Cable Tracer MT2202

This project used PIC24FJ64A004 and Bonus Part MCP6S26.

The Cable Tracer is a device to detect underground cables. It was designed and tested on un-energized electric power cables but could be used on any underground cable. A 125 Khz signal is injected into the underground cable. A pickup coil receives a sample of this field. The strength of this field will indicate the presence and direction of the cable.

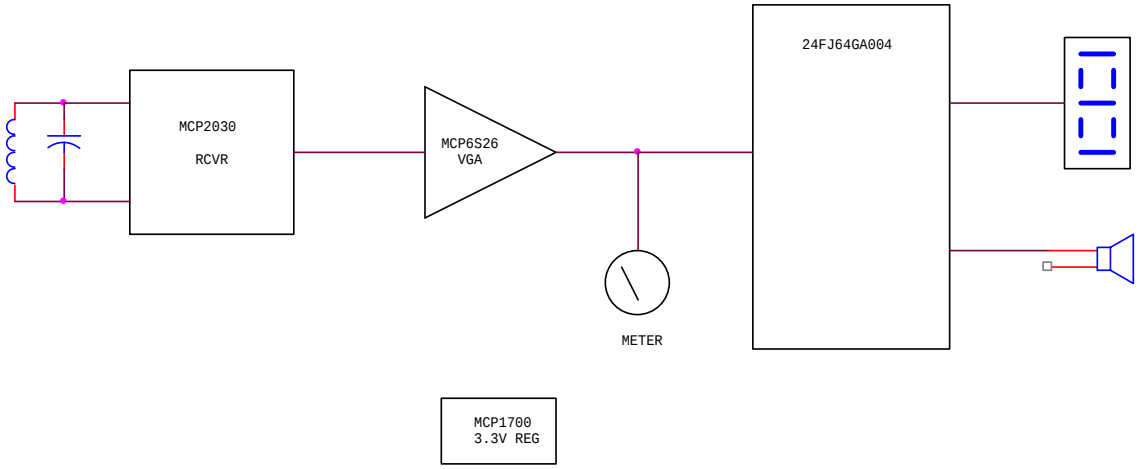
The Receiver consists of a PreAmp and Detector made from an RFID Chip (MCP2030). The RSSI signal strength output is amplified by a Variable Gain Amplifier (MCP6S26). The amplified output feeds an Analog Meter to provide a visual indication of the signal strength. The meter also has a sensitivity control which allows the meter to have enough sensitivity to indicate the direction of the cable by slightly rotating the pickup coil. The RSSI signal is fed to an A/D input on the PIC24FJ64GA004. It also feeds a Comparator input. The A/D channel is used to drive an audio Variable Frequency Oscillator based on a Timer Output to provide an audio indication of the signal strength similar to a Metal Locator. The Comparator is used to indicate a course signal strength that is used in conjunction with the A/D value to determine if the VGA should change the strength of the signal.

A 7 Segment LCD display is used to indicate the amplification factor of the VGA. An LCD is used to conserve battery power. The AC drive to the display is provided by an output port in conjunction with the firmware. This indication is used to determine if the signal source is at an appropriate level. With some operator practice it will also help indicate relative depth changes and/or cable breaks. There is also a Auto/Manual switch, Up, and Down buttons. In some cases Manual Operation of the amplification factor is useful.

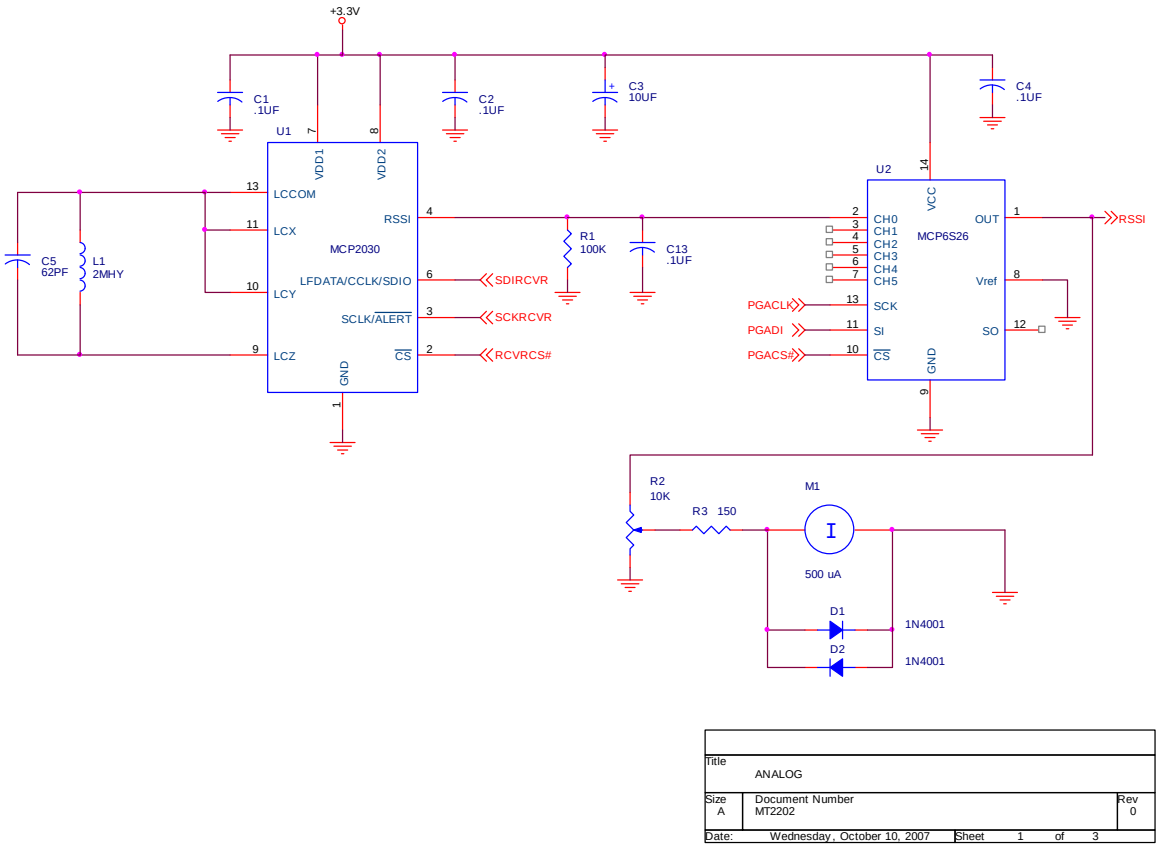
Another comparator is used to monitor Battery voltage. An LED is used to indicate a Low Battery condition.

The battery is four AA batteries to provide 6 volts. The 6 volt battery supplies the power to a 3.3V Low Drop Out regulator (MCP1700). The power consumption of the Cable Tracer is low enough to not cause heating of the LDO. The 6 volt source and LDO provide a reasonable amount of operational time before the battery discharges to an unusable level.

The Peripheral Pin Select was particularly useful in selecting the desired features of the microcontroller for this application.

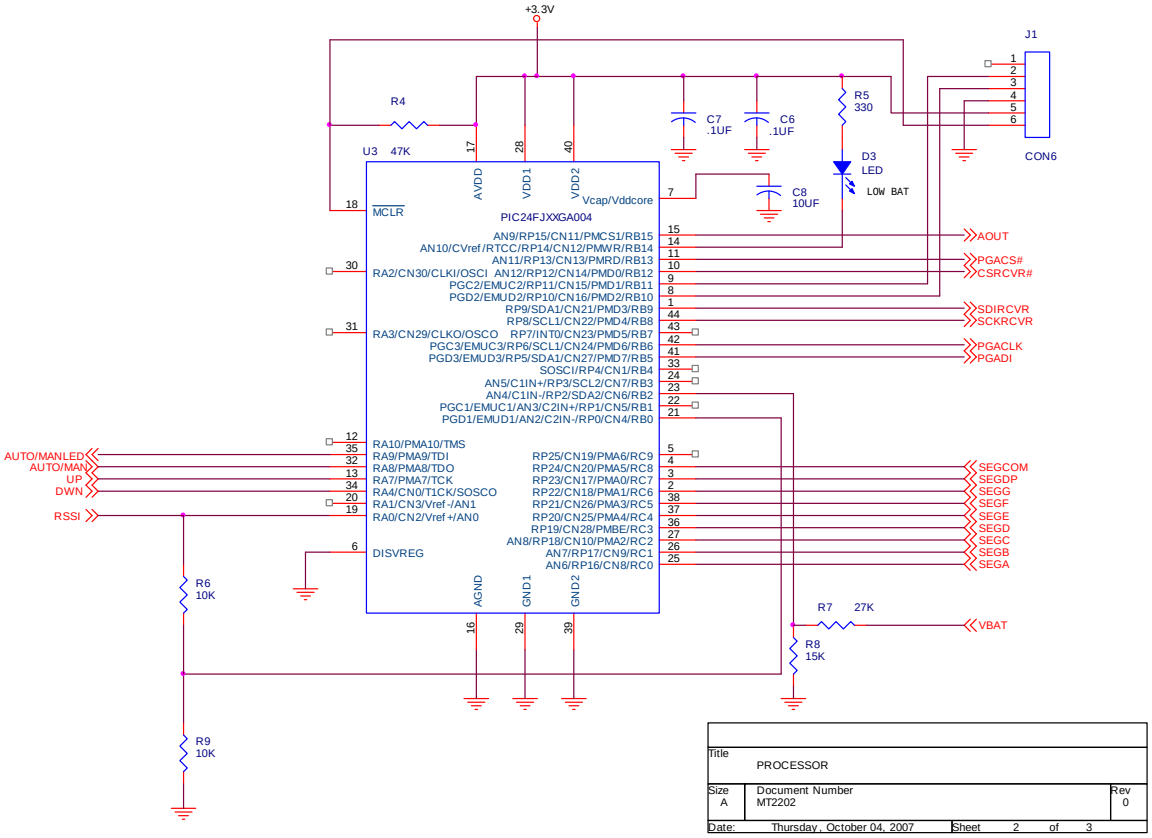


Cable Tracer MT2202

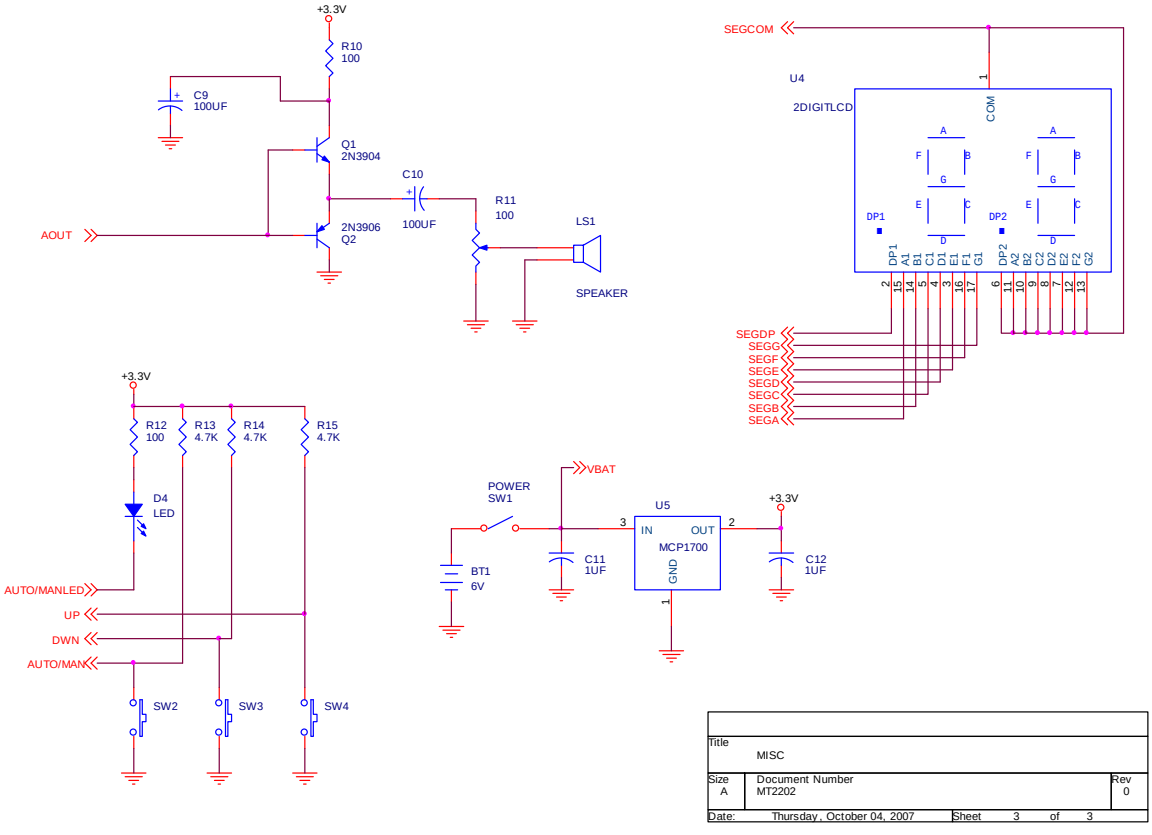


Title: ANALOG		
Size: A	Document Number: MT2202	Rev: 0
Date: Wednesday, October 10, 2007	Sheet 1 of 3	

Cable Tracer MT2202



Title	PROCESSOR	
Size A	Document Number MT2202	Rev 0
Date: Thursday, October 04, 2007	Sheet 2 of 3	



Title	MISC	
Size A	Document Number MT2202	Rev 0
Date: Thursday, October 04, 2007	Sheet 3 of 3	

```
while(1)
{
        ADConvert();                                    //Get RSSI Value
        loopct++;
        for (x=0; x < 0x7FFF;x++);                      //Delay
        if(AUTOMANSW == 0)                              //Push Button Pressed
                {
                while(AUTOMANSW == 0);                  //wait for release
                AUTOLED = ~AUTOLED;                     //Toggle Mode
                }

        if(AUTOLED == 1)                                //If MANUAL Mode
        {
                if(UPSW == 0)
                {
                        while(UPSW == 0);
                        if(gain < 0x4007)
                        {
                        gain++;                         //Boost Gain
                        PGACS = 0;                      //Select PGA
                        SPI1BUF = gain;
                        while(SPITBF == 1);             //Wait for empty
                        for (x=0; x<= 10;x++);          //Delay
                        PGACS = 1;
                        LCDout(gain & 0x0007);          //Out gain setting
                        }//end gain < 4007
                }//end UPSW
                if(DWNSW == 0)
                {
                        while (DWNSW == 0);
                        if(gain > 0x4000)
                        {
                                gain--;                 //Drop Gain
                                PGACS = 0;              //Select PGA
                                SPI1BUF = gain;
                                while(SPITBF == 1);     //Wait for empty
                                for (x=0; x<= 10;x++);  //Delay
                                PGACS = 1;
                                c2flag = 0;
                                LCDout(gain & 0x0007);
                        }//end gain > 4000
                }//end DWNSW
        }//end AUTOLED == 1
        else
                _C2EN = 1;                              //Enable for AUTO Mode

        if(loopct >= LOOP)
        {
                if((c2flag == 1)&(AUTOLED == 0))
//Gain too high
                {
                        if(gain > 0x4000)
                        {
                                gain--;                 //Drop Gain
```

```
                        PGACS = 0;              //Select PGA
                        SPI1BUF = gain;
                        while(SPITBF == 1);     //Wait for empty
                        for (x=0; x<= 10;x++);  //Delay
                        PGACS = 1;
                        c2flag = 0;
                        LCDout(gain & 0x0007);
                        _C2EN = 1;              //Re Enable after GAIN drop
                }//end gain>0x4000
        }//end c2flag == 1

        if((ADValue < GAIN_TRIP)&(AUTOLED == 0))    //Gain too low
        {
                if(gain < 0x4007)
                {
                gain++;                         //Boost Gain
                PGACS = 0;                      //Select PGA
                SPI1BUF = gain;
                while(SPITBF == 1);             //Wait for empty
                for (x=0; x<= 10;x++);          //Delay
                PGACS = 1;
                LCDout(gain & 0x0007);          //Out gain setting
                }//end gain < 4007
        }//end ADValue < GAIN_TRIP
loopct = 0;
}//end loopct >= LOOP


ADValue = ADValue & 0x03FF;             //10 bits
                                        //Oscillator PR4 = 1 = 32Khz
                                        // PR4 = 03FF = 60 Hz
                                        //Divide value by 2 for 120 to 15Khz
ADValue = 0x03FF - ADValue;             //Get as low to high on pitch
if(ADValue > 2)
        ADValue = ADValue / 2;
else
        ADValue = 1;
TMR4 = 0x0000;                          //Clear Reg, Otherwise might miss compare
PR4 = ADValue;                          //Load Oscillator
                                        //0x3FF 16 mS 60Hz
                                        //0x001 31.2Khz 32 uS
```

Cable Tracer MT2202