AN964

Software PID Control of an Inverted Pendulum Using the PIC16F684

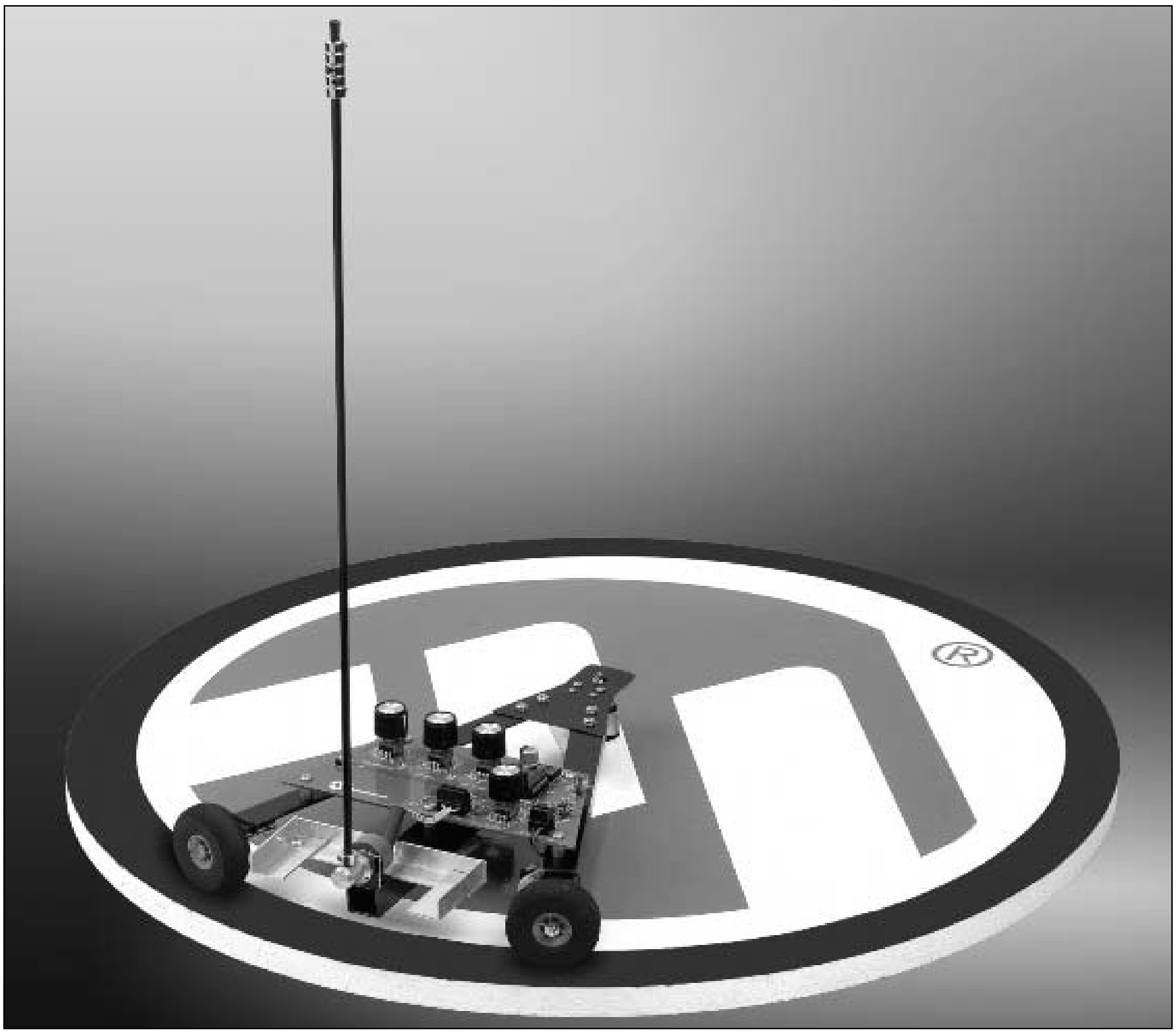
Author: John Charais
Ruan Lourens
Microchip Technology Inc.

INTRODUCTION

The purpose of this application note is to describe how a PIC16F684 can be used to implement a positional Proportional-Integral-Derivative (PID) feedback control in an inherently unstable system. An inverted pendulum is used to demonstrate this type of control.

The inverted pendulum consists of three main parts: the base platform, the pendulum and the controller board, as shown in Figure 1.

FIGURE 1: INVERTED PENDULUM



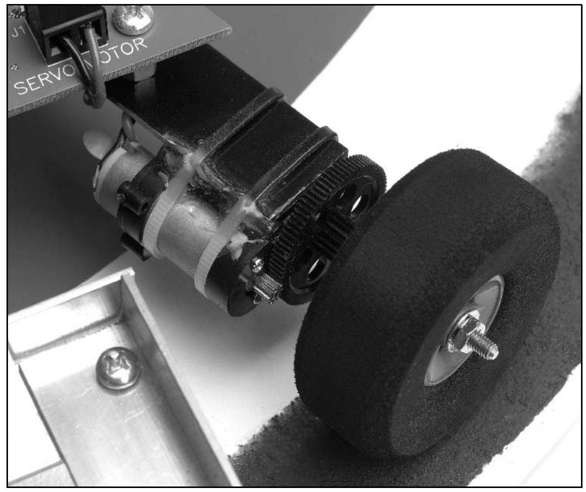
BASE PLATFORM

The base platform is a 3-point platform, 2 wheels (one of which is geared and attached to a DC motor) and an audio jack. When the DC motor is turned on, the base platform will rotate around in a circle with the center of the axis of rotation being the audio jack. The audio jack serves 2 purposes; first it is used as the axis of rotation for the base platform and second, it is used to bring commutated power to the controller board.

PENDULUM

The pendulum is attached to the base platform by a 360° free rotating potentiometer. The pendulum's base is attached to the potentiometer in such a fashion that when the pendulum is balanced (completely vertical), the potentiometer center tap is biased to $V_{REF}/2$. For the rest of this application note Θ will be used to denote the displacement angle of the pendulum with respect to the vertical axis.

FIGURE 2: MOTOR



CONTROLLER BOARD

FIGURE 3: CONTROLLER BOARD

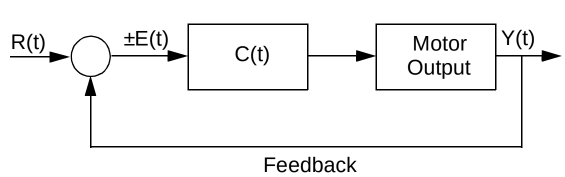
The controller board has 2 main functions, to measure Θ and to drive the DC motor. The power supply needed to run the system is dictated by the selection of the motor. The motor is controlled by an H-bridge which is driven by the PIC16F684 Enhanced Capture/Compare/PWM Module (ECCP). The outputs of the ECCP are connected to FET drivers that produce the proper drive voltages and reduce the transition times for the FETs in the H-bridge.

There are 5 potentiometers located on the controller board, 3 of which are used for adjusting the PID constants (K_P, K_I and K_D) and one to measure Θ. The fifth potentiometer is used in conjunction with the input filter's reference. The input filter is a low-pass Bessel filter with a cut-off frequency of 60 Hz and has a voltage gain of 6. A low-pass filter is needed to eliminate any high frequency noise on the angle measurement which the derivative term of the PID controller is extremely sensitive to. The Bessel filter is used because it has the best response to a step function. (Once the pendulum is balanced, a sudden displacement that causes it to become unbalanced will look like a step function.) The cut-off frequency was chosen to be at least twice the expected frequency of the pendulum. The gain of the filter was chosen to increase the resolution of the Analog-to-Digital (A/D) converter. With the 360° potentiometer and a 10-bit A/D converter, with no gain, one LSb equals 0.35°. With the gain set to 6, the displacement angle is limited to ±30° which gives a resolution of 0.059° per LSb. The fifth potentiometer controls the input filter's reference to produce a true 0° displacement angle when the pendulum is vertical. Without this potentiometer, any slight offset angle will cause the base to slowly increase its speed and eventually take the system into an unstable state. For more information on the controller board, see the schematics in **Appendix A: "Schematics"**.

PID

For the positional PID control system, Figure 4 is used to model the system.

FIGURE 4: PID CONTROL SYSTEM MODEL



R(t) = Reference of desired set point
Y(t) = Measured response
E(t) = Error
C(t) = Controller response

The desired set point R(t) of this system occurs when Θ = 0°. In this state, the pendulum is balanced. Since the desired response of the system is 0°, any angle measured other than 0° is the error or Y(t) = E(t).

In implementing the PID controller, there are 3 terms which are based off the error measurement.

Proportional Term: $K_P E(t)$ – where K_P is the proportional constant

Integral Term: $K_I \int_0^t E(t)dt$ – K_I is the integral constant

Derivative Term: $K_D dE(t)/dt$ – K_D is the derivative constant

EQUATION 1:

$$C(t) = K_P E(t) + K_I \int_0^t E(t)dt + K_D dE(t)/dt$$

In this system, the sign of the controller's output, C(t), will determine the direction in which the motor will turn. The magnitude of C(t) directly corresponds to the duty cycle of the PWM in the ECCP module, determining the speed at which the motor will turn.

PID IN A DIGITAL SYSTEM

Converting over to a digital system, Y(t) is measured by an A/D converter. In order to implement the PID controller, the PICmicro® microcontroller will have to do some approximations of integral and derivative terms. Starting with the derivative term, we can use the following difference equations for our approximation.

EQUATION 2:

$$dE(t)/dt \sim [E(n)-E(n-1)]/T_S$$

Where E(n) is the current error, E(n-1) is the previous error and T_S is our sampling period. Equation 2 is the approximate slope of the tangent line at E(t) (rise/run).

For the integral term use the approximation in Equation 3.

EQUATION 3:

$$\int_0^t E(t)dt \sim T_S \sum_0^N E(n)$$

With these approximations we can rewrite C(t) as shown in Equation 4.

EQUATION 4:

$$C(n) = K_P E(n) + K_I T_S \sum_0^N E(n) + K_D[E(n)- E(n-1)]/T_S$$

OR

$$C(n) = K(E(n) + (T_S/T_I) \sum_0^N E(n) + (T_D/T_S)[E(n)- E(n-1)])$$

Where $K_P = K$, $K_I = K/T_I$ & $K_D = K\ T_D$

MODELING THE INVERTED PENDULUM

In order to properly implement the control algorithm, the user needs to look at how the mechanical and electrical systems are going to interface together. Dynamic modeling the inverted pendulum is not a simple task. Here are some of the variables which need to be looked at in order to model the system:

- Bases' position
- Bases' velocity
- Bases' acceleration
- Bases' moment of inertia
- Bases' coefficient of friction
- Bases' mass
- Bases' length
- Earth's gravitational constant
- Pendulum's position
- Pendulum's velocity
- Pendulum's moment of inertia
- Pendulum's coefficient of friction
- Pendulum's mass
- Pendulum's length

In order to simplify all this, use one simple rule of thumb. Select a motor (with proper torque, rpm's and gear ratio to the drive wheel) that can accelerate the base platform as fast as the pendulum can fall. The angular acceleration of the pendulum with respect to the displacement angle is:

EQUATION 5:

$$\Theta'' = (g/R)\Theta$$

> **Note:** See **Appendix B: "Derivation of Equation 5"** for the derivation of this equation.

Since the acceleration of the pendulum is not constant, use the maximum acceleration of the pendulum when using this rule of thumb. The maximum acceleration of the pendulum will occur when Θ is at the largest angle, the controller will try and correct for (Θ_{MAX}). Θ_{MAX} is controlled by both hardware and software. The hardware boundary for Θ_{MAX} is set by the gain of the Bessel Filter; with a gain of 6, the limit is ±30º or ±0.523 radians. This can be further limited in the software. The software, accompanying this application note, further limits Θ_{MAX} to ±20º or ±0.349 radians. This is done to eliminate the possibility of hitting the hardware boundary.

With a pendulum length of 0.5 meters and Θ_{MAX} set to 20° or 0.349 radians.

From Equation 5:

$$\Theta''_{MAX} = (g/R)\Theta$$
$$\Theta''_{MAX} = (9.81/0.5)0.349$$
$$\Theta''_{MAX} = 6.845 \text{ radians/sec}^2$$
$$\Theta''_{MAX} = 3.425 \text{ meters/sec}^2$$

The motor used in this system that meets this criterion is from MAXX Products, Inc. The model number is EPU9, with an 8.6 to 1 gear ratio attached to a 2 inch wheel. The rated voltage for this motor is 4.8 to 7.2V, but for this example the motor will run at 12V. This is done to get a better response out of the motor at small duty cycles from the PWM. Running at approximately double the designated input voltage is not a concern because the motor will never be in a constant state where the duty cycle of the PWM is greater than 50%.

If finding a motor that meets this criterion is difficult, there are a few solutions. One solution is to decrease Θ_{MAX} in the software or increase the length of the pendulum; this will reduce the maximum acceleration of the pendulum. Another possible solution is to increase the coefficient of friction between the drive wheel and the base. If the motor is of ample size, the coefficient of friction of the drive wheel will be the limiting factor in how fast the base can accelerate. Change the coefficient of friction by changing the drive wheel to a different material or add an abrasive surface to the platform.

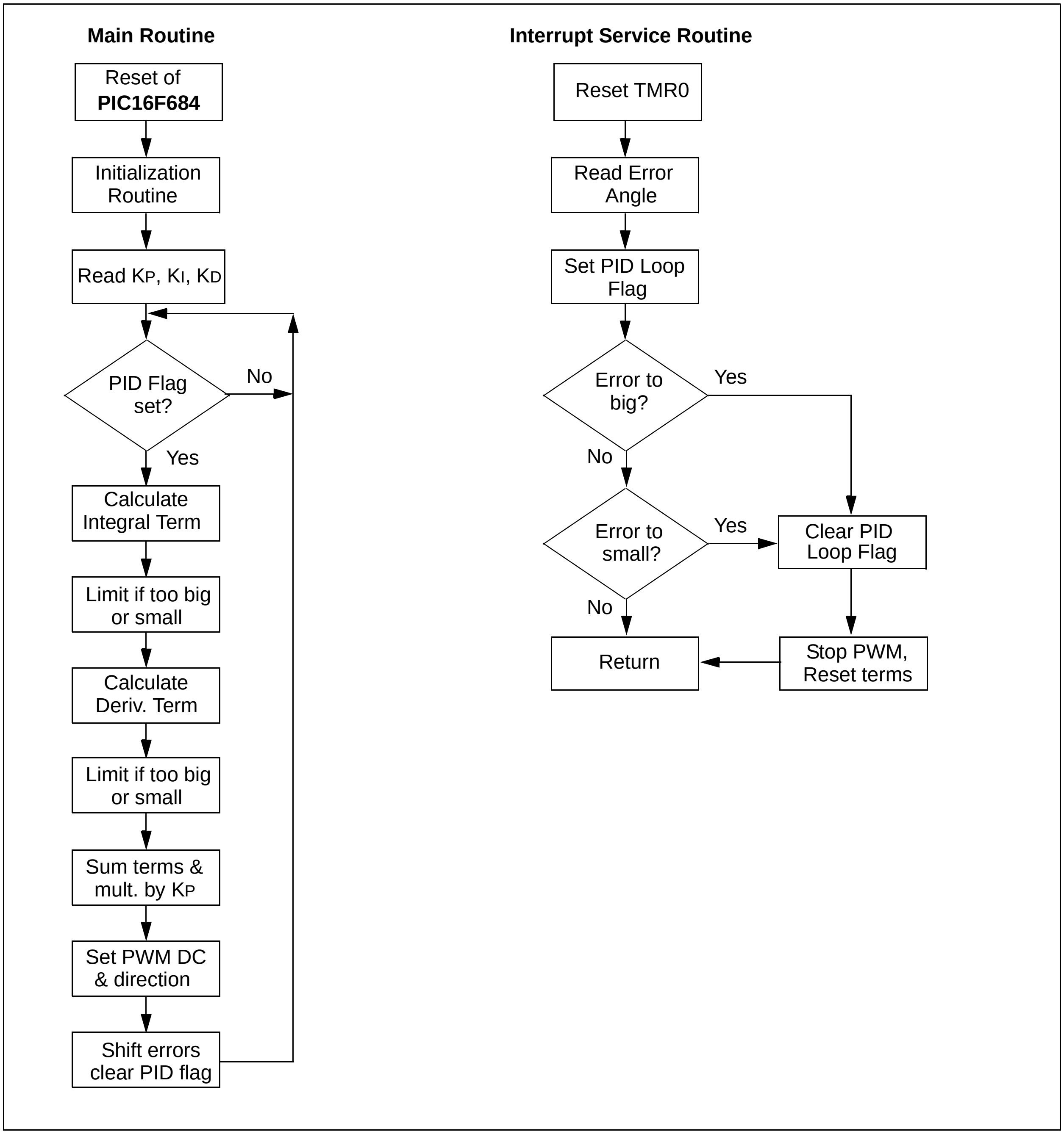
To get a rough estimate of how fast the PID loop needs to be updated, place an object, similar in length of the pendulum, on end in the palm of one hand and try balancing it. The object may or may not be possible to balance. The shorter the object, the harder it is to balance. In testing this method, the shortest length balanced for a sustained period was 0.5 meters in length. Balancing an object in such a fashion predominantly relies on the sense of sight. Since human vision can only process information at approximately 30 Hz, 30 Hz will be the baseline for the minimum speed the control loop needs to run at.

The frequency of the PID control loop is also going to be selected to simplify the math routines. The Integral term, in Equation 3, shows that each error term needs to be multiplied by the sampling period (which is the same as dividing by the sampling frequency). By choosing a sampling frequency in powers of 2's, a very fast divide routine can be done by using the right shift command, where each right shift is a divide by 2. It is very similar for the Derivative term, except the left shift would be used for a multiply by 2. Knowing this, choose 256 Hz as the sampling frequency. This is 8 times faster than the estimated minimum frequency and should allow plenty of room to vary the length of the pendulum, if desired.

C CODE FLOW CHART

The following flow charts show a simplified version of the C code for the inverted pendulum. For a full description of the code, see the ***AN964*** Source Code file.

FIGURE 5: FLOW CHART



SUBTLETIES IN THE SOFTWARE

The interrupt service routine is used to control the speed of the PID loop. The interrupt service routine is set to run off the Timer0 Interrupt. Timer0 is an 8-bit timer that will increment the TMR0 register every instruction clock. When the TMR0 register overflows, the Timer0 Interrupt Flag is set. The speed at which the interrupt should occur is every 3.9 milliseconds (1/256 Hz). Since we are using the internal 8 MHz internal oscillator, we will have a 2 MHz instruction clock or 0.5 µs per instruction. This yields that the interrupt should run every 7812 instructions. By setting the TMR0 prescaler to 32 and reloading 11 into the TMR0 register, in the interrupt service routine, the Timer0 interrupt occurs every 3.9 milliseconds or every 7808 instructions (255-11)*32.

As stated previously in this application note, there are 2 basic form of the PID that can be implemented and they are:

EQUATION 6:

$$C(n) = K_P E(n) + K_I T_S \sum_{0}^{N} E(n) + K_D[E(n)- E(n-1)]/T_S$$

OR

$$C(n) = K(E(n) + (1/T_I)\sum_{0}^{N} E(n) + T_D[E(n)- E(n-1)]/T_S$$

Where $K_P = K$, $K_I = K/T_I$ & $K_D = K\ T_D$

The later is used because a change in the proportional constant will not affect the pole response of the controller. If variations in the supply voltages were expected, such as battery powered applications, it would be possible to change the proportional constant on the fly to compensate for these supply variations. The relationship to the proportional constant and the supply voltage to the motor would be inversely proportional.

The derivative term is crucial in order to bring the inherently unstable system into stability. In any PID control the derivative terms acts as an anticipator. By checking the current error against the previous error, the controller can tell if the error term is getting bigger or smaller. If the error term is getting larger, the derivative term adds to the output of the controller much like that of the proportional and integral terms, but to a lesser effect. If the error term is getting smaller, this term will subtract from the output of the control in anticipation of an overshoot condition. Without the derivative term the system will always be unstable because there is no way to compensate for the overshoot condition. The following is the actual C code used to calculate the derivative term.

FIGURE 6: C CODE

```
//Calculate the differential term
    derivative_term = en0 - en3;
    if(derivative_term > 120){
       derivative_term = 120;
    }
    if(derivative_term < -120){
       derivative_term = -120;
    }
    derivative_term = derivative_term * kd;
    derivative_term =derivative_term>>5;
    //divide by 32
    }
    if(derivative_term > 120){
       derivative_term = 120;
    }
    if(derivative_term> -120){
       derivative_term = -120;
    }
```

When the approximation was made for the derivative term, it stated that the approximation would be:

EQUATION 7:

$$K_D dE(t)/dt = T_D\ [E(n)- E(n-1)]/T_S$$

Where $T_D = K_D/K_P$

By reviewing the code in Figure 6, one can determine that the approximation, shown in Equation 7, is not the approximation used. The actual equation that was used for calculating the derivative term is as follows:

EQUATION 8:

$$K_D dE(t)/dt = K_D\ [E(n)- E(n-3)]/32$$

Equation 8 more accurately written would be:

EQUATION 9:

$$K_D dE(t) = T_D[E(n)- E(n-3)]/(X\bullet 3\bullet T_S)$$
$$=K_D[E(n) -E(n-3)]/(K_P\bullet X\bullet 3\bullet T_S)$$

Where X is an unknown scaling factor

With a 10-bit A/D, K_D is measured as an integer between 0 and 1023. The X term allows for a fractional representation of K_D, not just a integer. (Similar scaling factors are also used when calculating the integral and proportional terms). The 3*T_S comes from the fact that we are tripling our sampling period by using E(n) – E(n-3) not E(n) – E(n-1). Why use E(n) – E(n-3) instead of E(n) – E(n-1)? The main reason for doing this is to limit the variation in the error angle measurement. There will always be an amount of uncertainty associated with the error measurement, some of which can be attributed to A/D error, Bessel filter throughput, mechanical vibration, etc. Since the uncertainty of the error measurements will be the same for all error terms, this uncertainty can be decreased by effectively tripling the sampling period. The real key is that the derivative term is still updated at 256 Hz rate. Doing so gives the benefit of the slower rate a more accurate derivative term, but at the desired faster sampling rate. The side affect of doing this is adding approximately 4 milliseconds of lag to the derivative term, which in this case is acceptable. K_P*X*3*T_S has been precalculated to be 32 to increase the speed of the PID loop.

TUNING THE PID CONTROLLER

Use the following steps to tune the PID constants:

1. Turn the K_P, K_I and K_D potentiometers counter clockwise as far as they will turn. This sets all the constants to zero.
2. Power the device using a 12V (minimum of a 3 amp) power supply.
3. Holding the Reset button down, lift the pendulum to the vertical position and release the Reset button and pendulum at the same time.
4. The pendulum should free fall and the base will not move. This verifies that all constants are properly read as zeros.
5. Increase the K_P constant by turning the potentiometer counter clockwise and repeat step 3.
6. Keep repeating steps 3-5 until there is a little oscillation in the base. If the K_P term is too small, the base platform will chase the top of the pendulum while Θ continues to increase. K_P will be too large if the drive wheel breaks free or the base oscillates at a high rate of speed.
7. Start increasing the K_I the same way as K_P until the pendulum can be balanced for several seconds under a constant oscillating condition. When the K_I is added, the base will now accelerate faster than the pendulum causing Θ to change from a positive angle to a negative angle (or vice versa). The pendulum will begin to fall backwards. The base should change directions and, again, accelerate faster than the pendulum until Θ changes signs and the whole cycle repeats. This is known as the Overshoot condition.
8. Increase K_D in the same manner as K_P and Ki until the Overshoot condition is gone and the pendulum remains balanced.
9. Once all overshoot is gone, the PID controller is tuned.

CODE CONVERSION TO ASSEMBLY

For those who prefer to program in assembly, there is an assembly file which can be used also. When programming in assembly, it is essential to make sure that the results of the math functions have the proper sign and the math registers never overflow. In order to speed up the PID loop, all the multiply routines have been limited to an 8x8 signed multiply routine with a 16-bit signed result. To do this, Θ was measured by the 10-bit A/D with the 2 LSb being ignored. The 8-bit A/D result was then converted to an 8-bit signed number by adding 128 decimal and ignoring the carry flag. The constants, K_P, K_I and K_D are all limited to a positive 8-bit signed number, or 0-127. Other than these 2 key changes, the assembly program follows the same flow chart as the C code.

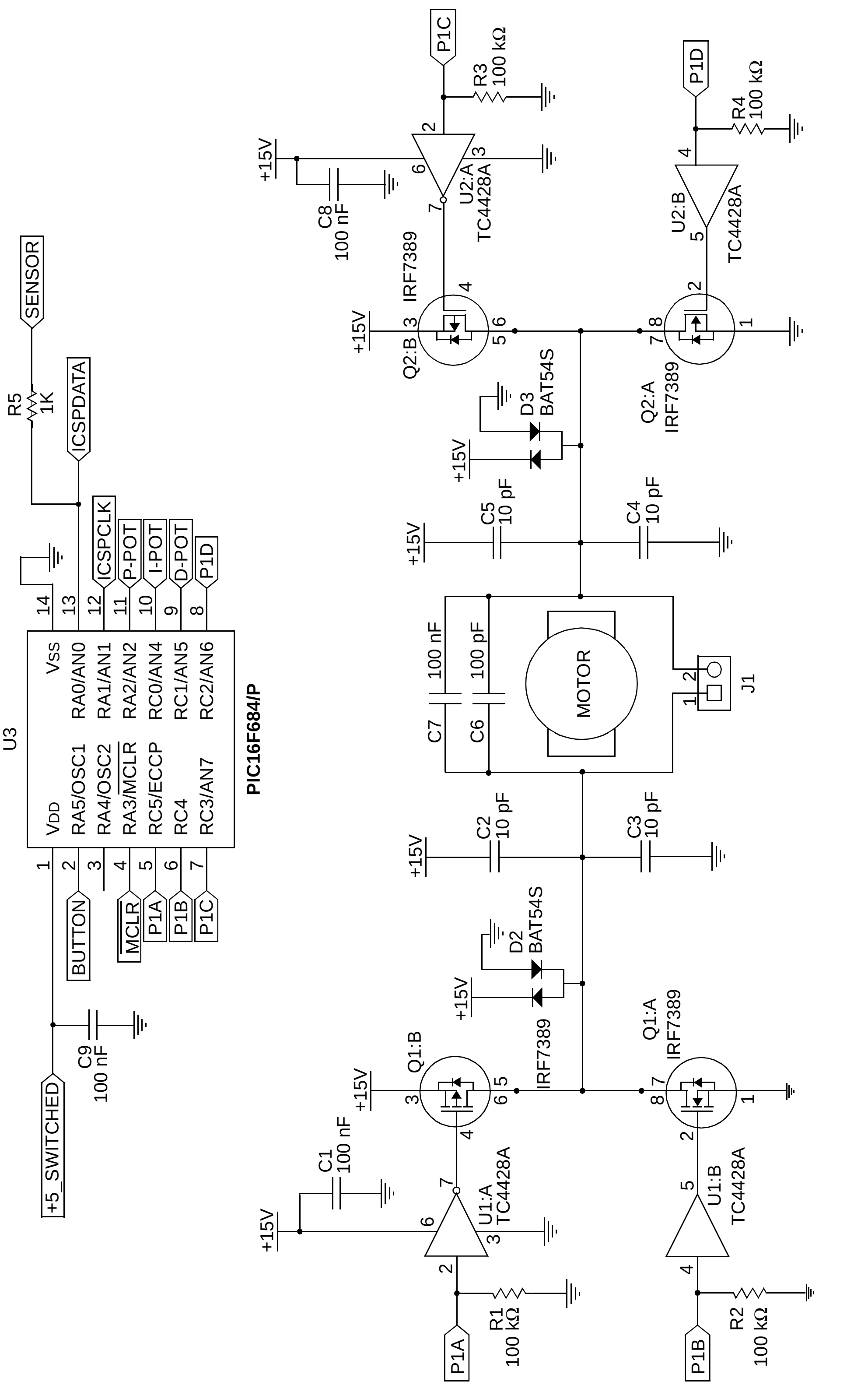
Note:	By limiting the multiply routines to 8x8 bit signed math, the Assembly code can execute the PID loop in approximately 215 µs where as the C code takes 1.4 ms.

CONCLUSION

By using the PIC16F684 device's ECCP and A/D modules we are able to demonstrate how to implement a positional PID controller to bring an inherently unstable system into stability. The keys to implementing this control is to have a basic understanding of the mechanical system, and identifying the derivative term would be a critical factor in the overall stability of the system. The other keys, with respect to the software, were making sure our registers never overflow, and picking the frequency of the PID loop that is a power of 2 so we could have a fast multiply and divide routine using the left and right shift.

APPENDIX A: SCHEMATICS

FIGURE 7:

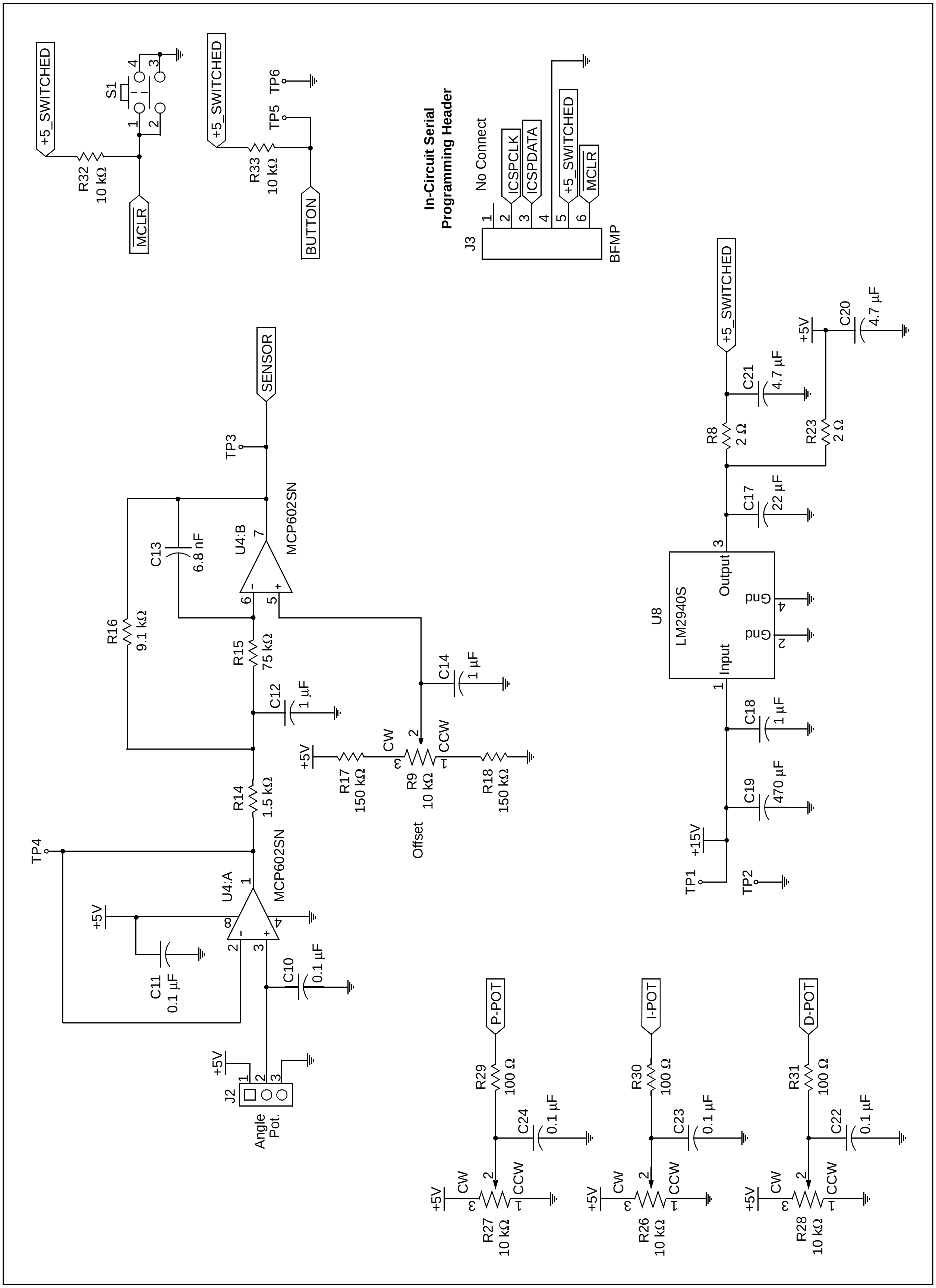


Note 1: Unless otherwise specified, resistance values are in Ω. Resistors are 5% tolerance. Capacitance values are in µF.

2: Device names and numbers shown here are for reference only and may differ from the actual number.

3: Items labeled with “A” are unpopulated, “B” are socketed but not populated, and “C” are socketed and populated.

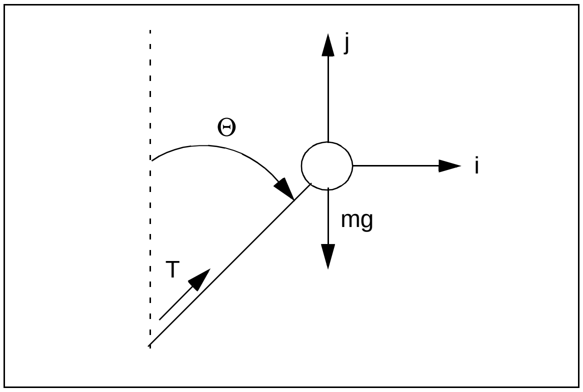
FIGURE 8:


+5_SWITCHED
S1
R32
10 kΩ
MCLR
+5_SWITCHED
TP5
TP6
R33
10 kΩ
BUTTON
In-Circuit Serial
Programming Header
No Connect
ICSPCLK
ICSPDATA
+5_SWITCHED
MCLR
J3
BFMP
SENSOR
TP3
U4:B
MCP602SN
C13
6.8 nF
R16
9.1 kΩ
R15
75 kΩ
C12
1 µF
C14
1 µF
+5V
CW
CCW
R17
150 kΩ
R9
10 kΩ
R18
150 kΩ
Offset
R14
1.5 kΩ
TP4
U4:A
MCP602SN
+5V
C11
0.1 µF
C10
0.1 µF
+5V
J2
Angle
Pot.
+5_SWITCHED
C21
4.7 µF
+5V
C20
4.7 µF
R8
2 Ω
R23
2 Ω
C17
22 µF
U8
LM2940S
Output
Gnd
Gnd
Input
C18
1 µF
C19
470 µF
+15V
TP1
TP2
P-POT
R29
100 Ω
C24
0.1 µF
R27
10 kΩ
I-POT
R30
100 Ω
C23
0.1 µF
R26
10 kΩ
D-POT
R31
100 Ω
C22
0.1 µF
R28
10 kΩ

APPENDIX B: DERIVATION OF EQUATION 5

The pendulum's motion is described in this appendix with one assumption. The assumption is that the pendulum is modeled as a point mass at the end of a massless rod.

FIGURE 9: PENDULUM FREE BODY DIAGRAM



Using 2 unit vectors, *i* and *j* to represent the horizontal and vertical vectors respectively, yields the following equations for the pendulum.

EQUATION B-1:

$$Position = R\sin\Theta i + R\cos\Theta j$$

Where R is the length of the pendulum

The pendulum's angular velocity is the derivative of the position with respect to Θ.

EQUATION B-2:

$$Velocity = R\Theta'\cos\Theta i - R\Theta'\sin\Theta j$$

The pendulum's angular acceleration is the derivative of the angular velocity w.r.t. Θ.

EQUATION B-3:

$$\begin{aligned} Acceleration &= (R\Theta''\cos\Theta i - R\Theta'^2\sin\Theta i) - (R\Theta''\sin\Theta j + R\Theta'^2\cos\Theta j) \\ &= R(\Theta''\cos\Theta i - \Theta''\sin\Theta j - \Theta'^2\sin\Theta i - \Theta'^2\cos\Theta j) \end{aligned}$$

From Newton's second law of motion, F = ma where the mass is the point mass of the pendulum and the accelerations is the angular acceleration of Pendulum, this yields Equation 4.

EQUATION B-4:

$$Force = mR(\Theta''\cos\Theta i - \Theta''\sin\Theta j - \Theta'^2\sin\Theta i - \Theta'^2\cos\Theta j)$$

With a free body diagram we can see that there are 2 forces acting on the pendulum, the tension from the rod and the force of gravity. This free body diagram yields the following equation.

EQUATION B-5:

$$Force = T\sin\Theta i + T\cos\Theta j - mgj$$

Equate Equation B-4 and Equation B-5:

EQUATION B-6:

$$T\sin\Theta\, i + T\cos\Theta j - mgj = m\,R(\Theta''\cos\Theta\, i - \Theta''\sin\Theta\, j - \Theta'^2\sin\Theta i - \Theta'^2\cos\Theta j)$$

Now separate into 2 different vectors equations and eliminate the vector notation.

EQUATION B-7: (i Vector)

$$T\sin\Theta = m\,R\Theta''\cos\Theta - mR\Theta'^2\sin\Theta$$

EQUATION B-8: (j Vector)

$$T\cos\Theta - mg = -m\,R\Theta''\sin\Theta - mR\Theta'^2\cos\Theta$$

Use these two simultaneous equations to eliminate the unknown T.

Multiply Equation B-7 by cosΘ.

EQUATION B-9:

$$T\sin\Theta\cos\Theta = m\,R\Theta''\cos^2\Theta - mR\Theta'^2\sin\Theta\cos\Theta$$

Multiply Equation B-8 by sinΘ.

EQUATION B-10:

$$T\sin\Theta\cos\Theta - mg\sin\Theta = -m\,R\Theta''\sin^2\Theta - mR\Theta'^2\sin\Theta\cos\Theta$$

Moving -mgsinΘ to the other side.

EQUATION B-11:

$$T\sin\Theta\cos\Theta = mg\sin\Theta - mR\Theta''\sin^2\Theta - mR\Theta'^2\sin\Theta\cos\Theta$$

Equate Equation B-9 and Equation B-11.

EQUATION B-12:

$$m\,R\Theta''\cos^2\Theta - mR\Theta'^2\sin\Theta\cos\Theta = mg\sin\Theta - mR\Theta''\sin^2\Theta - mR\Theta'^2\sin\Theta\cos\Theta$$

Divide by each side by mR.

EQUATION B-13:

$$\Theta''\cos^2\Theta - \Theta'^2\sin\Theta\cos\Theta = (g/R)\sin\Theta - \Theta''\sin^2\Theta - \Theta'^2\sin\Theta\cos\Theta$$

Collect like term.

EQUATION B-14:

$$\Theta''(\cos^2\Theta + \sin^2\Theta) = (g/R)\sin\Theta$$

By trigonometric definition $\cos^2\Theta + \sin^2\Theta = 1$ which yields Equation B-15.

EQUATION B-15:

$$\Theta'' = (g/R)\sin\Theta$$

Using the small angle approximation where $\sin\Theta = \Theta$ yields.

EQUATION B-16:

$$\Theta'' = (g/R)\Theta$$

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

Printed on recycled paper.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona and Mountain View, California in October 2003. The Company's quality system processes and procedures are for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Alpharetta, GA
Tel: 770-640-0034
Fax: 770-640-0307

Boston
Westford, MA
Tel: 978-692-3848
Fax: 978-692-3821

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

San Jose
Mountain View, CA
Tel: 650-215-1444
Fax: 650-961-0286

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8676-6200
Fax: 86-28-8676-6599

China - Fuzhou
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Shunde
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

China - Qingdao
Tel: 86-532-502-7355
Fax: 86-532-502-7205

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-2229-0061
Fax: 91-80-2229-0062

India - New Delhi
Tel: 91-11-5160-8631
Fax: 91-11-5160-8632

Japan - Kanagawa
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Taiwan - Hsinchu
Tel: 886-3-572-9526
Fax: 886-3-572-6459

EUROPE

Austria - Weis
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

Denmark - Ballerup
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Massy
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Ismaning
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

England - Berkshire
Tel: 44-118-921-5869
Fax: 44-118-921-5820

10/20/04