

Dispositivos de Entrada

Profa. M. Cristina

Profa. Rosane

março 2006 – março 2009

Dispositivos de Entrada

- Teclado
- Mouse
- *Trackball e Spaceball*
- *Joystick*
- Digitalizador (*tablet*)
- *Touch panel*
- *Light pen*
- *Data Glove*
- Voz

Data Glove



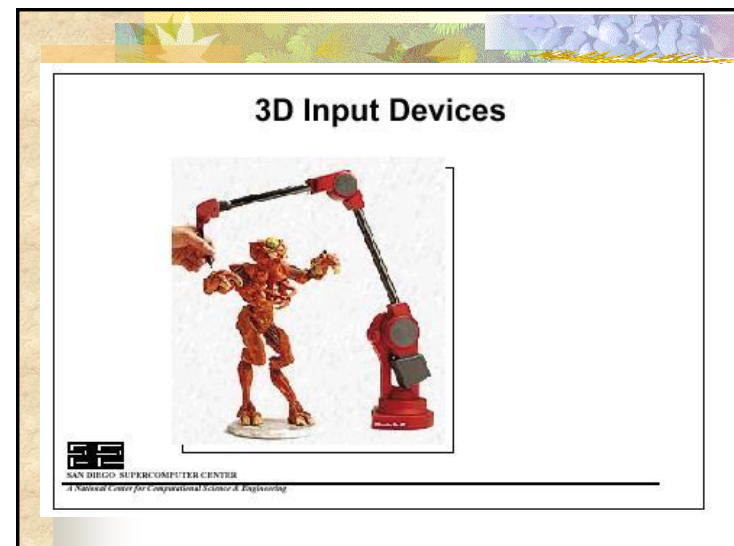
Data Glove

- Permitem 'pegar' um objeto virtual
- Conjunto de sensores detectam os movimentos da mão e dos dedos, bem como posição e orientação da mão

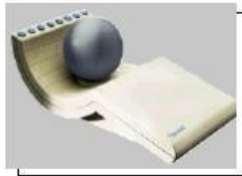


- ### Scanners 2D e 3D
- Permitem entrar coordenadas definidas em um espaço 2D e 3D
 - Digitalizar desenhos ou objetos
 - A partir dos pontos de entrada é reconstruído um modelo da superfície do objeto definido no espaço 3D

- ### Dispositivos de Entrada 3D
- Lêem uma posição 3D
 - Retornam 3 valores para o programa: tripla (x, y, z)
 - Alguns retornam também 3 ângulos de rotação
 - Ex. Digitalizador 3D, *spaceball*, *dataglove*



3D Input Devices



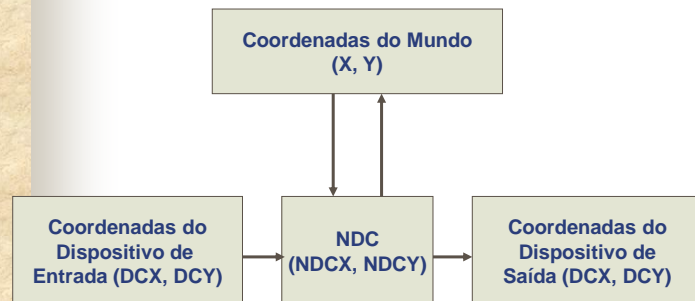

SAN DIEGO SUPERCOMPUTER CENTER
A National Center for Computational Science & Engineering

Force Feedback in 3D




SAN DIEGO SUPERCOMPUTER CENTER
A National Center for Computational Science & Engineering

Conversão de Coordenadas *mundo 2D e dispositivos*



Dispositivos de Entrada – Tipos Lógicos

- *Choice*
 - retorna uma escolha feita pelo usuário
 - Ex. teclado de funções, seleção de botão
 - Fornece algum tipo de *feedback* sensorial (luz, *clicks*, toque, ...)
- *Keyboard*
 - Retorna teclas com significados específicos
 - Letras, números, ...

Dispositivos de Entrada – Tipos Lógicos

- *Valuators*
 - Retorna um valor associado a algo
 - Ex. *knobs* (botões)
 - Pode especificar ganho, máximo e mínimo
- *Locators* (posicionadores)
 - Retornam a localização do cursor na tela
 - Ex. mouse, *trackball*, *tablet*,
 - Todos os posicionadores também podem funcionar como *valuators*
 - *Display-to-input ratio* (relação display-entrada)

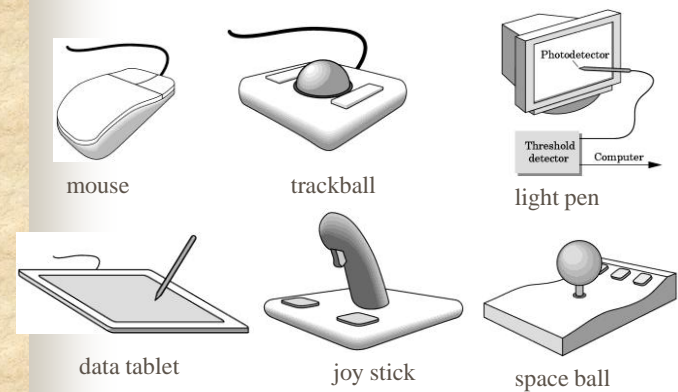
Dispositivos de Entrada – Tipos Lógicos

- *DTI ratio* (“ganho”)
 - quantidade de movimento do cursor na tela dividida pela quantidade de movimento da mão
- Valor alto: velocidade
- Valor baixo: precisão

Dispositivos de Entrada – Tipos Lógicos

- Formas de ler um dispositivo de entrada:
 - *Sampling* (amostragem): qual é a entrada nesse momento?
 - *Event-based*: aguarda até que o usuário forneça alguma entrada (execute alguma ação)

Physical Devices



Incremental (Relative) Devices

- Tablets devolvem um valor de posição diretamente ao sistema operacional
- Mouse, trackball e joy stick devolvem valores incrementais ao sistema operacional.
 - Valores precisam ser integrados para determinar os valores absolutos
 - Rotação dos cilindros do mouse
 - Rolagem da trackball
 - Podem ter sensibilidade variável

Dispositivos Lógicos (Periféricos do ponto de vista da programação)

- Considere os códigos C e C++:
 - C++: `cin >> x;`
 - C: `scanf ("%d", &x);`
- Qual é o periférico de entrada?
 - Não é possível dizer a partir do código.
 - Pode ser teclado, arquivo, saída de um outro programa.
- O código permite entrada lógica
 - O retorno é um número, independentemente do dispositivo

Dispositivos gráficos lógicos

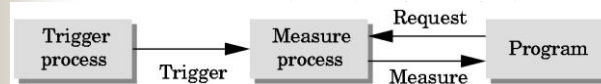
- Entradas gráficas são mais variadas.
- Duas APIs gráficas antigas (GKS, PHIGS) definiram seis tipos de entradas gráficas lógicas:
 - **Locator**: retorna uma posição
 - **Pick**: retorna ID do objecto
 - **Keyboard**: retorna strings de caracteres
 - **Stroke**: retorna arrays de posições
 - **Valuator**: retorna número em ponto flutuante
 - **Choice**: retorna um de n itens

Input Modes

- Request
- Event

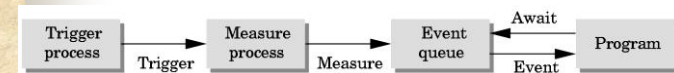
Request Mode

- Input provided to program only when user triggers the device
- Typical of keyboard input
 - Can erase (backspace), edit, correct until



Event Mode

- Most systems have more than one input device, each of which can be triggered at an arbitrary time by a user
- Each trigger generates an *event* whose measure is put in an *event queue* which can



Event Types

- Window: resize, expose, iconify
- Mouse: click one or more buttons
- Motion: move mouse
- Keyboard: press or release a key
- Idle: nonevent
 - Define what should be done if no other event is in queue

Callbacks

- Programming interface for event-driven input
- Define a *callback function* for each type of event the graphics system recognizes
- This user-supplied function is executed when the event occurs
- GLUT example:
glutMouseFunc (mymouse)
mouse callback function

GLUT callbacks

GLUT recognizes a subset of the events recognized by any particular window system (Windows, X, Macintosh)

- `glutDisplayFunc`
- `glutMouseFunc`
- `glutReshapeFunc`
- `glutKeyboardFunc`
- `glutIdleFunc`
- `glutMotionFunc`, `glutPassiveMotionFunc`

GLUT Event Loop

- Recall that the last line in `main.c` for a program using GLUT must be

```
glutMainLoop();
```

which puts the program in an infinite event loop

- In each pass through the event loop, GLUT
 - looks at the events in the queue
 - for each event in the queue, GLUT executes the appropriate callback function if one is defined
 - if no callback is defined for the event, the event is ignored

Realidade Virtual

- Pesquisa: Quais os dispositivos mais adequados para realidade virtual/aumentada?

Bibliografia

- Hearn, D. Baker, M. P. Computer Graphics with OpenGL, Prentice Hall, 2004 (Cap. 2)
- FOLEY, J.D. et al.- Computer Graphics Principles and Practice, Addison-Wesley, 2a. edição, 1990.
- Angel, Edward – Interactive Computer Graphics 4. Ed.