RC Model Aircraft Motor Control

Author: *Joseph Julicher*
Microchip Technology Inc.

INTRODUCTION

As modern MOSFET transistors are developed with lower ON resistance and smaller packages, Electronic Speed Controls (ESC) follow. By using a microcontroller and a few small MOSFETs, high performance speed controls can be built with advanced features in small packages. This application note will describe the design and construction of two small speed controls appropriate for flying model airplanes up to 6 lbs. Many of the control techniques in this application note can easily be used with larger DC motors in different applications.

Speed Control Features

A modern electric aircraft speed control must have the following features:

1. Low weight (everything in an airplane must be lightweight)
2. Low ON Resistance (minimal heat sink at high power ratings = low weight and long run times)
3. Safety Motor Start (the motor should not be armed until the throttle is at a minimum setting)
4. Gearbox protection (A geared motor should not accelerate quickly or gear damage could occur)
5. Safety Shutoff (if the controlling signal is lost, the motor should stop)
6. Battery Eliminator Circuit (BEC) (This allows the flight battery to power the radio equipment)
7. Low Voltage Shutdown (Disables the motor when the battery voltage drops to a low value which allows the radio to continue to function)
8. PWM (Pulse Width Modulation) frequency as high as possible. Be careful to keep electro-magnetic inductance (EMI) down by controlling rise time and switching rate. A short rise time will increase broadband radio frequency interference (RFI). A high switching rate will increase first order RFI.

The previous list is a minimum feature set. Some advanced features are:

1. Programmable throttle response (allows the pilot to map control position to motor output to match the plane).
2. Motor Brake (this will stop the motor so a folding propeller can fold to reduce drag in gliders).

Solutions

The broad range of features makes microcontrollers an obvious choice. Because different sized aircraft have different priorities on features, a variety of speed controls are required. This application note will show the design and construction for two different controllers (Versions 1 and 2).

TABLE 1: FEATURE COMPARISON

Version	Device	Battery Volts	Motor Current	BEC Amp	Low Voltage Detection	Brakes	Throttle Curve	PWM Rate kHz
1	PIC12C508A	20	20	0.5	limited	None	None	1.3
2	PIC16F628	20	20	1.0	Yes	Yes	Yes	32, 8, 2

RC Control Signals

The radio control modeling hobby has a long and varied history with many different control systems being used over the years. The current “standard” radio signal is a series of pulses with a nominal pulse width of 1 to 2 ms and a pulse period of approximately 20 ms. These values are not enforced and pulses range from 0.85 ms to 2.2 ms. Most radio equipment has the ability to adjust the pulse output a small amount, but it would be prudent to accept a large degree of variation in the signal. To accommodate equipment variations, many ESC manufacturers have included a “Training” mode to calibrate the ESC to a particular signal before each flight or during installation. Obviously, this can add some complexity to the hardware and software. Version 2 has a PC programmable throttle response. This has the added benefit of allowing the user to customize the response to the particular radio system. For the rest of this application note, the radio signal will be assumed to have a range of 1 – 2 ms, but the wider tolerance will be remembered. The throttle control on radio equipment is special because there is no return spring so the throttle control will remain at any given position. Additionally, the throttle control has a series of détentes that provide friction to hold the throttle control. These détentes restrict the throttle to a finite number of positions. Each radio is different but approximately 24 détentes are typical. To provide the feeling of a smooth throttle response, the throttle range should be divided into at least 48 steps. More steps will ensure a variety of radio systems have the same throttle feel.

ESC Testing

Before testing the motors, a test fixture was created to hold the equipment safely. The motors spin propellers very fast and can easily damage electrical equipment or cause injury. Power was provided by 8-cell NiCd battery packs or a bench supply. The servo control signal was provided by a radio simulator built with a PIC16F873. The code for the radio simulator will be provided with this application note. The test motor was a 7.2V Speed 400 brushed motor. A 125 x 110 push-on plastic propeller was used with the motor. This propeller is supplied at most hobby stores as a replacement propeller to a popular electric airplane.

PWM Motor Control

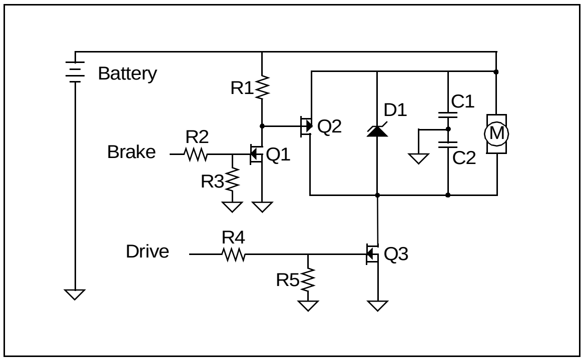
A brushed DC motor is a very simple device to control. The motor speed (RPM) is directly proportional to the voltage applied across the terminals. The motor torque is directly proportional to the current flowing through the motor. Motor voltage can be easily controlled by using a PWM switch to chop the current to the motor proportionally to the desired throttle setting. A simple PWM switch is an N channel MOSFET transistor connected between the motor and ground (Q3). If the gate threshold voltage and the gate capacitance are low then the MOSFET can be activated from a pin on the microcontroller. Because PICmicro® Microcontrollers can source 25 mA from their output pins, the 20 amp speed controls do not require any MOSFET driver chips, provided a suitable MOSFET can be found. To ensure that the motor does not energize during power-up, a large value resistor (approx. 10 KΩ) should be placed between the MOSFET gate and ground (R3, R5). This will hold the MOSFET off until the PICmicro Microcontroller finishes RESET and initializes the drive pin.

At this time, the drive design consists of a low resistance N channel MOSFET connected between the motor and ground. The MOSFET gate is driven by an output pin from a PICmicro Microcontroller. A small resistor (approx. 100 Ω) should be placed between the gate and the drive pin to limit transient current to the PICmicro MCU pin from discharge of the gate capacitance and to control the MOSFET rise time (R2, R4).

The simplest brake for a brushed DC motor is to short the terminals together. When a DC motor is rotated, it becomes a generator. If an electrical load is attached to the motor, an equivalent mechanical load is manifest at the motor shaft. By shorting the terminals, the motor is loaded with it's own internal resistance and the mechanical load is maximized. This will create a mechanical brake and stop the propeller. A brake is constructed with a P channel MOSFET across the terminals (Q2). When the gate voltage is brought low, it will engage the motor brake. The battery voltage is too high to drive the gate with the PICmicro Microcontroller so a separate drive circuit will be required (R1-3, Q1).

Presently, the drive design consists of a P channel MOSFET brake and an N channel MOSFET PWM switch. There is still a missing piece to the motor control puzzle. The motor armature is constructed from a large coil of wire, creating an inductor. When the PWM switch opens, the inductor will continue to source current across high impedance (the PWM switch) and cause very large voltages. If the ESC has a brake, the body diode of the brake FET could be used to control the inductive voltage. If the body diode is not strong enough or there is no brake, a Schottky diode must be placed across the motor terminals (D1). This will redirect the inductor current and prevent the high voltage spike. Small capacitors (approx. 0.1 μF) may also be required across the motor terminals to prevent EMI, but that will depend on the installation (C1, C2). Figure 1 shows the completed motor drive circuit.

FIGURE 1: RC MOTOR DRIVE CIRCUIT



Electric Aircraft Power Requirements

Before the components for the ESC can be selected, the requirements need to be identified. The general rule of thumb for electric flight is 50 watts per pound as a minimum for a sport model (or 3 watts per ounce). A popular foam airplane weighs 1.5 lbs. (24 ounces) requiring at least 75 watts. It is typically flown for 8 minutes with a 6V Speed 400-type electric motor and an eight cell 9.6V NiCd battery pack. This configuration draws around 12 amps at full throttle. This is a substantial amount of power that must be delivered to the motor as efficiently as space and weight allow. MOSFET transistors are available with 0.002 Ω on-resistance. If multiple MOSFET's are employed in parallel, the ESC motor losses can be reduced to a few hundred milliwatts. This is very desirable because a good MOSFET weighs a lot less than a good heat sink.

The largest potential power loss on the ESC is in the Battery Eliminator Circuit (BEC). This feature allows the radio equipment to be powered from the flight battery. Although a BEC will save weight and space, it can cause the following problems:

- Couple noise from the ESC into the radio power supply
- Cause control problems if the flight battery voltage drops too low
- Waste a lot of battery power with large battery voltages and high current servos

Most ESC's utilize a simple Low Dropout (LDO) regulator to drop the flight battery voltage to 5V. This is adequate with small batteries and low power servos. When the batteries are more than 8 cells and servo current exceeds 1 amp, a substantial portion of the battery power begins to be lost in the BEC. A typical servo requires a few hundred mA to hold a position. This holding current depends on the torque required to hold the load. A small airplane has very low flight loads so this is minimal. However, when the servo is moving to a new position, the current can become very large. Some fast servos have peak currents of over an amp. Even some cheap slow servos have high transient currents due to sloppy gears and inefficient motors. The 500 mA BEC current used for ESC Version 1 should be considered an absolute minimum. Most small installations in aircraft less than 2 lbs. should have no trouble. A larger 1 amp BEC would be much safer. Beyond 1 amp, most current systems use a separate battery to prevent the large power loss in the flight pack.

VERSION 1: THE MINIMAL ESC

Hardware

Now that the ESC problem and power requirements have been defined, it is time to build an ESC. The first ESC will simply control a motor up to 20 amps and 20V. The architecture of this device will be a PIC12C508A driving a pair of N channel MOSFETs in parallel. A small LDO regulator will provide 5V power to the PICmicro Microcontroller as well as 5V power to the external radio equipment. A Schottky diode will be used to protect the MOSFET. The schematic is shown in Appendix A. The physical size for this ESC is roughly the size of a quarter (see Figure 8).

Software

A microcontroller ESC has simple hardware but the software is what makes it work. Before the software is designed, the hardware constraints must be reviewed. These constraints are:

1. 4 MHz internal RC system clock which provides a 1 µs instruction rate.
2. "bit-banged" PWM with no glitches in the output
3. Output must go between 0 and 100% power.
4. There must be hysteresis to prevent spurious power bursts near 0%.
5. There must be approximately ¾ second rise time between 0 and 100% power (gearbox safety).
6. The ESC must not energize the motor until it has ensured that the throttle is in a 0% setting for ½ second.
7. The ESC should turn off the motor if it looses the control signal for approximately 50 ms.
8. There should be at least 48 steps in the throttle response.
9. The PWM frequency should be as high as possible within the constraints of EMI and switching loss.

With these constraints, the design choices made were:

1. Use 64 throttle steps. This makes the throttle control an even 6 bits.
2. Use the Watchdog Timer for the lost control RESET.
3. Valid receiver pulses will be between 1.15 to 1.85 ms.

With these decisions made, it is time to design the code. There are two modes in the code. The first mode is Arming. While the ESC is arming, it is waiting for 25 consecutive minimum width pulses. During the arming process, the motor is turned off.

The second mode is the Operating mode. While the ESC is in Operating mode, it is scaling the control pulses into PWM valves. The three ways to exit this mode are:

1. Remove the power.
2. Get a Brown-out Reset (BOR).
3. Get a Watchdog Timer Reset (WDT).

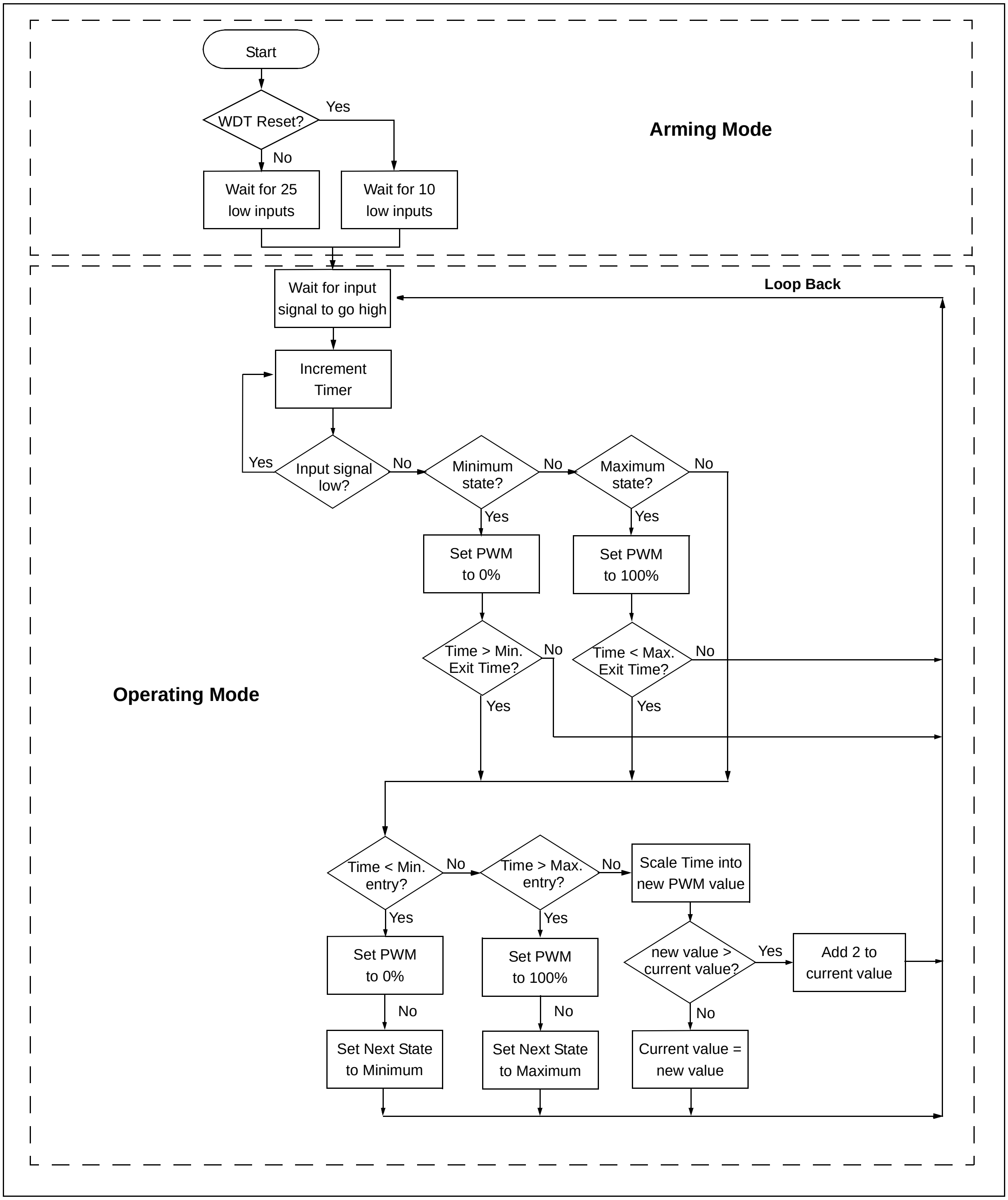
With power removed, the motor stops. This is the trivial case. The BOR or WDT Reset will cause the ESC to re-enter the Arming mode. In the case of a Watchdog Timer Reset, the time-out bit is consulted and the part will rearm with a shorter arming time. In either case, this will force the pilot to reduce the throttle to minimum to restart the motor.

The PWM algorithm used takes 6 instruction cycles to perform one PWM update. It takes 64 updates to perform one PWM period. The code allows 6 instruction cycles to exist between each PWM update. This places 12 cycles between each PWM update. With 64 updates and 12 µs per update, the PWM frequency is 1.302 kHz.

Figure 2 is a flowchart for the ESC operation.

Note: Every 6 instructions the PWM update macro must be called (see Example 1).

FIGURE 2: FLOW DIAGRAM OF ESC OPERATION



PWM Algorithm

The bit-banged PWM algorithm was simplified by choosing GPIO0 for the PWM output. By adding the PWM counter to the PWM value, the carry flag would be set or cleared according to the desired PWM signal. The carry flag was moved into GPIO0 with a simple rotate instruction. The do_pwm macro is shown in Example 1.

EXAMPLE 1:

```
; macro to perform 1 cycle of PWM
do_pwm     macro
   movlw   pwm_reload      ; preload the reload value
   decfsz  pwm_counter,f   ; decrement the counter value.
   movf    pwm_counter,w   ; if the counter is not 0, load W with counter
   movwf   pwm_counter     ; store w in the counter. this does an auto reload timer...
   subwf   pwm,W           ; pwm - counter sets the borrow flag
   rlf     GPIO,f          ; a left rotate places the borrow flag in GPIO
   endm
```

This PWM algorithm uses 6 CPU cycles. This must be performed at regular intervals to keep the output glitch free. The value of pwm_reload is configured at compile time. For Version 1, pwm_reload was configured for 64. If a longer period PWM was required that had more bits of precision, the pwm_reload value could be increased. To determine the PWM frequency, use Equation 1.

EQUATION 1:

$$PWM\ frequency = \frac{clock\ frequency}{((6 + cycles\ between\ PWM\ updates) \cdot pwm\_reload \cdot 4)}$$

$$PWM\ frequency = \frac{4MHz}{((6 + 6) \cdot 64 \cdot 4)}$$

$$PWM\ frequency = 1302Hz$$

where:

4 MHz	=	Clock Frequency
6	=	Cycles between PWM updates
64	=	*pwm_reload*

Other Features

SIGNAL LOSS STOP

The signal loss stop is done by resetting the Watchdog Timer (WDT) on every falling edge of the control signal. This is updated every 18-36 ms. By configuring the WDT with a divide by 4, the WDT period is 72.8 ms. This will cause a RESET after missing 2-4 pulses. The RESET will stop the motor and wait for 10 consecutive short pulses before resuming operation.

OUTPUT SCALING

The input is set to assume pulses smaller than 1.15 ms are 0% PWM and values greater than 1.85 ms are 100%. With a 12 µs polling time, there are only 58 discernible steps for the control signal. Because the output signal assumes 64 input steps, the output reaches 90.6% duty cycle then jumps to 100%. It would have been better to place the missing 10% at the bottom of the range where it takes large steps to get things turning.

LOW VOLTAGE CUTOUT

A desirable feature for a speed control with BEC is to be able to drop out the motor while there is still sufficient power to operate the receiver and glide home. This speed control does not have that feature, but the BOR can be considered a crude version. Should the BOR trip, the motor will stop. For some receivers, the brown-out voltage level is higher than the receiver's low voltage threshold. In this case, the ESC behaves correctly. With other receivers, there is no protection. If this is a feature that is required, make sure you add the correct circuitry to perform the function.

A complete code listing is in Appendix A.

FLYING

This speed control was mounted in two different aircraft for flight testing.

1. A twin motored aircraft powered by two-4.8V speed 400's was flown first. This airplane has a low voltage receiver that allowed the pilot to retain control after the ESC stopped the motor during a low battery condition. This airplane had a very noticeable hum at low throttle settings as both motors vibrated from Discontinuous mode operation. Discontinuous mode is where the current through the motor reaches 0 amp. This happens when the pulse width is much smaller than the pulse period so the current has time to reach 0. On a brushed motor, this will cause small pulsing in the motor armature. There are no adverse affects from this operation but it is noisy. The pilot thought the speed control flew well but noticed the lack of a propeller brake.
2. The second plane was a foam flying wing powered by a single 6V speed 400. This aircraft did not have as noticeable of a hum due to the single motor. After a slow 15 minute flight at half throttle, the speed control was only slightly warm to the touch. This indicates that power dissipation is very low in the MOSFET's and no heat sink is required. The radio equipment in this airplane is more sensitive to voltage so it is critical to get a low voltage cutout functioning in the next design. Fortunately, the foam wing was very resilient so no damage occurred when the receiver cutout. No noticeable range reduction was seen with either of the aircraft so the EMI should be considered acceptable.

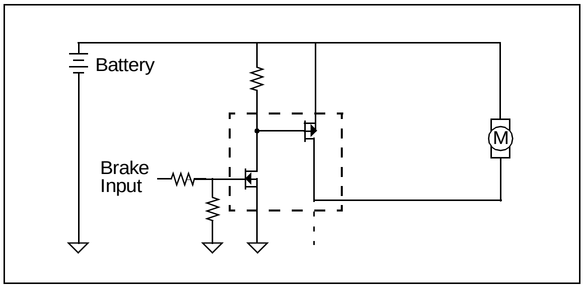
VERSION 2: RAISING THE RATES AND ADDING FEATURES

The basic speed control of Version 1 works well but has one serious problem. The low voltage cutout is critical for safe flying. Additionally, a motor brake and a programmable throttle response would be appreciated. Lastly, a higher rate PWM would be nice for low end throttle response and quiet operation. It is time to analyze these additional features and determine the basics of the new hardware.

Motor Brake

The motor brake has already been discussed in the PWM motor control chapter. Brakes will be added by employing a dual N/P MOSFET in a single package. The N channel will serve to pull the gate on the P MOSFET from the supply rail, turning on the MOSFET. This will add a single SO8 package and three resistors to the design.

FIGURE 3: MOTOR BRAKE CIRCUIT



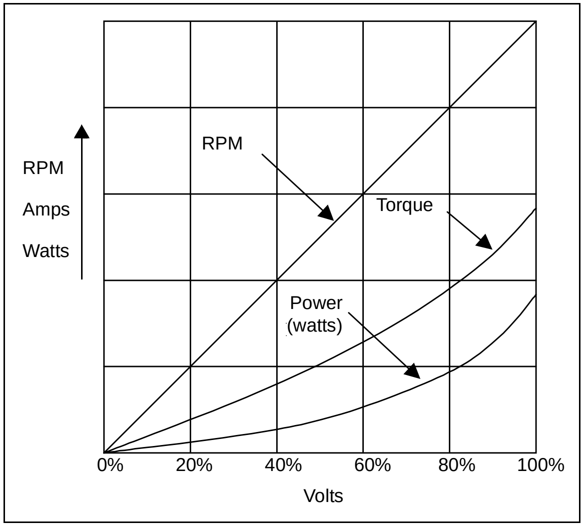
Low Voltage Cutout

This is the most important feature to be added. The best way is to use some sort of comparator to compare the battery voltage to a fixed reference. Since the motor must be disabled before the linear regulator starts to drop, the comparator can compare a divided battery voltage to a stable reference. A good LDO regulator requires ½ V to regulate, so the comparator and the resistor divider should be configured to require the battery to operate ¾ V higher than the LDO regulator output voltage.

Programmable Throttle Response

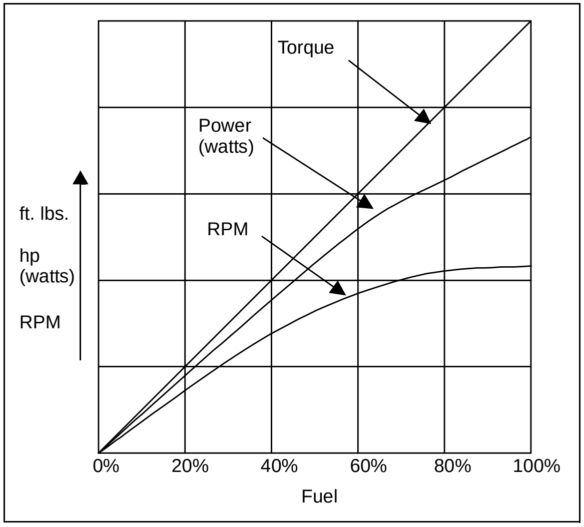
The previous ESC responded to throttle inputs by linearly adjusting motor voltage. This has the effect of linearly controlling the propeller RPM. But it does not linearly control power. An ideal response of an electric motor/propeller combination to RPM is as follows.

FIGURE 4: % DUTY CYCLE (THROTTLE COMMAND)



The actual numbers have been left off of the chart because they depend on the exact motor and propeller combination. The data will also change for a stationary or a moving propeller. This type of behavior makes sense based upon our understanding of electric motors. An electric motor’s RPMs are dependent on the EMF of the armature. The torque of the motor is dependent on the current through the armature. Unfortunately, this is totally different from a gasoline engine. When a pilot advances the throttle of a gasoline engine, they are adding fuel. The addition of fuel to a gasoline engine raises the engine torque. It has only a secondary effect on RPM that is dependant on load. This means that for a given throttle setting, the engine RPMs will change to match the available torque.

FIGURE 5: % FUEL (THROTTLE COMMAND)



The chart shows a representation of an ideal gasoline engine/propeller combination and how RPM and power are related to torque. One side effect of these responses is that an electric airplane will behave very differently from a gasoline airplane. Airplane performance is governed by available power. On full scale aircraft with constant speed propellers, the throttle sets the desired engine power and the constant speed propeller holds RPM's constant. The engine controller regulates the torque so engine power is held at the desired setting. Therefore, it would be desirable to match throttle output to a linear power response. If that is not possible, a linear torque response would be the next best option. Linear RPM is the worst option.

The best solution is to add current/voltage feedback and regulate power to the motor. The next best solution is to add current feedback and regulate torque to the motor. The third best solution is to imitate it by adding a lookup table and adjusting the output response in a non-linear way to approximate a linear power curve. Real aircraft are controlled by lookup tables developed during flight testing. By adding a lookup table and developing a programming tool, we can accommodate aircraft with non-standard control requirements.

The hardware requirements for a programmable throttle response is some form of EEPROM. A serial EEPROM could be added to Version 1 but the physical size of the ESC is starting to increase. A different device with adequate internal EEPROM should be chosen. The previous ESC used 64 steps of output to provide a smooth response. If the input is divided into 64 steps, a memory of 64 bytes will be required. Less memory could be used but then the output data would be packed across multiple bytes, complicating the code. Version 1 was constrained to 6 instructions between PWM updates. If it takes more than six instructions to retrieve a table value, the tables will need to be cached in RAM. Therefore a part with at least 64 bytes of additional RAM could be required.

PWM Issues

PWM control is a good way to control motor speed but it creates a few problems. The problems are switching loss, radio frequency interference, low speed motor control and audio noise.

SWITCHING LOSS

Switching loss occurs at each edge of the PWM waveform. An ideal PWM signal switches instantly, but real devices are not ideal so there is a small period of time where the MOSFET switch behaves linearly. During this period of time, the MOSFET starts to heat up. If the PWM frequency and the current are high, the MOSFET will heat up quickly. ESC Version 1 used a switching frequency of 1.3 kHz. At this rate, there was a slight heat increase in the MOSFET's. Measuring the rise and fall times showed 500 ns for rise time and 1000 ns for fall time. These rise and fall times can be expected with all the designs that use this MOSFET configuration driven by a pin from the PICmicro Microcontroller. If the switching frequency is 30 kHz, these rise and fall times can dominate the switching time at high duty cycles. Ways to reduce these switching losses are to improve the rise and fall times and to reduce the frequency of the PWM operation. The frequency reduction is especially important when drawing high currents from the battery in order to control EMI.

RADIO FREQUENCY INTERFERENCE

Radio frequency interference comes from the current switching transients. The sharp edges from a fast PWM switch cause broadband radio interference. The primary interference will be at the PWM frequency and it's harmonics. Using the 30 kHz example from before, RF power peaks can be expected at 30, 60 and 90 kHz. They don't stop there but continue up the spectrum towards infinity with reducing power. The power increases with the amount of current that is switched. In addition to this primary interference, the sharp edges in the switched current create broadband noise. If the broadband noise is too great, the radio receiver in the airplane will be affected. If the primary noise has a harmonic that the receiver is sensitive to, it will also cause problems for the radio receiver. The two rules are, keep harmonics away from sensitive frequencies and don't switch the current to quickly.

LOW SPEED MOTOR CONTROL

At low power/speed settings, the electric motor will not operate smoothly unless the PWM frequency is high enough to prevent Discontinuous mode operation. Discontinuous mode is where the current through the motor reaches 0 amp. This happens when the off time of the motor is long enough to allow the current through the motor to decay to zero. The simple solution is to raise the PWM frequency but leave the duty cycle constant. When this happens, the off time is reduced, but the percent off time is the same. The motor will now operate at very low speeds and be easy to control. For electric aircraft, this is not usually a problem because you do not fly at such low power settings, but it does have the side affect of reducing audio noise.

AUDIO NOISE

Switching current in the motor at 1 kHz causes the motor to buzz. This can be very loud at lower throttle settings. The buzzing comes from the motor armature vibrating as these short pulses of energy are passed through. This noise is loudest when the motor current is discontinuous. By switching at a higher frequency, this motor noise can be eliminated because the motor does not go discontinuous and the frequency is too high to hear.

AUDIO NOISE SOLUTION

The PIC16F62X has a hardware PWM circuit that will be used to drive the MOSFETS. When the PIC16F62X is configured for 7-bit PWM @4 MHz clock, the maximum PWM frequency is 31 kHz. By changing the prescaler of timer 2, the frequency can be shifted to 8 kHz and then to 2 kHz without affecting the PWM duty cycle. Using this feature, the ESC can switch frequency's on the fly to maximize the frequency at any operating point. This will minimize switching loses at high power while minimizing noise at low power. RFI is also minimized because high current pulses occur at lower frequencies.

Hardware

Version 1 was upgraded by adding the brake (R4, R8, R9, U4), upgrading the microcontroller (U1, Y1, C9, C10), and adding the programming connector (J2). The complete schematic is in Appendix A.

PC Interface

A PC interface is provided on this design to allow programmable throttle response and brake setpoints. The PC interface will be through the hardware UART. To initiate the PC mode, the PC will send a 0x80 at 9600 baud 8n1 format. Once communication has started, each byte of data is echoed.

When a carriage return is received (ASCII 13) the ESC will respond with a '>'. When debugging with a terminal program, this creates a prompt for typing additional commands. The available commands are listed in Table 2. All commands are terminated with a carriage return.

TABLE 2: AVAILABLE FORMATS

Name	Code	Description

Table I/O	T	Reads or Writes to the throttle table. Addresses beyond the table are ignored.
		Address 0 is the first byte of the throttle curve.
		To read the value at a location, type:
		`TAA<CR>`
		Where AA is the address of the desired data in hexadecimal.
		The ESC will respond with:
		VV
		>
		Where VV is the data in hexadecimal.
		To write a value at a location, type:
		`TAA=DD<CR>`
		Where AA is the address and DD is the data in hexadecimal.
		The ESC will respond with a prompt.

Address I/O	A	Reads or Writes to the non-volatile memory. Addresses beyond the memory are ignored. Address 0 is the actual Address 0 of the memory. All data can be read or written by this command if the memory map for the ESC is known. The data format is identical to the ‘T’ command.
Rx Low Pulse	c	The smallest receiver pulse recorded by the ESC is stored here. This is a read-only value. Typing ‘c’ will return the data.
RX High Pulse	C	The largest receiver pulse recorded by the ESC is stored here. This is a read-only value. Typing ‘C’ will return the data.
Brake Threshold	B	When the throttle is below this point, the brakes are activated. Typing ‘B’ will return the data. Typing ‘B=DD’ will set the brake point to the value DD. DD is a hexadecimal number.
Arming Threshold	R	When the throttle is below this point for 10 periods, the ESC will be armed. Typing ‘R’ will return the data. Typing ‘R=DD’ will set the Arming Threshold to the value DD. DD is a hexadecimal number.
Minimum Throttle Threshold	m	When the throttle is below this point, the output is 0% PWM. When the throttle is above this point, the output matches the curve. This creates 1 count of hysteresis. Typing ‘h’ will return the data. Typing ‘h=DD’ will set the minimum throttle point to the value DD. DD is a hexadecimal number.
Maximum Throttle Threshold	M	When the throttle is above this point, the output is 100% PWM. When the throttle is below this point, the output matches the curve. This creates 1 count of hysteresis. Typing ‘H’ will return the data. Typing ‘H=DD’ will set the maximum throttle point to the value DD. DD is a hexadecimal number.

The version information returned by the ‘V’ command is in the following format (See Table 3):

ff,xx.xx,mmddyyyy,hh,HH,XX,YY

TABLE 3: VERSION STRING FORMATS

Format	Description
ff	Supported Function Bits. This value is two bytes representing one hexadecimal byte. Bit 0 = supports c and C Bit 1 = supports B Bit 2 = supports R Bit 3 = supports h and H
xx.xx	Version string. Anything goes as long as it is 2 characters, a period, and two more characters.
Mmddyyyy	Two characters each for month and day followed by 4 characters for the year. Please use numbers.
hh	This is the number of counts of throttle input that corresponds to a pulse of 1msec. For an ideal radio system, this would be 0% throttle. It is assumed that 0 counts is less than 1 ms.
HH	This is the number of counts of throttle input that corresponds to a pulse of 2 ms. For an ideal radio system this would be 100% throttle. It is assumed that the maximum supported counts is greater than 2 ms. The total number of counts supported is supplied by the 'N' command.
XX	Number of points in the X axis of the lookup table.
YY	Maximum size of any value in the lookup table.

Software

The software for this ESC has two parts. The first part is the foreground application that runs the speed control algorithms. The second part runs in the background as a series of interrupt handlers. This part measures the servo pulses, watches for missing pulses and monitors the battery for low voltage. The foreground application is divided into three modes; Arming, ESC and PC mode.

BACKGROUND APPLICATION

The background application has three parts; Timer0 interrupt for detecting missing pulses, Comparator 0 interrupt for detecting a low battery, and Comparator 1 interrupt for detecting and measuring the incoming pulse.

Timer0:

Timer0 was configured to interrupt every 65 ms. This corresponds to slightly more than 3 servo input pulses. If the pulse measurement is successful, TMR0 is reset to hold off the interrupt. If the interrupt occurs, the interrupt routine sets a missing pulse flag. The foreground application will turn off the PWM and wait for the pulses to return. This will not cause a rearming sequence.

Comparator 0:

While both comparators share the same interrupt, each one is handled differently. Comparator 0 monitors the battery voltage and compares it to the internal voltage reference. The internal reference has been configured for 1.042V. The resistors (R1, R2) divide the battery voltage by 4.7. This combination causes the comparator to detect a low battery condition when the battery voltage reaches 4.896V. While the comparator is set (detecting the low voltage condition) the PWM signals are disabled. When the battery recovers, a rearming cycle is forced so the pilot must set the throttle to 0 before the motor will start. The pilot should be aware that the battery is low and that too high of a power setting will cause the low voltage cutout to happen again. This feature has been flight tested and works very well. It would be a good idea to add a programmable setpoint to this feature in the future.

Comparator 1:

This comparator interrupts when the servo signal rises and falls. When the signal rises, Timer1 is started. When the signal falls, Timer1 is stopped and the value is copied into the pulse-time variable. Timer1 is then reset and the wait resumes for a rising edge. The foreground application is responsible for all pulse scaling and PWM setting.

FOREGROUND APPLICATION

Arming Mode:

During the Arming mode, the ESC is determining if it is safe to enter ESC mode and begin motor operations. The ESC also monitors the serial port to determine if a PC calibration tool needs to communicate. During Arming mode, the interrupts are running to measure the incoming pulses. The Speed Control mode arms when 10 consecutive pulses are measured as smaller than the programmable Arming Threshold. The PC mode arms when the receiver receives a 0x80.

ESC Mode:

During ESC mode, the software continuously measures the servo control pulses and determines the correct PWM value for the motor. PWM values are determined through a lookup table programmed into the EEPROM memory by the PC service tool. The sequence of events is as follows:

1. Wait for a pulse measurement to complete.
2. Scale the pulse to a number from 0-99.
3. Check the pulse against the full throttle limit.
4. Check the pulse against the minimum throttle limit.
5. Look up the required PWM value.
6. Set the PWM.
7. Repeat, unless rearming is required.

PC Mode:

The PC mode simply waits for characters and processes them according to the protocol described earlier. A break character will cause a rearming.

Testing

Initial bench testing was performed without variable rate PWM. At the 32 kHz rate, high throttle settings caused the transistors and Schottky diode to get very hot. Later testing with the variable rate PWM showed a dramatic reduction in temperature, indicating increased efficiency and less power loss. Variable rate PWM is definitely a good way to go.

At low RPM settings, the high rate PWM provides very smooth and quiet operation. The motor made no discernible noise and turned smoothly at very low RPM's. These low RPM's would not fly an airplane, but other applications may find this level of control desirable.

Flight testing with a reverse exponential curve proved successful. The aircraft responded to throttle settings in a much more realistic manner. Half throttle resulted in half power. Low throttle caused a powered descent. Full throttle remained unchanged.

Conclusion

Microcontroller motor controls are ideal for many applications, especially electric flight. These controls can be made smaller and with more advanced features than would otherwise be possible. PICmicro Microcontrollers are ideal candidates in these applications due to their smaller size, ideal feature set, and simple development tools.

FIGURE 6: ESC CLOSE-UP (TOP VIEW)

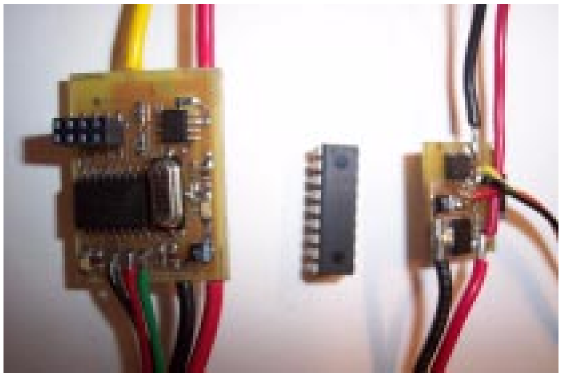
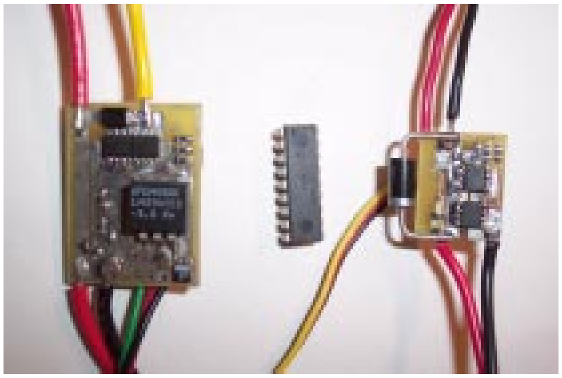


FIGURE 7: ESC CLOSE-UP (BOTTOM VIEW)

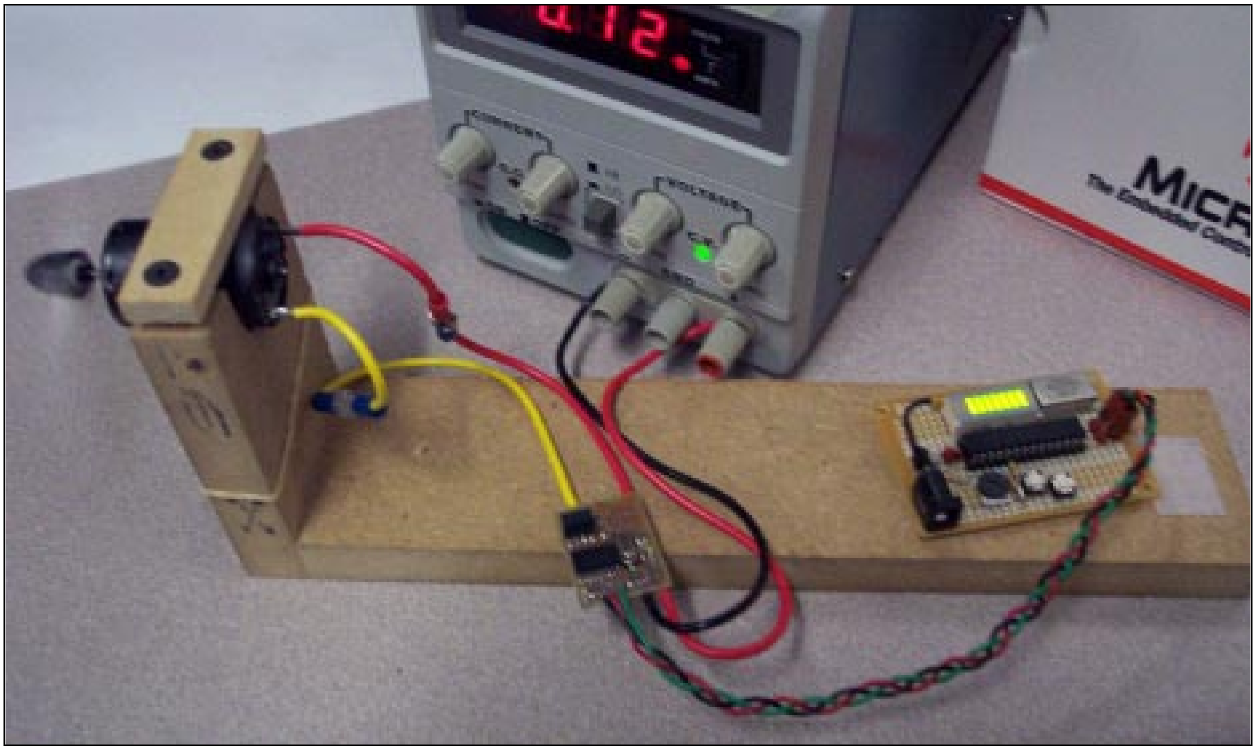


Note: 18-Pin DIP is used for size reference only (Figures 7 and 8).

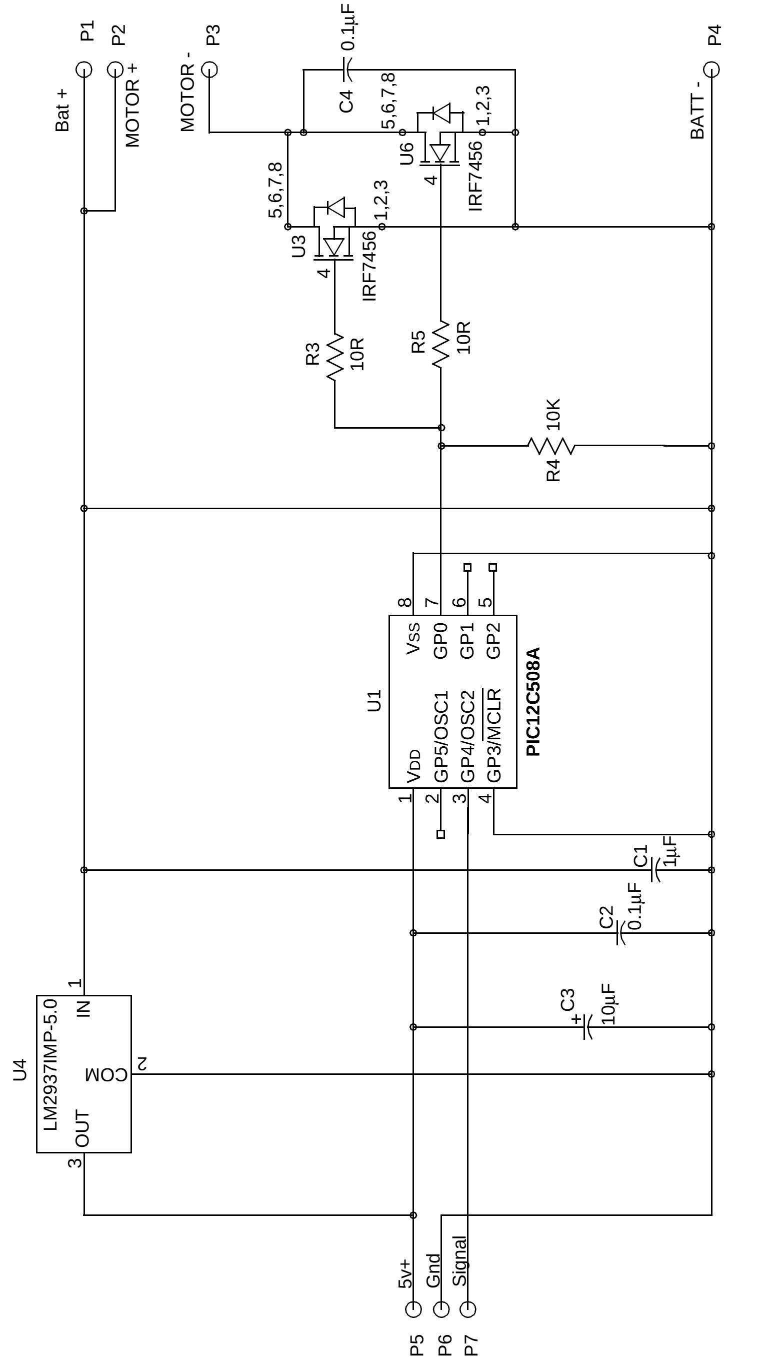
TABLE 4: MEMORY USAGE

Module	Program Memory (words)	File Registers (bytes)	EEPROM Data Memory (bytes)
Design 1	246	8	–
Design 2	1366	25	107

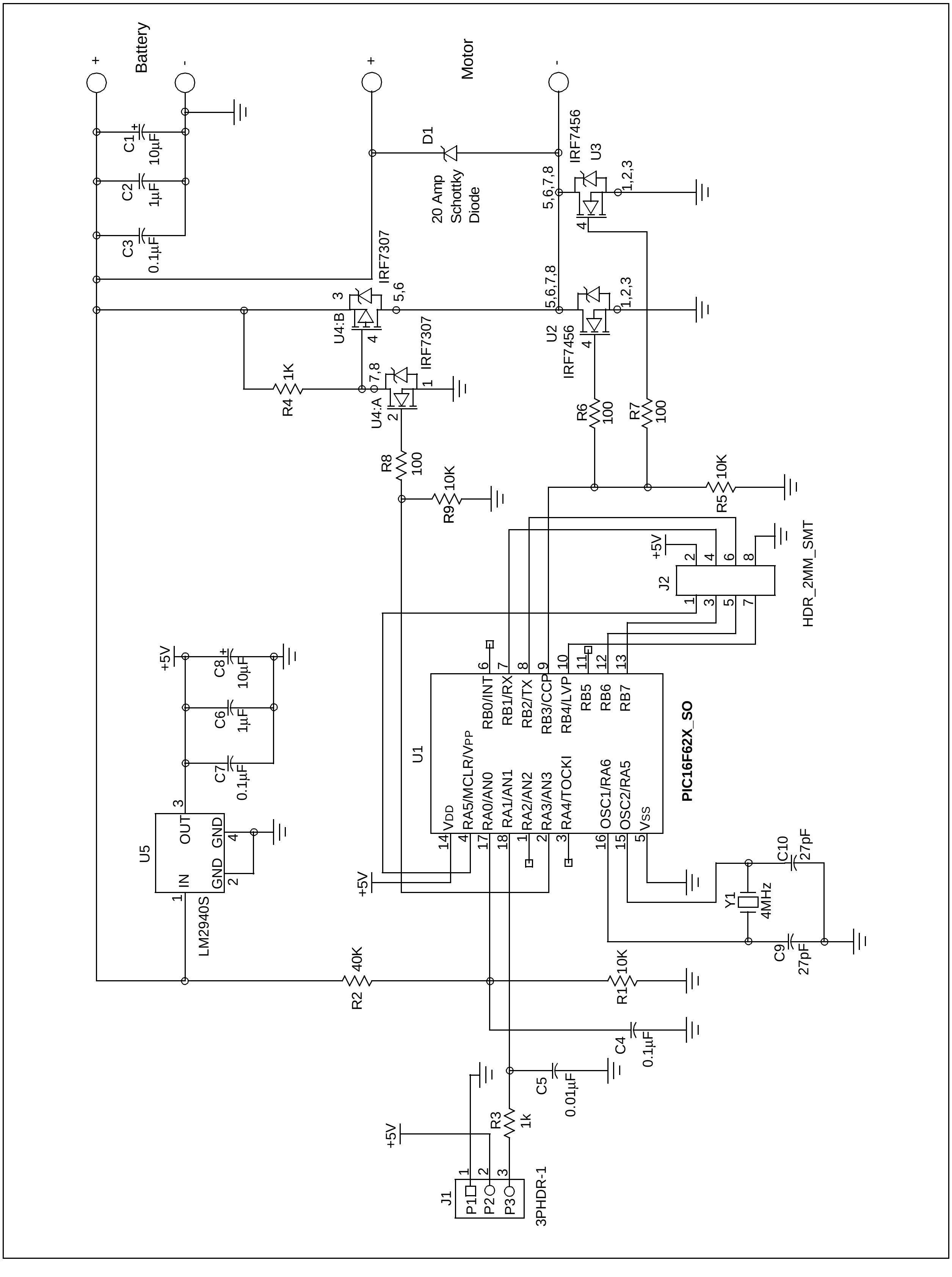
FIGURE 8: BENCH TESTING SETUP



APPENDIX A: SCHEMATIC - VERSION 1 (MINIMAL ESC)



APPENDIX B: SCHEMATIC - VERSION 2 (MOTOR BRAKE/VOLTAGE CUTOUT/ PROGRAMMABLE THROTTLE RESPONSE/HIGHER RATE PWM)



NOTES:

Note the following details of the code protection feature on PICmicro® MCUs.

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products 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 PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be 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 product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip’s products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, MXDEV, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

dsPIC, dsPICDEM.net, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXLAB, PICC, PICDEM, PICDEM.net, rfPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

Printed on recycled paper.

DNV Certification, Inc.
USA

ANSI · RAB
QMS
ACCREDITED

DNV

DNV MSC
The Netherlands
Accredited by the RvA

ISO 9001 / QS-9000
REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999 and Mountain View, California in March 2002. The Company’s quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, non-volatile memory and analog products. In addition, Microchip’s quality system for the design and manufacture of development systems is ISO 9001 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: 480-792-7627
Web Address: http://www.microchip.com

Rocky Mountain
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7966 Fax: 480-792-4338

Atlanta
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Boston
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848 Fax: 978-692-3821

Chicago
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423 Fax: 972-818-2924

Detroit
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Kokomo
2767 S. Albright Road
Kokomo, Indiana 46902
Tel: 765-864-8360 Fax: 765-864-8387

Los Angeles
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

New York
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

Toronto
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699 Fax: 905-673-6509

ASIA/PACIFIC

Australia
Microchip Technology Australia Pty Ltd
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

China - Beijing
Microchip Technology Consulting (Shanghai)
Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

China - Chengdu
Microchip Technology Consulting (Shanghai)
Co., Ltd., Chengdu Liaison Office
Rm. 2401, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-86766200 Fax: 86-28-86766599

China - Fuzhou
Microchip Technology Consulting (Shanghai)
Co., Ltd., Fuzhou Liaison Office
Unit 28F, World Trade Plaza
No. 71 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7503506 Fax: 86-591-7503521

China - Shanghai
Microchip Technology Consulting (Shanghai)
Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

China - Shenzhen
Microchip Technology Consulting (Shanghai)
Co., Ltd., Shenzhen Liaison Office
Rm. 1315, 13/F, Shenzhen Kerry Centre,
Renminnan Lu
Shenzhen 518001, China
Tel: 86-755-2350361 Fax: 86-755-2366086

China - Hong Kong SAR
Microchip Technology Hongkong Ltd.
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

India
Microchip Technology Inc.
India Liaison Office
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

Japan
Microchip Technology Japan K.K.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa, 222-0033, Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-6334-8870 Fax: 65-6334-8850

Taiwan
Microchip Technology (Barbados) Inc.,
Taiwan Branch
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

Austria
Microchip Technology Austria GmbH
Durisolstrasse 2
A-4600 Wels
Austria
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

Denmark
Microchip Technology Nordic ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France
Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany
Microchip Technology GmbH
Gustav-Heinemann Ring 125
D-81739 Munich, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy
Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

United Kingdom
Microchip Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

05/16/02