TI | Developer Conference
February 15-17, 2005 • Houston, TX

SEE THE FUTURE. CREATE YOUR OWN.

A Very Inexpensive Robot and Microcontroller Board based on the MSP430 for use in the Classroom

UNC CHARLOTTE

James M. Conrad
Associate Professor
UNC Charlotte
jmconrad@uncc.edu

Technology for Innovators™

Texas Instruments

Agenda

- Development – classroom efforts:
  - Motivation for an inexpensive controller
  - Hardware/software requirements
  - Design and implementation
- Final Board: Classroom use
  - Book/Board use in classroom
- Demonstration
- Additional design thoughts and conclusion

Problem Statement for Project

Design and build a microcontroller-based printed circuit board that:

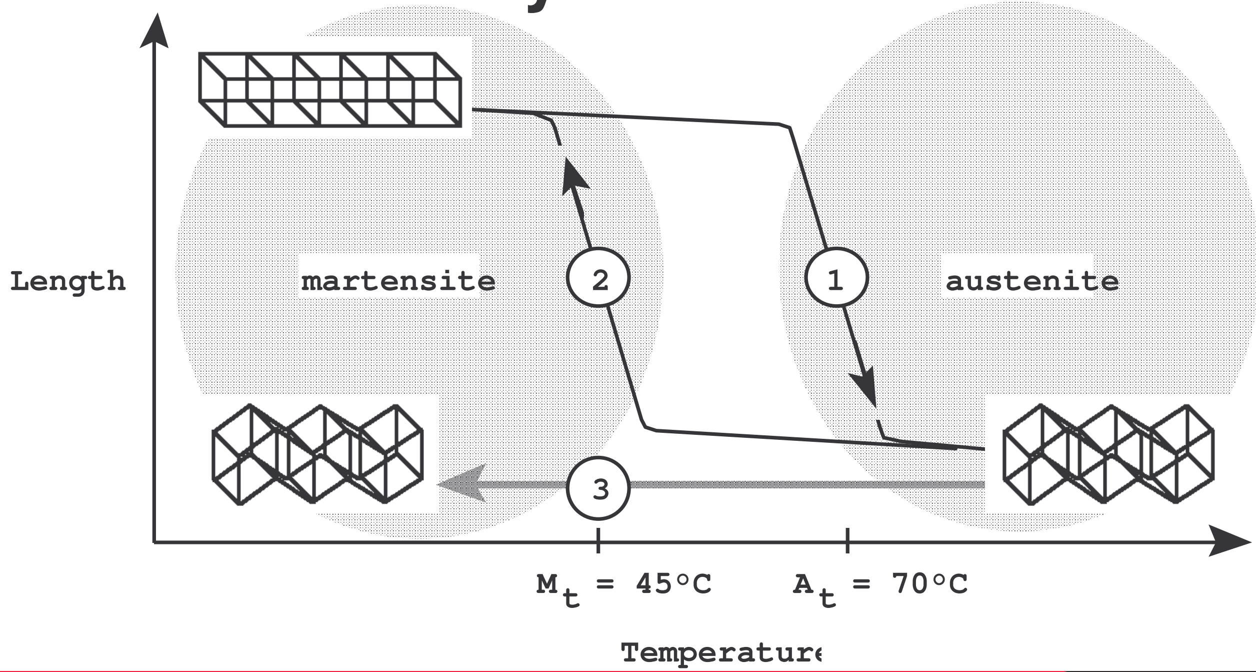
1. Will control the small robot Stiquito
2. Is inexpensive (<$8 to build)
3. Is programmable and expandable so it can be used in the classroom
4. Is suitable for 9^{th} grade and above

Stiquito - What Is It?

- Invented by Jonathan Mills, CS Department, Indiana University, in 1992
- Hexapod (six legs)
- Small - can sit on a credit card (75mm x 70mm x 25mm, 10g)
- Inexpensive ($3.00 in mass quantities), easy-to-build
- Can carry about 50g of weight
- Travels using a “Nitinol” muscle

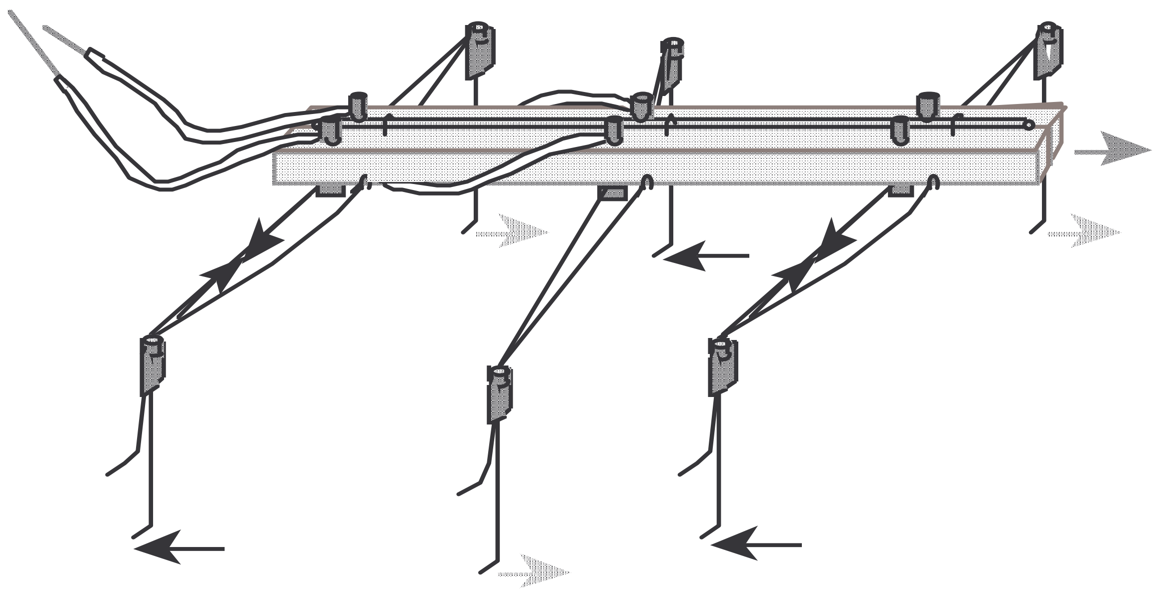
Nitinol - What Is It?

- Alloy of nickel and titanium
- Contracts when heated
- When cooled, must be “stretched” back to its original size
- Lasts millions of cycles



TI Developer Conference

Stiquito - How Does It Work?



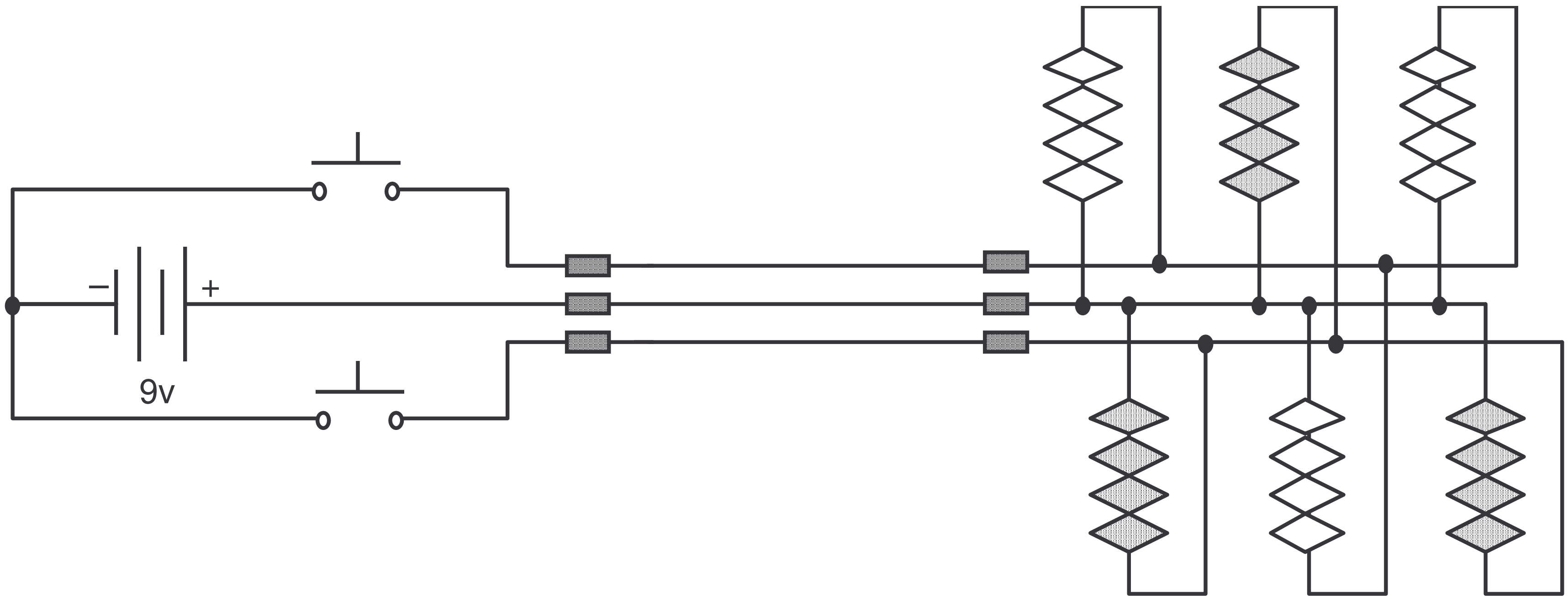
nitinol wire actuator contracts

leg catches surface as it bends backward

other legs slide forward

robot moves forward

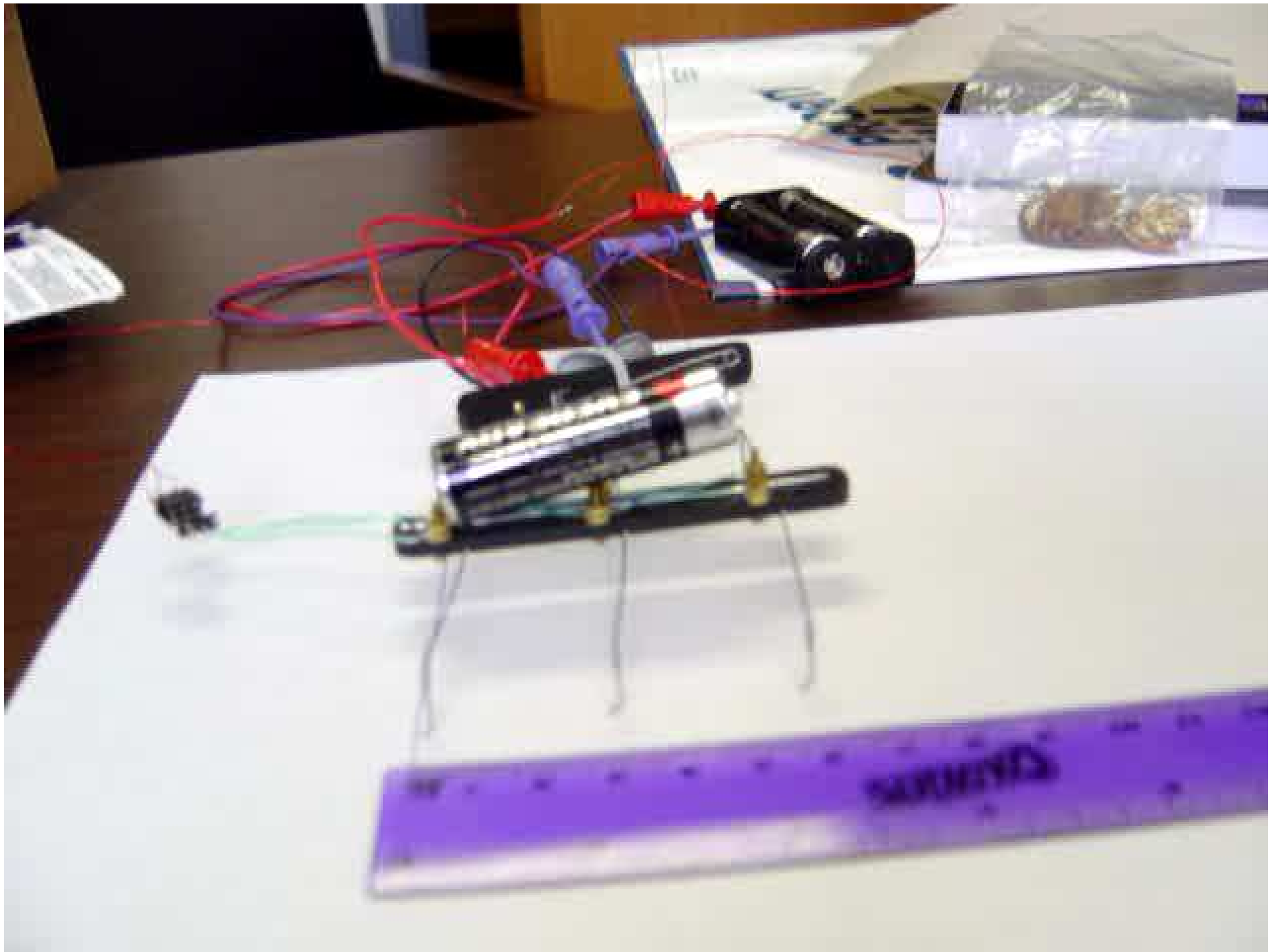
Controlling Stiquito Manually



- Simple operation, no components, two switches, requires tether

TI Developer Conference

The Manual Motion . . . .



Origin of Controller Board

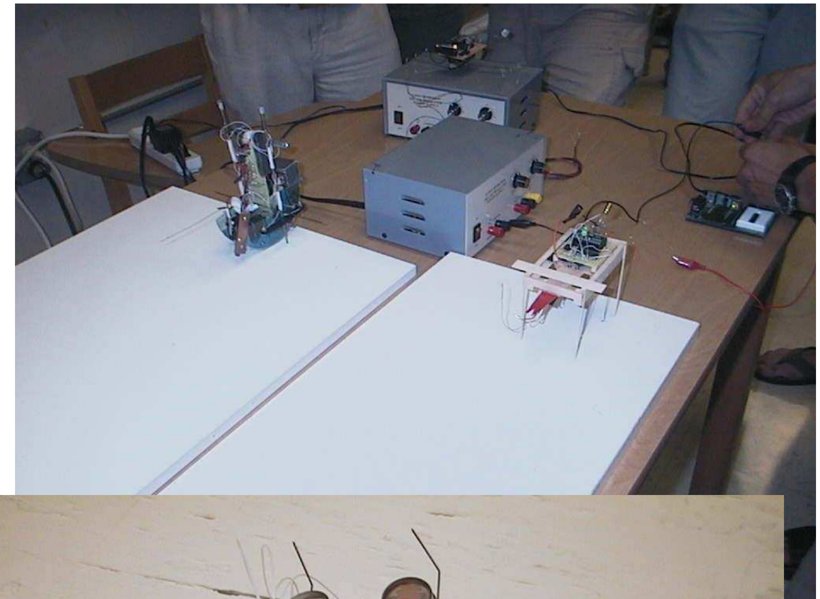
An engineering class at North Carolina State University was given the assignment to design and implement a functional robot for a race. The rules for the race and robot were that the robot:

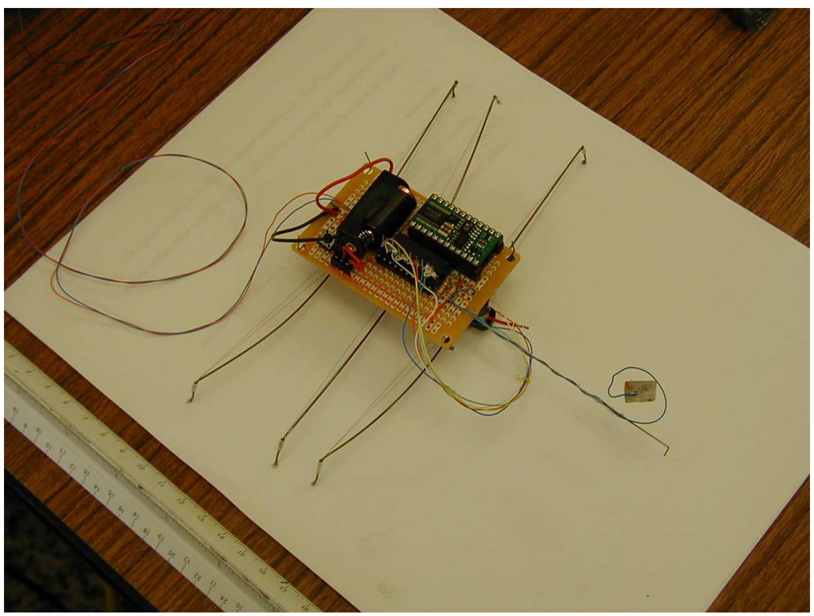
- must use Flexinol® (nitinol) for locomotion
- must use legs in its propulsion (two or more legs required)
- must walk four times its length in the fastest amount of time on smooth Formica
- must have an on-board microprocessor
- must measure no greater than 12 by 12 inches
- would be allowed to have either external (tethered) or attached power supply

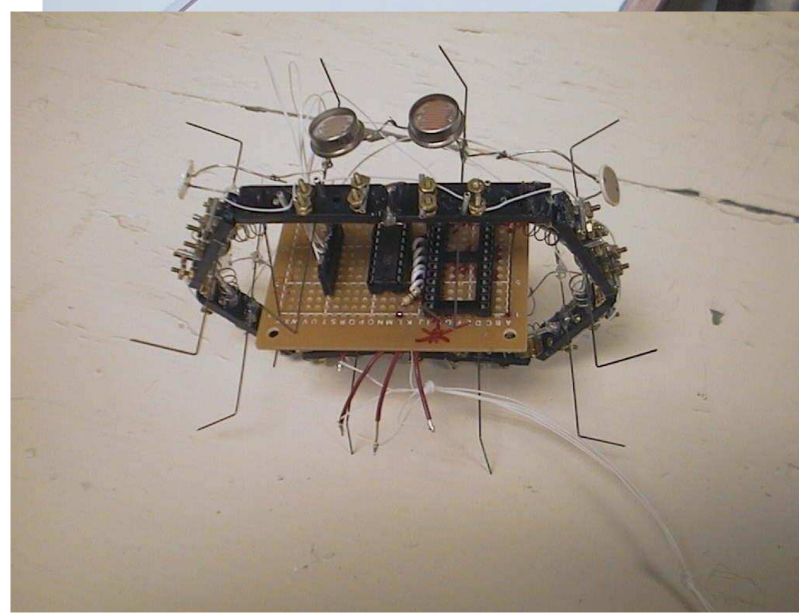
TI | Developer Conference

The Class Design Results

- **Several innovative designs were created, many were . . . . unusual:**







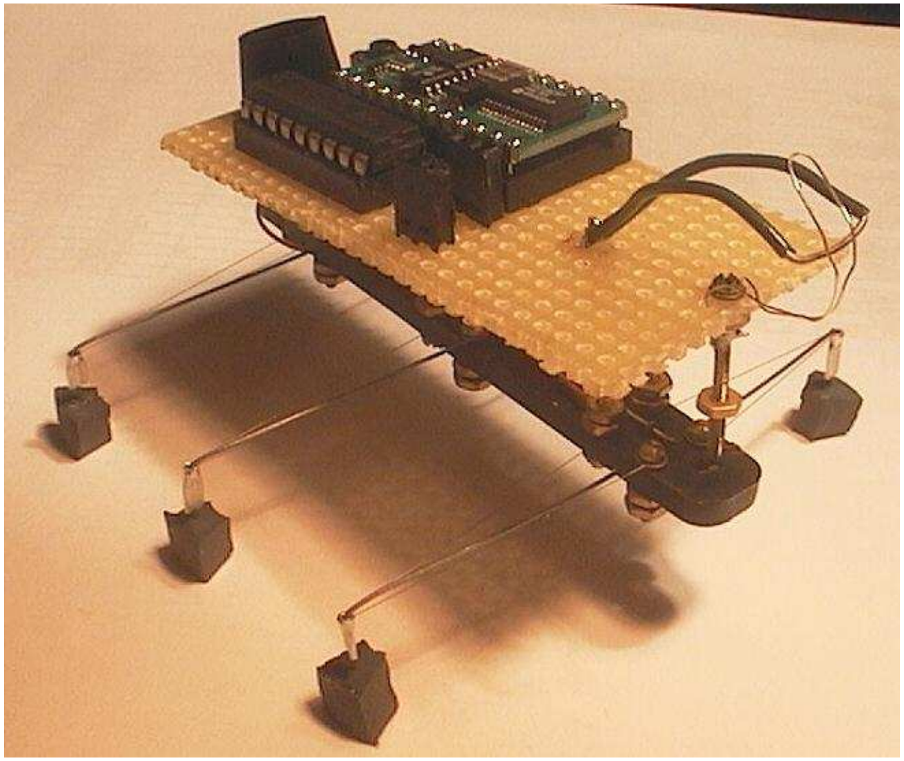
TI Developer Conference

The Class Made The Local News!

- Video clip from WRAL, Raleigh, NC

The Preferred Version

- The most successful version was the simple Stiquito with Parallax Basic Stamp 2.
- This microcontroller is not cheap!



Device Requirements

In order to design a product, one must first identify the requirements of the device. We determined that the Stiquito Controller board needed to have the following functionality and fulfill the following system requirements:

- The Stiquito Controller Board shall be designed as inexpensively as possible.
- All electronic parts on the board are to be the most cost-effective possible, with consideration of materials and assembly.
- It is anticipated that surface mount components will be needed.

TI Developer Conference

Device Requirements (2)

- At a minimum, the embedded system shall consist of a microcontroller, a transistor driver for the Flexinol® legs, a potentiometer for adjusting gait speed, a connection for power, and two LEDs for output of gaits
- The printed circuit board shall:
  - attach to the Stiquito Body
  - be a common, inexpensive epoxy-resin material with copper plating and solder masking
  - have a prototype area at one end. It shall have plated through-holes, 0.035" diameter, 0.1" spacing, the width of the board, at four rows long.

Microcontroller Requirements

The microcontroller shall:

- run with a supply voltage of 2.7 to 3.9 v
- have at least 512 bytes of internal programmable non-volatile memory storage (EPROM, EEPROM, Flash) and 32 bytes of RAM storage
- have the ability to be reprogrammed in the factory and by users. The printed circuit board shall have the necessary circuitry to support this reprogramming

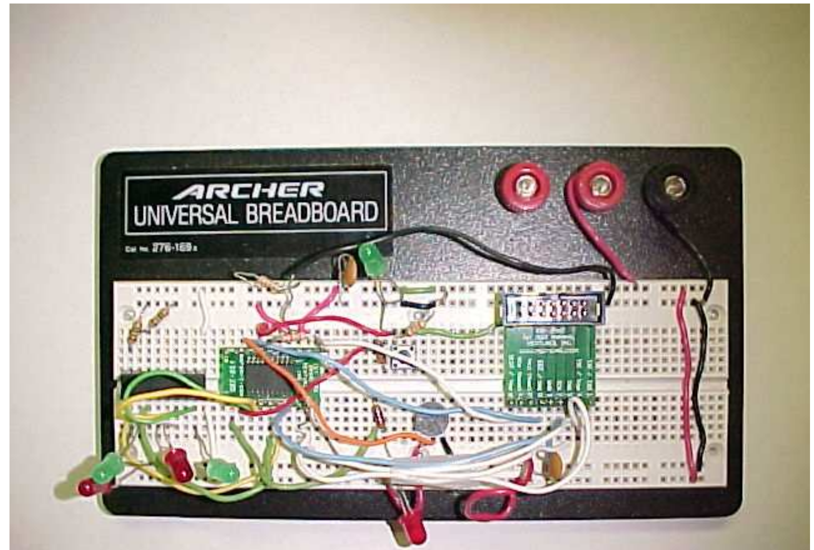
Microcontroller Requirements (2)

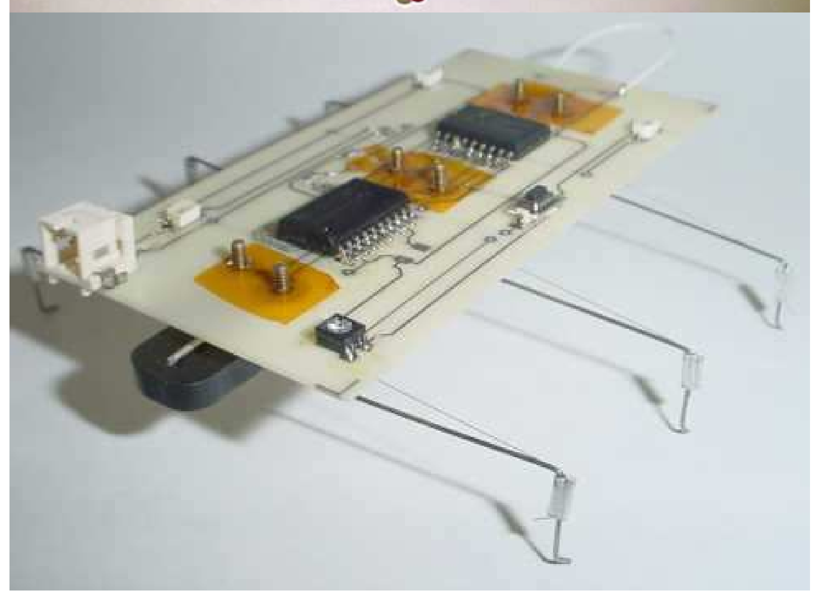
The microcontroller shall:

- have four outputs available for driving Flexinol® legs in a one-degree-of-freedom configuration and a two-degree-of-freedom configuration
- have two outputs available for driving LED's
- have at least one input for determining one or two-degree of freedom operation
- have at least one analog input for the measurement of a potentiometer for determining gait speed with at least one A/D converter for this analog input.

Senior Design Project

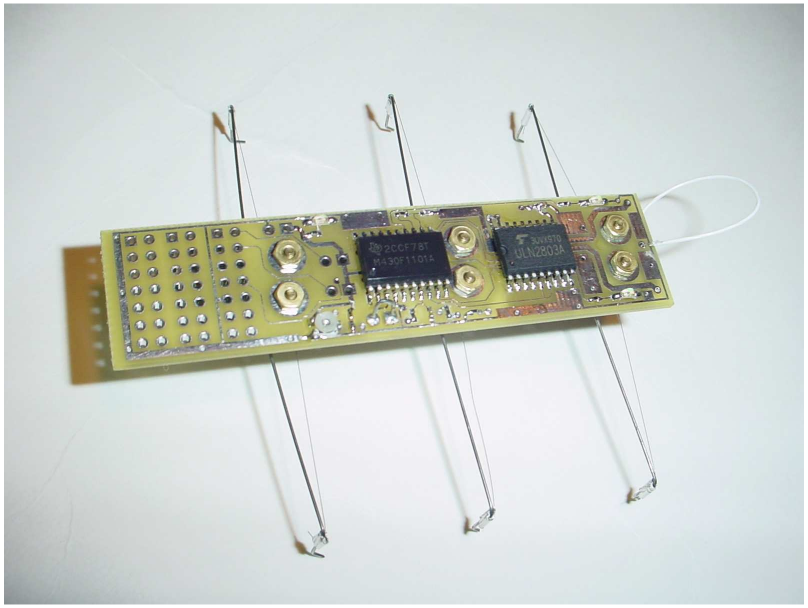
- After investigating low-cost microcontrollers by Microchip (PIC), Renesas, Texas Instruments, and others, we decided that the TI MSP430F1101 was an appropriate device.
- A bread board circuit and a prototype printed circuit board were created.





The Value of Prototyping

- Through other projects, students prototyped and programmed variations and discovered improvements.
- Through their investigations, they found that A/D on the MSP430F1101 was difficult to implement.
- They also ran out of flash space!



TI Developer Conference

Design Decision – New µC

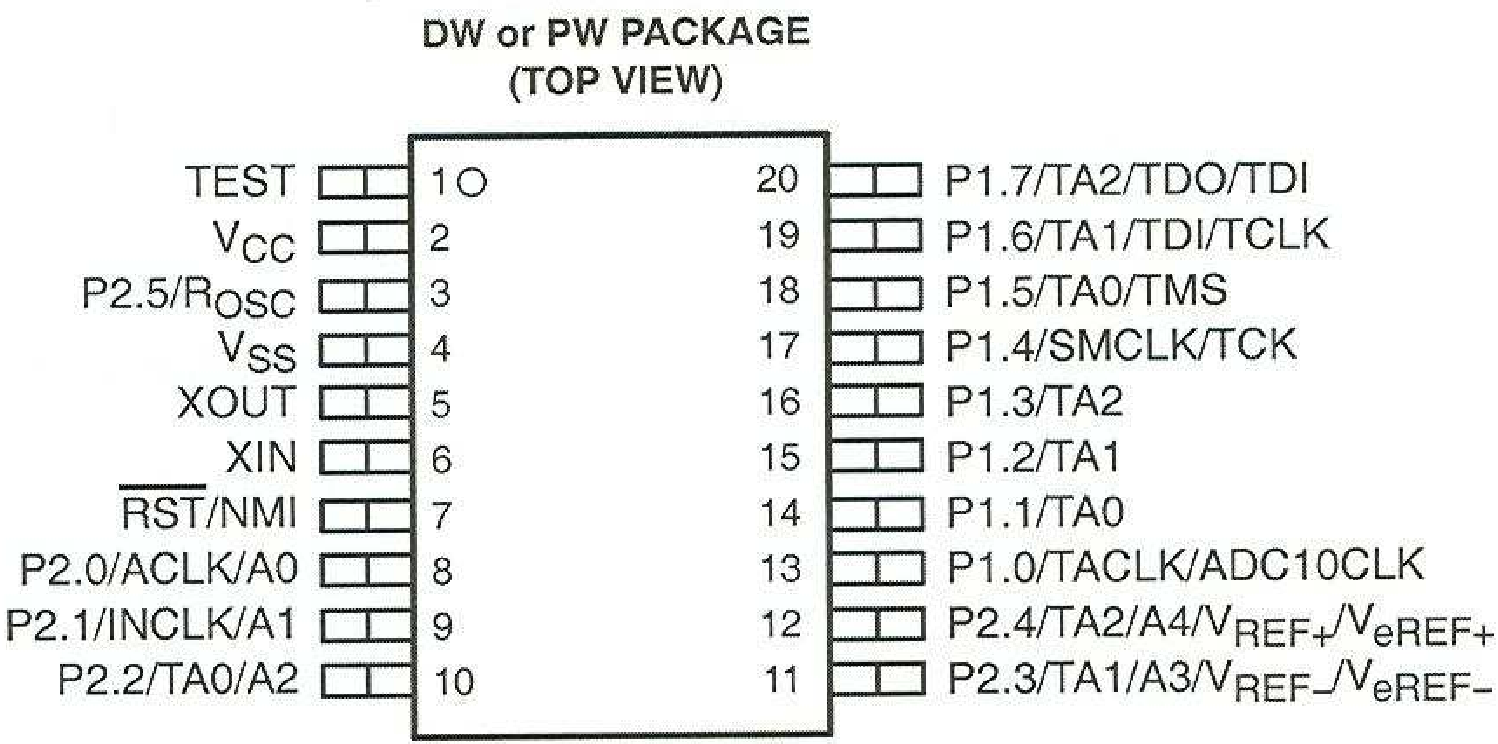
- The MSP430F1122, was selected to expand the size of flash memory, to add a true and simple A/D converter, and to add brown-out control.
- This is a good example of how, during the design process, building a prototype and evaluating the solution showed a significant change to the requirements was needed.

Features of the MSP430F1122

- Low Power Consumption
  - Active Mode ~200 µA
  - Standby Mode 0.7 µA
- 16-Bit Timer
- 10-Bit, 200-ksps A/D Converter
- Serial Onboard Programming (JTAG)
- Supply Voltage Brownout Protection
- 4KB + 256B Flash Memory (256B RAM)

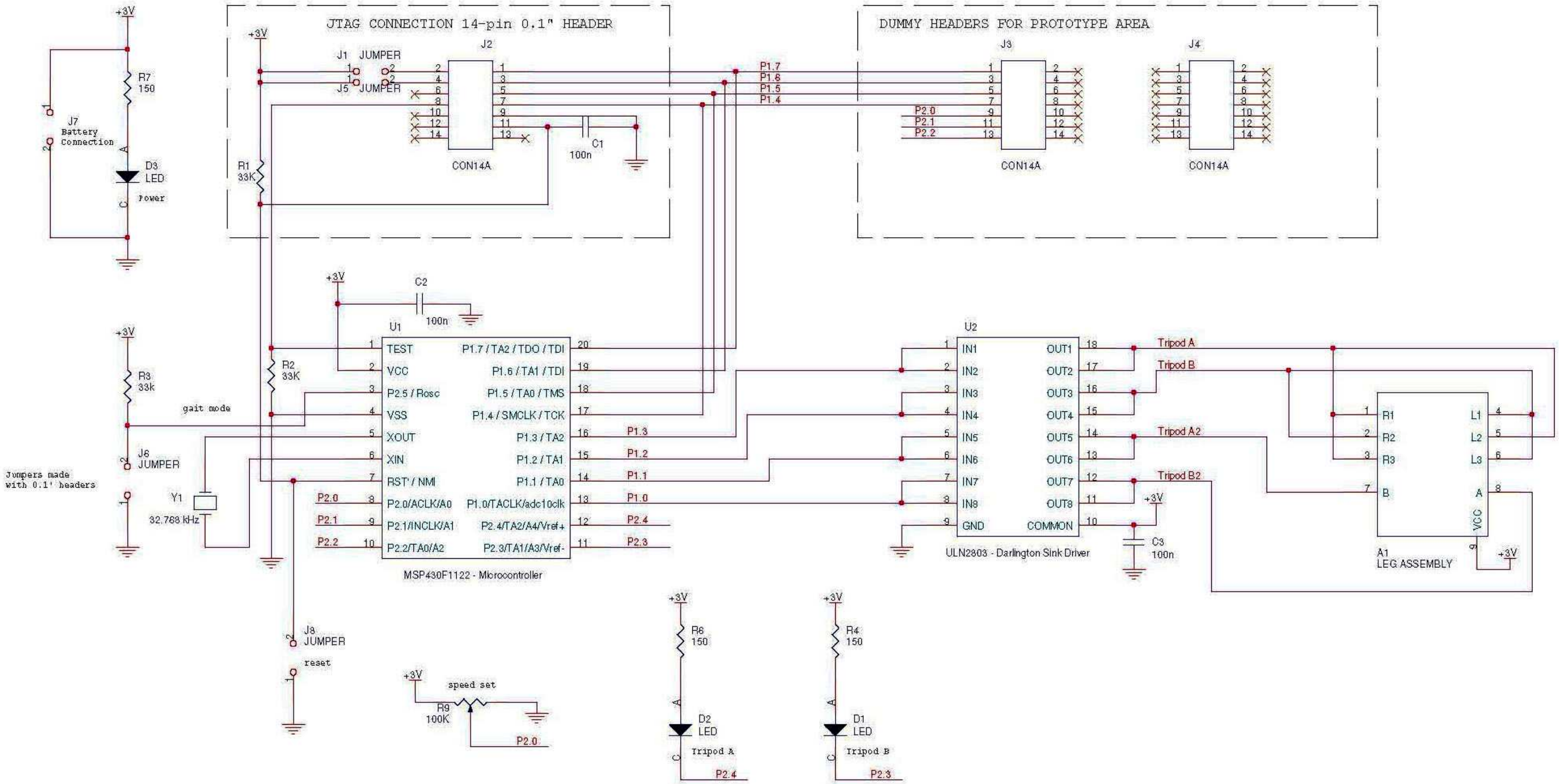
TI Developer Conference

Pin Out of the MSP430F1122

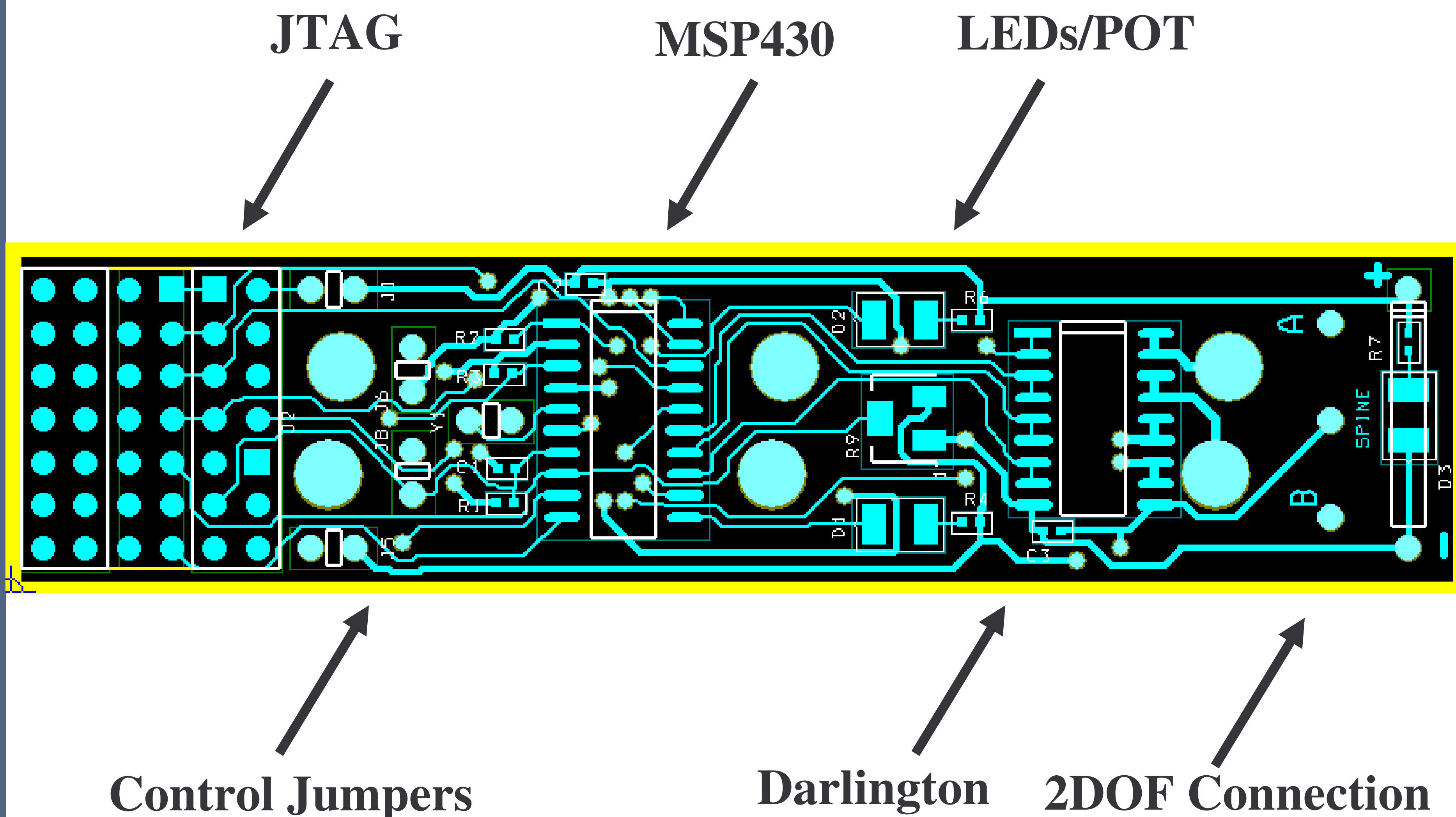


TI Developer Conference

Schematic of Board

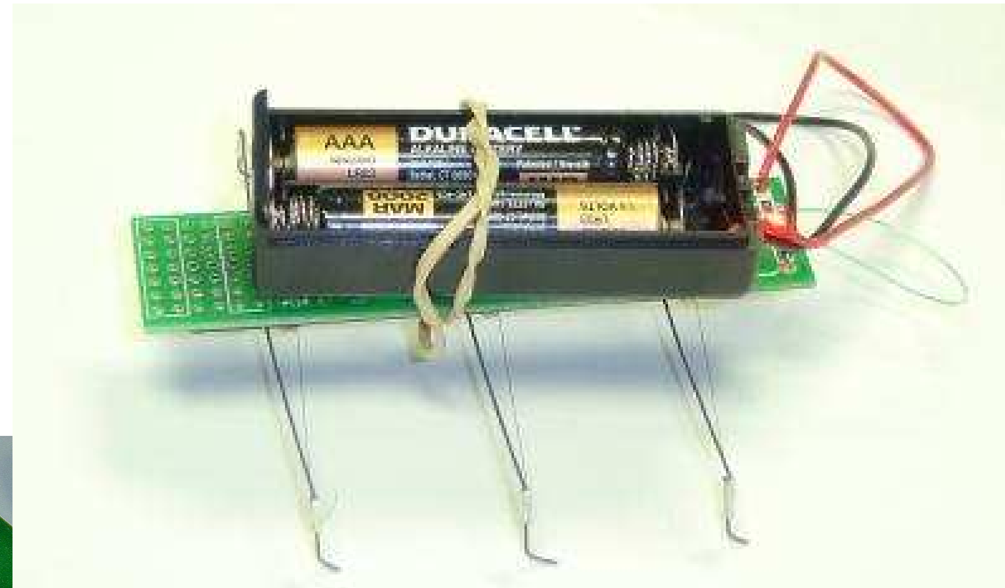


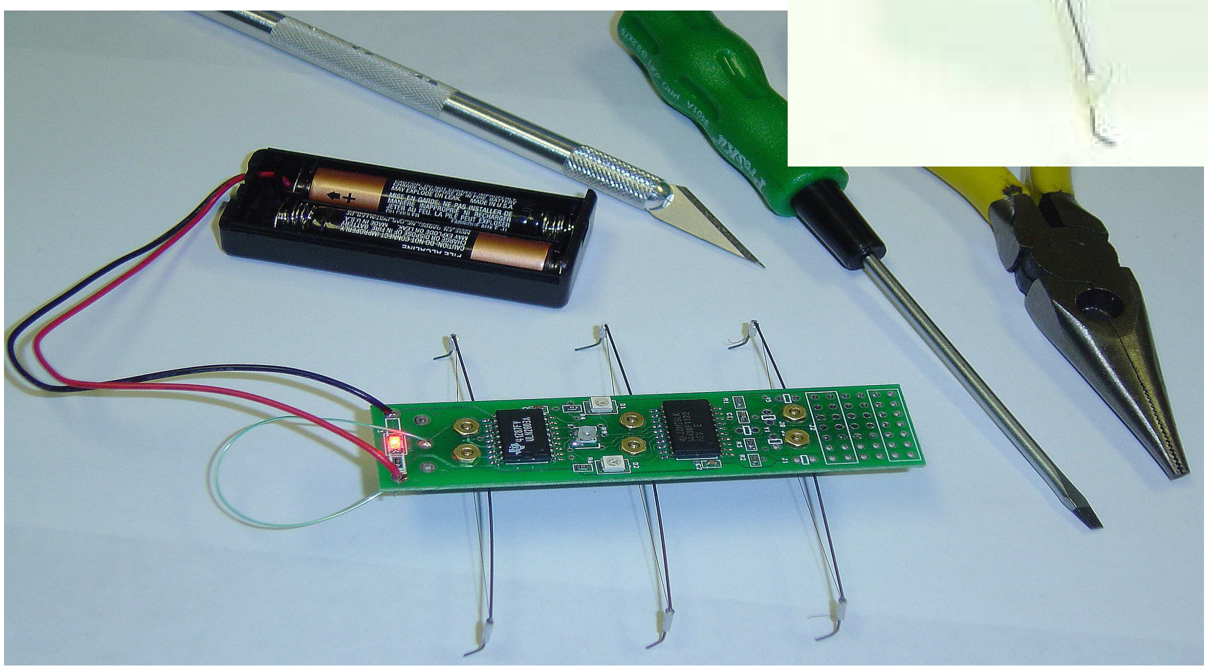
Board Layout



TI Developer Conference

The Final Classroom Robot





Low Power Mode

_BIS_SR(LPM0_bits + GIE);

- This command is executed after initialization, and instructs the processor to power down and wait for interrupts. The processor automatically "wakes up" whenever the timer interrupt occurs, and automatically returns to standby when the execution of that interrupt service routine is completed.

TI Developer Conference

Set Slow Speed

```
BCSCTL1 &= ~(RSEL0 + RSEL1 + RSEL2);
DCOCTL  &= ~(DCO0 + DCO1 + DCO2);
```

- These commands set different internal clocks to their slowest respective speeds.

Code for the Ports

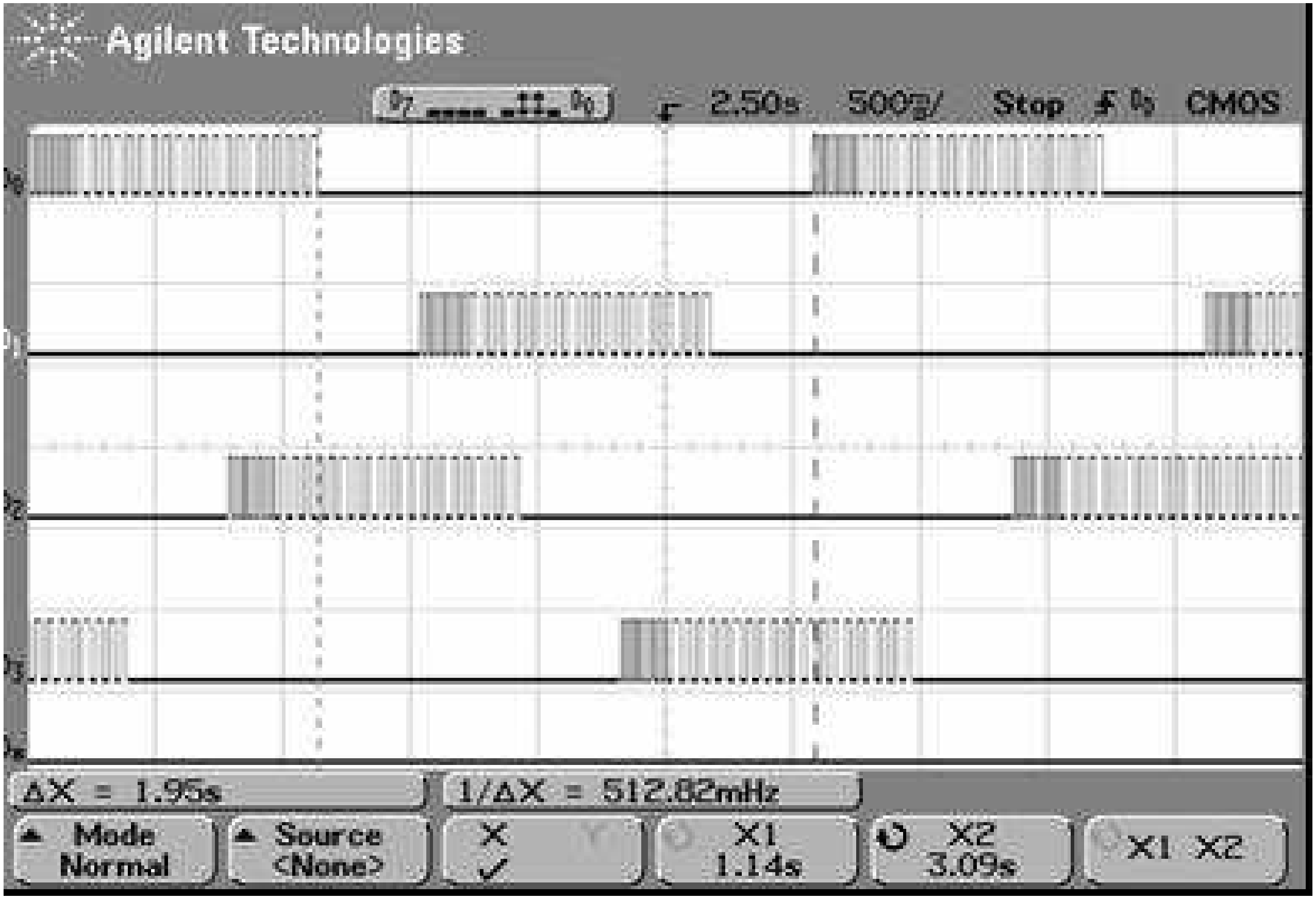
```
// P1 is set as follows:
// P1.1:  Tripod A 2-dof - output
// P1.0:  Tripod B 2-dof - output
// P1.2:  Tripod B - output
// P1.3:  Tripod A - output
// P1.4 till P1.7 : JTAG -Input
P1DIR = 0x0F;                    // Init P1.x
P1OUT = 0x00;                    //turn off legs and other outputs

// P2 is set as follows:
// P2.0: ADC input from POT to adjust speed
// P2.2 and P2.0 : routed to Prototype Area
// P2.3:  Tripod B LED - output
// P2.4:  Tripod A LED - output
// P2.5:  2-dof jumper - input
P2DIR = 0x1E;    // Init P2.x
P2OUT = 0x00;    //turn off legs and other outputs
P2OUT |= 0x18;  // turn off leds (inverse logic)
```

Using the A to D Converter

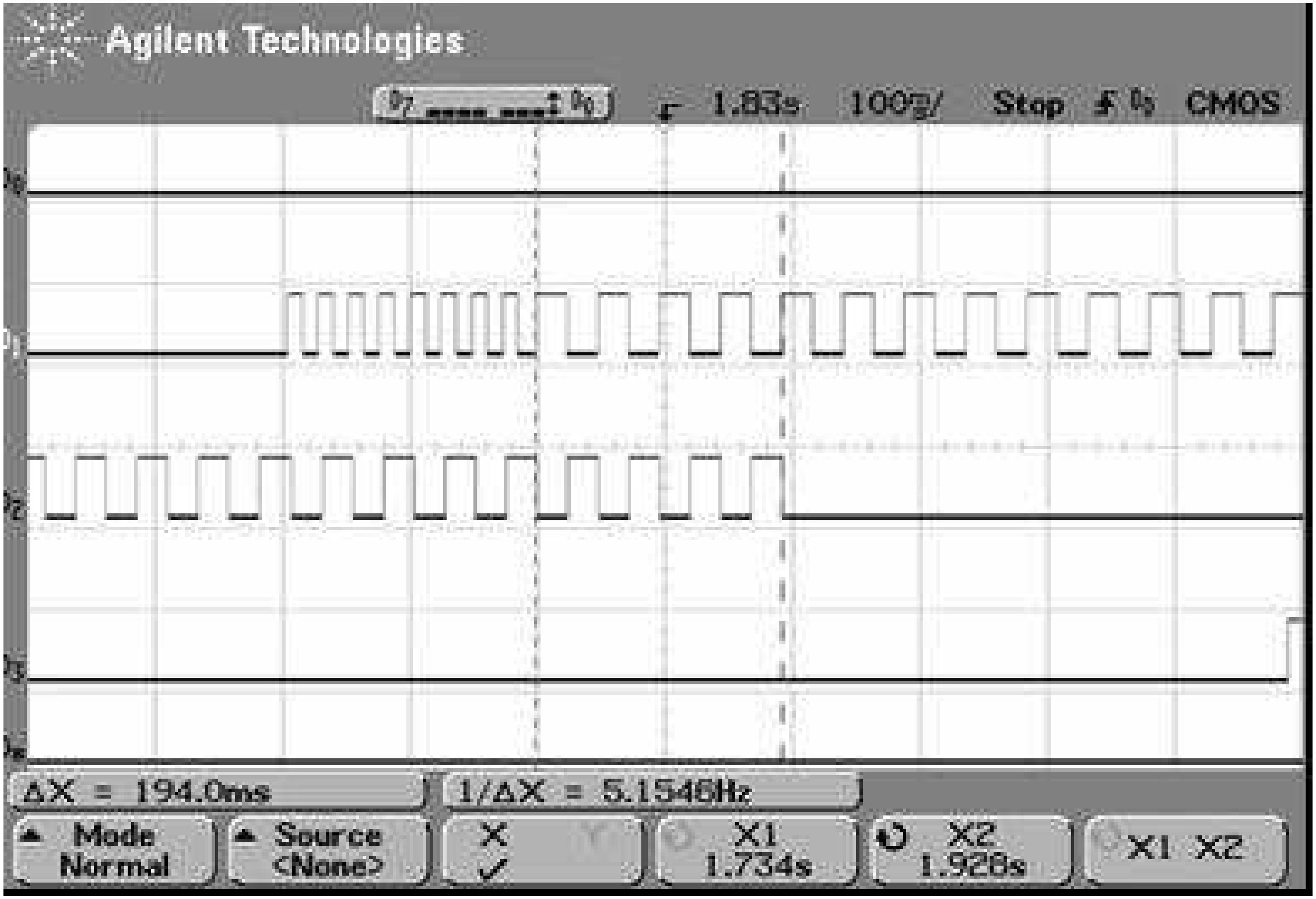
```
ADC10CTL0 = ADC10ON;            // ADC10ON
ADC10AE |= 0x01;                // P2.0 ADC option select
ADC10CTL0 |= ENC + ADC10SC;     // Sampling open

while ((ADC10CTL1 & ADC10BUSY) == 1);

m=ADC10MEM & 0x03FF;            // 10 bits ADC value
```

Operation – Two degree of freedom

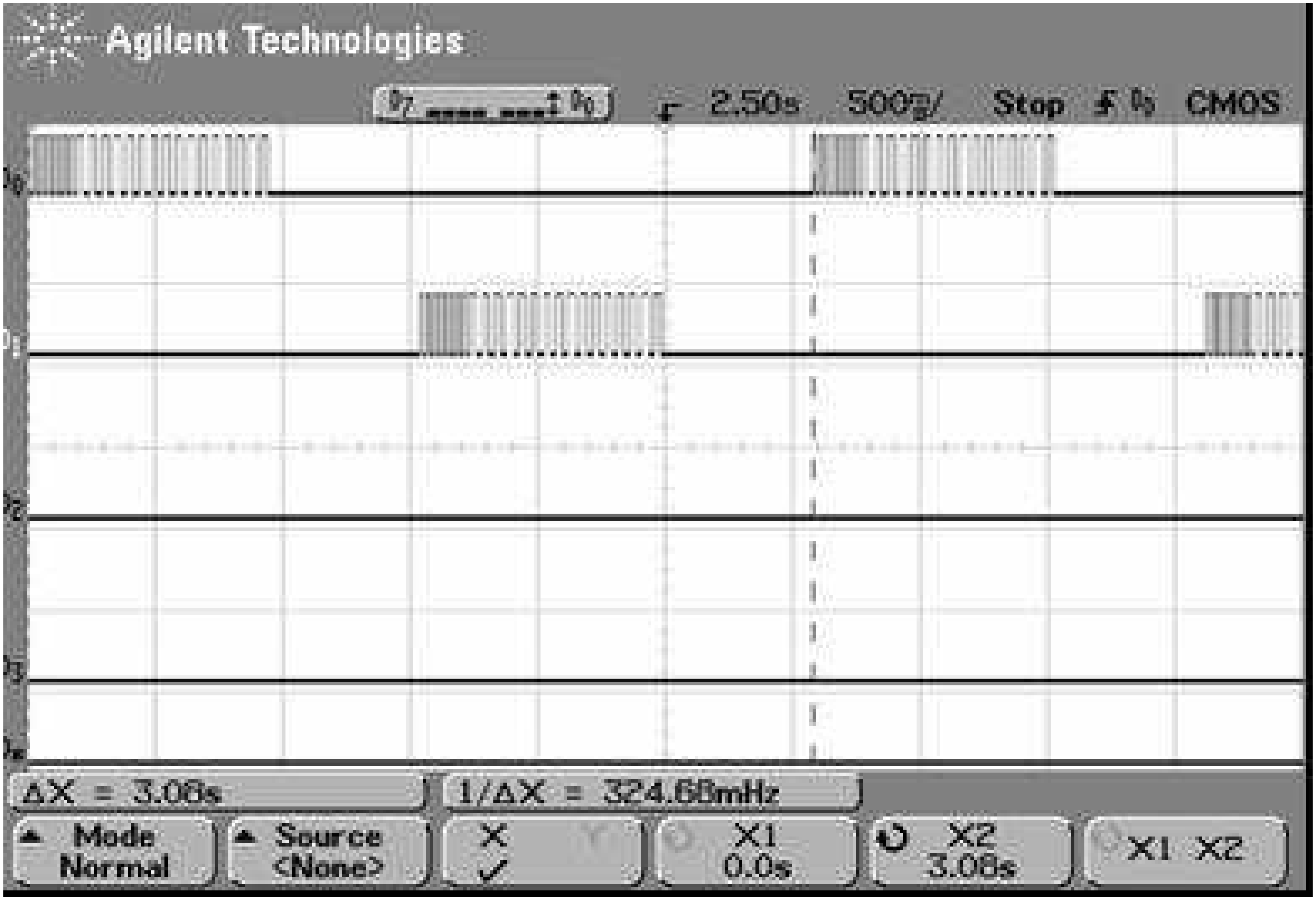
- Notice the two different tripods



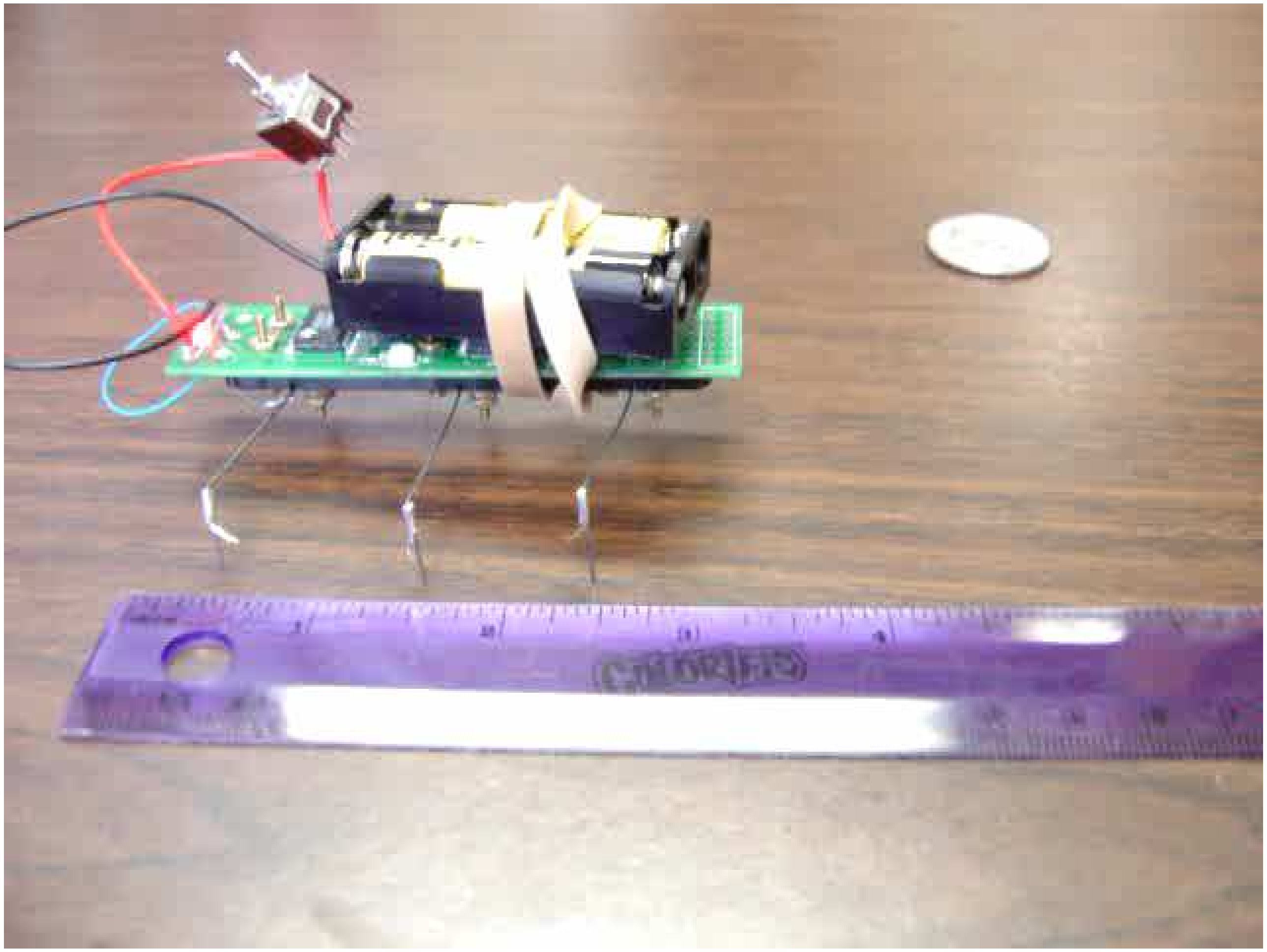
More Detail on Two Degrees



One degree of freedom

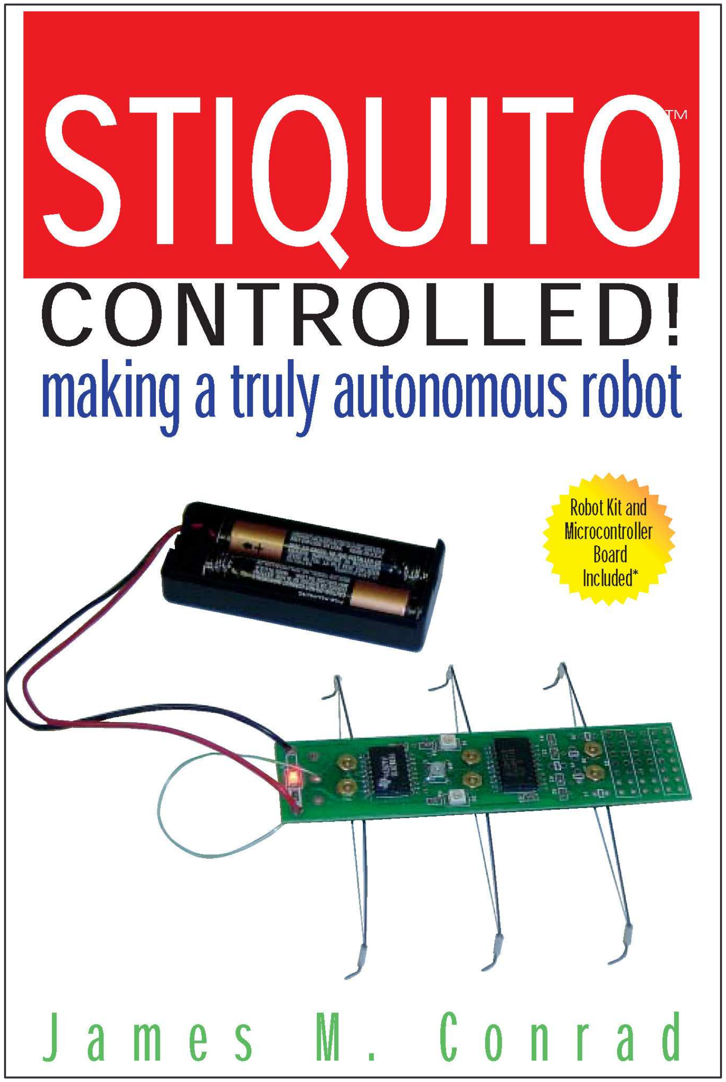


And Does it Walk?

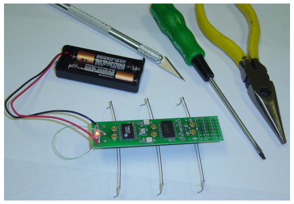


The Result of the Design Work

- Due to student efforts, the robot design was built and tested.
- Students wrote several papers
- All included in a new book, *Stiquito Controlled*
- Book can be used in classrooms
- Book includes robot and board



Book with Electronics

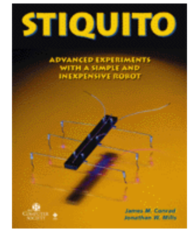


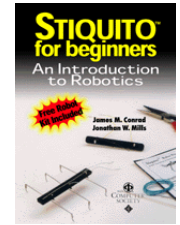
Stiquito(tm) Controlled! Making a Truly Autonomous Robot

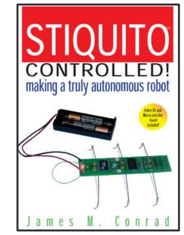
- **Chapter 1: An Introduction to Robotics and Stiquito**
- **Chapter 2: Introduction to Embedded Systems and the Stiquito Controller Board**
- **Chapter 3: PCB Layout and Manufacturing**
- **Chapter 4: Building Stiquito Controlled**
- **Chapter 5: Stiquito Programming using Texas Instruments MSP430F1122**
- **Chapter 6: A Two-Degree-of-Freedom Stiquito Robot**
- **Chapter 7: Optimizing the Stiquito Robot for Speed**
- **Chapter 8: More Stiquitos Controlled**

TI Developer Conference

Book Availability







- ***Stiquito**^tm^**: Advanced Experiments with a Simple and Inexpensive Robot***, **ISBN 0-8186-7408-3, Wiley Books.**
- ***Stiquito**^tm^ **for Beginners: An Introduction to Robotics***, **ISBN 0-8186-7514-4, Wiley Books.**
- ***Stiquito**^tm^ **Controlled! Making a Truly Autonomous Robot,*** **ISBN 0-4714-8882-8, Wiley Books, available everywhere.**

Why use Stiquito in the Classroom?

- Students learn best with hands-on activities.
- The walking mechanism is not complex. The analog circuit is not complex.
- Low cost - students pay for the supplies (the kit is in the book). Only one working robot needs to work between the three in a group.
- Immediate feedback on success or failure of design.
- Open ended: the kit is built. Now what?
- Exposure to several CompE and EE subareas (power, control, circuits, programming).

TI Developer Conference

How has UNCC Used Stiquito?

Have used in classroom to teach:

- Software development using Integrated Development Environments (design, coding, downloading, debugging)
- Hardware skills (soldering, interfacing, design, instrumentation)
- Power control/budgeting

How Others Use Stiquito

- New Jersey Institute of Technology, Biomedical Engineering sponsors a "Pre-Engineering Instructional and Outreach Program", which features Stiquito robot building
  - http://www.njit.edu/old/PreCollege/PrE-IOP/events.php
- Texas A&M has used Stiquito in its Introduction to Engineering and Problem Solving course (ENGR111)
  - http://crcd.tamu.edu/curriculum/engr111/stiquito/index.html
- Western Michigan, ECE 123 Mobile Robotics: An Introduction to Electrical and Computer Engineering
  - http://homepages.wmich.edu/~miller/ECE123.html
- Numerous Projects, i.e.
  - http://www.me.psu.edu/me415/fall99/stiquito/intro.html

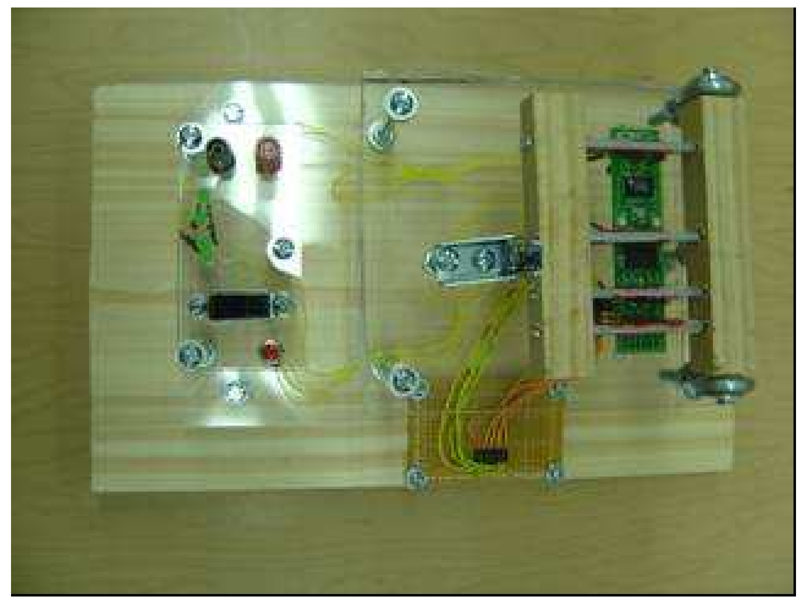
Possible Other Uses of Stiquito

- Once Stiquito has been introduced in the curriculum, it can be used to demonstrate other technologies learned in other classes:
- Advanced design: microcontroller systems
- Control systems
- Programming
- Manufacturing PCBs

- Stiquito can also be used for cross-disciplinary efforts (ME/EE/CompE)
- Create contests, races, competitions

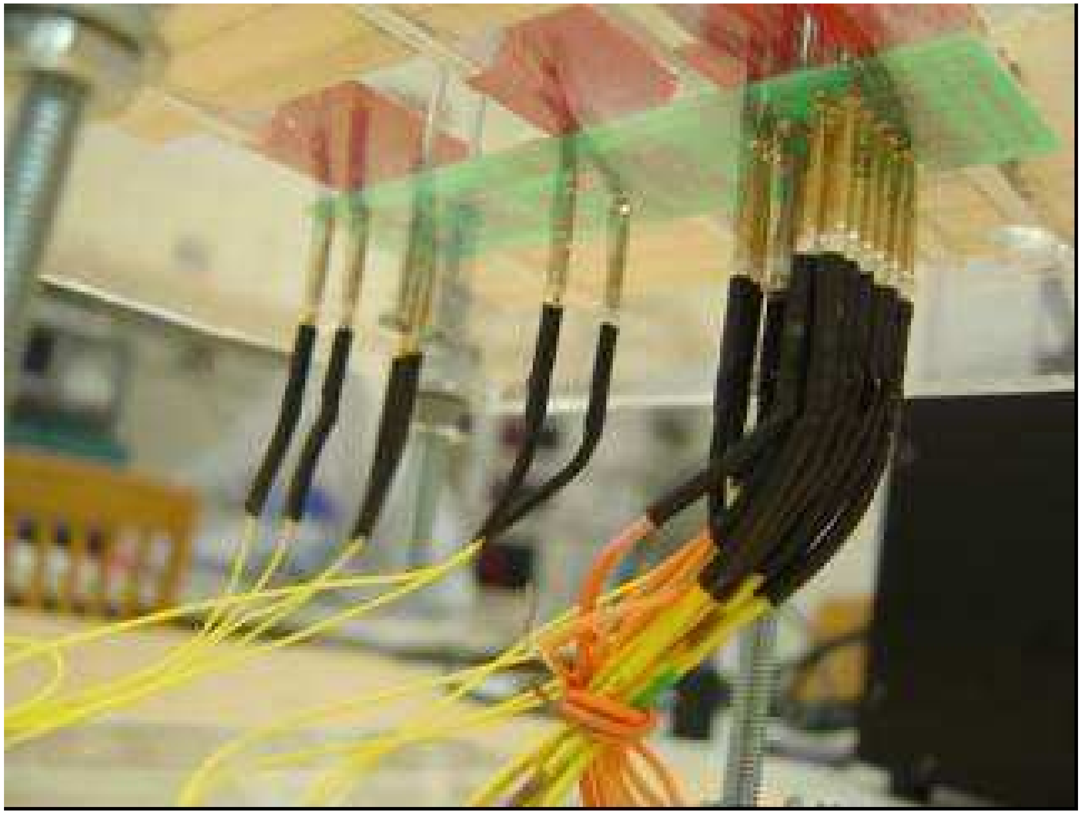
An Example of "Project"

- One class group created a test fixture to program and test the PCB
- Studied test theory, identified board test points, developed process, built fixture



Test Fixture

- A view from underneath the circuit board under test
- Most pins are associated with the JTAG port



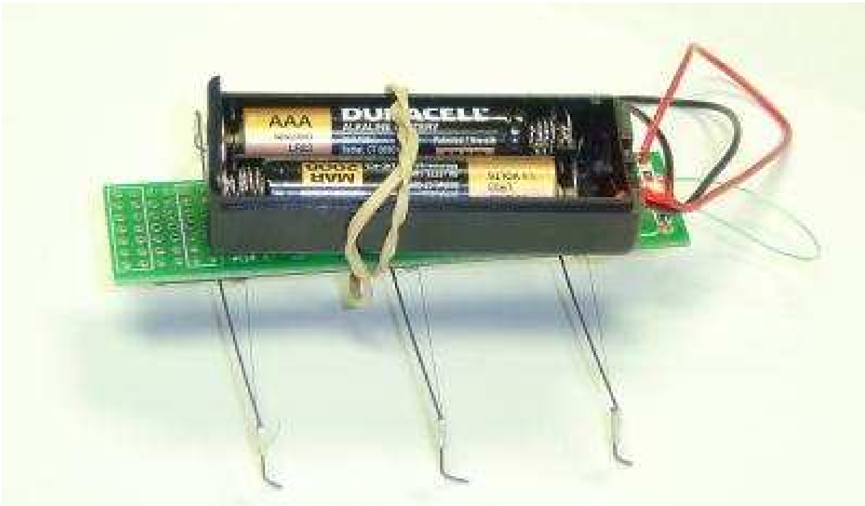
Demonstration of Robot

- Demo: Development/download of code
- Demo: Walking Stiquito – 1-degree-of-freedom
- Demo: Walking Stiquito – 2-degree of freedom

TI Developer Conference

Conclusion

- Stiquito is an inexpensive platform to learn about engineering concepts.
- Making Stiquito walk requires knowledge of different engineering disciplines.
- Stiquito can be controlled manually or by a microcontroller



- A book with an enclosed PCB and Stiquito are available for classroom use

TI Developer Conference

Future Efforts

- Improve/optimize gait
- Publish lab exercises with attached sensors, actuators, and communications
- Make easier to build
- Make smaller (robot and electronics)
- Make new insects:

Get the Most Out of the MSP430

- Exhibit demonstration on Tuesday, Feb 16, 12-2 p.m.
- www.coe.uncc.edu/~jmconrad
- TI MSP430 University support:

  Go to www.ti.com, then navigate to

  TI Home->Microcontrollers->MSP430 MCUs->University Program
- MSP430 day, April 30, 2005, in many cities in the US. Visit the TI Training website:

  http://focus.ti.com/docs/training/traininghomepage.jhtml

James M. Conrad
Associate Professor
UNC Charlotte
jmconrad@uncc.edu

Get to market faster with TI products, support and partners.

UNC CHARLOTTE