Lego MindStorms™ : Not Just for K-12 Anymore

Frank Klassner

Villanova University

Robotics in CS UnderGrad Curricula

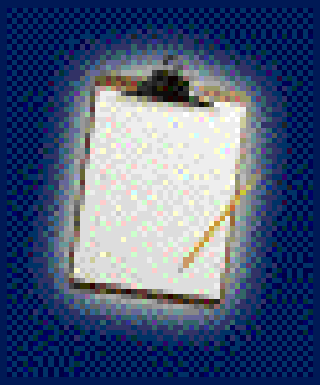
- Students Intrigued
  - Better Learning?
- Motivates Variety of Topics
  - "Not Just for Robotics Anymore"
  - Embedded Experience
- Zeitgeist Is Right?
  - CC2001

Applicable CC2001 Knowledge Areas

- Programming Fundamentals
- Algorithms & Complexity
- Architecture
- Operating Systems
- Programming Languages
- Intelligent Systems
- Net-centric Computing

What's Needed?

- Reasonable Cost
- Fast CPU & ++RAM
- Programming Language Variety
- Adaptable Chassis
- Sensor Variety
- Networkability



Why MindStorms?

- Cost
  - 1/4 Handyboard, 1/10 ActivMedia
- Flexibility
  - reusable pieces, variety of sensors/effectors
- Student Interest
  - Childhood experience
- Professional Curiosity

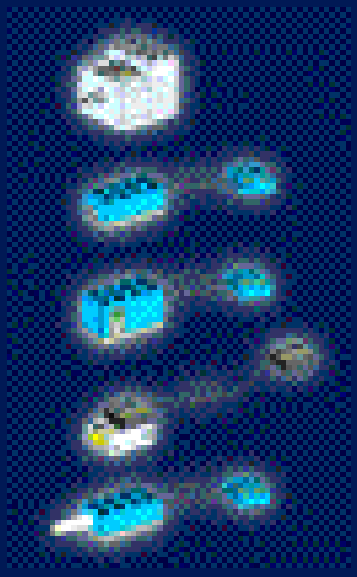
The MindStorms Platform

- 750+ building blocks/kit
- Programmable RCX
  - 3 sensor ports, 3 effector ports
  - IR port, 8-bit message buffer
  - Hitachi H8 CPU (16-bit)
  - 32KB onboard RAM
  - Firmware/VM - oriented
  - 6 AA batteries
  - GUI programming environment

- Desktop Communication
  - Serial and now USB transceiver
  - 2400 baud

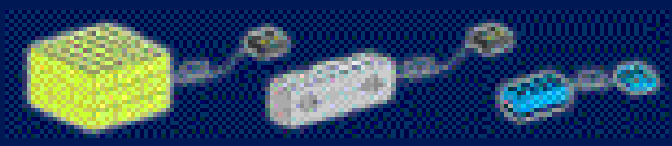
The MindStorms Sensor/Effector Suite

- Lego
  - Motor
  - Light
  - Rotation
  - Touch
  - Temperature
  - (Camera)

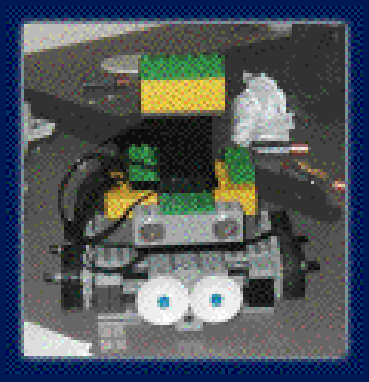


The MindStorms Sensor/Effector Suite

- 3rd Party (www.HiTechnic.com)
  - Compass
  - Ultrasound distance
  - IR proximity
  - Color
  - Motor Mux'er



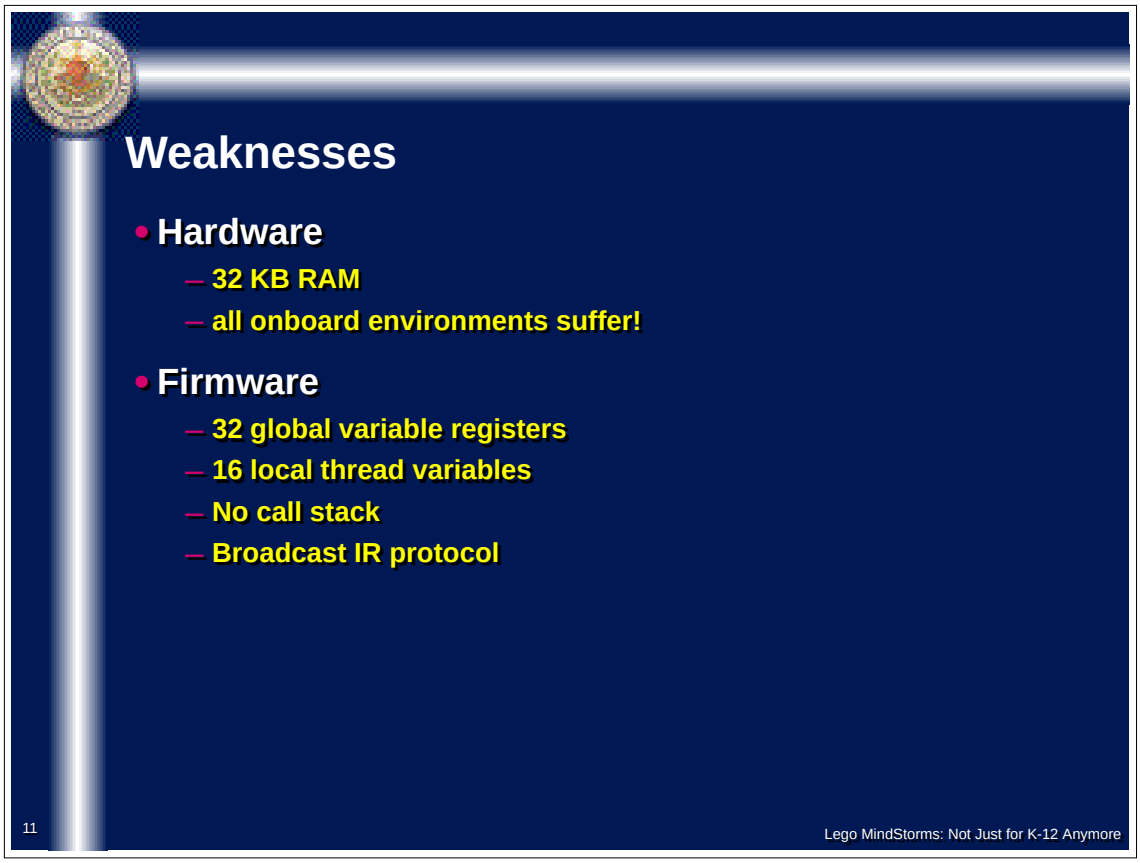
MindStorms Strengths

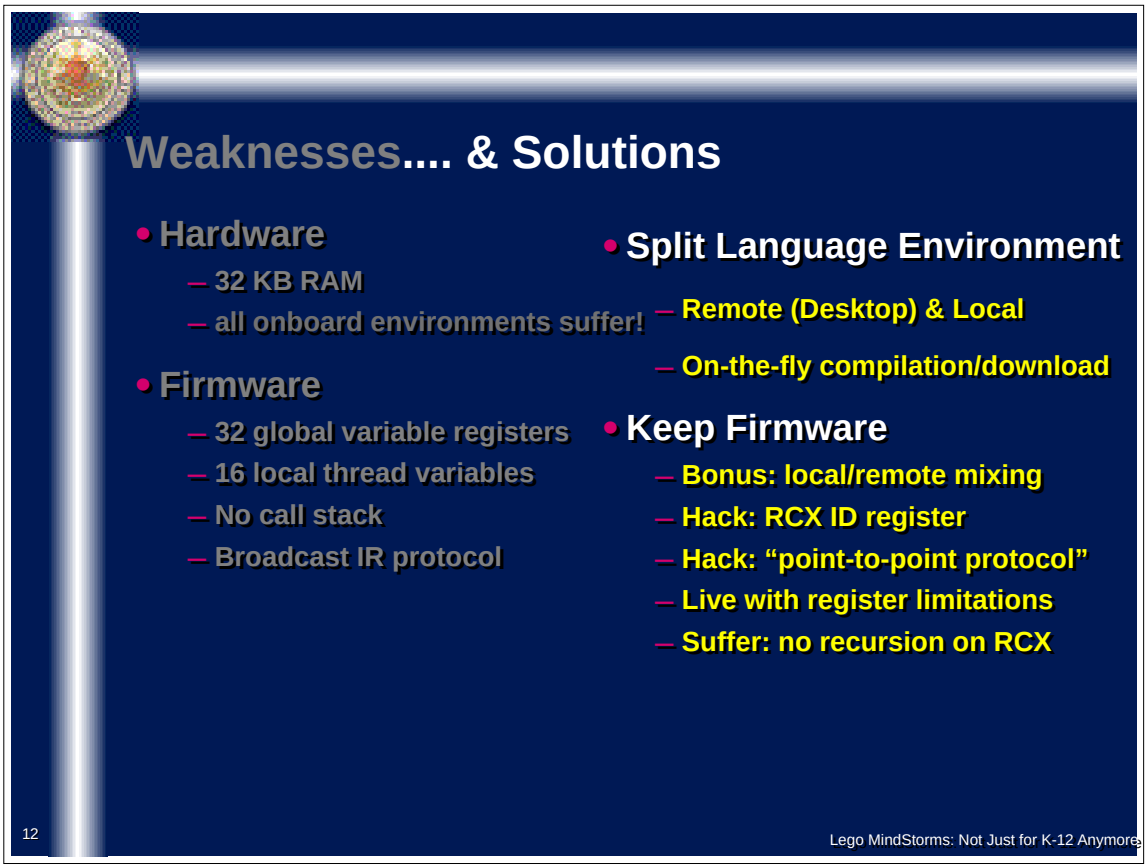


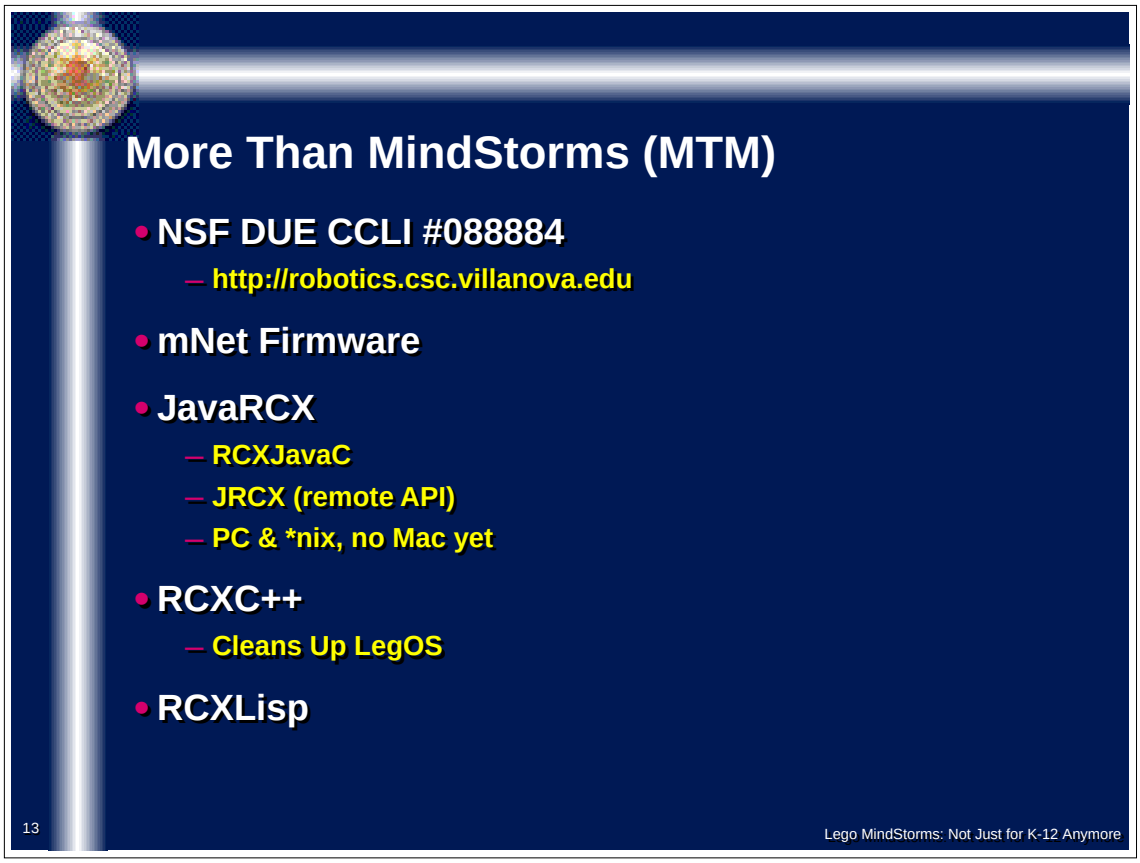
- Hardware
  - Sturdy!!!
  - Safe for CS majors :-)
  - Variety of Sensors
- Firmware (Virtual Machine)
  - Thread support
  - Remote-control mode
  - Local-execution mode

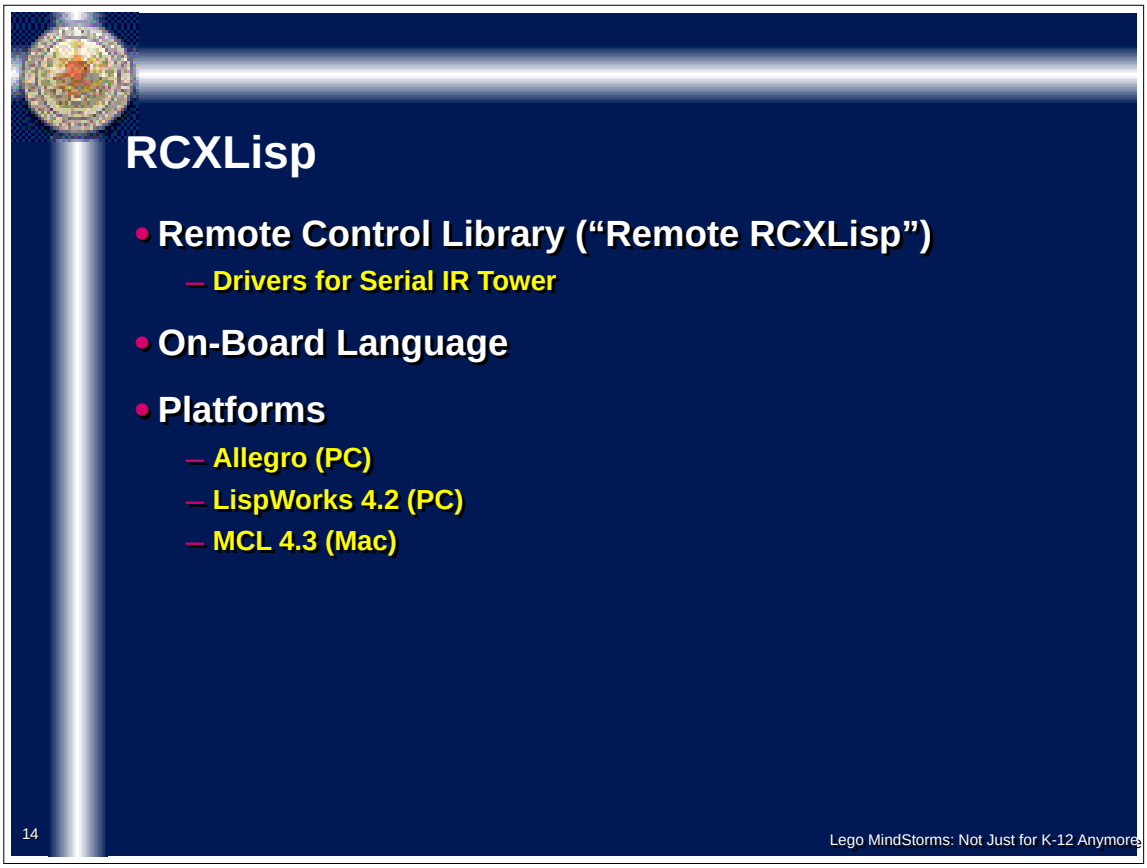
Other Non-Lego Programming Solutions

- NQC (Baum)
  - Lego firmware-based
- LeJOS (Solarzano)
  - Native JVM
- LegOS (Noga)
  - C kernel
- Lego Scheme (Jadud)
  - Native!


Weaknesses
• Hardware
– 32 KB RAM
– all onboard environments suffer!
• Firmware
– 32 global variable registers
– 16 local thread variables
– No call stack
– Broadcast IR protocol
11
Lego MindStorms: Not Just for K-12 Anymore


Weaknesses.... & Solutions
• Hardware
– 32 KB RAM
– all onboard environments suffer!
• Firmware
– 32 global variable registers
– 16 local thread variables
– No call stack
– Broadcast IR protocol
• Split Language Environment
– Remote (Desktop) & Local
– On-the-fly compilation/download
• Keep Firmware
– Bonus: local/remote mixing
– Hack: RCX ID register
– Hack: “point-to-point protocol”
– Live with register limitations
– Suffer: no recursion on RCX
12
Lego MindStorms: Not Just for K-12 Anymore

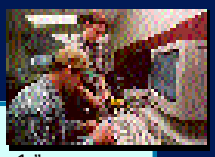

More Than MindStorms (MTM)
• NSF DUE CCLI #088884
– http://robotics.csc.villanova.edu
• mNet Firmware
• JavaRCX
– RCXJavaC
– JRCX (remote API)
– PC & *nix, no Mac yet
• RCXC++
– Cleans Up LegOS
• RCXLisp
13
Lego MindStorms: Not Just for K-12 Anymore


RCXLisp
• Remote Control Library (“Remote RCXLisp”)
– Drivers for Serial IR Tower
• On-Board Language
• Platforms
– Allegro (PC)
– LispWorks 4.2 (PC)
– MCL 4.3 (Mac)
14
Lego MindStorms: Not Just for K-12 Anymore

Remote RCXLisp

- Macros
  - with-open-port, with-open-rcx-stream
  - using-rcx
  - “remote” setf
- Remote Functions (~40)
  - alivep, battery-power, change-rcx-id, play-tone, clock, set-clock, set-effector-state, set-sensor-state, shutdown, start-timer, sensor, var, set-var, start-rcx-thread, stop-rcx-thread, set-transmit-range, ...
- On-the-Fly Compiler
  - rcx-compile-file, rcx-compile-formlist, load-executable, read-executable
- Multiple RCXs from Same Port (Threadsafely!)

Remote RCXLisp Example



```
(DEFUN full-speed-ahead (s dir)
“Make RCX go at speed S in direction DIR until touch sensor on its ‘2’ port returns a 1.”
  (LET ((result 0))
    (with-open-com-port (out  2) ;we assume serial port 2 is where the IR tower is plugged in
      (with-open-rcx-stream (str out :rcx-unit 10)
          (set-effector-state ‘(:A :B :C) :power :off str) ;turn everything off
          (set-effector-state ‘(:A :C) :speed s str)
          (set-effector-state ‘(:A :C) :direction dir str) ; :forward, :backward, or :toggle
                  ; no motion will occur until the next call to set-effector-state
          (set-sensor-state 2 :type :touch :mode :boolean :stream str)
          (set-effector-state ‘(:A :C) :power :on str)
          (LOOP ;this loop will repeat forever until sensor 2 returns a 1
            (SETF result (sensor 2 :default str))
            (WHEN (AND (NUMBERP result) ;prevent error if “sensor” call returns nil.
                       (= result 1))
                (RETURN)))
          (set-effector-state ‘(:A :C) :power :float str)))))
```

Remote RCXLisp Examples

```
(DEFUN full-speed-ahead2 (r s dir)
"The rcx in R goes at speed S in direction DIR until touch sensor on its '2' port returns 1."
  (LET ((result 0))
      (using-rcx r
           (set-effector-state '(:A :B :C) :power :off)
           (set-effector-state '(:A :C) :speed s)
           (set-effector-state '(:A :C) :direction dir)
           (set-sensor-state 2 :type :touch :mode :boolean)
           (set-effector-state '(:A :C) :power :on)
           (LOOP
             (SETF result (sensor 2))
             (WHEN (AND (NUMBERP result) ; prevent error if "sensor" call times out.
                        (= result 1))
               (RETURN)))
          (set-effector-state '(:A :C) :power :float)))))

(DEFUN test ()
   (with-open-port (p 1)
       (with-open-rcx-stream (rcx1 p :rcx-unit 1)
           (with-open-rcx-stream (rcx2 p :rcx-unit 2)
               (full-speed-ahead2 rcx1 5 :backward)
               (full-speed-ahead2 rcx2 1 :forward)))))
```

17

Lego MindStorms: Not Just for K-12 Anymore

Remote RCXLisp Examples

```
(DEFUN test-one (port)
  (with-open-rcx-stream (rcx port :retries 4 :rcx-unit 1)
       (FORMAT t "~%RCX1 var 3--->~d~%" (var 3 rcx))
       (play-tone 800 250 rcx) (play-system-sound 4 rcx) (set-var 3 80 rcx)))

(DEFUN test-five (port)
  (with-open-rcx-stream (rcx port :retries 4 :rcx-unit 5)
        (FORMAT t "~%RCX2 var 3--->~d~%" (var 3 rcx))
        (play-tone 300 250 rcx) (play-system-sound 3 rcx) (set-var 3 100 rcx))))

(DEFUN trial1 ()
"This function will invoke each test in sequence."
  (with-open-port  (portstream  1)
     (test-one portstream) (test-five portstream)))

(DEFUN trial2 ()
 "This function creates a thread for each test. Each test uses the same port."
  (with-open-port (p 1)
     (LET ((a (mp::process-run-function "fn1" nil #'test-one p))
           (b (mp::process-run-function "fn2" nil #'test-five p)))
      (LOOP
         (WHEN (AND (NOT (mp::process-alive-p a)) (NOT (mp::process-alive-p b)))
           (RETURN))))))
```

18

Lego MindStorms: Not Just for K-12 Anymore

Local RCXLisp

- Forms
  - defmacro, defvar, defconstant
  - defthread, defregister
- RCX-Specific Functions
  - most “remote” functions, sans stream arg
  - send-message: contact other RCX or desktop
  - sleep: pause a thread for a time period
- Common-Lisp Functions & Features
  - integer & bitwise arithmetic, comparison, base prefixes, let/let*
  - cond, if, when, loop, return, dotimes

Local RCXLisp Example

```
(defconstant *receiver* 1)
(defregister 4 *LIMIT* 16)

(defthread (signaller) ( )
  (LOOP
    (send-message 78)
    (sleep 15) ;; leave IR port silent for short time in case desktop is sending a message.
  ))

(defthread  (alpha :primary t)   ( )
  (LET ((diff1 0)
        (diff2 0))
   (set-sensor-state *receiver* :type :light :mode :raw)
   (SETQ diff1 (ABS (- (sensor *receiver* :raw) (sensor *receiver* :raw))))
   (SETQ diff2 (ABS (- (sensor *receiver* :raw) (sensor *receiver* :raw))))
   (start-rcx-thread signaller)
   (LOOP
     (WHEN (>= (ABS (- diff1 diff2)) *LIMIT*)
       (play-tone 500 1))
     (SETQ diff1 diff2)
     (SETQ diff2 (ABS (- (sensor *receiver* :raw)
                         (sensor *receiver* :raw)))))))
```

Communication & Coordination

- Local RCXLisp
  - defregister, accept-only, send-message
- Remote RCXLisp
  - set-var, set-message
- Remote Monitoring
  - threading feels natural
  - variety of vendor approaches to threads problematic

Experiences

- Inside the Classroom

- Outside the Classroom
  - 15 Laboratory Projects (PF, OS, AI)
  - Workshops (SIGCSE & CCSC)

Future Development

- USB IR Support
- Remote Programming (all langs)
  - timeout handling for RCXstreams
- mNet Firmware
  - support local event handling opcodes
  - add thread prioritizing support
- LeJOS mixed local/remote control
  - extend RCXJavaC
- Native Lisp or Lisp VM on RCX
  - Brooks' L?