A climate chamber for simulating a temperature and humidity gradient across a wall or roof

Tim Padfield
Department of Building and Energy, Technical University of Denmark
Conservation Department, The National Museum of Denmark

28/12/2000

Form and Purpose

The climate chamber (figure 1) is an open topped cylinder made entirely of stainless steel. It is 800 mm across and 500 mm deep. The chamber can be used in two ways: A test specimen can be mounted to form the lid of the chamber, allowing investigation of the effects of a gradient in temperature and relative humidity on a porous structure. Alternatively the top can be sealed with a metal plate, so that the cylinder encloses the specimen under test in an airtight space.

The temperature and relative humidity (RH) of the chamber can be controlled between about 30^{o}C and 10^{o}C and from about 40% to 95% RH. A unique facility is that the water moving into or out of the chamber, or into and out of specimens sealed within the chamber, can be measured: the humidity control unit within the chamber measures how much water it collects or releases. Basically, water is evaporated from, or condensed into a weighed water tank whose temperature is controlled by a Peltier heat pump. The apparatus is shown in figure 2.

The physical layout

The test specimen can be placed entirely within the chamber. It can be a vertical slab resting on the floor (maximum dimensions about 450 x 450 x 50 mm) or it can be suspended horizontally, resting on a grid which is supported by studs welded to the side of the chamber (max 790 mm dia. x 150 deep). The chamber has a second set of studs near the top, so that a specimen can be laid across the top of the cylinder, in place of the metal lid (790 mm x 1000 mm max. depth).

There is a removable tray which rests on studs just above the base of the chamber. This tray bears the water flux control system, climate sensors, ventilators and electrical connections. This can be lifted up for routine servicing and can be entirely disconnected for thorough servicing.

This 'reaction chamber' is enclosed by an annular outer chamber which contains a circulating air stream whose temperature is controlled by an electric heater and a finned copper tube containing recirculating cold water. The temperature of the entire vertical wall of the inner chamber is thus controlled at an even temperature. The heat transfer surface is so large that there is only a small variation in temperature over the surfaces of the inner chamber. This means that the RH can be very high, about 95% if the test specimen is insulating, without risking condensation on the walls. The base of the inner chamber is well insulated by 300 mm of dense expanded polystyrene.

The cold water comes from a separate, conventional cooling unit. The water temperature is manually adjustable between 15 and 6 degrees. The water flow through the coil in the annular space is controlled by a magnetic valve.

There is a second cold water system which cools the heat pump which controls the chamber humidity. The temperature of this water is held two degrees above the dew point in the chamber, so its temperature will vary. This is achieved by recirculating the water through a tank which is cooled by a branch of the primary cold water supply. The water temperature in the tank is controlled by a magnetic valve in the primary cooling supply.

The electrical layout

Electrical services are brought in through three airtight connectors in the base of the chamber. A fourth cable serves the outer chamber. In the inner chamber there is one cable for control measurements. The second cable carries power to the humidity controller and to the sensors. The third cable (8 pairs of wires) is for test measurements in and around the specimen. All cables are brought to terminals on the movable tray that holds the air conditioning apparatus and the climate sensors. The cables are long enough that the tray can be raised for servicing. The cables can be disconnected at the chamber floor, so that the tray can be removed entirely for major repairs.

The control sensor signals are brought from the inner chamber directly to the terminals of the data logger, to minimise electrical noise. The power cable, however, divides: some leads connect directly to the data logger while others are connected at a terminal block near the data logger. This terminal block distributes power supplies and digital signals. Test measurement wires are brought first to the connection block and then into the data logger. This allows these signals to come either through the cable mounted in the base of the chamber or directly from sensors mounted in specimens laid over the top of the chamber.

The control and measurement hardware

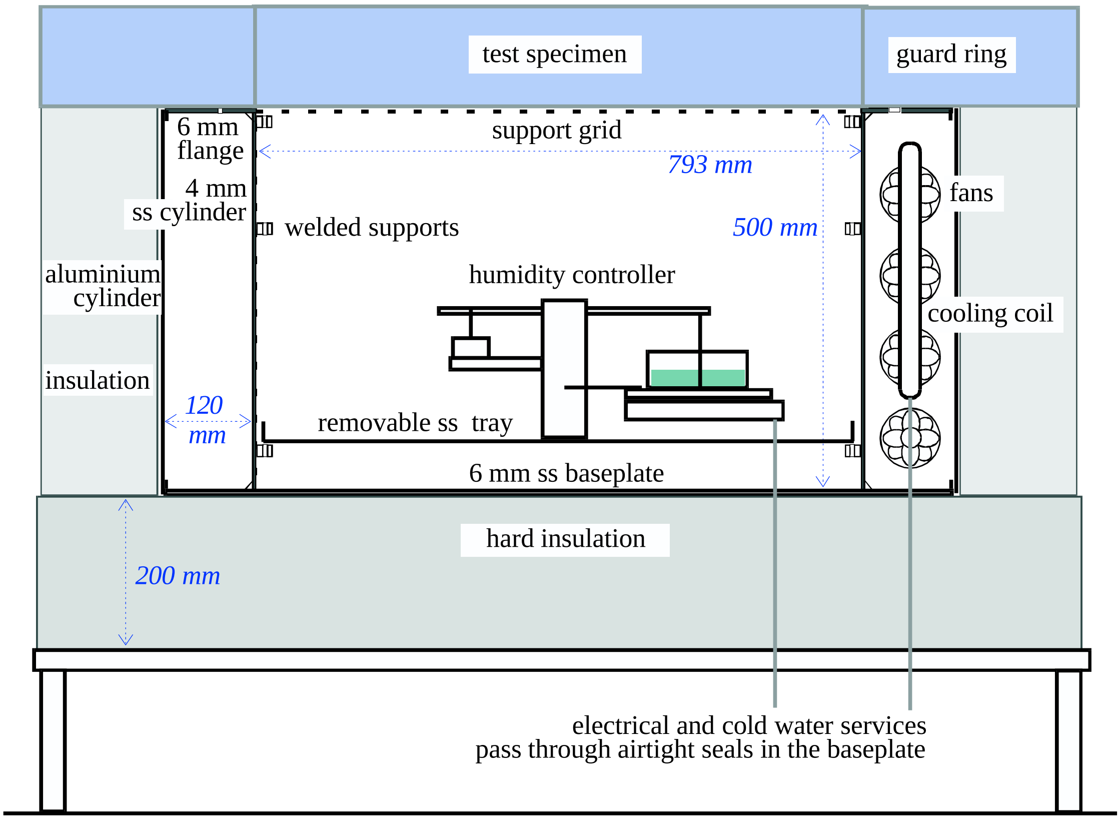
The chamber temperature is measured by a four wire platinum resistance sensor. The dew point is measured by a cooled mirror dew point sensor, which also uses a four wire platinum resistance sensor. Other temperatures needed for control are measured by type K thermocouples: cooling water temperature, heat sink temperature (to check for overheating of the heat pump) and condenser temperature (just for testing performance). The tank of the humidity controller is suspended from an aluminium beam, whose deflection under load is measured by a strain gauge bridge. The beam is tilted by an electrically driven crank a few seconds before each measurement, so that the tank and the attached heat pump are raised from the heat exchanger and hang free on the beam. The electrical power to the heat pump is brought in through thin flexible stainless steel strips.

The control and measurement program

The various sensors are connected to an Agilent (Hewlett Packard) data logger. The data logger is connected to a computer where a continuously active program collects the data, analyses it and sends control signals back through the data logger to the various water valves and devices in the chamber. The data logger accepts instructions and requests in a standard ASCII text format used in the instrument industry, called the SCPI command syntax. This is described in the user's guide to the data logger. Configuring the program is described in the next chapter.

Figure 1

Climate chamber with measured water vapour exchange



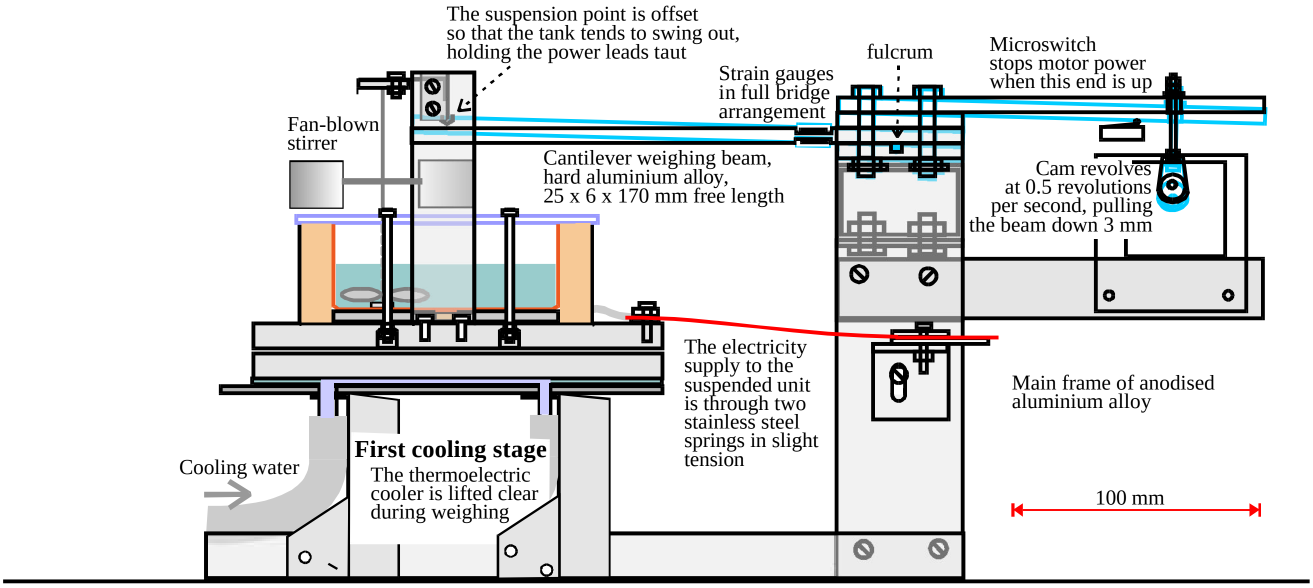
The test chamber is a cylindrical well which contains the humidity control apparatus.
The temperature in the well is controlled by a cooling coil in the outer annular chamber.
The specimen is laid on a grid and enclosed by a thin wall continuing the line of the chamber wall.
Outside this wall is a thermal guard ring of the same material.

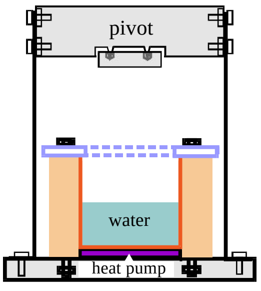
Figure 2

Humidity controller

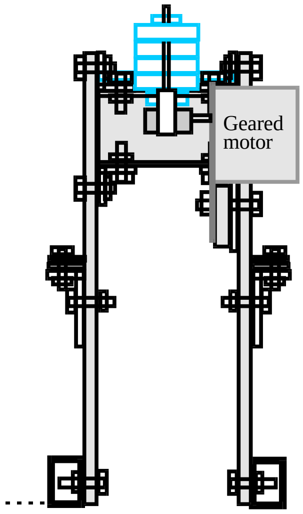
Tim Padfield 28.12.2000

Water vapour is condensed into, or evaporated from the weighed container, according to the water temperature. This is controlled by a Peltier heat pump under the water tank.
The lower side of the Peltier heat pump is held at the temperature of the circulating cooling water. The entire assembly of water tank and heat pump is raised from the heat exchanger for weighing


The suspension point is offset so that the tank tends to swing out, holding the power leads taut
Fan-blown stirrer
Strain gauges in full bridge arrangement
fulcrum
Microswitch stops motor power when this end is up
Cantilever weighing beam, hard aluminium alloy, 25 x 6 x 170 mm free length
Cam revolves at 0.5 revolutions per second, pulling the beam down 3 mm
The electricity supply to the suspended unit is through two stainless steel springs in slight tension
Main frame of anodised aluminium alloy
First cooling stage
The thermoelectric cooler is lifted clear during weighing
Cooling water
100 mm



Suspended water tank and thermo-electric heat pump



The control program for the climate chamber

The operating system is Linux. The programs and the setup files are written in Python, which is a 'c-like' scripting language, well suited to dealing with the message format that the data logger uses. The computer can be controlled over the internet. It will give some starting pains for people used to Windows programs but Python is easy to learn and it gives a satisfyingly direct control over the experiment.

The control program consists of a main body of code ***climchxx*** (where xx is a version number). It reads climate data collected in the data logger at one minute intervals. It calculates the necessary adjustments to keep the climate on the course defined in two setup files, which you will have to alter to suit your purpose.

The first setup file is ***climch_cc.py***. It specifies the course of the climate within the chamber. The instructions are in the form of a dictionary, with pairs of identifiers and corresponding numerical values. You may only change the values, that is the number immediately after the colon on each line. The comment after this number describes the purpose of the command and the limiting values. Some variables are marked #! as a warning against changing numbers that can have unpredictable effects on the stability of the climate control. This file is read by the main program every ten minutes. It is a good idea to increment the ID number (one of the dictionary items) every time you change this file, or you will quickly get in a tangle.

The second setup file is ***climch_rc.py***. It contains a sequence of instructions in the data logger language for setting up the logger at the start of a run. It also contains a list of data logger channels whose measurements are to be read and stored. You should not need to change this file during an experiment, so it is only read once, when the program starts. Note that some instructions are vital to the operation of the system and should not lightly be changed, these are noted in the file with a comment #! If you are using a separate logger to collect your experimental data, you don't need to alter this file.

> The main program file ***climchxx*** uses these two setup files and two other files which you should not edit:
> The file ***climch_defs.py*** contains functions for refining the raw data and for controlling the temperature and humidity in the chamber. You should not attempt to change these functions without a real understanding of the control and measurement process.
> The file ***serport.py*** contains code for communicating through the serial port. You must not change this.

There are backup versions of all files in the read only directory ***/backup.*** There are starting versions of the setup files in the directory ***/templates.*** Copy ***climch_cc.py*** and maybe ***climch_rc.py*** from there to the ***/home/chamber*** directory, edit them and then run the program. Keep a copy in your own home directory for reference.

To run an experiment, after making your personal versions of the files ***climch_rc.py*** and ***climch_cc.py*** (with exactly these names), you just type ***climchxx*** at the command prompt.

When you have checked that everything is behaving correctly, you should stop the program and restart it to run continuously in the background, even when you have logged off the computer. The command is: >***nohup climchxx &***

> This means '**no h**ang **up** on logout', the **&** means 'run in the background'

This program writes the collected data every ten minutes to a file called ***data.txt***.

While the program is running in the background it sends diagnostic data to a file called ***nohup.out***. This gets very large. Once you are confident that all is working well you can delete this file, after which the program will take the hint and not write to it any more.

The program has no elegant stopping mechanism. Just hit> *control-c* to kill it. If it is running in the background: use> *ps ax* to see the list of processes. Find the number, *nnnn* corresponding to climchxx. Type> *kill nnnn*

How do I know what instructions to write in climch_rc.py?
All communication with the data logger uses sentences listed and described in the data logger handbook. An examples follows shortly.

Running the program remotely
The program and the utility programs (described later) are designed for remote operation, because many experiments last a long time and need only occasional checking to ensure that all is well. The computer can be reached by ***ssh*** (**s**ecure **sh**ell), which allows you to take over the computer from anywhere in the world. If you just want to collect the data file, use ***scp*** (**s**ecure **c**o**p**y). Setting up these services on your own computer requires some specialised knowledge so ask the system manager.

Editing the setup files
The easiest way for newcomers to Linux to copy and edit files is to use 'Midnight Commander': >*mc* at the command prompt. This has a simple text editor and a simple command for copying files.

Here is a very short and simplified ***climch_cc.py*** file. Note that # starts a comment which continues to the end of the line. The \ symbol at the end of a line continues the logical line onto the next screen line.

The instructions start with some identification and control instructions, such as ID number for this run, the temperature cycle period and amplitude. These data are presented in the form of a *dictionary* with a set of *name:value* pairs. The pairs are separated by commas and the entire dictionary is enclosed in curly brackets { }. Here is a shortened version to show the layout. You will only need to alter the number immediately after the colon on each line. If you cannot guess what the description means, you probably don't need to alter the number.

```
# list of instructions

# First section is a dictionary of Control Constants
Cc = { # don't edit this line
'run_id': 1, # any integer
'elapsedTime': 0, # time into the cycle
'temperature': 20,
'vapourPeriod': 24,
'vapourHalfAmplitude': 5,
'sgWeightCorrection': 1.032, #!change only after calibration
'peltierTare': 103.4, # starting weight of water
'pidDiv': 0.3, #! you shouldn't need to alter this
'pidMult': 3.0 #! or this!
} # don't edit this line
```

The template file contains all the variable names and default values, with brief explanations. For the moment it is only necessary to explain that in this shortened example the temperature is constant at 20C, the vapour flux is swinging +/-5 grams in a 24 hour sine cycle. The other names refer to calibration constants which seldom need to be changed.

> **Some details about how entries in 'climch_cc.py' affect the control system**
> The cooling and humidity loads on the system vary greatly according to the experimental specimen and the climate regime that is imposed on it. The computer does not just switch the heater on and off according to whether the actual temperature is above or below the desired value. The chamber uses proportional control: the heating is active for a portion of each minute. This portion varies according to the temperature error. If the fractional 'on' time is proportional to the error, the temperature will oscillate about the correct value. A second correction modifies the 'on' time according to the speed of approach to the target. In addition the history of the approach to the correct value is taken into account. The new 'on' time therefore depends on the previous 'on' time, the size of the error and the rate of approach to the desired value. The various constants are defined in the file ***climch_cc.py*** You can play with these to get the smoothest climate change. Suggested starting values are listed in the comments. Generally the 'trend' constant should be about ten times the 'error' constant. Note that the heat pump uses a continuously varying voltage. It does not switch on and off because this will cause rapid death through thermal shock. For this reason you should not start and stop the program more often than is really necessary to set up and test the variables.

the other instruction file, ***climch_rc.py,*** contains a set of instructions to set up the data logger

```
# List of initiation strings for data logger
# These are run only once at program start

InitiateMeasurements = [ # don't edit this line
'syst:pres',
'conf:temp tc, k, (@301,302,318)\n',
'trig:sour imm\n',
'trig:count 1\n'
] # don't edit this line
```

Each line listed above is a single instruction in SCPI jargon. `'syst:pres\n'` resets the datalogger from whatever mixed up situation it was in. The line `'conf....'` sets two of the data logger channels to measure temperature with type K thermocouples. The last two lines define how the measurement process is started and how many repeat measurements are required at each measuring moment. Each line is an instruction, exactly as listed in the data logger user's handbook but enclosed in quotes and with a terminal \n to indicate the end of the instruction. Instructions are separated by commas.
The file ends with a list of instructions to get the data from the logger:

```
# List of data measurements
DataMeasurements = [
'data:last? 1, (@301)\n', #!
'data:last? 1, (@302)\n', #!
'data:last? 1, (@318)\n'
]
```

This list must have only channel numbers (@3nn) that have been defined at the top of the file, though there can be fewer. The order of the first few measurements is important for controlling the chamber. These protected measurements are marked with a #! and must not be changed.

This is a very simplified instruction file, just to show the structure. In practice many more initiation instructions would be sent, because the data logger is capable of a considerable amount of pre-processing, to convert the measured voltage across the strain gauge into grams of water in the tank, for example. It is important that the data returned by the data logger is the actual engineering unit, if at all possible. This saves confusion later when one wonders what correction factors to apply in a spreadsheet.

Choosing and testing the instructions with diagnostic programs

It is easy to make mistakes in the logger instructions, or misunderstand how the logger responds to instructions. There is a program ***logchat*** which allows you to send instructions from the keyboard in exactly the same format as the data logger language and receive the response, if any, exactly as the data logger formats it.

This program soon gets tedious as a way of building up a configuration for the logger. There is a program ***climch_interact*** which loads a configuration file ***climch_interact_rc.py*** before handing control over to the keyboard. It also has some time saving macros for simplifying frequently used instructions. The configuration file is similar to the beginning of the ***climchxx_rc.py*** file. You can use this program to build up the initialisation sequence for the data logger. When it is working correctly, copy this file into ***climch_rc.py*** (remember to add to the end of this instruction list the data channels to measure).

The program ***climch_safe*** closes down the chamber safely: setting the Peltier voltage to zero and then stopping the fans, the water flow and some electrical services.

> There is a program for sending a voltage directly to the heat pump (Peltier) system. This is ***pelttest***. It takes a voltage from the keyboard and shows the bit pattern that it sends to the amplifier. Finally it measures the voltage produced. You should not need to use this program.

Appendix 1. The control and measurement program

What the main program does

The program begins by sending a stream of setup instructions to the data logger. These are simply strings in the data logger language that are taken directly from the file climch_rc. You can check these instructions on the screen: they are sent quite slowly. If the data logger doesn't understand, it will beep. Then you will have to ask the datalogger what is wrong (see the datalogger handbook). If you cannot understand your mistake you may need to use a utility program to talk directly to the data logger.

The program then goes into an endless loop. This is the sequence of events:

1. Stop the fan in the chamber and raise the beam which weighs the water tank by sending 5V to the motor for 0.8 seconds (bypassing the microswitch, which is open).

2.Wait a few seconds for air movement to stop.

3. Start a logger scan.

4. Read the measured data into an array.

5. Re-start the fan and lower the beam, by sending a 1.5 second pulse of 5V to the motor, through the microswitch, which cuts of the voltage when the water tank rests on its heat exchanger.

The program now operates on the collected data in this sequence:

6. Check that the beam was up by checking that the weight is not negative.

7. Look at the chamber dewpoint and at the heat sink cooling water temperature. Close the cooling water valve if the temperature is less than two degrees over the dew point.

8. Look at the temperature of the heat sink. If it is over 25 degrees the cooling circuit has probably failed. A flag is set that causes the Peltier voltage to be set to zero in instr.10, after which the program terminates.

9. Work out the intended temperature from instructions in the *climch_cc.py* file. Calculate also the intended temperature for the next minute. These values, plus the actual temperature and the previous

actual temperature, are used to work out a time of heating (which will be negative if cooling is required). This time is later used to control the heating and cooling system.

10. Work out the intended humidity (relative humidity or water content of tank) and the next value. Compare this with the actual value and the previous actual value and compute a voltage to the Peltier heat pump (which is negative for cooling).

11. Convert this voltage into a serial digital signal and then send it to the power amplifier.

12. If the time is at a ten minute mark, store the current values in the output file 'data.txt'

13. Overwrite the array of previous values with the present values.

14. The final act in the cycle is the temperature control. The heating time has been computed, but we are already well into the minute cycle, because of the time used to measure and to compute and send the humidity control signal. The program therefore decides which is the shorter control period: with the cooling valve (or the heating coil) open or closed. The shortest period is run first so that the longer period can still be running while the program is busy with the next measurement cycle.

Note for programmers:

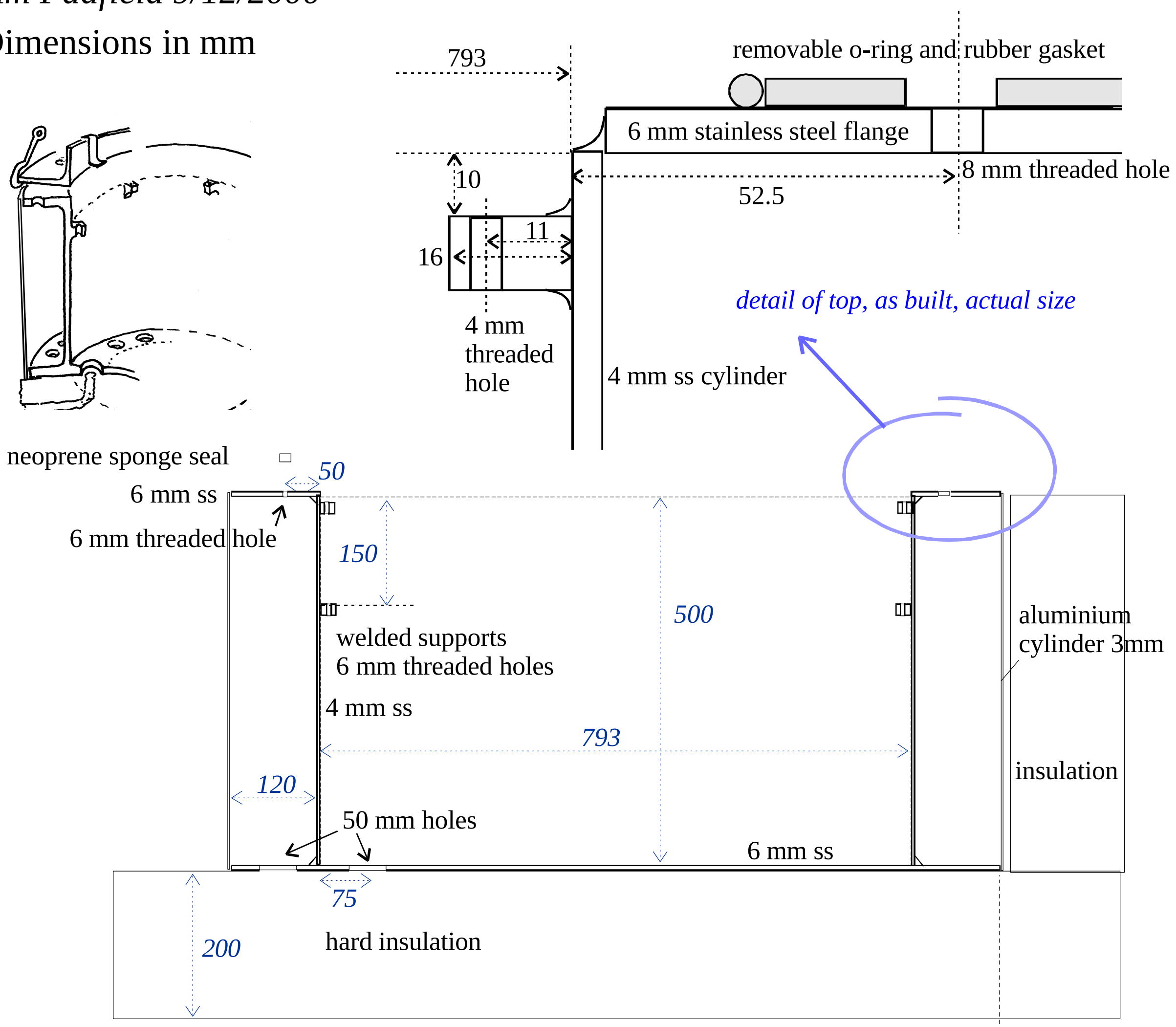
The low level communication system (unix version)

The program uses the low level (unbuffered) data stream through the serial port, The serial communications functions are read into the program on start up. Every communication event consists of sending a message and receiving the reply, reading one character at a time and breaking out of the listening loop after 5 seconds of silence from the datalogger. This complexity is necessary because HP does not support Unix connection to the data logger. The datalogger does not reply to all instructions, and therefore does not always send the newline character that usually marks the end of Unix messages.

Climate chamber

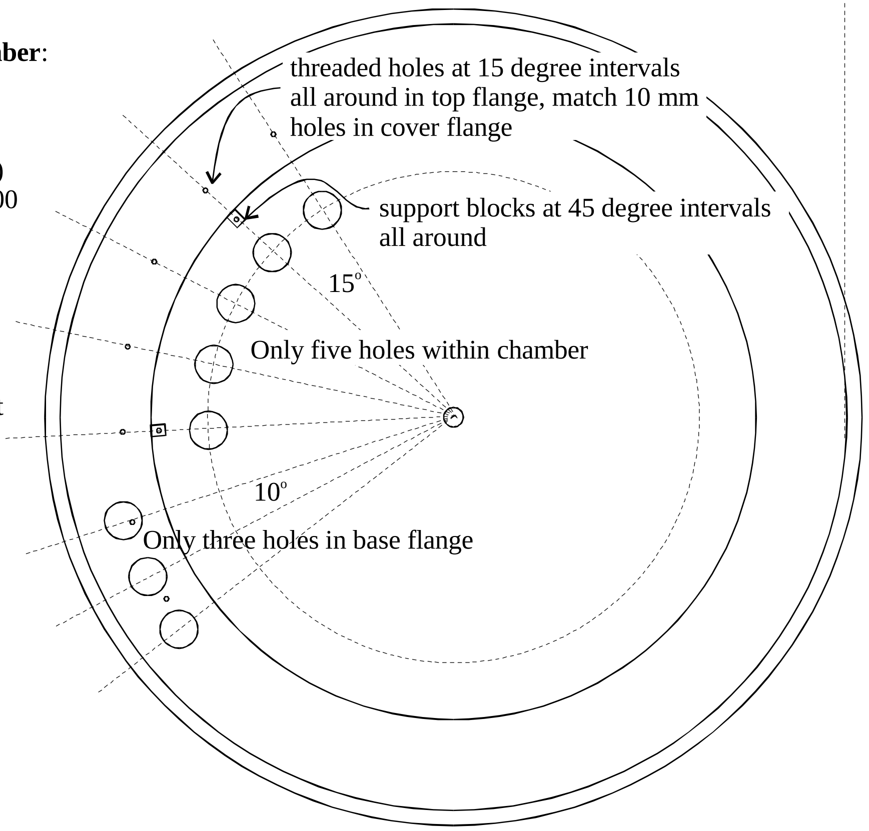
Tim Padfield 9/12/2000

Dimensions in mm



Holes within chamber:
1 cooling water in
2 cooling water out
3 Power circuits:
TEC, Fan, (heater)
4 tc and sg and pt100
signal lines
5 Air line for
dewpoint check

Holes in flange:
1. Cooling water in
2. cooling water out
3. Power to fan

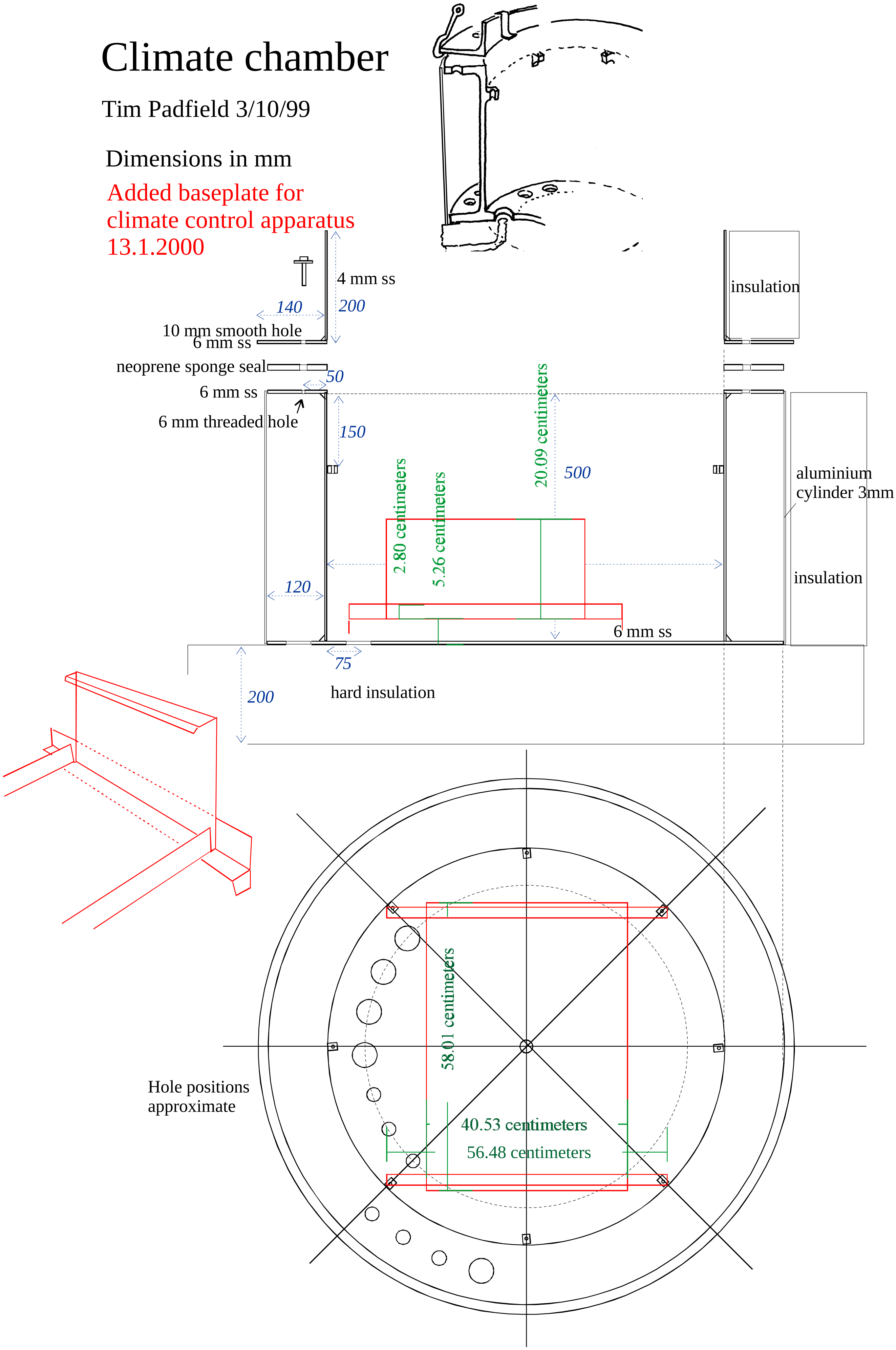

threaded holes at 15 degree intervals
all around in top flange, match 10 mm
holes in cover flange
support blocks at 45 degree intervals
all around
15°
Only five holes within chamber
10°
Only three holes in base flange

Climate chamber

Tim Padfield 3/10/99

Dimensions in mm

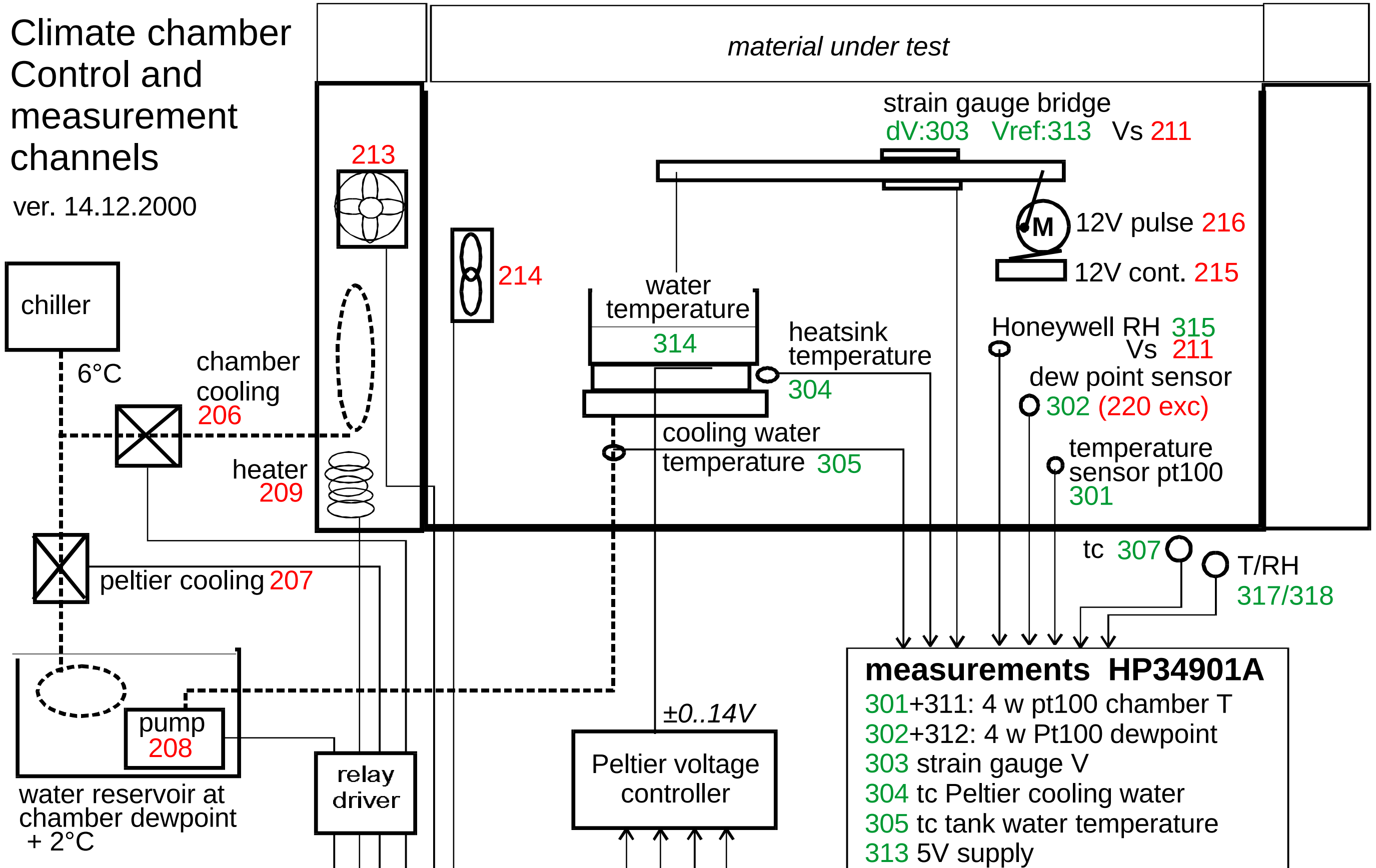
Added baseplate for climate control apparatus 13.1.2000



Hole positions approximate

Climate chamber Control and measurement channels

ver. 14.12.2000



material under test

strain gauge bridge

dV:303 Vref:313 Vs 211

12V pulse 216

12V cont. 215

213

214

water temperature

314

heatsink temperature

304

Honeywell RH 315

Vs 211

dew point sensor

302 (220 exc)

temperature sensor pt100

301

cooling water temperature 305

chiller

6°C

chamber cooling

206

heater

209

peltier cooling 207

tc 307

T/RH

317/318

pump

208

water reservoir at chamber dewpoint + 2°C

relay driver

±0..14V

Peltier voltage controller

measurements HP34901A

301+311: 4 w pt100 chamber T
302+312: 4 w Pt100 dewpoint
303 strain gauge V
304 tc Peltier cooling water
305 tc tank water temperature
313 5V supply
314 tc heat sink alarm
315 Chamber RH (hwl)
316 Peltier volts (abs)
307 tc(K) direct to logger
317.318 T & RH room
306.308 T & RH probe A
309.310 T & RH probe B
319.320 T & RH probe C

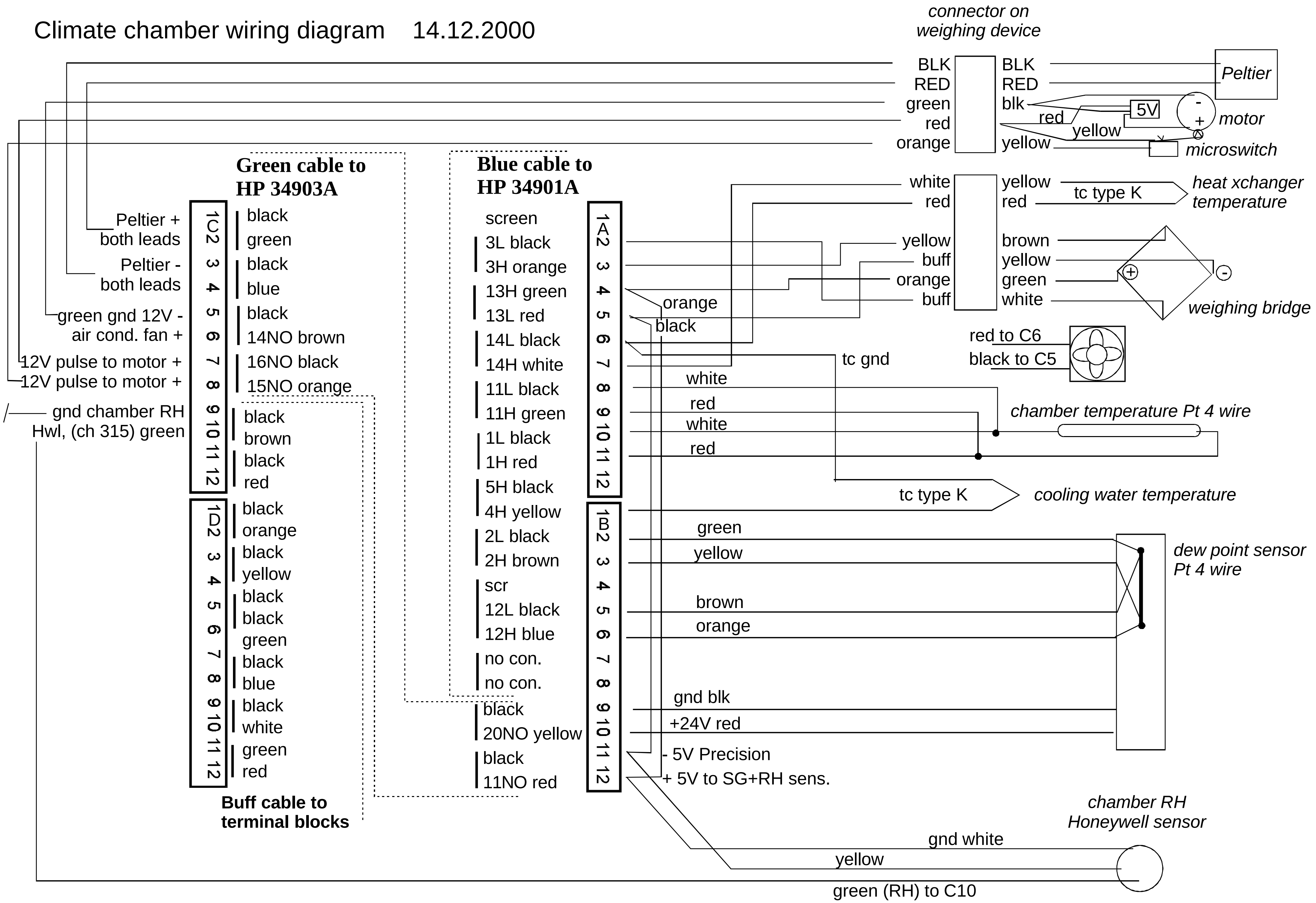
control contact closures HP34903A

202 clock
203 digit
204 latch
205 polarity

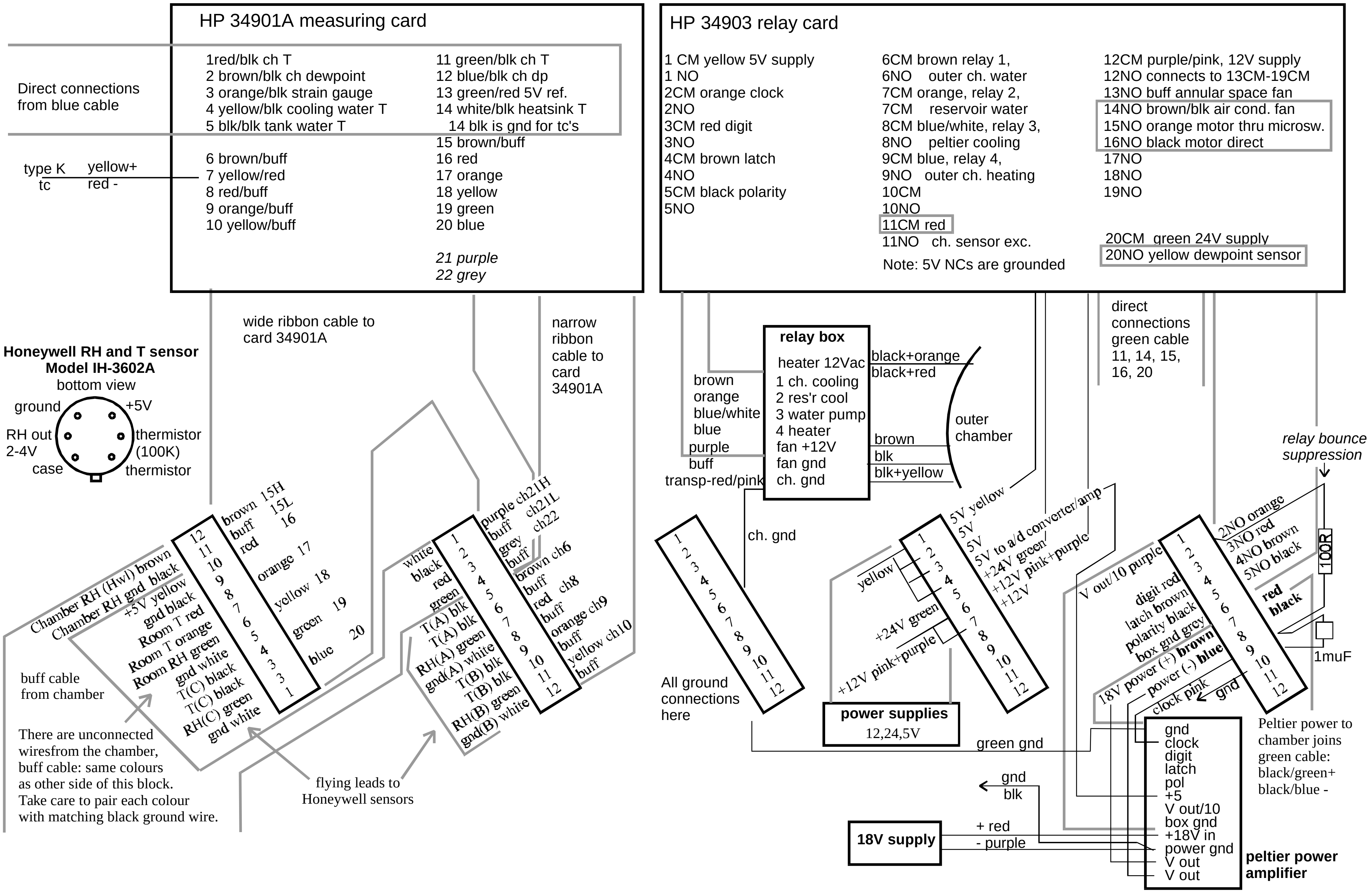
201 enable 5V to logic, RF and strain gauge
212 enable 12V to fans
220 enable 24V to dp sensor only
206 5V control chilled water, outer chamber cooling coil
207 5V control chilled water to heat exchanger reservoir
208 5V control TEC cooling water pump
209 5V control heater in outer chamber
210 unused
211 5V exc. to RH sensors
213 12V to fan in annular space
214 12V to air conditioner fan (inner chamber)
215 12V pulse to lower tank
216 12V pulse to raise tank
220 24V to dew point sensor

Digital signal cycle:
start with 204 open (latched)
repeat eight times:
202 close (set clock)
203 close(0) or open(1)
202 open (transmits to shift register)
then:
204 close (transfers register to DAC)
204 open (latches DAC)

Climate chamber wiring diagram 14.12.2000



Data logger connections ver. 14.12.2000



HP 34901A measuring card

1red/blk ch T
2 brown/blk ch dewpoint
3 orange/blk strain gauge
4 yellow/blk cooling water T
5 blk/blk tank water T
6 brown/buff
7 yellow/red
8 red/buff
9 orange/buff
10 yellow/buff

11 green/blk ch T
12 blue/blk ch dp
13 green/red 5V ref.
14 white/blk heatsink T
14 blk is gnd for tc's
15 brown/buff
16 red
17 orange
18 yellow
19 green
20 blue

21 purple
22 grey

Direct connections
from blue cable

type K tc yellow+ red -

HP 34903 relay card

1 CM yellow 5V supply
1 NO
2CM orange clock
2NO
3CM red digit
3NO
4CM brown latch
4NO
5CM black polarity
5NO

6CM brown relay 1,
6NO outer ch. water
7CM orange, relay 2,
7CM reservoir water
8CM blue/white, relay 3,
8NO peltier cooling
9CM blue, relay 4,
9NO outer ch. heating
10CM
10NO
11CM red
11NO ch. sensor exc.

Note: 5V NCs are grounded

12CM purple/pink, 12V supply
12NO connects to 13CM-19CM
13NO buff annular space fan
14NO brown/blk air cond. fan
15NO orange motor thru microsw.
16NO black motor direct
17NO
18NO
19NO

20CM green 24V supply
20NO yellow dewpoint sensor

wide ribbon cable to
card 34901A

narrow
ribbon
cable to
card
34901A

direct
connections
green cable
11, 14, 15,
16, 20

Honeywell RH and T sensor
Model IH-3602A
bottom view

buff cable
from chamber

There are unconnected
wiresfrom the chamber,
buff cable: same colours
as other side of this block.
Take care to pair each colour
with matching black ground wire.

flying leads to
Honeywell sensors

relay box

heater 12Vac
1 ch. cooling
2 res'r cool
3 water pump
4 heater
fan +12V
fan gnd
ch. gnd

All ground
connections
here

power supplies
12,24,5V

18V supply

relay bounce
suppression

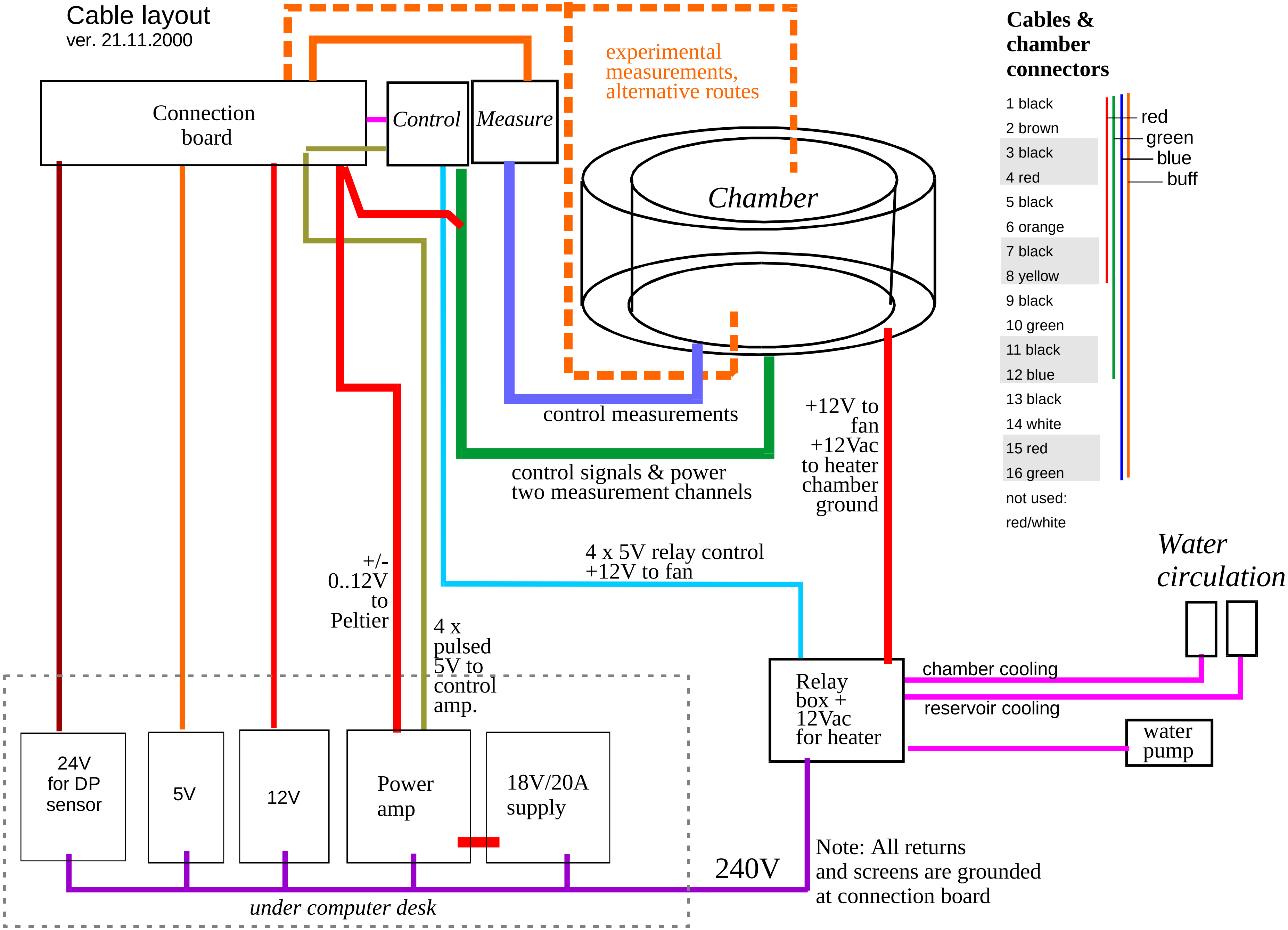
gnd
clock
digit
latch
pol
+5
V out/10
box gnd
+18V in
power gnd
V out
V out

peltier power
amplifier

Peltier power to
chamber joins
green cable:
black/green+
black/blue -

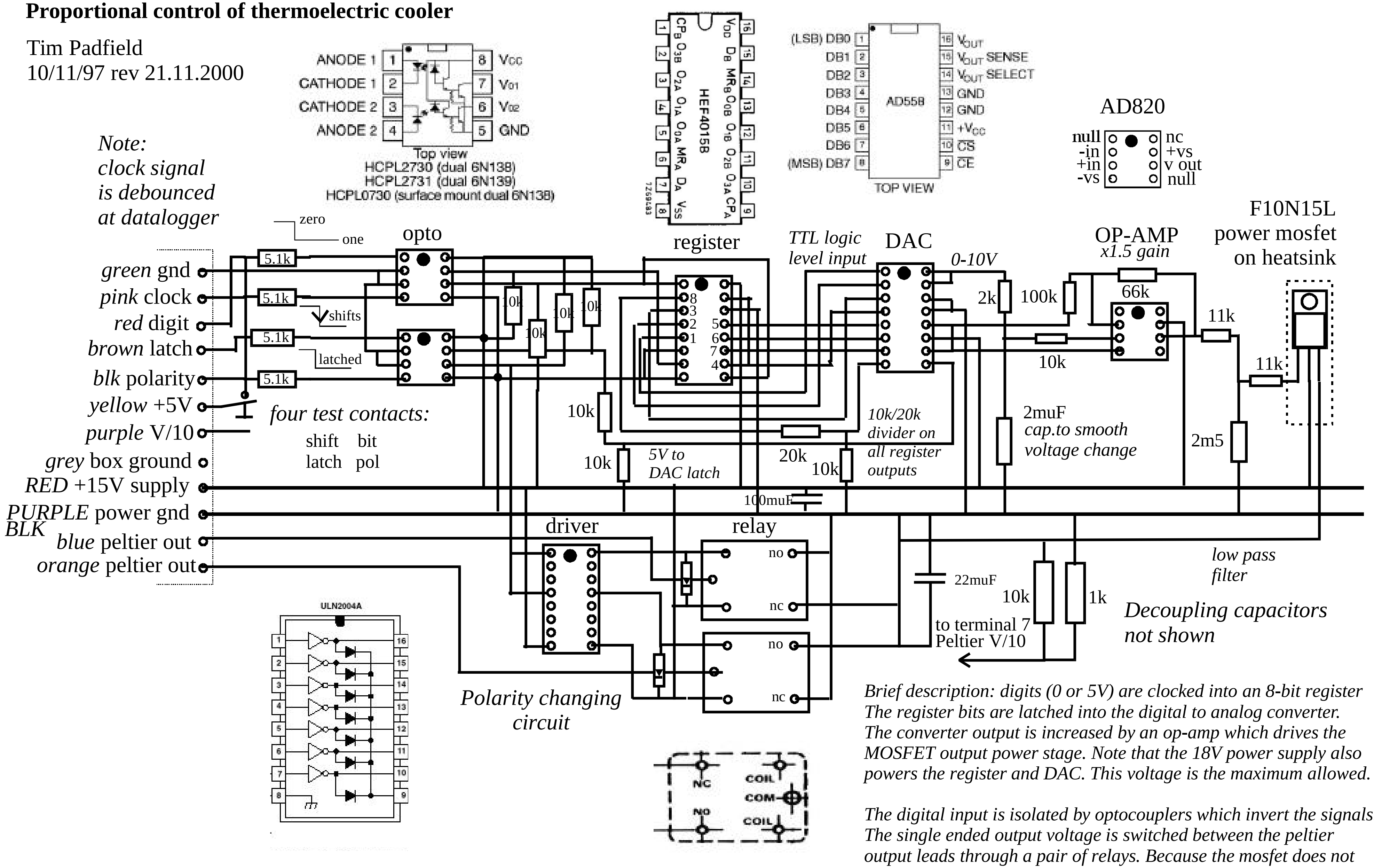
Cable layout

ver. 21.11.2000



Proportional control of thermoelectric cooler

Tim Padfield
10/11/97 rev 21.11.2000



Brief description: digits (0 or 5V) are clocked into an 8-bit register The register bits are latched into the digital to analog converter. The converter output is increased by an op-amp which drives the MOSFET output power stage. Note that the 18V power supply also powers the register and DAC. This voltage is the maximum allowed.

The digital input is isolated by optocouplers which invert the signals. The single ended output voltage is switched between the peltier output leads through a pair of relays. Because the mosfet does not conduct at low driver voltage, the digital algorithm must compensate for this dead band.