The ATPC-X42
All-Terrain Power Chair

Niaz Khan
Selome Mandefro
Alex Mann
Vikram Shenoy
Autodesk CAD Model

- This is the current state of the Autodesk model of the ATPC-X42
Autodesk CAD Drawing

- Different views of the CAD model
Chassis

- Pre-made chassis taken from a Quickie S-626 power chair

- Material: Aluminum
- Dimensions: 26 1/8” L x 18.5” W x 2” D
Modified Subunit: Seat Mount

- 1 inch spacers, seen in red, will be inserted in the seat mount to widen the base of the seat from 14.5” to 16.5”
Seat Mount Spacer

- Will fit in between the rails and crossbars of the seat mount
- Material: Aluminum – easier to cut to specs than steel
- One-inch wide on either side of both crossbars
New Subunit: Front Casters

- Wider front casters are needed to accommodate wider front tires
- Material: Steel – must be able to withstand forces of ground impacts
- Current casters: 3”; New casters: 3.5”
Motors

- Current motors in Quickie S-626 power chair are:
  - Reliance Electric
  - Model #: E679
  - Part #: 679-011-039
  - Power: 2hp

- Were in working order when chair was picked up from NEAT Marketplace

- Chair tested on hill in front of Bronwell Building
  - Strong enough to go down and up under own power
Tires

- Front tires – rubber tubeless, off-road tires
  - 10” diameter x 3” width
  - Increased stability due to greater ground contact
  - Will fit into new front casters

- Rear tires – also rubber tubeless, off-road tires
  - 16” diameter x 4” width
  - Greater size and ground contact than current tires, which are rubber disks
Circuit

TO CONTROLLER

BATTERY BOX

70 AMP CIRCUIT BREAKER PUSH TO RESET

BLACK

BLACK

BLACK

ORANGE

FRONT BATTERY

REAR BATTERY
Circuitry

- The previous highlight the circuitry used for the All Terrain Power Chair.
- The batteries are connected in series – 24V
- The batteries use quick connectors to the circuit for easy replacement.
- There will be an auto actuation circuit that will be built to enable automatic actuation of the seat.
Joystick

- The main joystick controls the chair’s motion.
- It has the on/off switch as well and has the main stick used to control movement.
- It allows for precise movement depending on the tilt of the joystick.
- Speed control is also available from about 1 mph to the maximum 7 mph.
- There is a bottom charger/programming port.
Controller

- Penny & Giles Pilot+
- Powered via the 24 volt batteries and controls both motors and receives signals from joystick
- The controller can be reprogrammed using a Qtronix Programmer and connects to the charging port on the joystick.
- There isn’t code that is actually written for convenience
Actuator

- Linak Actuator
- Operates with 24V, 4.5A
- Max Load 4000N
- Duty Cycle 10% (6min/hr)
- Adjusts tilt of seat with the use of hinges on the chassis
- Has its own controller that connects to the main P&G controller
- Will be used in auto adjusting circuit
Batteries

- 2 PowerSonic 12V batteries
- Group24 75Ahr
- Capable of powering the chair for the entire day without slowing down
- Sealed rechargeable Gel Cell battery for ease of use and quality
Battery Cage

- The batteries will sit in the pre-made battery cage of the Quickie S-626
- It is made of steel and is strong enough to support the weight of each 51 lb battery
Charger

- Lester Electrical 24V Dual mode automatic battery charger
- Delivers 8 Amps to charge the batteries
- Is able to plug directly into any wall socket for ease
- LED indicators tells the user when the batteries are <80%, >80% and 100%.
- Also reports any faults if any
Self Adjusting Actuation

- Use of an accelerometer (ADXL335) will be used to determine tilt.
- With the use of a microcontroller the voltage read from the accelerometer will be translated into tilt information using AD conversion.
- The actuator will then adjust accordingly.
The above circuit will be built and will contain the PIC along with the accelerometer. The wires from the actuator switch will be bypassed into the circuit when the user wants and auto adjustment will be done.
Schematic

- Microcontroller – PIC16F874
  - Will decipher the voltage from the accelerometer and convert that to tilt. The tilt will then control the actuator to auto adjust

- Accelerometer – ADXL335
  - An accelerometer that is powered off 3V. It will relay a voltage to the PIC that will be converted to tilt

- Transistors
  - They will be used as switches to control the actuator

- Oscillator
  - Necessary for timing of the PIC
The above shows what the PCB layout will look like
Tilt Sensor Code

- The code for the tilt sensor will be written in C and coded into the PIC
- The PIC will be responsible for analog to digital conversion of the accelerometer
- The tilt will then be calculated and the actuator will be controlled to accommodate for the tilt
- When the tilt reaches the critical angle (15°) a buzzer will sound to indicate the hill is too steep to traverse
Tilt Sensor Code

```c
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <pic.h>
#include <htc.h>

__CONFIG(DUNPROT & PWRTDIS & XT & WDTDIS & BORDIS & LVPDIS);
#ifndef _XTAL_FREQ
  // Unless already defined assume 4MHz system frequency
  // This definition is required to calibrate __delay_us() and __delay_ms()
  #define _XTAL_FREQ 7372800
#endif

void InitLCD(void);
void DisplayC(unsigned char position, const char *str);
void DisplayCharacter(unsigned char pos, unsigned char c);
float seatlevel, degree, y, volt, ydiff, yg, radians;

void Initial(void)
{
  ADCON1 = 0x44; // Select PORTA pins for ADC or digital I/O
  ADCS0 = 1; // Use A/D FOSC/8
  TRISA = 0x0B; // Set I/O for PORTA
  TRISB = 0xE1; // Set I/O for PORTB
  TRISC = 0xB7; // Set I/O for PORTC
  TRISD = 0x00; // Set I/O for PORTD
  TRISE = 0x04; // Set I/O for PORTE
  PORTD = 0; // Turn off LEDs
  InitLCD();
}
```
void ADConvert(void)
{
    CHS0 = 1; // Use channel AN1
    ADON = 1; // Turn A/D on
    __delay_us(30); // delay 30 usec to settle A/D acquisition
    ADGO = 1; // Start conversion
    while (ADGO); // wait for ADGO to go off signalling end of conversion
}

int main(void)
{
    float volt = (y/1023.0)*3; // 3 volt is the power supply, y is the returned value for 10 bits (0-1024)
    float ydiff = volt/zeroG; // zeroG can be measured using a multimeter at angle =0
    float yg = ydiff/accelaration;
    float radian = asin(yg);
    float degree = (180/PI)*radian;
    Initial();
    DisplayC(0x80,"Analog / Digital");
}

for (;;)
{
    ADConvert();
    if (degree > 90);
    P1B1T2EX= 1;
    return;
    elseif (degree < 90);
    P3B23INT1 =1;
    return;
    else (degree > 80 && degree > 100);
    PP2B2A10 = 1
    return;
}
## Budget

<table>
<thead>
<tr>
<th>Item</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quickie S626 Power Chair</td>
<td>$100</td>
</tr>
<tr>
<td>2 Power Sonic 12V Rechargeable Batteries</td>
<td>$150 ea.</td>
</tr>
<tr>
<td>Jay J2 Deep Contour Seat Back and Seat Cushion</td>
<td>$25</td>
</tr>
<tr>
<td>Lester Electrical 24V Dual Mode Battery Charger</td>
<td>$20</td>
</tr>
<tr>
<td>Harness and Foot Plate</td>
<td>$5</td>
</tr>
<tr>
<td>Rear Tires</td>
<td>$40 ea.</td>
</tr>
<tr>
<td>Rear Wheels</td>
<td>$18 ea.</td>
</tr>
<tr>
<td>Front Tires</td>
<td>$40 ea.</td>
</tr>
<tr>
<td>Front Wheels</td>
<td>$12 ea.</td>
</tr>
<tr>
<td>Microcontroller</td>
<td>$10</td>
</tr>
<tr>
<td>ADXL335 Accelerometer</td>
<td>$25</td>
</tr>
<tr>
<td>Metal for Fabrication of Joystick rotator and foot plate</td>
<td>$25</td>
</tr>
<tr>
<td>Miscellaneous Supplies (Screws, nuts, bolts, grease, paint)</td>
<td>$50</td>
</tr>
</tbody>
</table>
Budget Overview:

- Total Allotted: $1300
- Total Expected Cost: $780
- Total Spent So Far: $450

<table>
<thead>
<tr>
<th>Free Items</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Armrests</td>
<td></td>
</tr>
<tr>
<td>Actuator</td>
<td></td>
</tr>
<tr>
<td>Joystick</td>
<td></td>
</tr>
<tr>
<td>Upper and Lower Body Posture Constraints</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>$780</td>
</tr>
<tr>
<td><strong>Total Allotted</strong></td>
<td>$1300</td>
</tr>
</tbody>
</table>