Anexos

A Scripts en MATLAB para decodificar la señal FSK

A1 data_extract_fsk.m

```
% ------------------------------------------------------------------------
% data_extract_fsk.m
%
% Master Thesis 'Exploiting Wireless Sensors'
% KTH@Wireless
% Author: Francisco Javier Sánchez Galisteo
% 12/12/2011
% ------------------------------------------------------------------------
% This script loads a file recorded with USRP and extract the data
% (in bits) from it. After executing, the variable "bits" contains the
% different bitstreams captured, each one in a different column.


clc;
close all;
clear all;

%Data
fs=2000000;
nbits=64;
t_trama=0.00371;
muestras_bit=round(t_trama*fs/nbits);
muestras_trama=muestras_bit*nbits;
sec=read_complex_binary('FILENAME.dat');

% 1: We detect when the frames start
pot=sum(sec) / length(sec);
comienza_trama=zeros(length(sec),1);
umbral=400;
v_umbral(1:length(sec))=umbral;
i=1;
while i < length(sec)-1
    if ( sec(i) >= umbral )
        comienza_trama(i)=500;
        i=i+3.75*fs;
    else
        i=i+1;
    end
end


% 2: We store the data to extract the bitstream
indices=find(comienza_trama==500);
datos=zeros(muestras_trama+1,length(indices));
for ind=1:sum(comienza_trama)/500
    datos(:,ind)=sec(indices(ind) : indices(ind)+muestras_trama);
end
```

```
% 3: We extract the bits
pot_datos(1:size(datos,2))= sum(datos(:,1:size(datos,2)))/
length(datos(:,size(datos,2)));
bits=[];


for ind2=1:length(pot_datos)
    databit=[];
    for indf=1:muestras_bit:muestras_trama
        bit=datos(indf:indf+round(muestras_bit),ind2);
        fft_s=abs(fft(bit));
        [p,l]=pkpicker(fft_s(1:round(length(fft_s)/2)),1,1);
        if l<6
            databit=[databit 0];
        else
            databit=[databit 1];
        end
    end
    bits=[bits; databit];
end



% 4: Representing the data
bits=bits';
t_muestreo=round(muestras_bit/2:muestras_bit:muestras_trama);
t=0:1/fs:length(datos)/fs -1/fs;

for ind3=1:length(pot_datos)
    figure
    plot(t,datos(:,ind3)/10000,'r');
    hold on
    stem(t_muestreo/fs,bits(:,ind3)*max(s)/10000);
    title('One frame, data extracted')
    xlabel('Samples')
    hold off
    pause;
end
```

A2 frames_dif.m

```
% ------------------------------------------------------------------
% frames_dif.m
%
% Master Thesis 'Exploiting Wireless Sensors'
% KTH@Wireless
% Author: Francisco Javier Sánchez Galisteo
% 12/12/2011
% ------------------------------------------------------------------
% This script needs to be executed after 'Data_extract_fsk.m'. The
% variable 'bits' generated in that script is analized here to detect
% the different frames recorded, in order to eliminate the frames
% repeated, since they do not add information. After executing, the
% variable 'frames_diferentes' is the number of different frames, and
% the variable 'v' contains those different frames.

clc
close all

bits_total=bits;
num_columnas=size(bits_total,2);

if (num_columnas==1)
    frames_diferentes=1;
else

    no_repetido=true;
    frames_diferentes=[];
    for i=1:num_columnas-1

        for j=i+1:num_columnas
            if bits(:,i) == bits(:,j)
                no_repetido=false;
            end
        end

        if no_repetido
            frames_diferentes=[frames_diferentes,i];
        end
        no_repetido=true;
    end
    frames_diferentes=[frames_diferentes,i+1];
end

v=[];
for indice=1:length(frames_diferentes)
    v=[v bits(:,frames_diferentes(indice))];
end
```

A3 bit_diferente.m

```
% ------------------------------------------------------------------
% bit_diferente.m
%
% Master Thesis 'Exploiting Wireless Sensors'
% KTH@Wireless
% Author: Francisco Javier Sánchez Galisteo
% 12/12/2011
% ------------------------------------------------------------------
% This script compares the different frames, stored in the variable
% 'bits', to determine which bits are equal and which ones are
% not between different transmissions. The variable 'bit_dif' contains
% the position of bits that are not equal in all the bitstreams
% analized.


clc
close all

bits_total=bits;
num_columnas=size(bits_total,1);

bit_dif=[];
for i=1:num_columnas
    if (sum(bits_total(i,:)) ~=0) & (sum(bits_total(i,:))
~=size(bits_total,2))
        bit_dif=[bit_dif; i];
    end
end
```

B Archivo “Temperaturas.xls”

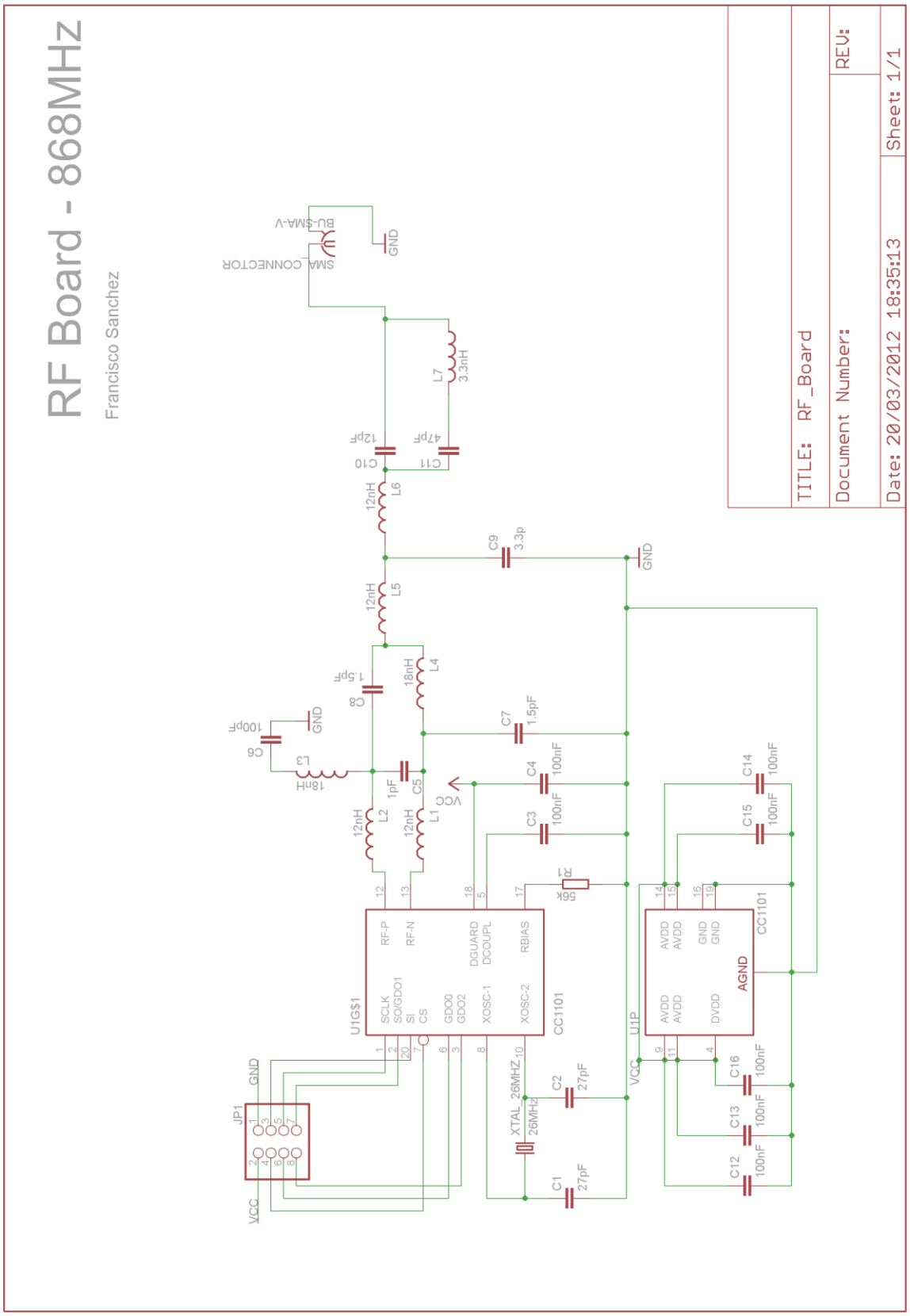
TEMP	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
-12,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	0	1	1	0	1	1	0	1	1
-11,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	1	0	1	1	0	1	0	1	0	1	0	0	1	1	0	0	1
-11,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	1	1	0	1	1	0	1	0	1	0	0	1	0	0	0	0	0	0
-11,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	1	0	1	0	1	1	0	1	0	1	0	0	0	0	1	1	0	1	0
-11,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	1	1	0	0	1	1	0	1	0	1	0	0	0	1	1	0	1	1	1
-11,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	1	1	0
-9,5	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	1	0	0	0	1	1	0	1	0	1	0	1	0	0	0	1	0	1	1
-9,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	0	1	1	0	0	1	1	0	1	0	1	0	0	1	0	1	0	0	1	0
-9,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0	1	0	0	1	0	1	1	0	1	0	1	0	1	1	0	0	1	0	1	1
-8,7	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	1	0	0	1	0	0	1	1	0	1	0	1	0	1	0	1	1	1	1	1	1
-8,5	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	1	0	1	0	0	0	1	1	0	1	0	1	0	1	1	1	0	0	1	0	1
-8,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	0	1	1	0	0	0	1	1	0	0	0	0	1	1	0	1	0	1	0	0	1	0	1	0	0	0	1
0,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	1	0	1	0	1	0	0	1	1	0	1	0	0	0
0,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	1	0	1	0	1	0	1	1	0	1	1	1	0	0
1,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	1	1	1	0	0	0	1
1,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	1	1	0	0	1	1	0	1	0	1	0	1	1	0	1	1	1	1	1
2,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	0	0	0	1	1
14,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	0	1	0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	0	0	0	1	1	0	0	0	1
15,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	0	1	0	1	0	1	0	1	0	1	0	0	0	1	1	0	1	0	1	0	1	1	0	0	1	0	0	1
23,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	0	1	1	0	1	0	0	0	1	1	0	1	0	1	0	1	1	1	0	0	0	1	1
25,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	0	0	0	0	1	1	0	1	0	1	0	0	0	1	1	0	1	0	1
25,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	0	0	0	0	1
25,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	0	1	0	0	1	1	0	1	0	1	0	1	1	1	0	1	1	0	0
25,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	0	1	1	0	1	1	0	1	0	1	0	0	0	0	1	1	0	0	0
25,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	1	0	0	0	1	1	0	1	0	1	0	1	0	1	1	0	1	1	0
25,5	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	0	1	0	0	0	0	1	0
25,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	0	1	1	0	1	0	1	0	0	1	1	0	1	1	1	1
25,7	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	0	1	1	1	0	1	1	0	1	0	1	0	1	0	0	1	1	0	1	1
25,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	1	0
25,9	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	0	1	1	0	0	1	0	1	1	0	1	0	1	0	1	1	1	1	0	1	1	0
26,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0	0	0	0	1	1	1
26,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	0	0	1	0	1	1	0	1	0	1	0	0	1	1	1	0	0	1	1
26,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	0	1	0	0	1	1	0	1	0	1	0	0	1	0	1	1	1	1	0
26,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	0	1	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0
26,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	1	0	0
26,5	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	0	1	1	0	1	0	1	0	1	1	1	1	0	0	0	0
26,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	0	1	0	1	1	0	1	1	1	0	1
26,7	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	0	1	1	0	1	0	1	0	0	0	1	0	1	0	0	1
26,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	1	0	0	0	0	1	1	0	1	0	1	0	1	0	1	1	0	0	0	0
26,9	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	0	1	1	0	1	0	0	1	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0	0
29,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1	0	1	0	1	0	0	1	1	1	1	1
30,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	0
31,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	1	0	0	0	1	0	1	1	0	1	0	1	0	0	1	1	1	0	1	1	1
31,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	1	0	0	1	1	0	1	1	0	1	0	1	0	1	0	1	0	1	1	1	0
31,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	1	0	1	1	0	0	1	1	0	1	0	1	0	1	1	0	1	1	0	0	1
31,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	1	1	0	0	0	0	1	1	0	1	0	1	0	1	0	1	1	0	1	0	0
31,9	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	0	1	1	0	0	1	0	1	1	0	1	0	1	0	0	1	0	0	0	0	0	0
32,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	0	0	0	0	1	0	1	1	0	1	0	1	0	1	1	0	0	0	1	0	1
32,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	0	0	0	1	1	0	1	1	0	1	0	1	0	0	0	0	1	1	1	0	0
32,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	0	0	1	1	0	0	1	1	0	1	0	1	0	0	1	1	0	1	0	1	1
32,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	0	1	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	1	1	0
32,9	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	0	1	0	0	1	0	1	1	0	1	0	1	0	1	1	1	1	0	0	1	0
33,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	0	0	0	0	0	1	1	0	1	0	1	0	0	1	0	1	1	1	1	1
33,1	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	0	0	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	1
33,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	0	0	1	0	0	1	1	0	1	0	1	0	1	0	0	0	0	1	1	0
33,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	1	0	0	0	1	1	1	0	0	1	1	0	1	0	0	0	1	1	0	1	0	1	0	1	1	0	1	1	1	0	0
46,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	1	0	1	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0	1	0	1	1	0	0
46,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	1	1	0	1	0	1	0	0	1	0	0	0	1	1	0
46,7	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	0	0	1	1	0	0	1	1	1	0	1	1	0	1	0	1	0	0	1	1	0	1	0	1	1
47,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	0	0	1	1	1	0	0	0	0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	1
50,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	1	0	1	0	1	0	0	1	0	1	0	1	0	1
50,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	1	1	0	1	0	1	0	1	1	1	0	0	0	0	1
51,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	0	0	0	0	1	1	0	1	0	1	0	1	0	1	1	1	0	0	0
51,3	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	0	1	1	0	1	1	0	1	0	1	0	1	0	0	1	0	1	0	1
51,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	1	1	0	0	1	1	0	1	0	1	0	1	1	1	0	0	0	1	0
51,9	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	1	0	0	1	0	1	1	0	1	0	1	0	0	1	1	1	1	0	1	1
52,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	0	1	0	0	1	1	0	1	0	1	0	1	1	0	1	0	0	1	1
52,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	1	0	1	0	1	0	1	0	0	0	1	0	0	1
52,7	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	0	0	1	1	1	0	1	1	0	1	0	1	0	1	0	1	0	0	1	0	0
53,0	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	1	0	0	0	0	0	1	1	0	1	0	1	0	0	1	1	0	0	1	0	0
53,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	1	0	0	1	0	0	1	1	0	1	0	1	0	1	0	1	1	1	1	0	1
53,5	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	1	0	1	0	1	0	1	1	0	1	0	1	0	0	0	0	1	0	0	1	1
53,7	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	1	0	1	1	1	0	1	1	0	1	0	1	0	1	1	0	0	1	0	1	0
53,9	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	0	1	1	1	0	0	1	0	1	1	0	1	0	1	0	1	0	1	0	0	1	1	1
54,2	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	0	1	0	0	1	1	0	1	0	1	0	1	0	0	0	0	1	1	0
54,4	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	1	0	0	0	1	1	0	1	0	1	0	1	1	0	1	1	1	0	0
54,6	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	1	0	0	0	1	1	0	0	1	1	0	1	0	1	0	0	0	0	0	0	1	0	1
54,8	1	0	1	0	1	0	1	0	0	0	1	0	1	1	0	1	1	1	0	1	0	1	0	0	1	0	0	1	1	1	0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0	0	0	0	1	1	0	1	0	1	0	0	1	1	0	1	0	0	0

C PCB receptor RF

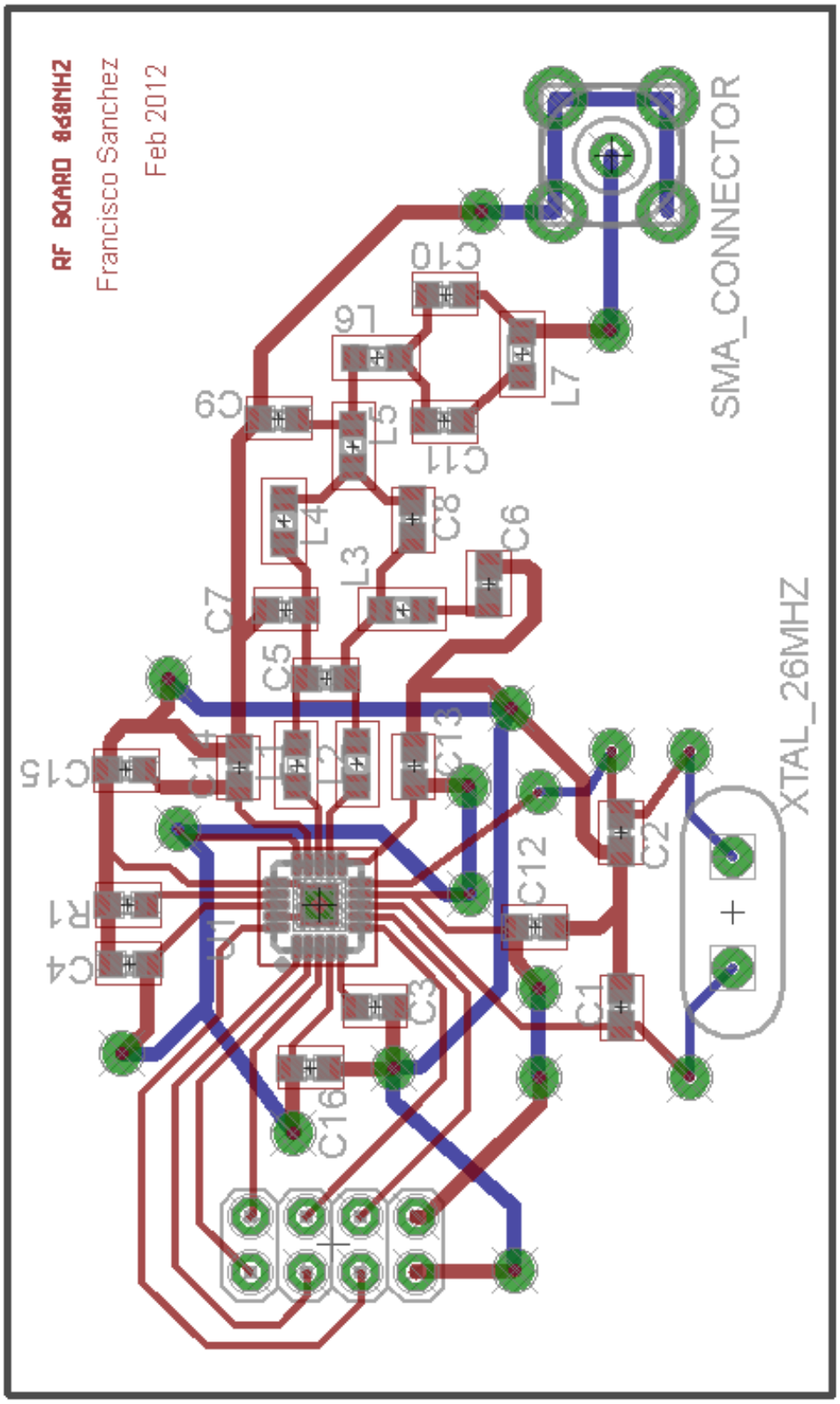
C1 Lista de componentes

PART	TYPE	Digikey part	Qty	Value
C1,C2	CAPACITOR 0603 (1608 metric)	445-1274-1-ND	2	27pF
C3,C4,C12,C13,C14,C15,C16	CAPACITOR 0603 (1608 metric)	399-1095-1-ND	7	100nF
C5	CAPACITOR 0603 (1608 metric)	445-5009-1-ND	1	1pF
C6	CAPACITOR 0603 (1608 metric)	445-1281-1-ND	1	100pF
C7,C8	CAPACITOR 0603 (1608 metric)	445-5013-1-ND	2	1.5pF
C9	CAPACITOR 0603 (1608 metric)	445-5025-1-ND	1	3.3pF
C10	CAPACITOR 0603 (1608 metric)	445-1270-1-ND	1	12pF
C11	CAPACITOR 0603 (1608 metric)	445-1277-1-ND	1	47pF
XTAL_26MHZ	CRYSTAL 26MHz (HC49/US)	887-1029-1-ND	1	
SMA_CONNECTOR	SOCKET, SMA, STRAIGHT PCB		1	
R1	RESISTOR 0603	RMCF0603FT56K0CT-ND	1	56k
L1,L2,L5,L6	INDUCTOR 0805 (metric 2012)	535-10490-1-ND	4	12nH
L3,L4	INDUCTOR 0805 (metric 2012)	535-10492-1-ND	2	18nH
L7	INDUCTOR 0805 (metric 2012)	535-10487-1-ND	1	3.3nH
U1	CC1101	296-21981-1-ND	1	
JP1	CONNECTOR 2x4 FEMALE		1	

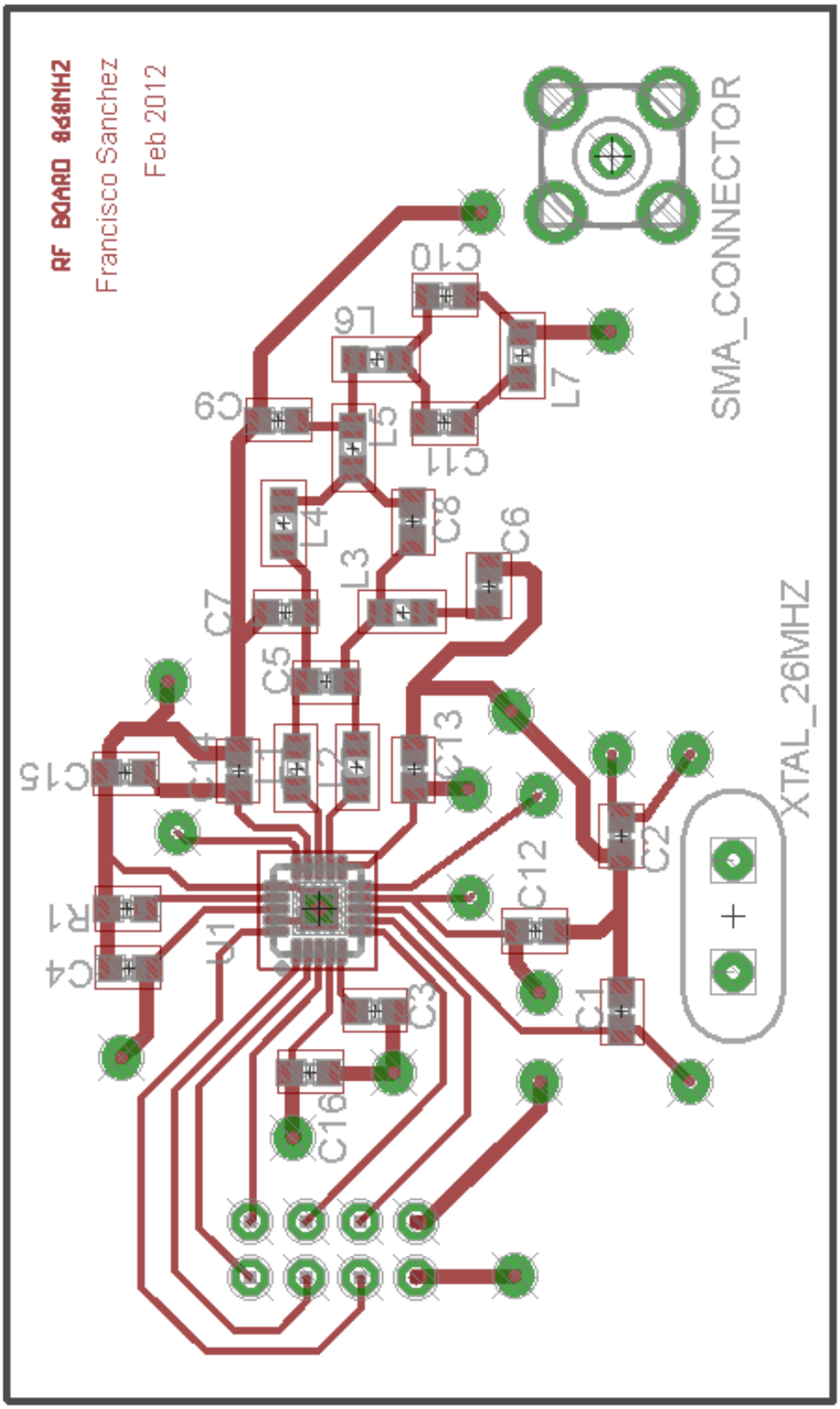
C2 Esquemático



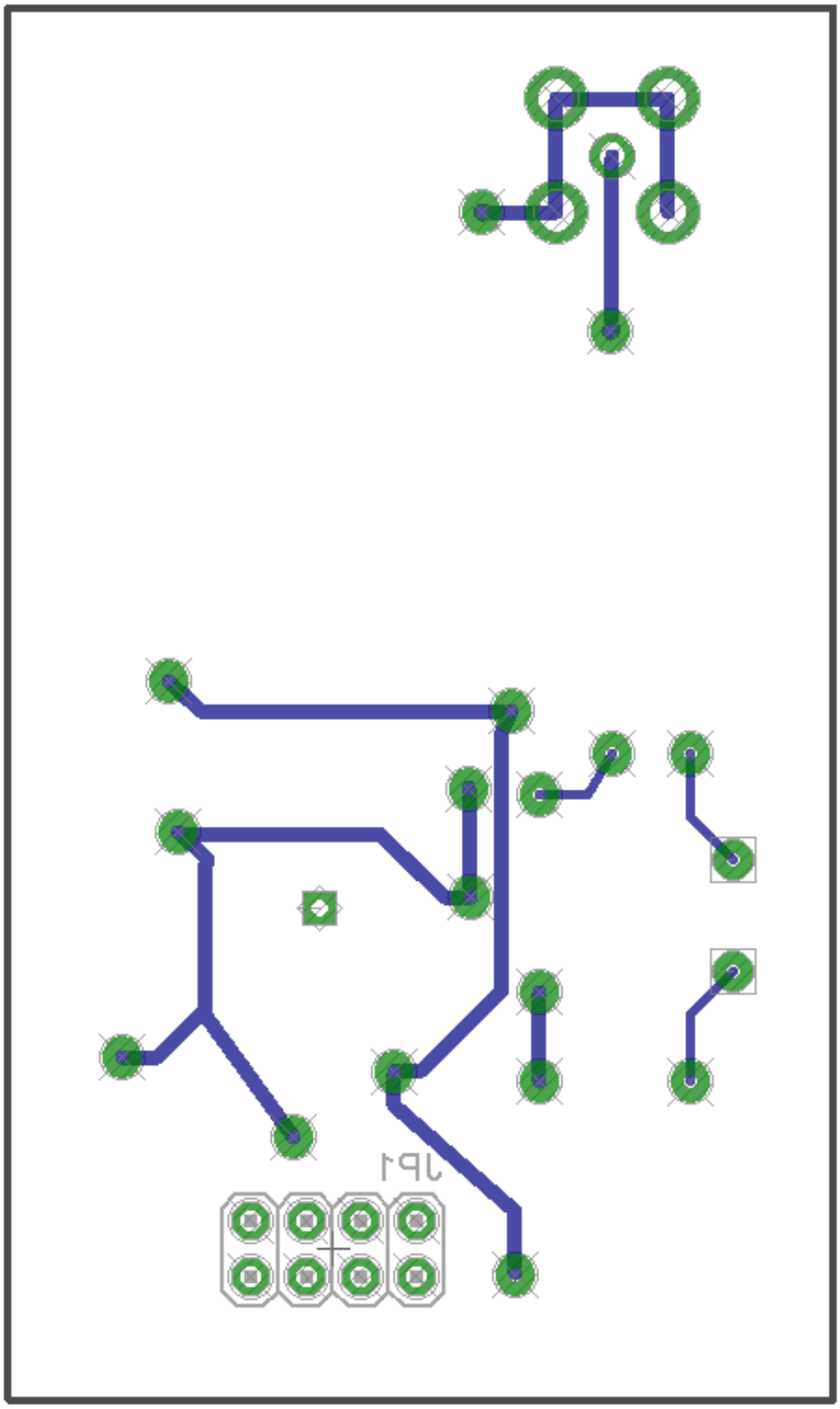
C3 Layout


RF BOARD 868MHZ
Francisco Sanchez
Feb 2012
SMA_CONNECTOR
XTAL_26MHZ
C10
L6
C9
L5
L7
C11
L4
C8
C6
L3
C7
C5
C15
C14
C13
C12
C2
R1
C4
C3
C1
C16

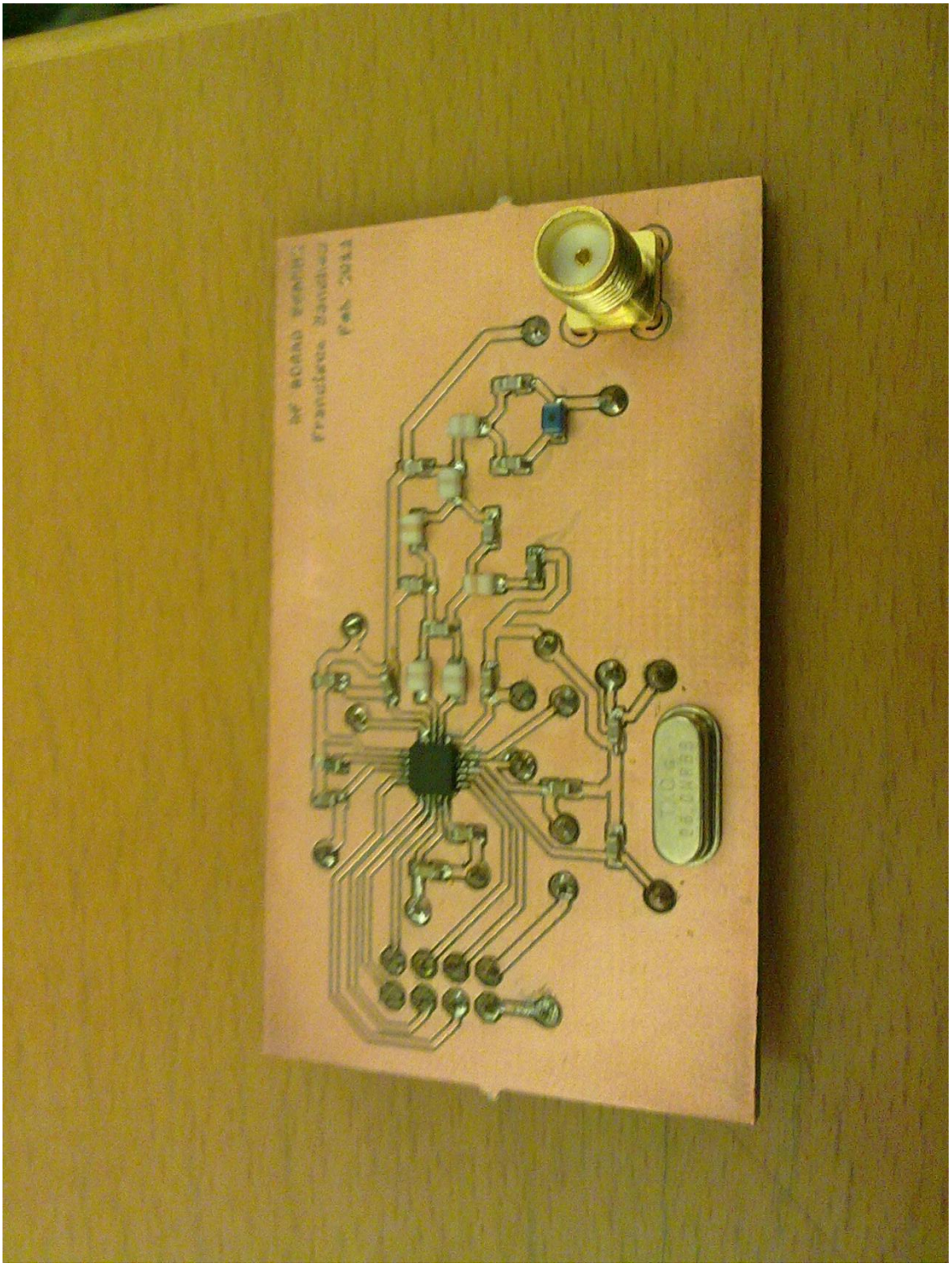
C4 Capa superior



C5 Capa inferior



C6 Imagen final



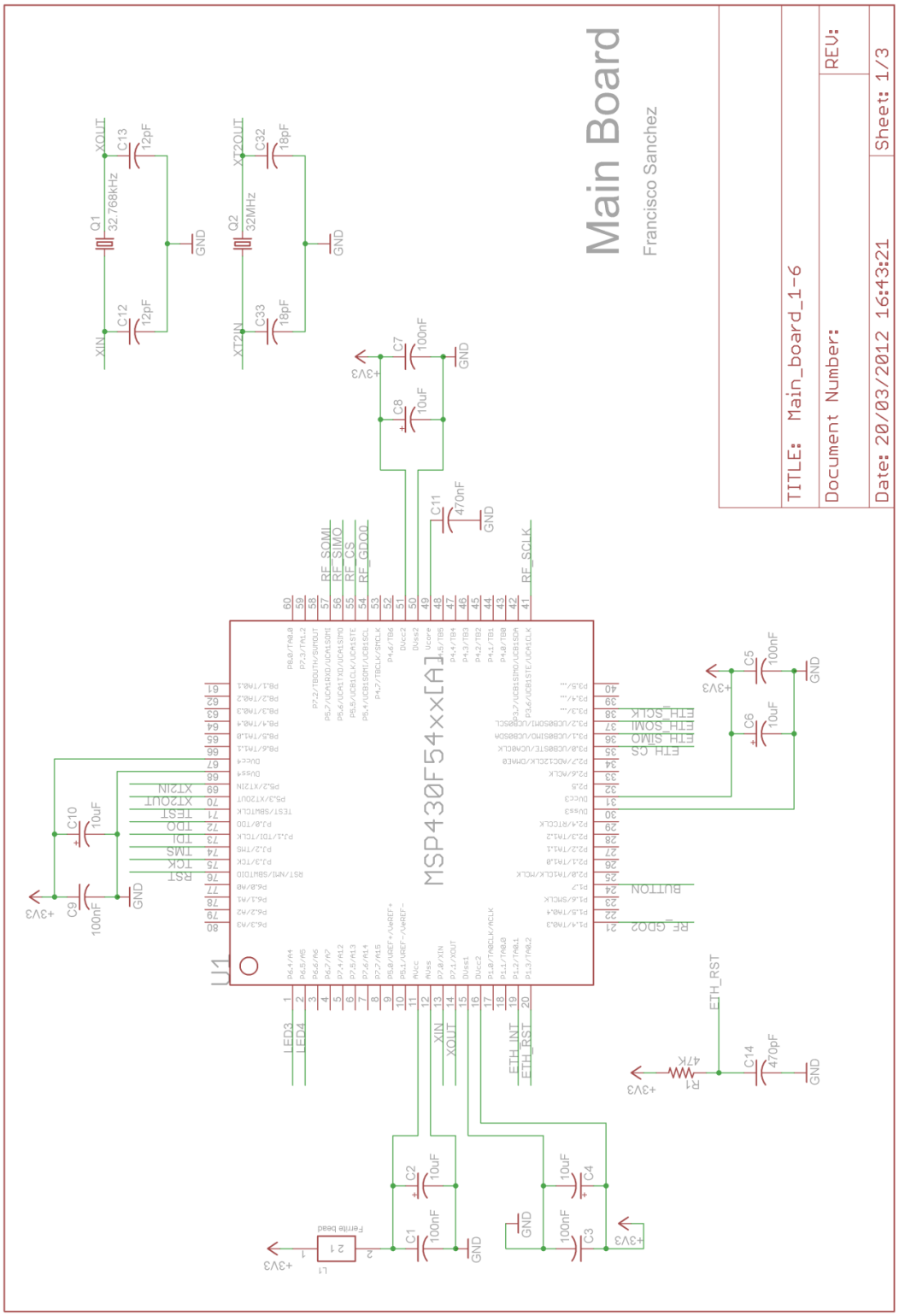
D PCB principal

D1 Lista de componentes

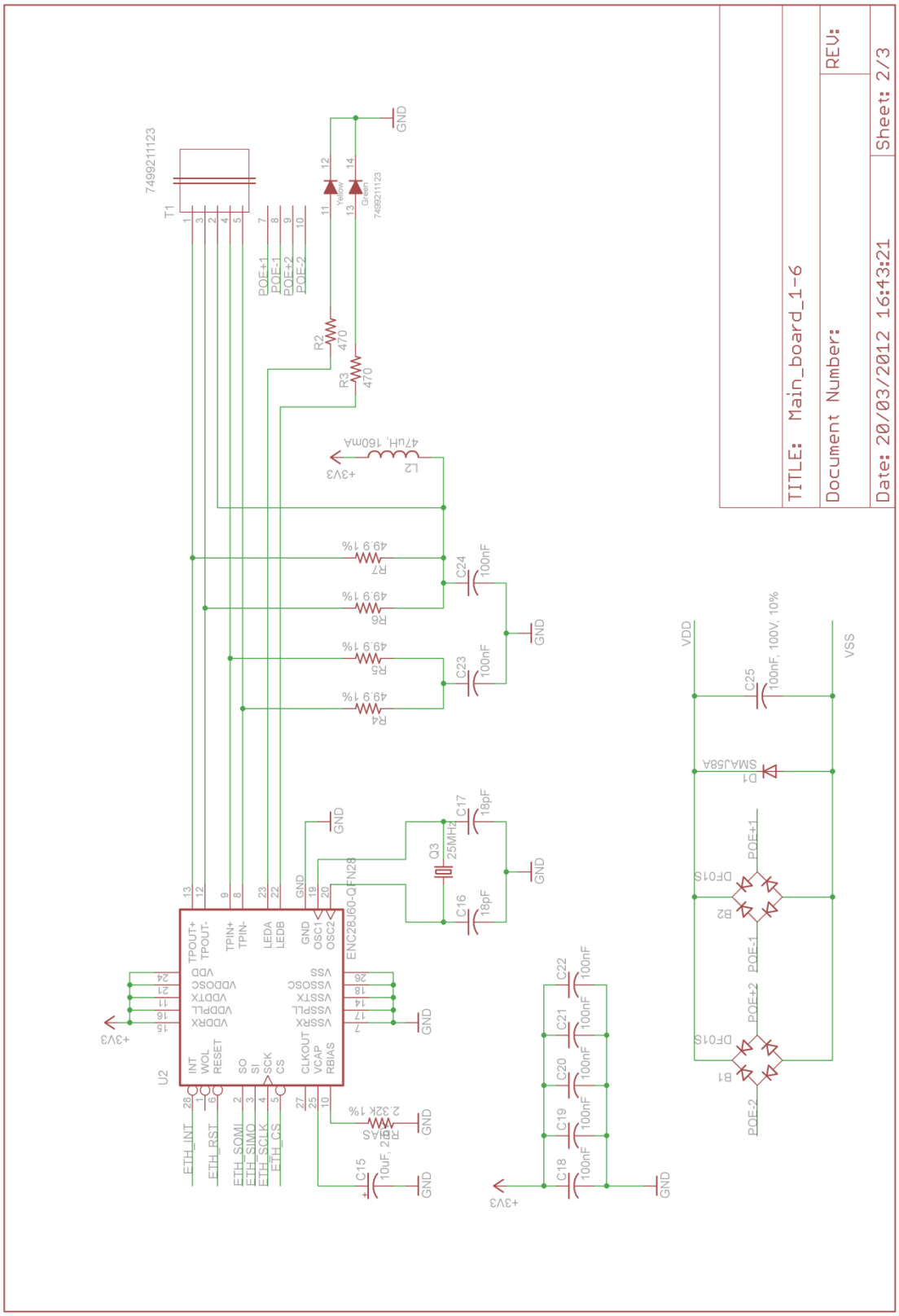
PART	TYPE	Digikey part	Qty	Value
B1,B2	BRIDGE RECTIFIER DF01S	DF01STR-ND	2	
C1,C3,C5,C7,C9,C18,C19,C20,C21, C22,C23, C24, C27,C28	CAPACITOR 0603 (1608 metric)	399-1095-1-ND	14	100nF
C11	CAPACITOR 0603 (1608 metric)	445-3454-2-ND	1	470nF
C12,C13	CAPACITOR 0603 (1608 metric)	445-1270-2-ND	2	12pF
C14	CAPACITOR 0603 (1608 metric)	445-1289-1-ND	1	470pF
C16,C17,C32,C33	CAPACITOR 0603 (1608 metric)	445-1272-2-ND	4	18pF
C2,C4,C6,C8,C10,C15	CAPACITOR 1206 (3216 metric)	493-2351-1-ND	6	10uF
C25	CAPACITOR 0603 (1608 metric)	445-5202-1-ND	1	100nF
C26	CAPACITOR THROUGH HOLE, 5-10,5	P5313-ND	1	100uF
C29	CAPACITOR THROUGH HOLE, 3,5-8	P5193-ND	1	100uF
C30	CAPACITOR THROUGH HOLE, 3,5-8	493-1702-ND	1	470uF
C31	CAPACITOR 1210 (3528 metric)	493-4090-1-ND	1	100uF
D1	DIODE SMAJ58A (DO214A)	SMAJ58ALFTR-ND	1	
D2,D3,D5	DIODE 11DQ09-ND (DO41-10)	11DQ09-ND	3	
D4	DIODE ZENER 60V (SOD123)	MMSZ5264BT1GOSTR-ND	1	
JP1,JP2	CONNECTOR 1x3 MALE		2	
JP3	CONNECTOR 2X4 MALE		1	
JP4	CONNECTOR 2X7 MALE		1	
L1	FERRITE BEAD (1806)	240-2541-1-ND	1	
L2	INDUCTOR 0805 (metric 2012)	587-2068-1-ND	1	47uH
L3	INDUCTOR PULSE ENGINEERING PE53120	553-1588-5-ND	1	1000uH
L4	INDUCTOR 0805 (metric 2012)	587-2046-1-ND	1	22uH

LED1	LED, SMD 1206, RED	160-1457-1-ND	1	
LED2,LED3	LED, SMD 1206, GREEN	160-1456-1-ND	2	
LED4	LED, SMD 1206, YELLOW	160-1458-1-ND	1	
Q1	CRYSTAL 32,768 KHz (TC26H)	300-8303-ND	1	
Q2	CRYSTAL 32MHz (HC49/US)	300-8518-ND	1	
Q3	CRYSTAL 25MHz (HC49/US)	300-8513-ND	1	
R1,R13,R15	RESISTOR 0603	RMCF0603FT47K0CT-ND	3	47k
R10,R11	RESISTOR 0603	P330HCT-ND	2	330
R12,R14	RESISTOR 0603	RMCF0603FT10K0CT-ND	3	10k
R2,R3,R8	RESISTOR 0603	P470HCT-ND	3	470
R4,R5,R6,R7	RESISTOR 0603	P49.9HTR-ND	4	49.9
RBIAS	RESISTOR 0603	P2.32KHTR-ND	1	2.32k
RCLASS	RESISTOR 0805	P953CTR-ND	1	953
RDET	RESISTOR 0603	P24.9KHTR-ND	1	24.9k
RLIM	RESISTOR 0603	P442KHTR-ND	1	442k
RPULLUP	RESISTOR 0603	P24KGTR-ND	1	24k
S1, S2	PUSH BUTTONS DTSM6	679-2383-2-ND	2	
T1	RJ45 JACK 7499211123	732-1862-ND	1	
U1	MSP430F5437a	MSP430F5437AIPN-ND	1	
U2	ENC28J60	ENC28J60-I/ML-ND	1	
U3	TPS2375	296-17061-ND	1	
U4	TL2575HV-33	296-21838-1-ND	1	
X1	CONNECTOR DC INPUT		1	

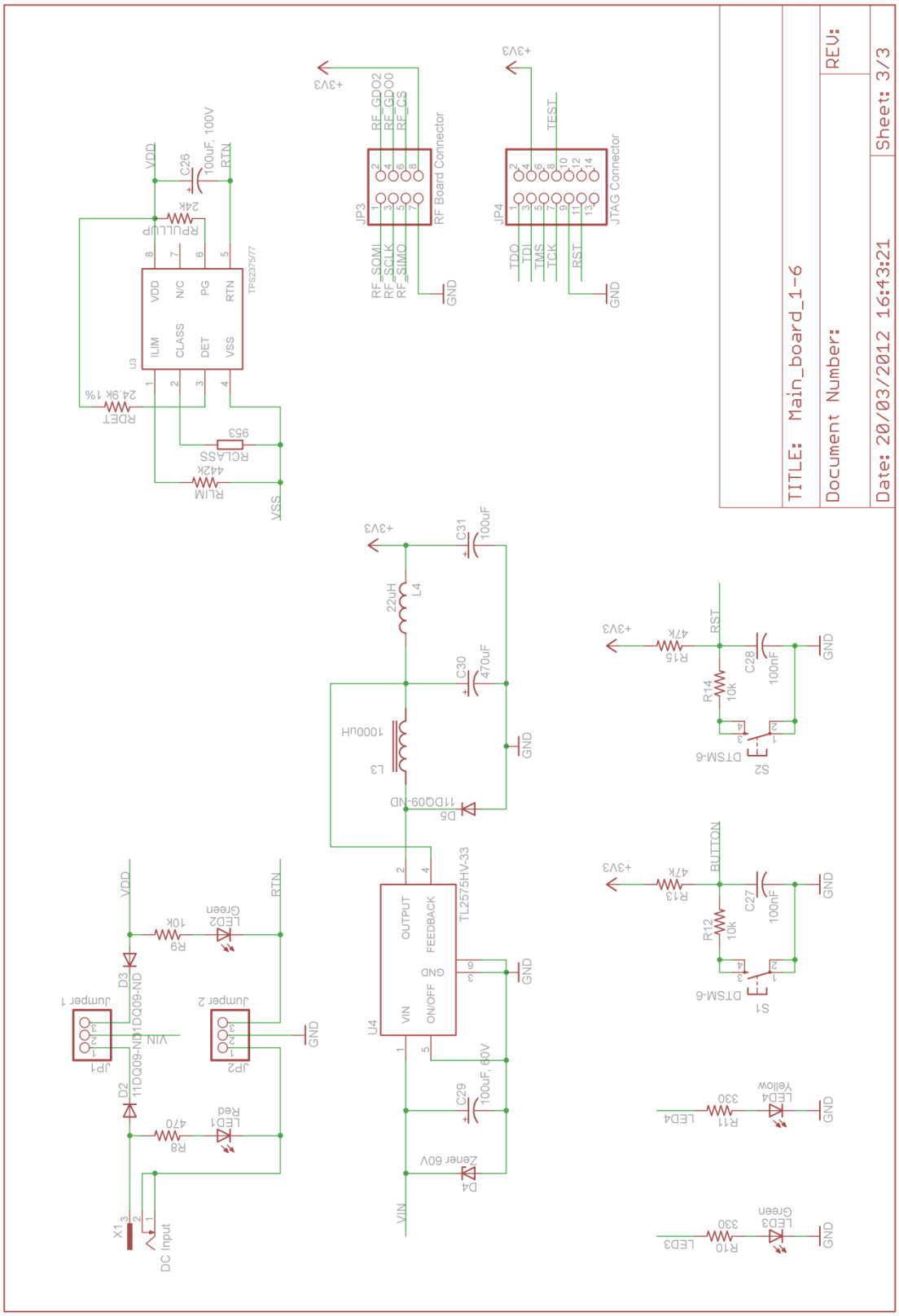
D2 Esquemático (1/3)



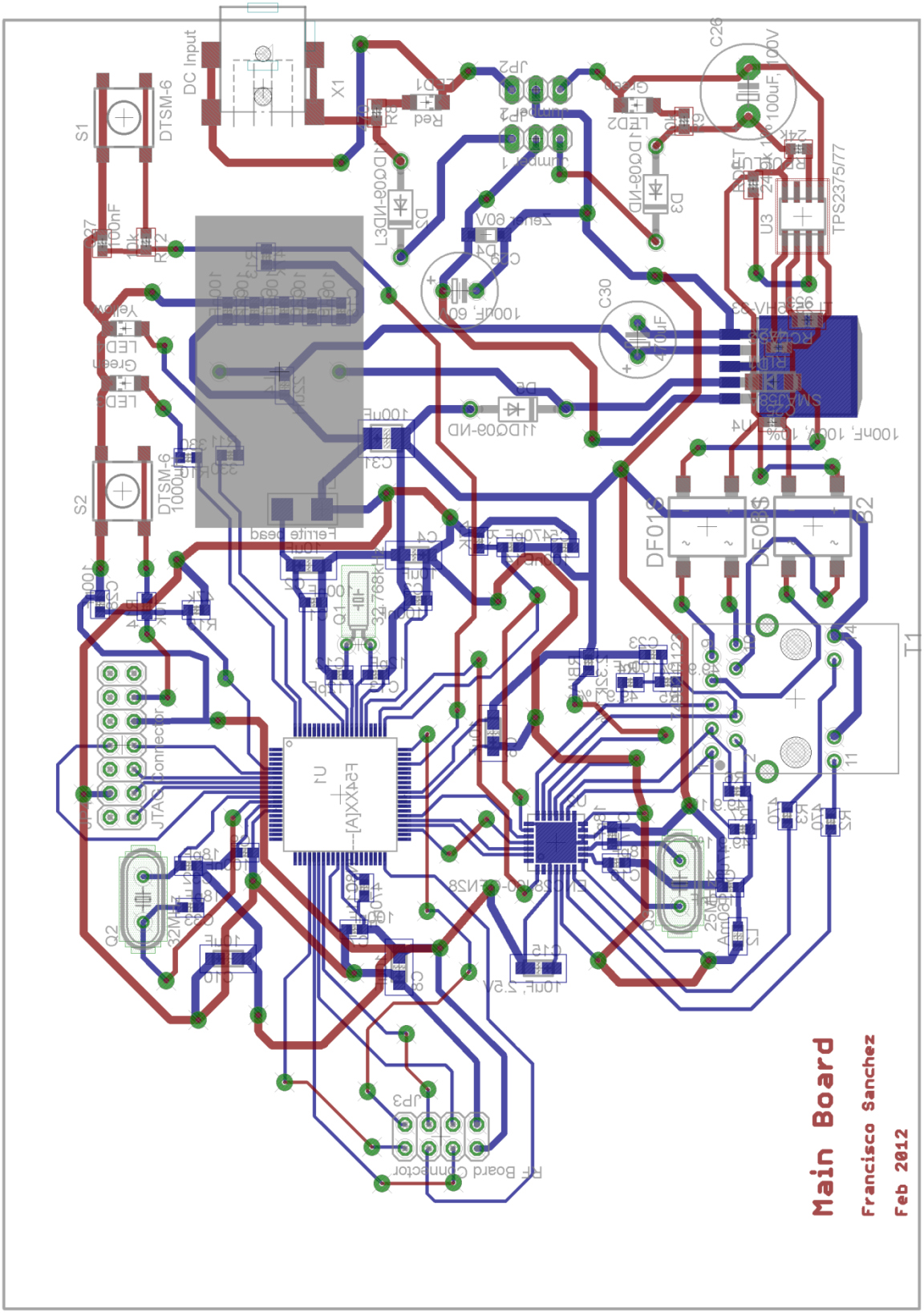
D3 Esquemático (2/3)



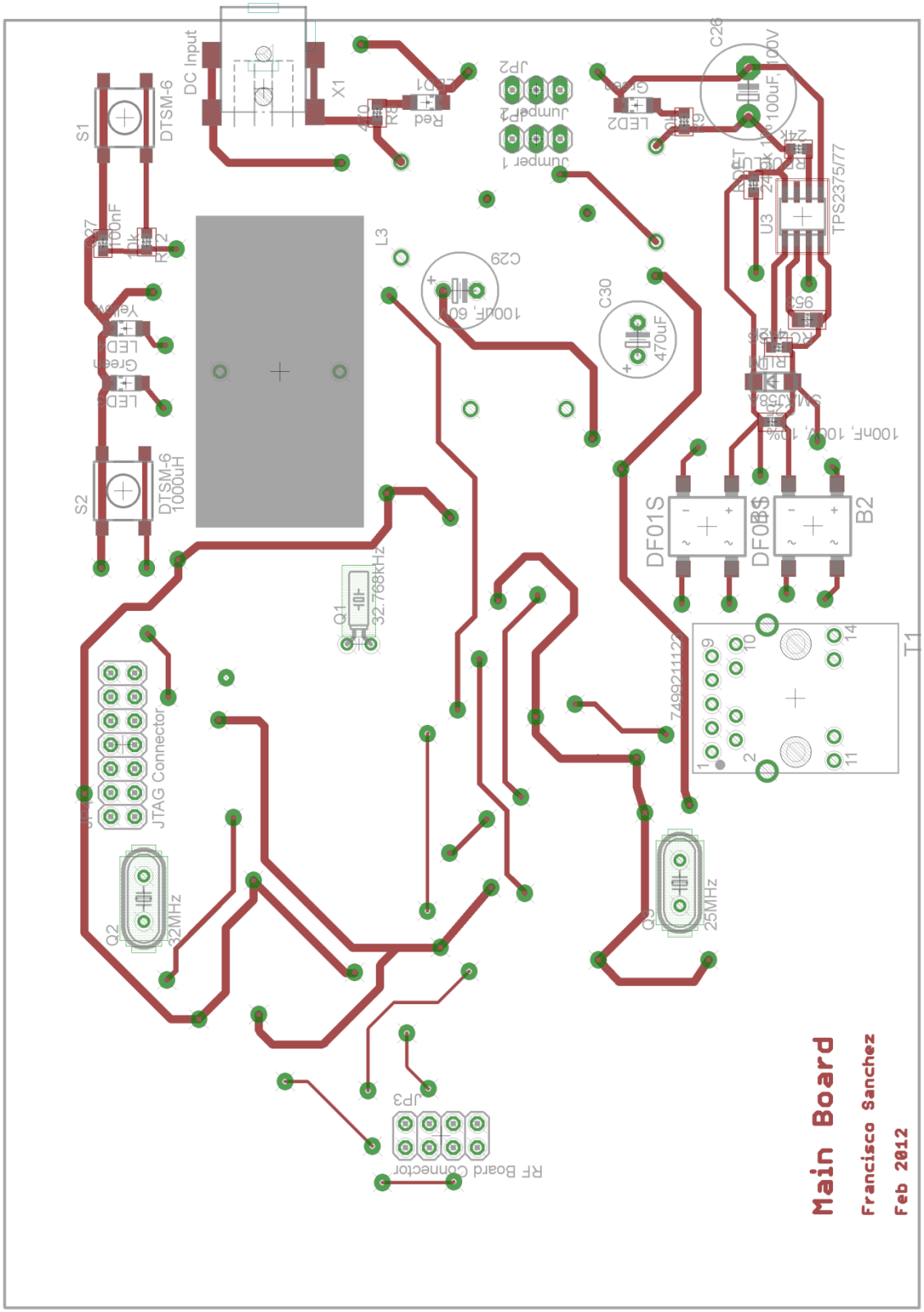
D4 Esquemático (3/3)



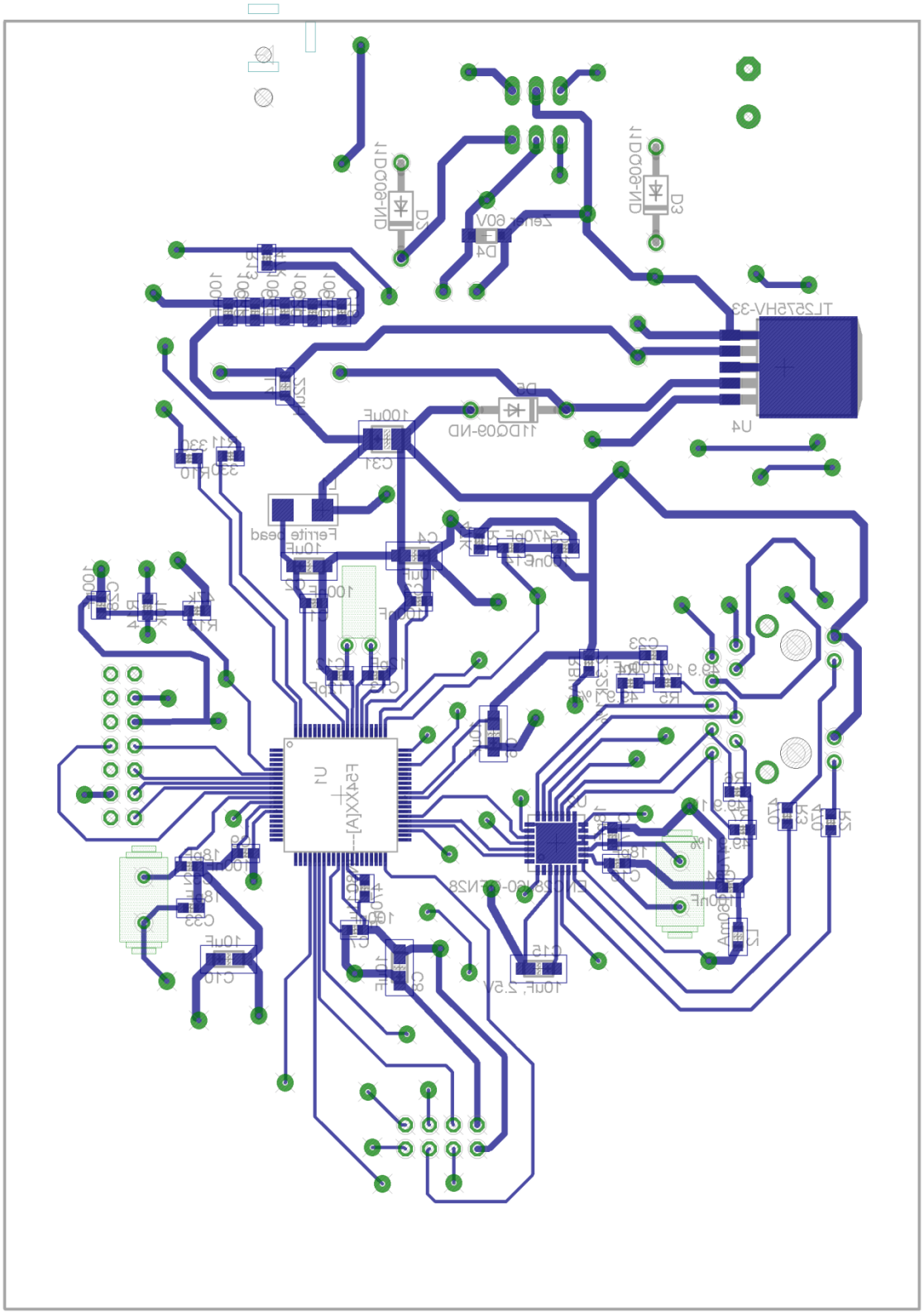
D5 Layout



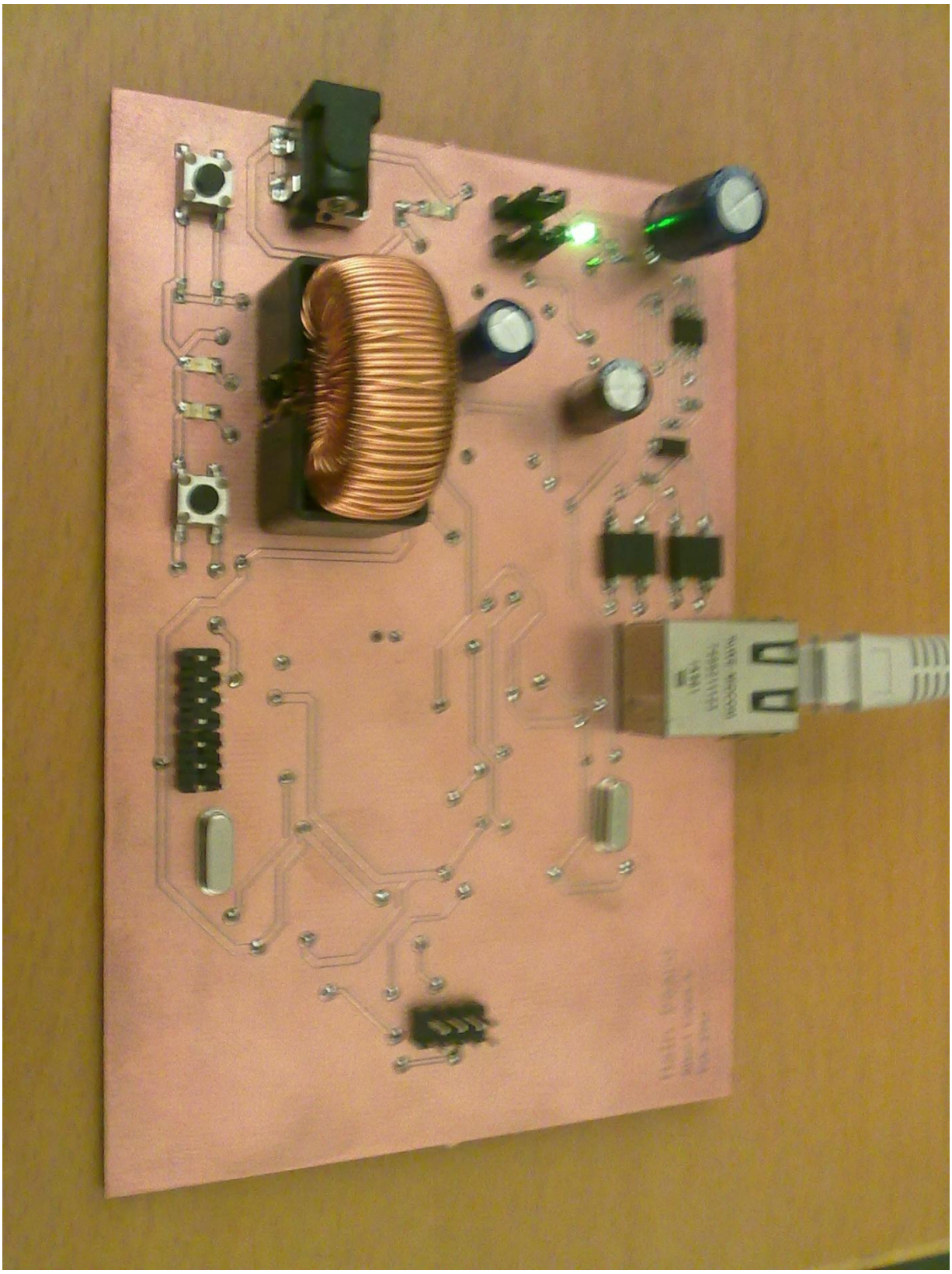
D6 Capa superior



D7 Capa inferior



D8 Imagen final



E Software para test de placa

E1 Main.c

```
//*******************************************************************
//        MSP430F5437a
//        Francisco Sanchez
//        Mar 2012
//        Built with Code Composer Studio v4
//*******************************************************************************************

#include <msp430.h>
#include "config.h"
static unsigned char toggle=0;

void main(void)
{
        WDTCTL = WDTPW + WDTHOLD;               // Stop WDT
        InitializeButton();
        InitializeLeds();
        volatile unsigned int i;

        LED_OUT ^= LED1;
        i=0;
        while (i!=50000) i++;

        LED_OUT ^= LED1+LED2;
        i=0;
        while (i!=50000) i++;

        LED_OUT ^= LED1+LED2;
        i=0;
        while (i!=50000) i++;

        LED_OUT ^= LED1+LED2;
        i=0;
        while (i!=50000) i++;

        LED_OUT ^=LED2;
        i=0;
        while (i!=50000) i++;

        __enable_interrupt();                   // Enable interrupts.

        /* Main Application Loop */
        while(1)
        {
                if (toggle==1){
                        LED_OUT ^= LED1 + LED2;
                        i=0;
                        while (i!=50000) i++;
                        }
        }
}
```

```
// PORT1 interrupt service routine
#pragma vector=PORT1_VECTOR
__interrupt void PORT1_ISR(void)
{
        if (toggle==1 )
                toggle=0;
        else
                toggle=1;
                P1IFG=0x00;
                volatile unsigned int  c=0;
        while (c!=50000) c++;
}
```

E2 Config.h

```
#ifndef CONFIG_H_
#define CONFIG_H_

#define LED1                BIT4
#define LED2                BIT5
#define LED_OUT             P6OUT
#define LED_DIR             P6DIR

#define BUTTON              BIT7
#define BUTTON_IN           P1IN
#define BUTTON_OUT          P1OUT
#define BUTTON_DIR          P1DIR
#define BUTTON_REN          P1REN
#define BUTTON_IES          P1IES
#define BUTTON_IE           P1IE
#define BUTTON_IFG          P1IFG

void InitializeButton(void);
void InitializeLeds(void);

#endif /*CONFIG_H_*/
```

E3 Config.c

```
#include "config.h"
#include <msp430.h>

void InitializeButton(void)                    // Configure Push Button
{
  BUTTON_DIR &= ~BUTTON;                       // In
  BUTTON_REN |= BUTTON;                        // Pull-up/down enabled
  BUTTON_OUT |= BUTTON;                        // Pull-up
  BUTTON_IES |= BUTTON;                        // High to Low
  BUTTON_IFG &= ~BUTTON;
  BUTTON_IE |= BUTTON;
}

void InitializeLeds(void)                      // Configure LEDs
{
  LED_DIR |= LED1 + LED2;
  LED_OUT &= ~(LED2);
  LED_OUT |= LED1;
```