early controllers used analog computing based on mechanical and pneumatic devices. Analog computation is is very well suited for solving differential equations. Controllers were implemented using analog computing and analog computers were used to verify the design of control systems by simulation. Feedback was also used extensively in analog computing to obtain high precision from components with poor precision. Development of control and computing benefitted significantly from this cross fertilization.

Although digital computers were available in 1950 (check the date) it took some time before they had a major impact on control. The reason was that the early digital computers were large and expensive and slow.

Use of digital computing for control started with the development of computerized process control in the late 1950s. The development accelerated with the emergence of minicomputers in the 1960s. The major impact of computer came, however, with the emergence of the microprocessor in the 1970s. This made digital computing cost effective and available even for individual control loops. It is now the major technology for implementing controllers. There are however some areas where analog computing is still used. . For very fast systems digital controllers can be implemented using field programmabe gate arrays. Analog computing is still used for applications that require fast controllers. They are also used in MEMS devices because analog technology can be implemented with less silicon surface than digital controllers.

### Communication

Sensors and actuators are seldom located in the same place. Advanced control systems also compute control actions from several sensors. There are often good reasons to have the computing device in a location that is different from the sensors or the actuators. It is therefore necessary to communicate the information from the sensors to the computing device and from the computing device to the actuators. The distances required can vary significantly. A CD player is about 0.1 m. Systems for a car have sizes of a few meters. Climate control of buildings have sizes of hundreds of meters. For large paper mills the distances can be of the order of km. Power systems cover several nations and the Internet covers the whole planet. The distances are even larger for the space missions. Many different technologies can be used for communication, mechanical motion can be transmitted using lever, pressure can be communicated in pipes, electrical signals in wires. It is increasingly common to use communication networks to reduce the wiring. A wide variety of networks can be used, electrical, optical and wireless.

The functions of sensing, computing and actuation were combined early controllers, like the centrifugal governor. As controllers developed these functions were separated. This implied that the functions could be fart apart and it became necessary to communicate between the devices. This was very common in process control where many controllers were combined in a control room that also had indicators and recorders to give operators an overall view of a complete system. This development made it necessary to standardize the communication. The first attempt at this was done using pneumatic signal transmission where dimensions of tubes and signal levels were standardized, typically to 3 to 15 psi. Later when electronics replaced pneumatics electrical signals were used forcommunication. The signals were typically in the range of 0–20 mA or 4–20 mA. The main reason for using 4 mA instead of 0 mA as the lower limit is that many transmitters are designed for two-wire connection. This means that the same wire is used for both driving the sensor and transmitting the information from the sensor. It would not be possible to drive the sensor with a current of 0 mA. The main reason for using current instead of voltage is to avoid the influence of voltage drops along the wire due to resistance in the (perhaps long) wire.

When digital computers became electrical wires were replaced by communication networks. Wireless communication is also used. A large number of protocols are used in different applications.

## 10.3 Analog Control

In analog controllers the signals are represented by physical quantities, electrical currents and voltages, pressure, position of mechanical linkages or angeles. Many sensors give signals in such forms and computations is performed by exploiting properties of physical devices. Analog computing emerged simultaneously with control in the late 1930s and 1940s. It had immediately a strong impact on control. It was used extensively both to implement controllers and to simulate systems. A characteristic feature of analog computing is that is solves differential equations in continuous time. Analog computing is thus well suited to implement controllers. It is still used extensively, particularly in situations which require fast control. It is also used in MEMS because analog controllers can often be implemented using less silicon surface than digital controllers.

Analog computing can be traced back to Lord Kelvin who developed a machine to compute tidal tables. Kelvin used ball and disk integrators to perform integration. A major step forward was the mechanic differential analyzer developed by Vannevar Bush at MIT in 1927. The machine was based on the ball and disk integrator as a dynamic element and a torque amplifier to provide high gain. The original differential analyzer could solve six differential equations. A number of similar machines were developed in many other universities in the 1930s and 1940s. The MIT machine was used by Nichols to develop tuning rules for PID controllers, see **??**.

The development of the operational amplifier is the key to analog computing. This is a very flexible amplifier with high gain that permits very effective use of feedback. Combined with passive elements in the form of resistors and capacitors it gives a simple and effective way of implementing controllers. Simple controllers can be implemented in a straight forward way and there are systematic methods to implement complicated controllers.

Analog computing improved drastically with the development of the operational amplifier. This is a very flexible amplifier with high gain that permits very effective use of feedback. Lineare differential equations can be solved very conveniently by combining the amplifier with passive elements in the form of resistors and capacitors. These possibilities led to commercial analog computers which were used extensively by industry and university. Analog computing was also a simple and effective way of implementing controllers. The first analog computers used electronic tubes. The technology developed rapidly by exploiting the advances in electronics which provided transistors and integrated circuits.

Efficient integration is the key to analog computing. Integration was

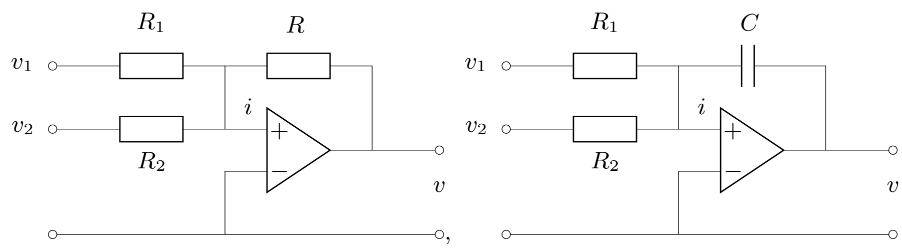


Figure 10.5: Schematic diagrams of a summer left and an integrator right that are implemented using operational amplifiers.

originally performed by a mechanical device, the ball and disk integrator. A major advance of analog computing was made by implementing integration with feedback around an electronic amplifier. This is particularly easy to do using operational amplifiers. Recall fromSection **??** that the operational amplifier is a flexible and has high gain. Combined with passive elements in the form of resistors and capacitors it gives a simple and effective way of implementing controllers. An integrator and a summer can be implemented as illustrated in Figure 10.5 using an operational amplifier with a capacitor as feedback and resistors as elements. Consider for example the summing circuit. Assuming that the current $i$ is zero it follows that the current through the resistor $R$ is the sum of the currents through resistors $R_1$ and $R_2$, hence

$$\frac{v_1}{R_1} + \frac{v_1}{R_1} = -\frac{v_2}{R}$$

Solving for $v_2$ gives

$$v_2 = -\frac{R}{R_1}v_1 - \frac{R}{R_2}v_2 \tag{10.2}$$

Now consider the integrator in Figure 10.5. Let $V_1(s)$, $V_2(s)$ and $V(s)$ be the Laplace transforms of the signals $v_1$, $v_2$ and $v$. Assuming that the current $i$ is zero it follows that the current through the capacitor $C$ is the sum of the currents through resistors $R_1$ and $R_2$. Using the concept of operator impedances introduced in Section C.3 we find

$$\frac{V_1(s)}{R_1(s)} + \frac{V_1}{R_1} = -V(s)sC$$

Recall that the operator impedance of a capacitor is $1/(sC)$. Solving for $V(s)$ we get

$$V(s) = -\frac{1}{R_1C}V_1(s) - \frac{1}{R_2C}V_2(s)$$

Transforming back to the time domain gives the following input output relation

$$y(t) = -\frac{1}{R_1C}\int_0^t v_1(\tau)d\tau - \frac{1}{R_2C}\int_0^t v_2(\tau)d\tau \tag{10.3}$$

Systems that can handle more inputs are obtained simply by adding more input resistors to the circuit. Notice that the coefficients $R/R_1$, $R/R_2$ are positive. The negative sign is required because effective use of an operational amplifier requires negative feedback. With summers and integrators it is thus straight forward to implement any linear controller (10.1). Extra amplifiers may be necessary for sign changes.

## 10.4 Computer Control

Digital computing emerged in the mid 1940s, a milestone being the stored program machine developed by John van Neumann in collaboration with Eckert and Mauchly at the Moore Schools of Electronic Engineering. The computers have gone through a very dramatic development where performance has doubled every 18 months. Control was one of the applications that was considered. The early digital computers were, however, large, bulky and expensive. Implementation required investments of millions of dollars. The initial investment in computer control is thus quite large but the incremental cost for adding another control loop is small. For this reason digital control was feasible only for very large systems. The first installation was a system for process control of an oil refinery in 1959. Use of digital control grew rapidly as computers become smaller and cheaper. The introduction of minicomputers led to major advances in process control. With the advent of the microprocessor digital control became a commodity and it is now the most common way to implement controllers.

Since a computer only can operate on numbers it is necesary to have sensors that can deliver numbers or analog to digital converters that can convert analog signals to numbers. It is also necessary to have actuators that can act on numbers or digital to analog converters that convert a number to an analog signal that drives the actuator. A schematic diagram of a digital controller is shown in Figure 10.6.

### Sampling

Since a digital computer operates sequentially it is necessary to sample the sensor signals. Sampling is often done periodically. The choice of sampling rate is essential. It is necessary to sample at a rate that is compatible with

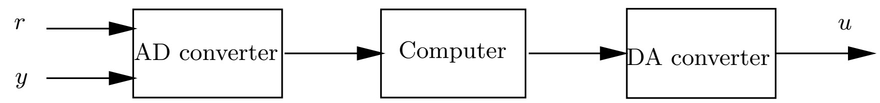


Figure 10.6: A schematic diagram of a digital controller.*Has to be redrawn*

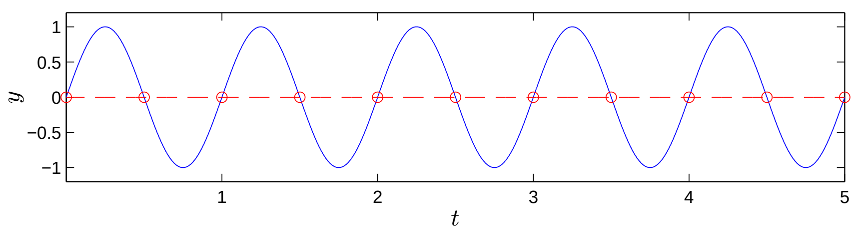


Figure 10.7: Sampling the signal $y = \sin 2\pi t$ at the Nyquist frequency 2 Hz.

the dynamics of the control loop. A sample is a very poor representation if there are rapid variations in a signal between the sampling instants. It is therefore necessary to filter a signal before it is sampled.

A simple way to determine a lower bound of the sampling rate is to consider sinusoidal signals. Figure 10.7 shows a sinusoid with frequency one Hz that is sampled at two Hz. The figure shows that it is not possible to separate the sinusoid from the signal that is zero from the samples. The sampled sinusoidal signal has the zero signal as an alias. The situation illustrated in Figure 10.7 is actually the critical case. Consider periodic sampling with frequency $\omega_s$. In order to reconstruct the signal from the sample it must be required that the signal does not contain any components with frequencies higher than $\omega_N = \omega_s/2$. Components with frequencies higher than the Nyquist frequency will appear as low frequency signals after sampling. This phenomena which is called aliasing is illustrated with an example.

*Example* 10.1 *(Aliasing).* Assume that the signals $v(t) = \sin{(\omega + n\omega_s)t}$ and $y(t) = \sin \omega t$ are sampled at times $h$, $2h$, $\ldots$, $kh$, $\ldots$, where $\omega_s = pi/h$ is the Nyquist frequency and $\omega < \omega_N$. Let $y_k = y(kh)$ and $v_k = v(kh)$ denote the sampled signals. We have

$$
\begin{aligned}
v_k &= \sin{(\omega + n\omega_s)kh} = \sin \omega t \cos \omega_s kh + \cos \omega t \sin \omega_s kh \\
&= (-1)^n \sin \omega kh = (-1)^n y_k
\end{aligned}
$$

The signal $v(t)$ with frequency $\omega + n\omega_s$ thus has the alias $y(t)$ for even $n$ and $-y(t)$ for odd $n$. Both signals have the frequency $\omega$. After sampling the signal $v(t) = \sin(\omega + n\omega_s)t$ thus appears as a signal with all its energy at the frequency $\omega < \omega_N$.

The anti-alias filter is typically an analog low pass filter of second or higher order. Inadequate antialiasing filters is a common error in digital control systems. It is necessary to keep in mind that the filters have to be changed when the sampling period is changed.

The aliasing effect can create significant difficulties if proper precautions are not taken. Frequencies above the Nyquist frequency must be reduced by an analog filter before the sampling. This filter, which is called a prefilter or an antialias filter is essential in computer controlled systems. Filters of first or second order are commonly used in simple applications but it may be necessary to use filters of high order when there is much measurement noise just above the Nyquist frequency. The filter must also be redesigned when the sampling period is changed.

## Digital Computing

The computations required by a controller is to solve the differential (10.1). Digital computers can only perform arithmetic operations like addition, multiplication and division. It is therefore necessary to make approximations to solve differential equations. A simple method is to approximate the derivative by a difference. Replacing the derivative in (10.1) by a forward difference gives

$$\begin{aligned}\frac{x(t+h)-x(t)}{h} &\approx f(x(t),y(t),r(t))\\ u(t) &= g(x(t),y(t),r(t))\end{aligned}$$

The computations required for an approximate solution of the differential equation are thus given by

$$\begin{aligned}u(t) &= g(x(t),y(t),r(t))\\ x(t+h) &= x(t)+hf(x(t),y(t),r(t))\end{aligned}\tag{10.4}$$

The simple forward difference approximation works very well if the sampling period is short. There are some advantages in using other approximations as is illustrated by the following example.

*Example* 10.2 *(Other Approximations).* Consider the PD controller with filtered derivative described by

$$U(s) = \Big(k + \frac{k_d s}{1 + sT_f}\Big)E(s)$$

In the time domain this can be written as

$$T_f \frac{du}{dt} + u = (kT_f + k_d)\frac{de}{dt} + ke$$

Approximating the derivative with a forward difference gives

$$T_f \frac{u(t+h) - u(t)}{h} + u(t) = (kT_f + k_d)\frac{e(t+h) - e(t)}{h} + ke(t)$$

The control is thus updated by the following difference equation

$$u(t+h) = \Big(1 - \frac{h}{T_f}\Big)u(t) + (kT_f + k_d)(e(t+h) - e(t)) + ke(t)$$

This difference equation is stable if $h < T_f$. For a general purpose PD controller it is highly desirable to have code that works also if the derivative action is eliminated. The natural way to do this is to set $T_d = 0$. The algorithm given above will then be unstable for all values of $h$.

If the derivative is instead approximated by a backward difference we find

$$T_f \frac{u(t) - u(t-h)}{h} + u(t) = (kT_f + k_d)\frac{e(t) - e(t-h)}{h} + ke(t)$$

Solving for the control gives

$$u(t) = \frac{T_f}{h + T_f}u(t) + \frac{kT_f + h}{T_f + h}(e(t) - e(t-h)) + \frac{kh}{T_f + h}e(t)$$

This difference equation is stable for all values of $h$ and $T_f$.

There is an extensive theory for computer control systems. Many approximations can be avoided by observing that the control signal is constant over the sampling period. For linear controllers all approximations can actually be eliminated. The details are given in the extensive literature on computer control.

### Real Time Issues

To provide proper real time response the computations must also be synchronized by a clock. The control algorithm can be organzed as follows.

1. Wait for interrupt from the clock.
2. Read analog input $y$ and reference $r$.
3. Compute control $u$
$$u = g(x, y, r).$$
4. Send $u$ to AD converter to generate analog output.
5. Update the controller state
$$x = x + hf(x, y, r).$$
6. Go to Step 1.

The controller has the state $x$ that is stored between the computational steps. Since digital computers operate sequentially there will always be a delay. With the implementation given above, the delay is the time required for the operations 2, 3 and 4.

### Controller Implementation and Computational Delay

The design of a controller for the sampled-data system results in a controller that can be represented in state-space form

$$\begin{aligned} x_c(k+1) &= Fx_c(k) + Gy(k) + G_c u_c(k) \\ u(k) &= Cx_c(k) + Dy(k) + D_c u_c(k) \end{aligned} \tag{10.5}$$

where $x_c(k)$ is the state of the controller. The controller (10.5) can be interpreted as an observer combined with a feedback from the observed states of the process. The controller can, however, also be a realization of any dynamical controller that has been designed. If the controller is designed in input-output form then the it can be written as

$$R(q)u(k) = T(q)u_c(k) - S(q)y(k) \tag{10.6}$$

In this form the states of the controller are the old inputs, outputs, and reference signals that have to be stored to compute the control signal. This implies that (10.6) normally is not a minimal representation of the controller.

The number of states is in a direct implementation $\deg R + \deg S + \deg T$, while the minimal representation will have $\deg R$ states. One reason to have a non-minimal representation can be that the inputs, outputs, and reference signals have to be stored for other reasons, such as data analysis. Since the complexity of the controller, in most cases, is quite low then the extra storage of some variables does not have any practical consequences.

When discussing the implementation of the controller we can without loss of generality choose the form (10.5). An algorithm for the controller at time $k$ consists of the following steps:

1. A-D conversion of $y(k)$ and $u_c(k)$.

2. Computation of the control signal $u(k) = Cx_c(k) + Dy(k) + D_c u_c(k)$.

3. D-A conversion of $u(k)$.

4. Update the state $x_c(k+1) = Fx_c(k) + Gy(k) + G_c u_c(k)$.

Notice that the implementation of the control algorithm is done such that the control signal is sent out to the process in Step 3 before the state to be used at the next sampling interval is updated in Step 4. This is to minimize the *computational delay* in the algorithm. It is possible to further reduce the amount of computations between the A-D and D-A conversions by observing that the term $Cx_c(k)$ can be precomputed, which gives the following algorithm

1. A-D conversion of $y(k)$ and $u_c(k)$.

2. Computation of the control signal $u(k) = z(k) + Dy(k) + D_c u_c(k)$.

3. D-A conversion of $u(k)$.

4. Update the state $x_c(k+1) = Fx_c(k) + Gy(k) + G_c u_c(k)$ and precompute $z(k+1) = Cx_c(k+1)$ to be used at the next sampling instance.

For a single-input-single-output controller this results in two multiplications and two additions that have to be performed between the conversions.

Depending on the real-time operating system and the conversion times in the A-D and D-A converters the computational delay, see Figure 10.8, may vary from sample to sample. This will be even more pronounced if the control algorithm also includes iterations or optimization steps. One way to reduce the influence of the computational delay is to introduce a full sample interval delay in the controller, i.e. to allow $u(k)$ to be a function

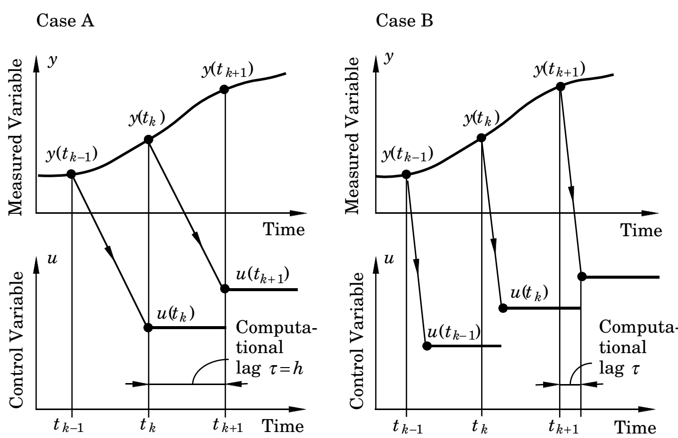


Figure 10.8: Two ways of synchronizing inputs and outputs. In case A the signals measured at time $t_k$ are used to compute the control signal to be applied at time $t_{k+1}$. In case B the control signals are applied as soon as they are computed.

of the process output and the reference signal up to time $k-1$, see Case A in Figure 10.8. The computational delay is now more deterministic but unnecessarily long, which normally is not good for the performance of the closed loop system.

## Controller Representation and Numerical Roundoff

Even if (10.5) and (10.6) are general forms for implementation of the controller there is a degree of freedom in the choice of the states of the controller. Different state representations have different numerical properties. Assume that we want to implement the digital filter or controller

$$y(k) = H(q^{-1})u(k) = \frac{b_0 + b_1 q^{-1} + \cdots + b_m q^{-m}}{1 + a_1 q^{-1} + a_2 q^{-2} + \cdots + a_n q^{-n}}\, u(k) \qquad (10.7)$$

The filter $H(q^{-1})$ can be implemented in different ways

- Direct form
- Companion form
- Series or parallel form

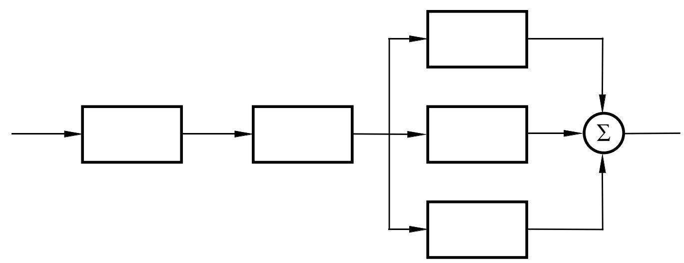


Figure 10.9: Illustration of series and parallel form of implementation of a digital filter where each block is a system of first or second order.

- $\delta$-operator form

The *direct form* is the non-minimal representation

$$y(k) = \sum_{i=0}^{m} b_i u(k-i) - \sum_{i=1}^{m} a_i y(k-i)$$

This and all other representations that directly are using the polynomial coefficients $a_i$ and $b_i$ are very sensitive for numerical errors if the polynomials have multiple roots close to the unit circle. Small perturbations in the coefficients or in the numerics can then lead to instabilities in the computations. Better realizations are series or parallel forms illustrated in Figure 10.9, where each block represents a first or second order filter.

The poles of a sampled-data system are approaching one when the sampling interval is decreased. This implies that there will be a cluster of poles that are close to the stability boundary when the sampling interval is short. To avoid these numerical difficulties it is possible to make a short-sampling-interval modification. Using (**??**) it follows that the matrix $F$ in (10.5) is close to a unit matrix and that the vectors $G$ and $G_c$ are proportional to the sampling interval for short sampling intervals. This implies that there can be several orders of magnitude in difference between $F$, $G$, and $G_c$. The state equation in (10.5) can be rewritten into the equivalent form

$$x_c(k+1) = x_c(k) + (F-I)x_c(k) + Gy(k) + G_c u_c(k) \tag{10.8}$$

The last three terms of the right hand side can now be interpreted as an correction of the state that are of the same magnitude. The correction is added to the previous state. The modification introduced in (10.8) is similar to the introduction of the $\delta$-operator which is defined as

$$\delta f(kh) = \frac{f(kh+h) - f(kh)}{h}$$

The $\delta$-operator can be interpreted as a forward difference approximation of the differential operator $p = d/dt$.

*Example* 10.3 *(Numerical sensitivity).* Bosse har exempel.

### A-D and D-A Quantization

The previous section illustrated how numerical quantization can influence the performance of the system. With double-precision the arithmetic computations are typically done with a resolution of 64 bits. The A-D and D-A conversions are done with much less accuracy and will have a larger influence on the performance. The resolutions of the A-D and D-A converters are typically 12–16 bits and 8–12 bits, respectively. It is better to have a good resolution of the measured signal, i.e. at the A-D converter, since a control system is less crucial for a quantized input signal. Compare the situation where on-off control is used. With such a crude input quantization it is still possible to get a reasonable closed loop performance.

Nonlinearities such as quantizations will normally introduce oscillations or limit cycles in the system. The magnitude of the oscillations will decrease with the quantization step. It is normally difficult to make an exact analysis of the influence of the converter resolutions even if crude results can be obtained using the method of describing function. The final choice of the resolutions has to be based on simulations or crude estimates.

### User Interfaces

A controller must also have facilities for man-machine interaction for example to change the reference value and to change controller parameters. To do this in a safe way the computations are coordinated by a real time operating system that ensured proper timing and that parameters are not changed in the middle of computations.

### Computer Architecture

- Process control computers
- Signal processors

- Microcontrollers
- PLCs
- Field programmable gate arrays

## 10.5 Biological Control

All living species have a large number control systems, their implementation have both similarities and differences compared to technical systems. Lots of feedback loops. A good idea may be to go through a muscle. Kolla Clarac och NI.

### Sensing

Biological systems have many highly specialized cells that act as sensors by detecting external stimuli. There are sensors for light, sound, smell and tactile sensors which respond to touch or pressure. There are also sensors in the muscles which respond to stretch or pressure. Primary sensor cells can combined with other cells and feedback to provide very sophisticated sensing system. Vision and hearing are typical examples of such sensor systems. Even if sensors appear in many different configuration the basic principles used are often similar.

A primitive light sensor is simply a well with cells that are light sensitive. It generates a signal that responds to the intensity of the light that reaches the well. More complex sensors uses a collection of light sensitive cells and lenses or mirrors, see Figure 10.10. There are many different arrangements, lenseyes with a retina as in the human or facet eyes where the optical part is split into a large number of lenses.

There is a wide variety of vision systems in the animal world, many of them are superior to the human eye in color- and light-sensitivity and sharpness. Hair cells are used as detectors of fluid motion, see Figure 10.14. These cells bend when surrounded in a fluid that moves. The bending causes a closed channel to open, entry of positively charge ions generates an electrical output signal. (fig sid 391).By using stiff cells of different lenghts combined with signal processing it is possible to obtain remarkable frequency resolution. The vestibular system provides information about the orientation of the head, see Figure 10.12. It consists of three tubes called the semicircular canals, that are approximately in the horizontal, vertical and sagittal planes. The tubes are combined at the ultricle. The canals are

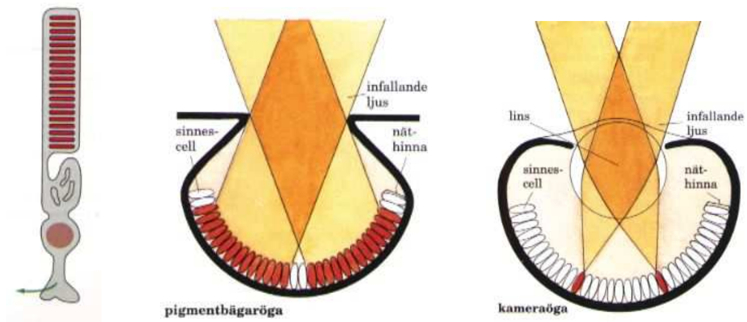


Figure 10.10: Schematic picture of a vision sensors, a rod cell in the retina (a), a primentbagar;ga (b) eye and a lens eye (c). These cells have photosensitive discs which contain the light-sensitive pigment rhodopsin. Light generates an electric signal which is transmitted via nerve cells.

filled with liquid. A gelatinous massm called the cupola, is inserted in the canal. Hair cells in the cupola detect bending which is an indication of the angular velociy of the head. The ultricle is provided with a device consisting of a membrane with embedded stones supported by a gelatineous mass with haircells. This device acts like a damped inverted pendulum and provides information about tilt and acceleration. Notice that the haircells are the primary sensory devices. The primary sensors in biological systems are often combined with other cells tro form signaling cascades. Such systems can have very high gain. The olefactory system of a dog can react to a few molecules only. The signaling cascades can also adapt to very large signal ranges.

### Actuation

There are several types of actuators in biological systems. Muscles is one type of actuators which makes it possible to exert external forces. Humans have three types of muscles, sceletal muscles, smooth muscles and heart muscles. The smooth muscles are present in internal organs, such as the digestive tract, veins and arteries. They are composed of thing long cells with one nucleus, see Figure 10.13.

The sceletal muscles are used to move the joints in the body. These muscles consists of a large number of muscle fibres which large cells with many nuclei, see Figure 10.13. The cells are typically several centimeters long with diameters around 50 $\mu m$. The fibres have an intriguing structure

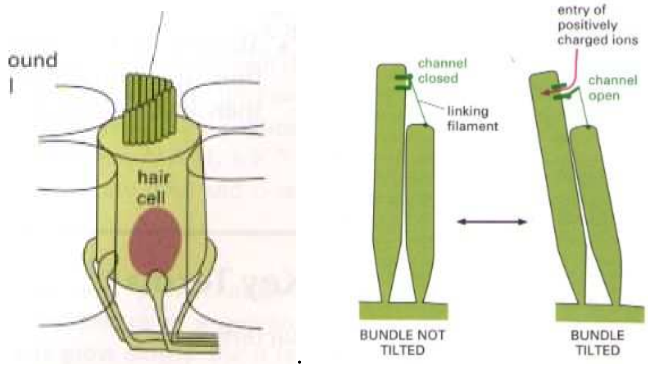


Figure 10.11: Schematic picture of a hair cell in the inner ear (left) and an enlargement (right) which shows the detection mechanism.

with many myofibrils which permit contraction of the muscle. Since the muscles operate by contraction bidirectional motion is generated by using pairs of muscles. There are also stretch sensors in the muscles which are used for feedback.

The cardiac muscles are specialized cells that produce the hart beat. Neighboring cells interact electrically to create the periodic motion required to pump the blood.

The smooth muscles and the heart muscle are controlled by the autonomous nerve system. The sceletal muscles can be controlled conciously.

Apart from muscles there which generate forces there are also actuators in the form secretary cells that act by secreting hormones and other substances into the blood.

### Communication and Computing

There are several ways to communicate in biological systems. Hormones produced in glands can be distributed widely by secretion into the bloodstream. Molecules can diffuse to neighboring cells. There are special mechanisms for direct membrane to membrane contacts.

Neurons, see Figure **??**, are special cells for computing and communication. The neuron consists of a cell body, dendrites and an axon. The cell body is nucleous which contains the biochemical system essential for to keep the cell alive. The dendrites are tree-like tubes that extend from teh cell

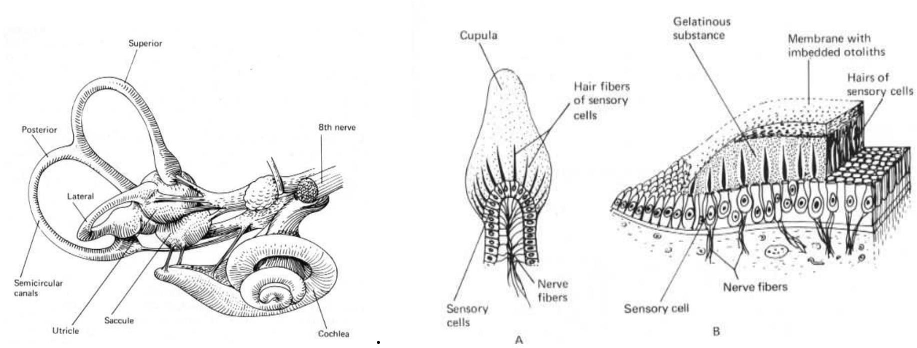


Figure 10.12: Schematic picture of the vestibular system (left), the cupola which is the primary sensor based on hair cells (middle) and the utricle (right).

body. They serve as receptors for incoming signals. The axon is the the output device that serves as a communication device to other neurons. The output singal of a neuron consists of pulses which are transmitted electrically along the axon. Some axons are provided with a sheath, which gives faster signal transmission. The axon ends with terminal fibres so that it can be connected to many other neurons. Information to other neurons is transmitted at synapses. The axon terminals are terminal buttons which contain tiny structures called vesicles. A vesicle can hold thousands of molecules which can be released into the receiving neuron. Transmission occurs when an electrical pulse reaches the buttons. The released molecules change the postsynaptic potential in the body of the receiving cell. A pulse is generated when the cell voltage reaches a threshold. Some inputs are excitory, they stimulate pulse generation, other inputs are inhibitory.

A human brain is believed to have around $10^{11}$ neuron. A typical neuron may have from $10^3$ to $10^4$ synapses. The axons can be very long, more than 1 m in the human. Even if the basic structure of the neurons are similar they can have different structure and different functions.

- Explain how computations can be made with first order systems and summation.

### Control System For Muscle Contraction

How sensing, actuation and computing is combined in a system can be illustrated by the components involved in the knee reflex. Contraction of

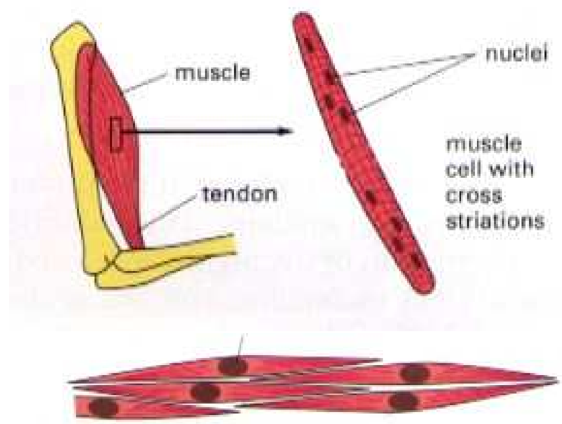


Figure 10.13: Sceletal muscles (top) and smooth muscles (bottom).

muscles are initiated by special motoneurons, see Figure **??**. The axons of these neurons are covered with glia cells, which increases the speed of signal transmission. The axons are connected to the muscle cells. The muscle cells also have stretch sensitive neurons that are imbedded in the muscles. When the knee is hit the stretch sensitive neuron is excited. The signal is passed to many motoneurons in the spinal cord as illustrated in Figure 10.15. The motoneurons activate the muscles to generate the rapid knee kick. Information is also sent to the brain as indicated in Figure 10.15. human.

### Mathematical Models

Neurons can be viewed as multi-input single-output systems. The primary signal forms is pulses or spikes. There are inhibitory and exhibitory inputs. A crude model is obtained by giving the average pulse rate as a function of the average pulse rates of the excitory and inhibitory inputs.

A simple static model gives the average pulse rate as a function of the postsynamptic potential. Many forms have been suggested, one example is the Naka-Rushton model

$$\begin{aligned} y = f(u) &= \begin{cases} \frac{au^n}{b^n+u^n}, & \text{for } u \geq 0 \\ 0, & \text{for } u < 0 \end{cases} \\ u &= \Sigma_k e_k - \Sigma_k i_k \end{aligned} \tag{10.9}$$

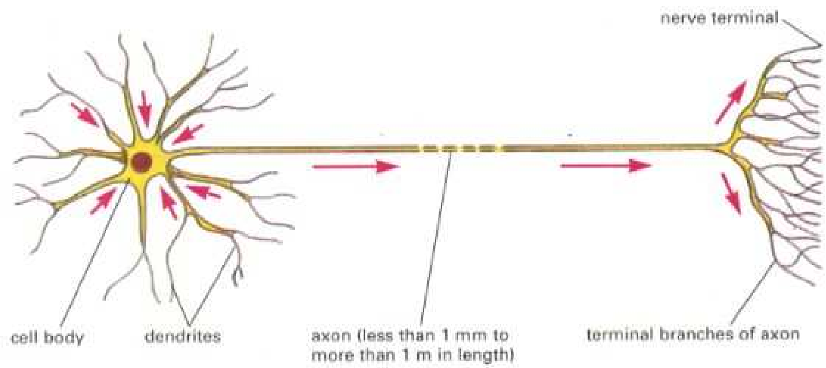


Figure 10.14: Schematic picture of a neuron. From Alberts et al 1997.

where $y$ is the pulse rate of the output, and $u$ is the stimulus intensity. The total stimulus activity is difference between the the sum of the pulse rates from the excitory inputs $e_k$ and the pulse rates of the inhibitory connections $i_k$. The parameter $a$ is the maximum pulse rate, and $b$ the input stimuli which gives half the maximum rate.

A simple dynamical model is obtained by adding first order dynamics, hence

$$\frac{dy}{dt} = \alpha(y - f(u)) \tag{10.10}$$

The model given by Equations (10.9) and (10.10) can be represented by the simple diagram in Figure 10.16.

The models given by Equations (10.9) and (10.10) are simple models that capture some of the input-output of the neuron but they are not closely related to physiology. The Hodgkin-Huxley model is more complicated that captures more of the physiology. This model is describes the complicated phenomena in the cell membrane of the neuron. The key element is to account for passage of sodium and postassium ions across the membrane. The cell body is approximately 10 times richer in potassium and about 10 times leaner in sodium. There is leakage by diffusion through the cell membrane. The concentration differences is maintained by pumps for sodium and potassium. The pumps are influenced chemically by the transmitter molecules secreted by the vesicles. Different synapses have different effects. A consequence of the concentration differences inside and outside the cell is that the potential of the cell is about 70 mV negative relative to the external liquid.

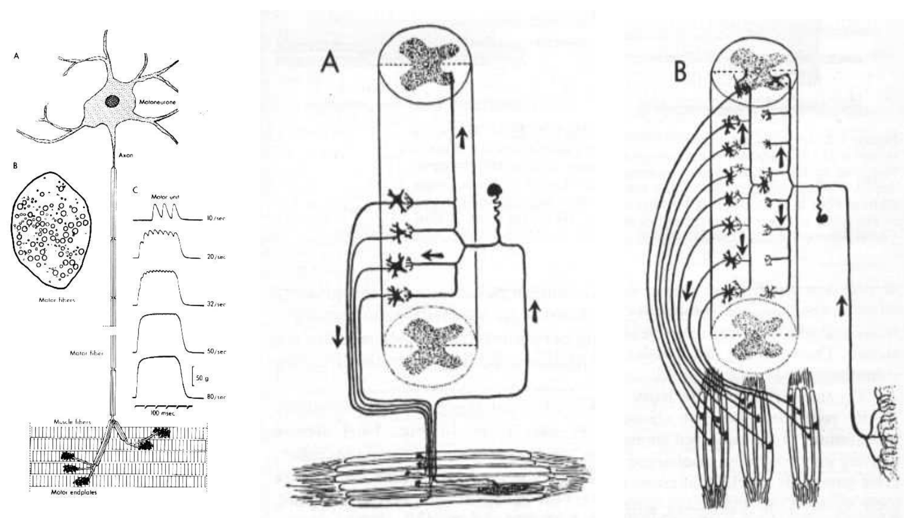


Figure 10.15: Schematic picture of a motoneuron (left), schematic diagram of the pathways involved in the knee reflex (middle) and the pathways for reaction to other stimuli (right).

Figure 10.16: Block diagram of the simple neuron models given by Equations (10.9) and (10.10). The inputs and the outputs represent average pulse rates.

The Hodgkin-Huxley equation is a charge balance for the cellmembrane. Let V be the potential difference across the cell membrane, $I_{Na}$, $I_K$ the currents generated by the sodium and potassium pumps and $I_l$ the leakage current. A charge balance then gives

$$C\frac{dV}{dt} = -I_{Na} - I_K - I_l \tag{10.11}$$

The currents are given by $I = c(V - E)$ where $c$ is the conductance and $E$ the equilibrium potential for the ions which is given by Nernst's law

$$E = \frac{RT}{zF}\log\frac{C_{out}}{C_{in}}$$

where $C_{in}$ and $C_{out}$ are the concentrations inside and outside the cell, $z$ the charge of the ion, $R$ and $F$ the thermodynamic gas constant and $F$ the Faraday constant. The ion concentrations also obey differential equations. The net result is a set of four nonlinear differential equations. These differential equations capture many important properties observed in real neurons.

- The HH model predicts generation of limit cycles spikes.
- Computation can be made with first order systems instead of integrators.
- Why two signal channels inhibitory and excitatory
- Differences and changes.
- Muscles of different types
- Sensors with derivative action
- Gain changes and adaptation

## 10.6 Summary

- Why is pulses used as a signal form in biological systems
- Why is electorchemistry not used more for sensors
- Why stable first order systems instead of integrators?

In this section we have described how control systems are implemented. There are four separate functions: sensing, actuation, communication and computing. It has been shown that the functions can be implemented using a wide variety of technologies. It is interesting to observe that feedback plays an essential role in sensing, actuation and computing. By exploiting the nice properties of feedback it is possible to obtain systems with well defined properties from components with highly variable nonlinear characteristics. For example it is possible to optain precise analog computing from linear passive components and highly nonlinear amplifiers. Pulses are used in devices such as relay based temperature controllers, Figure **??**, the delta-sigma modulator, Figure **??** and in biological system, Figure **??** both as devices for computing and communication. There is however very little theory available for such systems. It is also interesting to observe that there is a superficial similarity between analog and neural systems in the sense that multi-input single output devices play a central role as summers and integrators in analog systems and as neurons in biological systems. There are however also significant differences. Integrators in ananlog are identical but neurons can differ significantly. There is also a major difference in complexity. A large analog system may havee 100 integrators with 5 inputs each but a the human brain has $10^{11}$ neurons and a neuron may have $10^3$ inputs.

## 10.7 References

J. C. Eccles The understanding of the brain. Mc Graw Hill New York 1973.

Alberts, B and D. Bray and A. Johnson and J. Lewis and M. Raff and K. Roberts and P. Walter Essential cell biology An introductin to the molecular biology of the cell Garland Publishing Inc New York 1997

H.R. Wilson Spikes decisions and actions. Dynamical foundations of neuroscience. Oxford University Press, Oxford 1999

Scientific American The Brain Freeman New York 1979.

D.H. McBurney and V. B. Collings Introduction to sensation/perception. Prentice hall Englewood Cliffs NJ 1977.

T. R. H. Sitzer The digital differential analyser. Chapman and Hall, London 1968.