Date: 12-30-11

Subject: PICPCAP Version 4 Release Notes

Re: Version 4

From: JGH

Summary

The following lists a summary of the changes:

- Added the new decoding algorithms. Currently fixed at up to 8 touches, can be adjusted to do less or more, RAM dependent. The new decoding algorithms are less sensitive to thresholds
- The new source code can be compiled for processors PIC18F, PIC24F, and PIC32MX. The source contains three projects, one for each
- The majority of the code is generic to PCAP, these projects allow us to make general changes to the functionality of PCAP code across all processors without editing each processor type independently keeping all PCAP code the same
- Processor specific code is now located in pcap_hardware_processor.c, pcap_hardware_processor.h, and sensor_processor.h. Any changes specific to the processor you are using should be located in the file specific to your processor. Create a new project and files for additional processors
- ROM lookup tables (LUT) for each processor are employed making setting up new processors and using current ones much simpler
- Memory models are flat meaning all processors now use the same RAM structures and two dimensional arrays
- Not all features are currently implemented for all processor types, these will be added as the projects mature
- All sensor specific settings are located in the processor specific header file sensor_processor.h

Base Projects

The projects are based on the following base processor/hardware; refer to appendix A for the schematics:

- PIC18F46K22
- PIC24FJ64GB106
- PIC32MX250F128D

The following table lists current support for each processor, a check mark indicates it is included.

Processor	I2C	UART	USB	EEPROM	Sleep	Short Checking	Sensor Support
PIC18F	√	√		√		√	AMT P3002
PIC24F	√		√	√			AMT P3002
PIC32MX	√						AMT P3002

Setting up and Building a Project

To setup and build a project do the following:

- Open one of the project files based on your processor and hardware. There are 3 of them: PIC18PCAP, PIC24PCAP, and PIC32PCAP
- At the top of main.h define the processor you are using, there are 3: PCAPPIC18F, PCAPPIC24F, and PCAPPIC32MX

The projects are shown in figure 1.

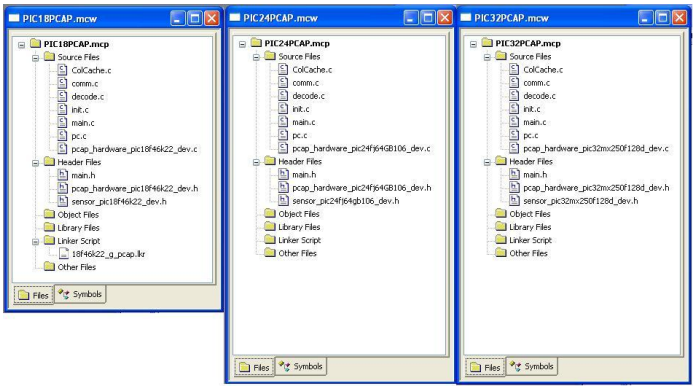


Figure 1. Projects

Note each project is the same other than 3 files: pcap_hardware_processor_dev.c, _hardware_processor_dev.h, and sensor_processor_dev.h. Note also the addition of the linker script for the PIC18F project. The linker script removes the dependence on banked memory an the PIC18F allowing generic RAM defines across all processors.

Processor Specific Files

The processor specific files are used to contain any differences between the processor types such as initialization and timers. As mentioned before, the actual PCAP code is generic, the specifics of setting up and acquiring the PCAP signals are dependent on the hardware and processor type.

Macros

Keep macros short, they are difficult to read and difficult to debug. Macros are typically used when setting up things like timers, CTMU, and communications. For example to set timer 1 on a PIC24 uses

the instruction T1CONbits.TON = 1 and on a PIC32 the instruction T1CONSET = 0x00008000. This is a good use of a macro.

C files

The specific C file, pcap_hardware_processor_dev.c, contains function specific to a processor such as setup and acquiring the PCAP signal. These files contain functions used for each processor type.

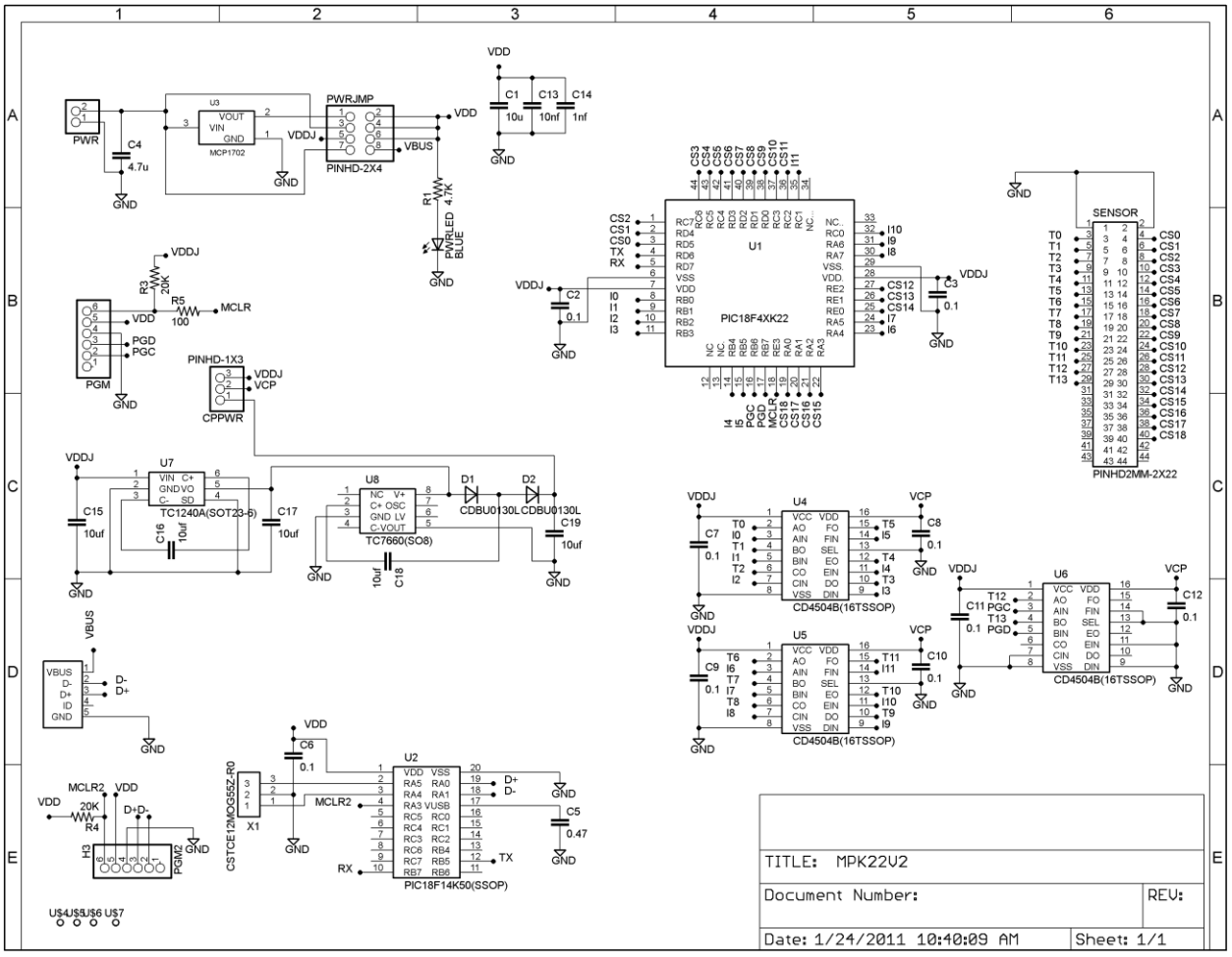
RAM memory Sizes

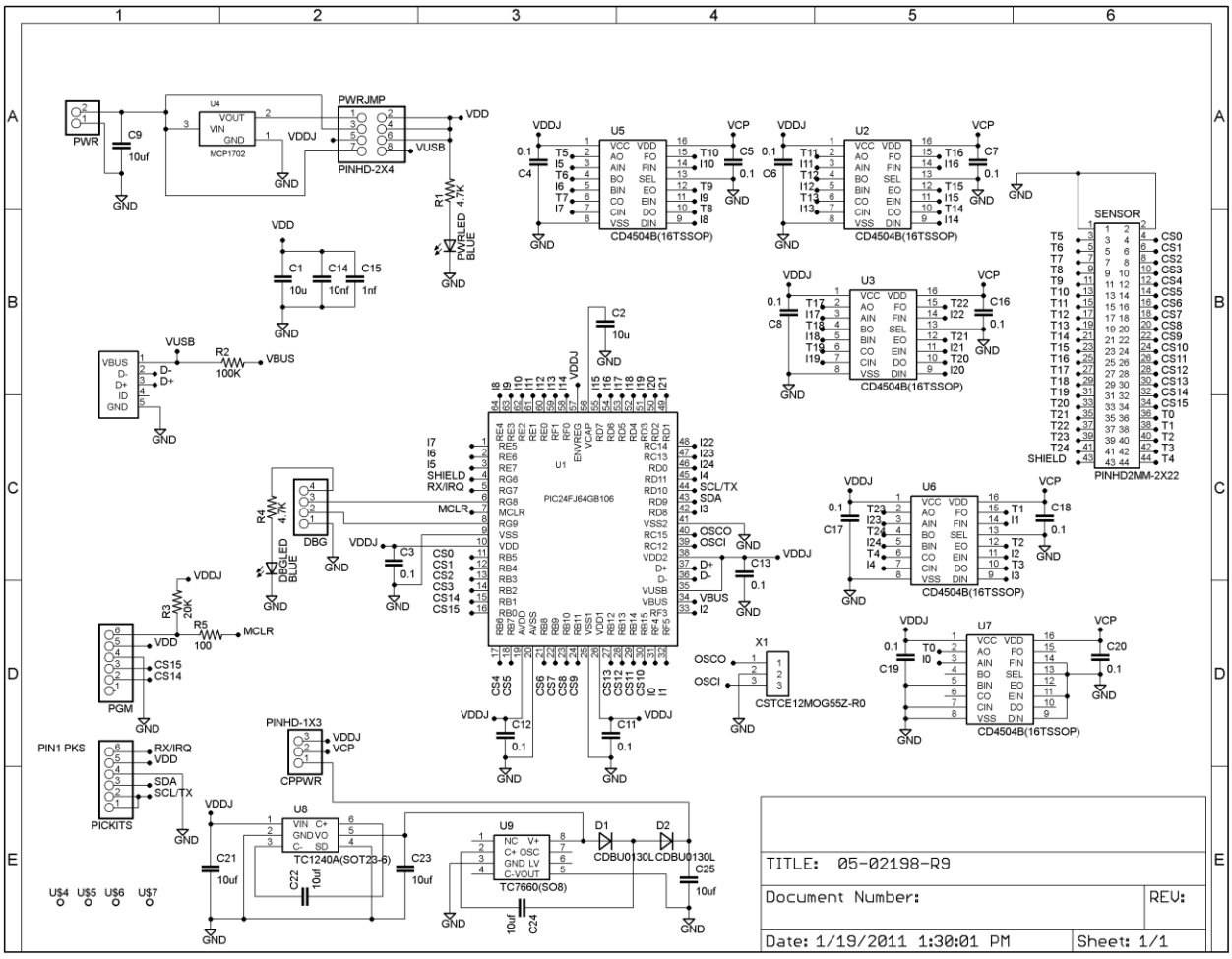
Be careful when defining new variable sizes, an unsigned int on a PIC32 is 32 bits while on a PIC18 or PIC24 it is 16 bits.

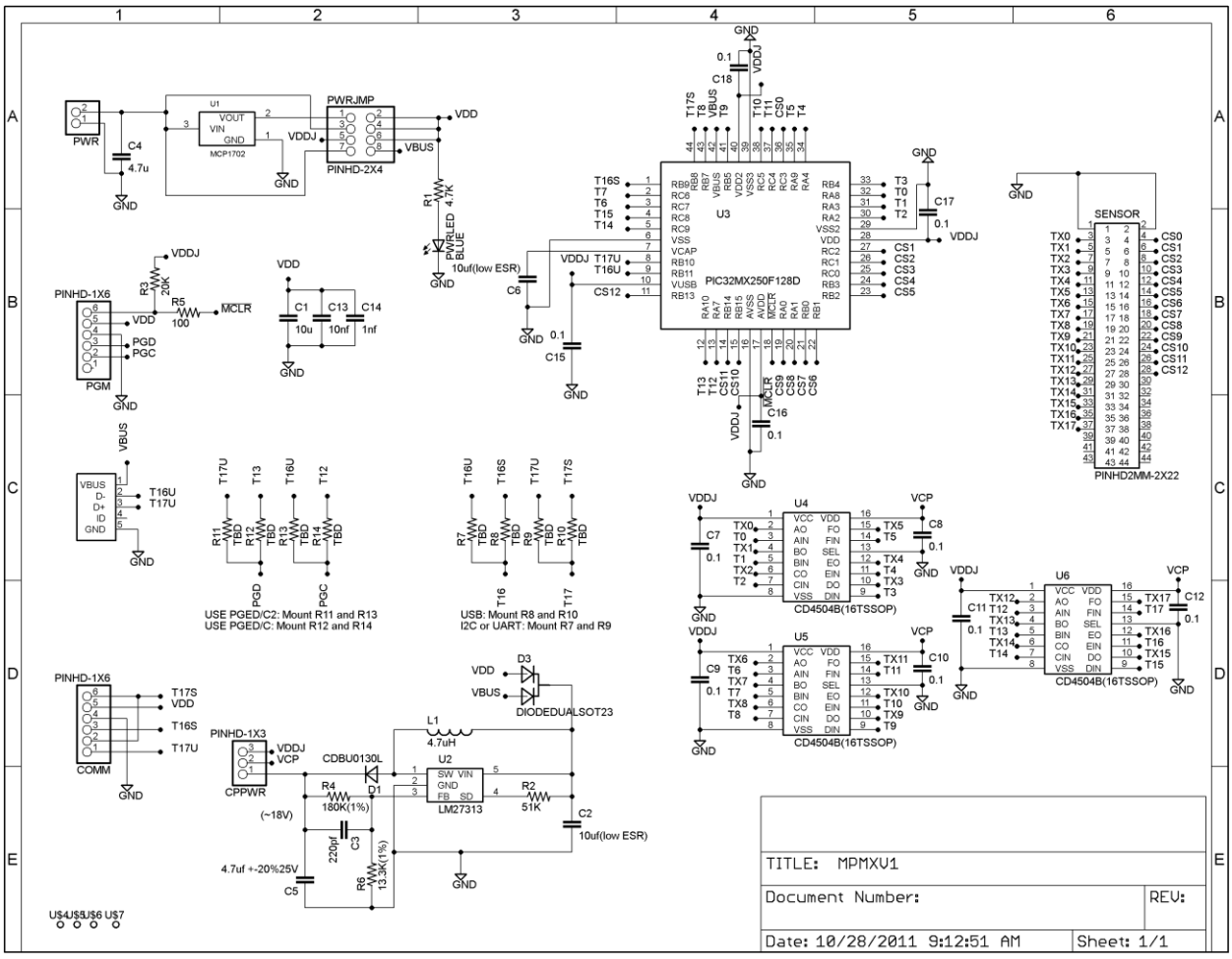
Flat RAM Model

The PIC24 and PIC32 already use a flat memory model, the PIC18 requires a modified linker script exposing all RAM as flat memory. Be aware of the linker script when moving to other processors in the 18F and 16F domain.

Appendix A




PWR
C9
10uf
U4
VOUT
VIN
GND
MCP1702
VDDJ
PWRJMP
PINHD-2X4
VDD
VUSB
R1
4.7K
PWRLED
BLUE
C1
10u
C14
10nf
C15
1nf
VBUS
R2
100K
D-
D+
ID
U5
U2
U3
CD4504B(16TSSOP)
VCP
C2
10u
SENSOR
PINHD2MM-2X22
SHIELD
RX/IRQ
MCLR
SCL/TX
SDA
OSCO
OSCI
U1
PIC24FJ64GB106
DBG
DBGLED
BLUE
R4
4.7K
R3
20K
R5
100
PGM
CS15
CS14
PIN1 PKS
PICKITS
RX/IRQ
SDA
SCL/TX
PINHD-1X3
CPPWR
X1
CSTCE12MOG55Z-R0
U6
U7
U8
TC1240A(SOT23-6)
U9
TC7660(SO8)
D1
D2
CDBU0130L
C21
10uf
C22
10uf
C23
10uf
C24
10uf
C25
10uf
U$4
U$5
U$6
U$7
TITLE: 05-02198-R9
Document Number:
REV:
Date: 1/19/2011 1:30:01 PM
Sheet: 1/1


PWR
C4
4.7u
U1
VOUT
VIN
GND
MCP1702
PWRJMP
PINHD-2X4
VDD
VDDJ
VBUS
R1
4.7K
PWRLED
BLUE
10uf(low ESR)
C6
C15
0.1
PIC32MX250F128D
U3
C18
C17
C16
MCLR
PINHD-1X6
PGM
R3
20K
R5
100
C1
10u
C13
10nf
C14
1nf
PGD
PGC
VBUS
T16U
T17U
R11
R12
R13
R14
TBD
USE PGED/C2: Mount R11 and R13
USE PGED/C: Mount R12 and R14
R7
R8
R9
R10
USB: Mount R8 and R10
I2C or UART: Mount R7 and R9
SENSOR
PINHD2MM-2X22
U4
U5
U6
CD4504B(16TSSOP)
VCP
C7
C8
C9
C10
C11
C12
PINHD-1X6
COMM
T17S
T16S
PINHD-1X3
CPPWR
(~18V)
CDBU0130L
D1
D3
DIODEDUALSOT23
L1
4.7uH
U2
LM27313
R4
180K(1%)
R2
51K
C2
10uf(low ESR)
C3
220pf
4.7uf +-20%25V
C5
R6
13.3K(1%)
U$4 U$5 U$6 U$7
TITLE: MPMXV1
Document Number:
REV:
Date: 10/28/2011 9:12:51 AM
Sheet: 1/1