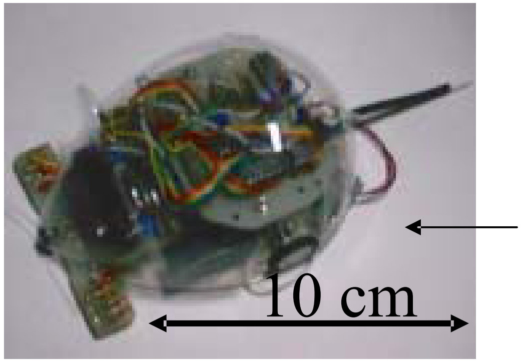
Open servo

Index

1. Outline – a hobbyist's look at robot building -
2. Proposal – I need a net connected servo -
3. Content – If you make it like this, robots can be made easier
4. Conclusion – nothing happens if you don’t make it -

Outline – a hobbyist's look at robot building -

- hobbyist robot events are becoming more and more high level, ROBO-ONE (2 legged walking robot event) etc.
- high level robots contain more than 20 servos within one body, more than 6 acceleration sensors, a video camera … servos/motors are rapidly increasing in use
- BTW a servo is a motor – the robots source of movement
- as a result of individual servo wiring, a robot has become a pile of cables



> an answer to a mountain of cabling?

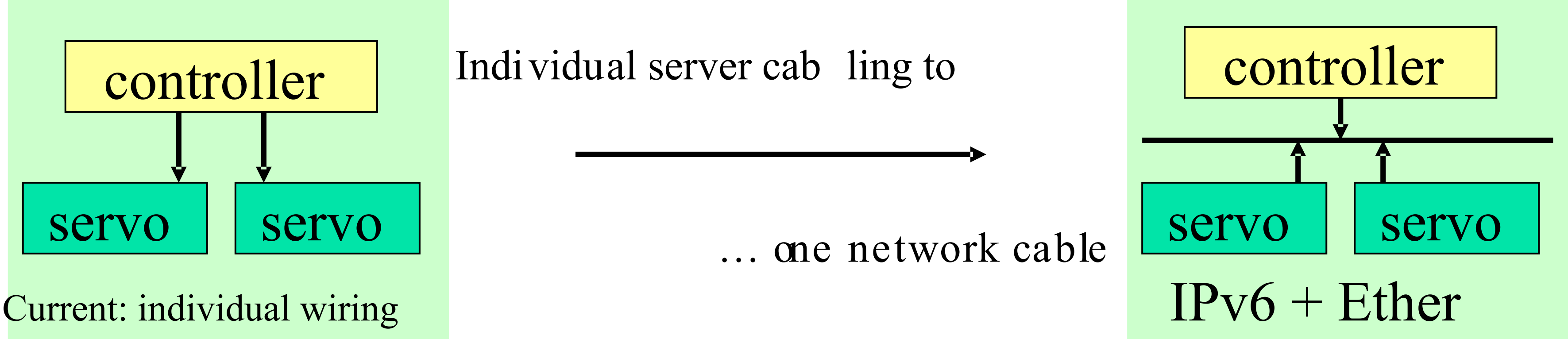
> yes: a robot like a human needs a nervous system

2 motors, 1 sensor made up a robot
in the past.. Good fun!

One of my creations: smallest, lightest body - robot sumo (as of 1999)
reckless!! kit base sumo robot: 200g, 12cm(not including tail) -

Proposal – I need a net connected servo -

- nervous system? > TCP/IP network can be used
- IPv6 (physical layer:Ethernet) connected servo and sensor
  - servo and sensor functioning with network and power only
- Why IPv6? Is v4 not suitable?
  - there is a problem in using an IPv4 servo at a hobbyist robot event
    - address assignment: ROM configuration doesn't allow part swapping
    - DHCP > robot usage environment is severe. - regularly turn off power to servo or momentary lose power - IP address is lost > control is lost…



Content – If you build it like this, robots can be made easier

□„ API for robots have already been developed but …

- not OpenSource (specs are not publicly available)
- library etc. not readily available (specs only not sufficient)
- hobbyists cannot use these – let’s make our own!

- have to build the following
- software
  - IPv6 Protocol Stack uClinux based) for an embedded board
  - servo and sensor control spec s and API (library) development
- hardware
  - □„ IPv6 ready servo (realized by SIMM Linux board embedded in a radio controlled servo)

Conclusion – nothing happens if you don’t make it -

- Advantages of a net-enabled servo/sensor
  - a mountain of wiring reduced to 1 communication cable
- to this point USB servo’s have been predominant – Why IPv6?
- Reasons for using IPv6
  - common specs can be made not dependent on the physical layer USB is still OK for the physical layer)
  - address assignment: Plug&Play becomes a reality for part
  - global: if its on the network, control can be handled from the other side of the world (distributed control can be realized)
  - routing: router can be made to do low level control handling, network architecture and dispersed control handling structure matches well
- to start, let’s try and make it!!