controller (+interfaccia CAT) per loop magnetica realizzato con Arduino UNO

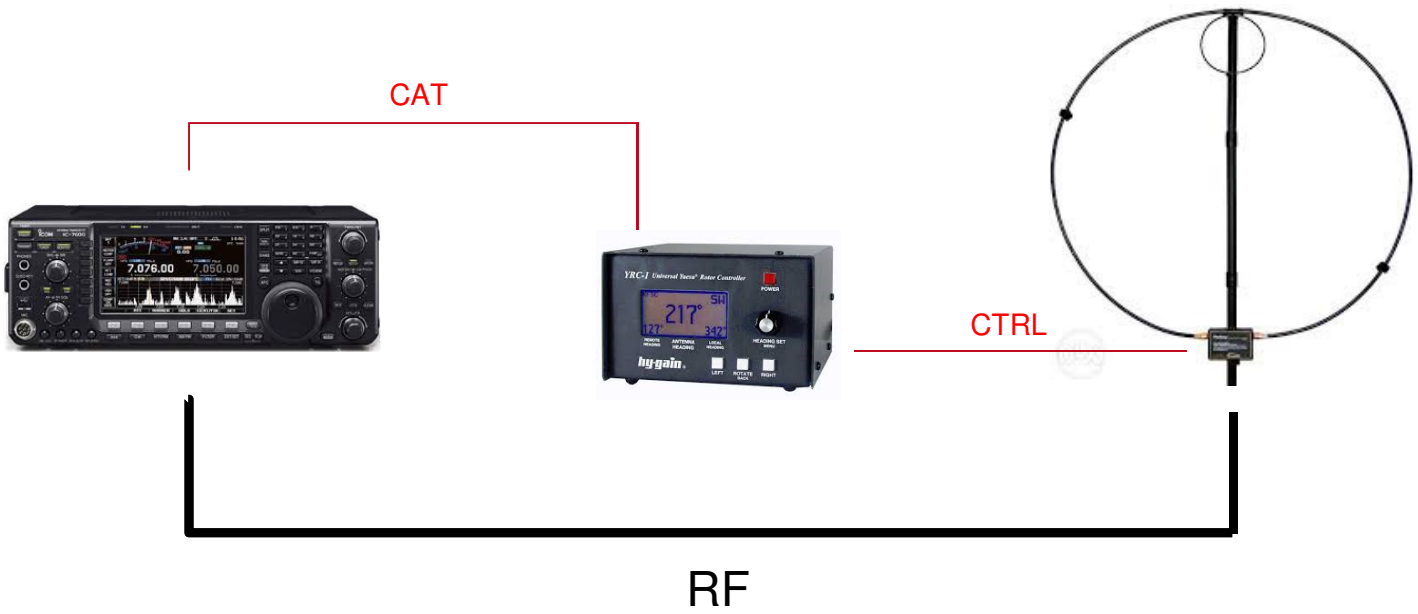
Perugia, gennaio 2018

IKØPUL, Andrea

Progetto

OBBIETTIVO:

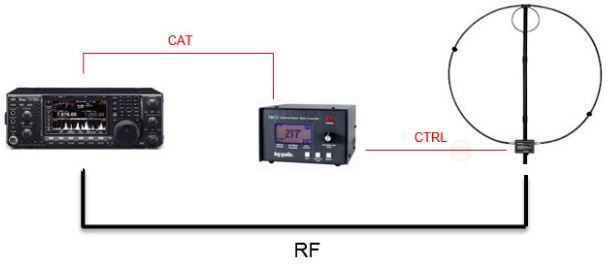
realizzare un sistema (con Arduino) che permetta di sintonizzare in tempo reale una antenna loop magnetica seguendo il VFO dell' RTX, senza alcun intervento dell'operatore…



Progetto

PREREQUISITI:

conoscenza generale della loop magnetica



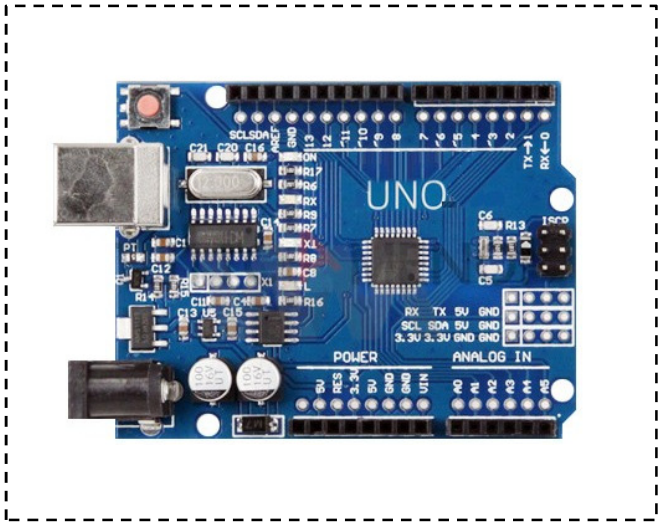
REQUISITI:

1. controller per il tuning a distanza del loop (posizionamento della capacità)
2. interfaccia *CAT* tra controller ed RTX (qualsiasi RTX)
3. tuning *automatico* (o semi automatico) del loop quando sposto il VFO
4. potenza 100 W max
5. frequenza operativa: 12mt -> 80mt

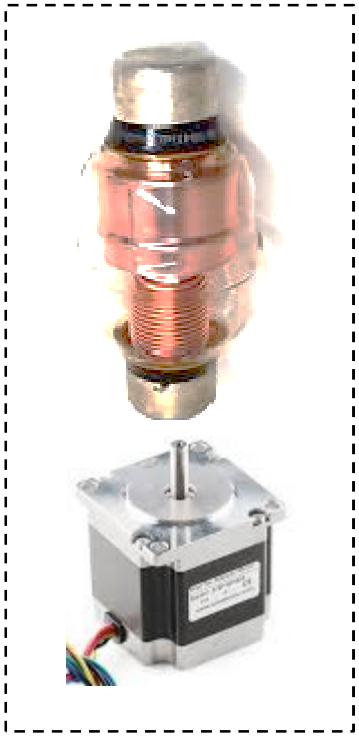
TUNING a Distanza

questo requisito implica il *posizionamento* (preciso e fine) del condensatore variabile della loop da parte del *controller* posto in stazione

Arduino UNO



CONTROLLER



condensatore variabile sottovuoto 250pF

motore passo passo

POSIZIONAMENTO
(attuatore antenna)

TUNING a Distanza / posizionamento

Possibili realizzazioni:

consensatore a farfalla
da 0 a Cmax in 180°

servo motori
(tipicamente da 0° a 180°)

OK per QRP

Motori DC
(tipicamente da 0° a 180°)

OK per tuning manuale

TUNING a Distanza / posizionamento

A.R.I. Associazione Radioamatori Italiani Sezione di Perugia "IQ0WN Mario Benevenga"

RISOLUZIONE del POSIZIONAMENTO

250 pF in 200 step/giro x 22 giri = **0,0568 pF / step**

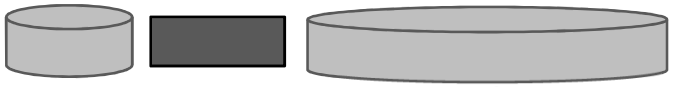
con passo 1/8 di step (1600 step/giro, 22 giri x 1600 = 35200 step):

250 pF in 1600 step/giro x 22 giri = **0,0071 pF / step**

condensatore
variabile
sottovuoto 250pF
5 KV, **22 giri**

TX up to 100 W

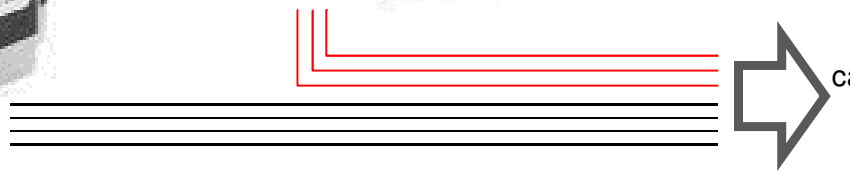
12 denti



26 denti

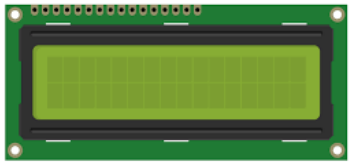
motore
passo passo
NEMA 23
200 step per 360°
(1,8°/step, in full step)
Coppia 1,1Nm
Corrente 2,5A

Potenziometro
A filo 1Kohm
in **10 giri**
(set point iniziale
+ ctrl di fine corsa)

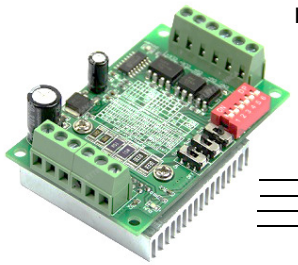


cavo al controller
4 + 3 fili

TUNING a Distanza / controller



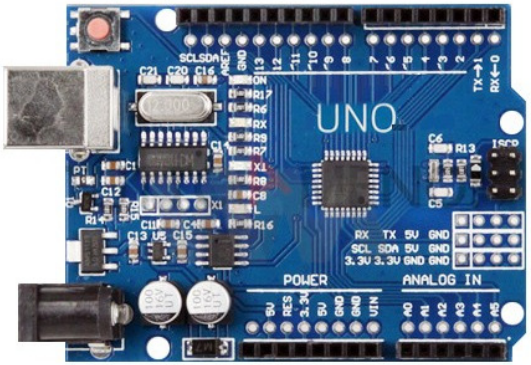
Display LCD



Driver motore p.p.
mod.TB6560,3A max

al motore
passo passo

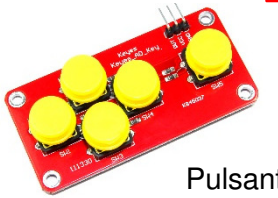
pin DIGITALI



pin ANALOGICI

Potenziometro
setpoint

SWITCHING
TRIMMER 220V-12V 5A



Pulsantiera
(partitore resistivo)

LOGICA di CONTROLLO / Teoria

Fino ad ora abbiamo descritto un dispositivo che permette di associare (e controllare) in modo remoto:

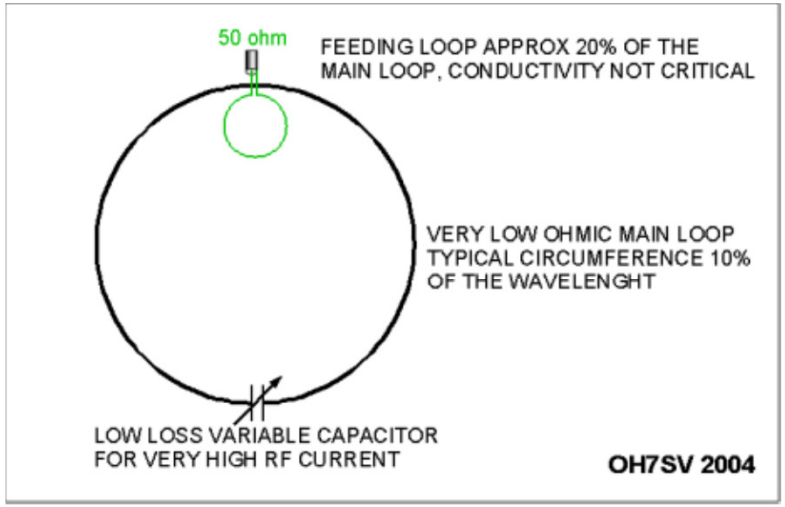
numero di step motore p.p. #STEP [1;35200] ⟺ valore capacità Cvar [5,250] pF

LOGICA di CONTROLLO / Teoria

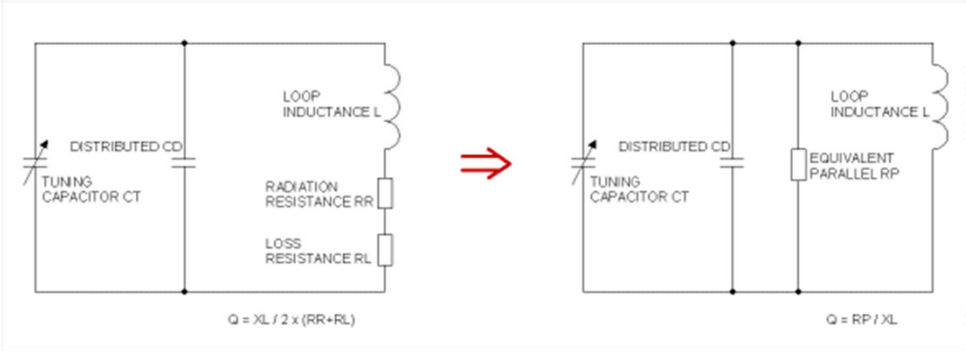
Magnetic loop antenna calculator for round copper construction 1.8.2004 OH7SV

	Enter here
Frequency (MHz)	7,1
Loop conductor length (m)	4,9
Conductor diameter (mm)	25
RF power (W)	100

	Results
Inductance (uH)	4,107
Capacitance (pF)	122,3
XL = XC (ohms)	183,2
Distributed C (pF)	13,2
Tuning capacitance (pF)	**109,2**
Main loop diameter (m)	1,560
Feeding loop diameter (m)	**0,312**
Radiation resistance (ohms)	0,036
Loss resistance (ohms)	0,043
Parallel resistance (kohms)	211
Efficiency (%)	45,6
Compared to ideal loop (dB)	**-3,41**
Loaded Q	1150
Bandwidth -3dB (kHz)	**6,2**
Capcitor rms voltage (V)	**4591**

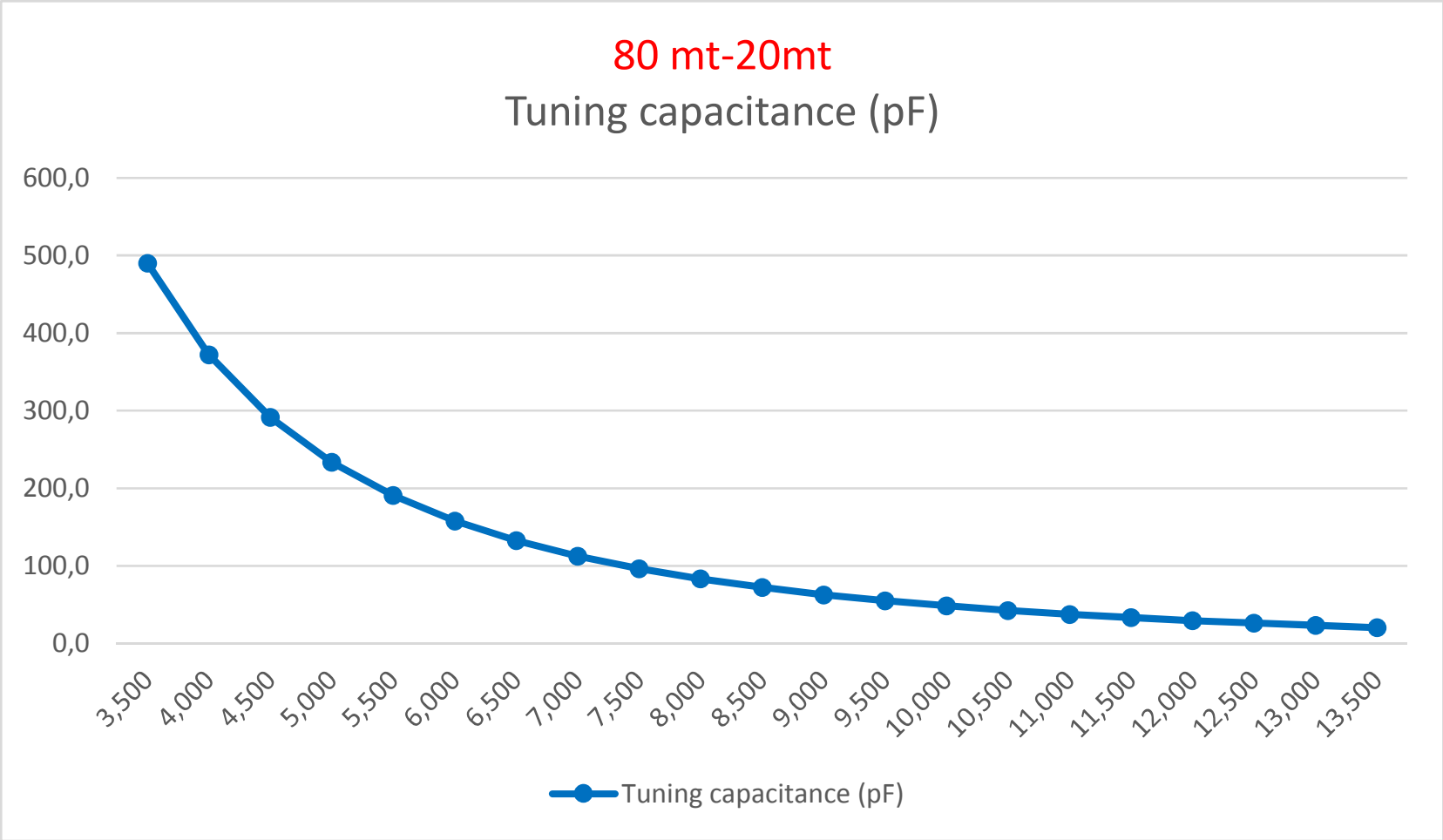


$$L = 62.3e^{-9} l \left(7.353 \log \frac{8l}{\pi d} - 6.386 \right)$$

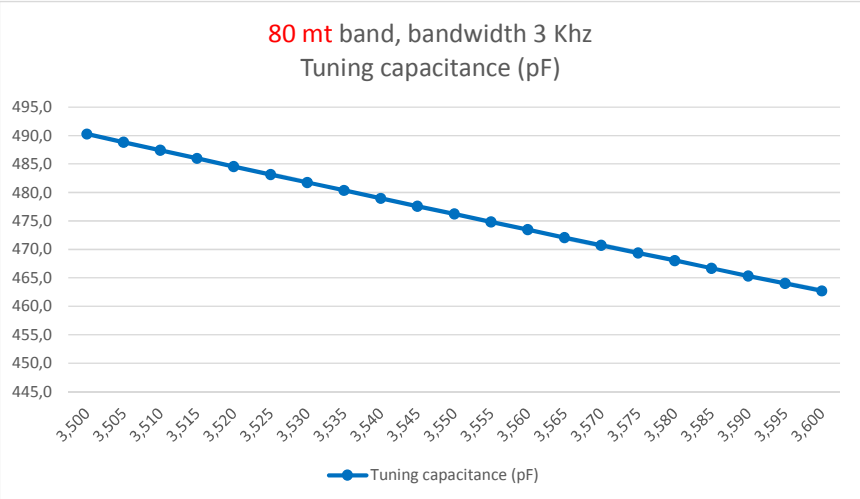


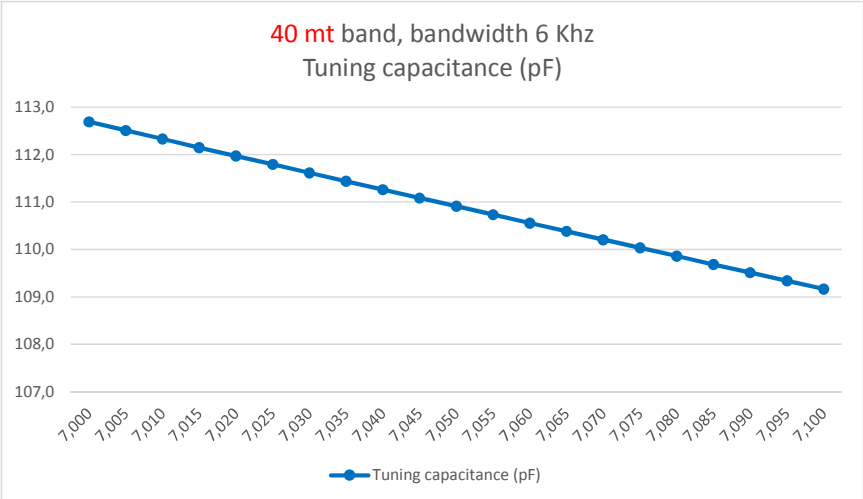
Nota: diametro loop 1,56 mt, diametro conduttore 25 mm

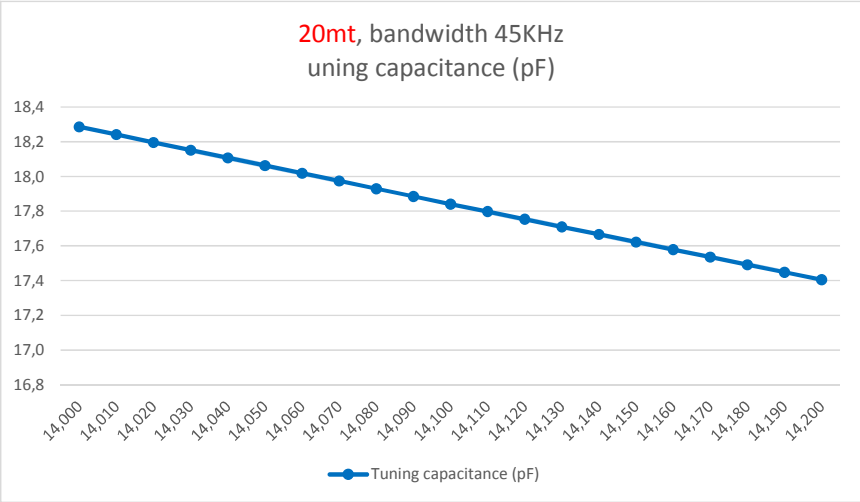
LOGICA di CONTROLLO / Teoria

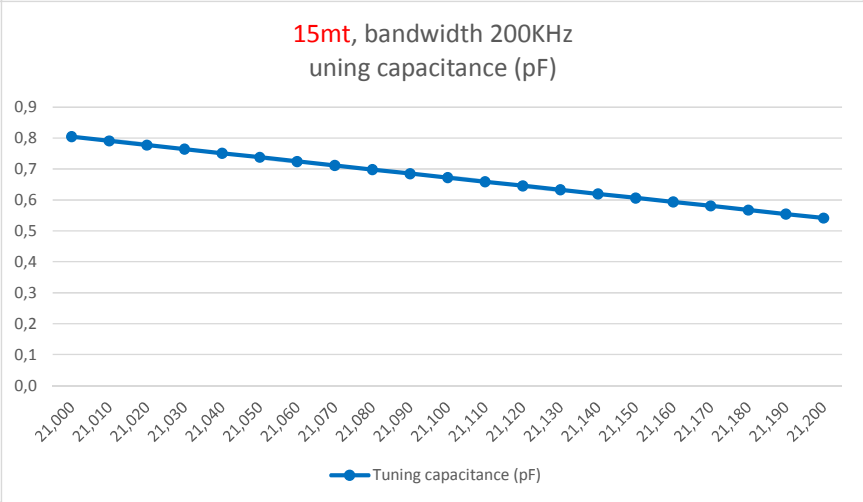


LOGICA di CONTROLLO / Teoria









LOGICA di CONTROLLO / Pratica

Si definisce un banco di memorie e si effettua la *CALIBRAZIONE*
per tali frequenze (= il posizionamento o # di step del motorino passo passo)

F.min	step	F.max	step
3500	*St-1-low*	3700	*St-1-high*
7000	*St-2-low*	7200	*St-2-high*
10100	*St-3-low*	10150	*St-3-high*
14000	*St-4-low*	14350	*St-4-high*
18068	*St-5-low*	18168	*St-5-high*
21000	*St-6-low*	21450	*St-6-high*
24890	*St-7-low*	24990	*St-7-high*

per qualsiasi frequenza intermedia, il posizionamento si *calcola* in modo lineare

CALIBRAZIONE

Per semplicità realizzativa, la ***CALIBRAZIONE*** è un processo **MANUALE**:

si determina lo step di minor SWR colleganto il loop ad apposito analizzatore di antenna oppure leggendo l' SWR sull'RTX

Altre realizzazioni permettono di effettuare una operazione **AUTOMATICA**:

esempio 1) il controller è dotato di lettore di SWR

esempio 2) il controller è in grado di mandare in TX l' RTX e leggere l'SWR rilevata dall'RTX stesso (es. FT-817/857)

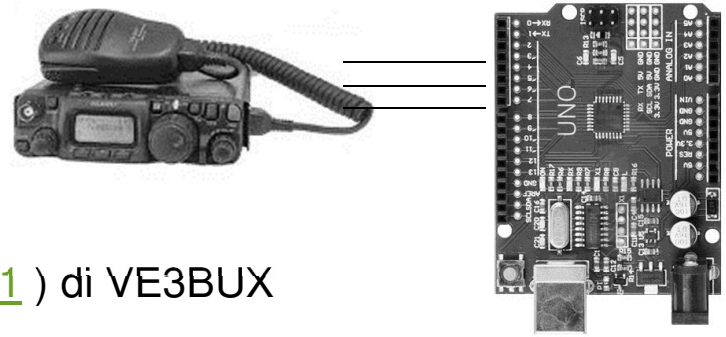
in entrambi i casi il controller può effettuare una scansione ed autocalibrarsi

LOGICA di CONTROLLO

Fino ad ora abbiamo descritto un dispositivo che permette di associare (e controllare) in modo remoto:

Frequenza Operativa FOp MHz ⟺ numero di step motore p.p. #STEP [1;35200] ⟺ valore capacità Cvar [5,250] pF

CAT: opzione 1, Yaesu FT-8x7



RTX CAT/ACC GND -> Arduino GND
RTX CAT/ACC TX -> Arduino pin 3
RTX CAT/ACC RX -> Arduino pin 2

FT857D library (http://ve3bux.com/?page_id=501) di VE3BUX

```
extern SoftwareSerial rigCat(2,3); // rx,tx
```

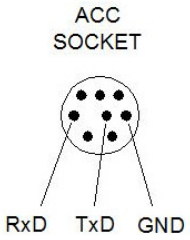
OSSERVAZIONI:

Yaesu FT-8x7 ACC implementation una segnalazione tipo RS-232 MA hanno uscite a livello di TTL (0,+5V) quindi collegabili direttamente ai pin digitali di Arduino senza alcun tipo di adattamento.

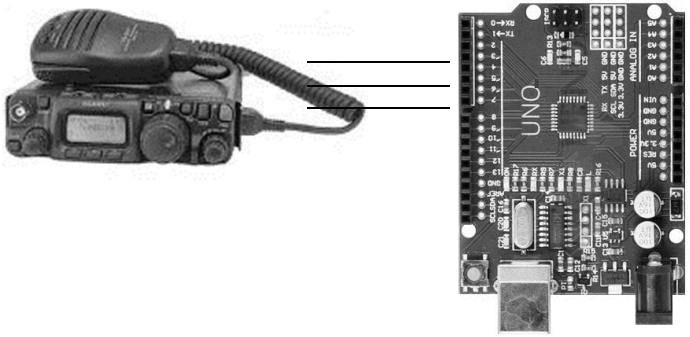
La libreria permette di controllare tutte le funzioni dell'RTX quindi permette di implementare la *CALIBRAZIONE AUTOMATICA*

esempio https://www.alain.nyc/2012/02/10/ft-817-automatic-loop/

CAT: opzione 1, implementazione



RTX CAT/ACC GND -> Arduino GND
RTX CAT/ACC TX -> Arduino pin 3
RTX CAT/ACC RX -> Arduino pin 2

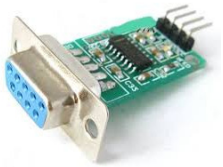


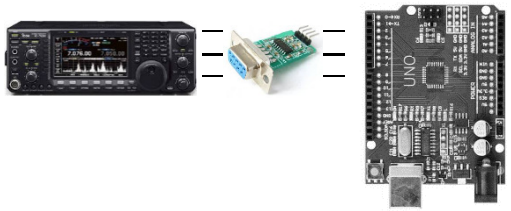
```
#include <SoftwareSerial.h>
#include "FT817.h"

FT817 rig;  // CAT object
frequenza ="";

void setup() {

  //inizializza CAT
  SoftwareSerial mySerial(2,3);
  rig.assignSerial(mySerial);
  rig.begin(9600);
  delay(500);

}

void loop() {

    //leggo la frequenza dell'RTX
    frequenza=String(rig.getFreqMode());
    Serial.println(frequenza);
    delay(100);

}
```

FT857D library (http://ve3bux.com/?page_id=501) di VE3BUX

CAT: opzione 2, any RIG RS232

Le porte standard seriali RS232 usano altri valori di tensione e bisogna aggiungere un piccolo convertitore di livello (coso circa 2/3 eur), ma in generale serve un RX + TX:

RS-232
RTX CAT/ACC GND ->  -> Arduino GND
RTX CAT/ACC TX -> -> Arduino pin 3
RTX CAT/ACC RX -> -> Arduino pin 2




```
extern SoftwareSerial rigCat(2,3); // rx,tx
```

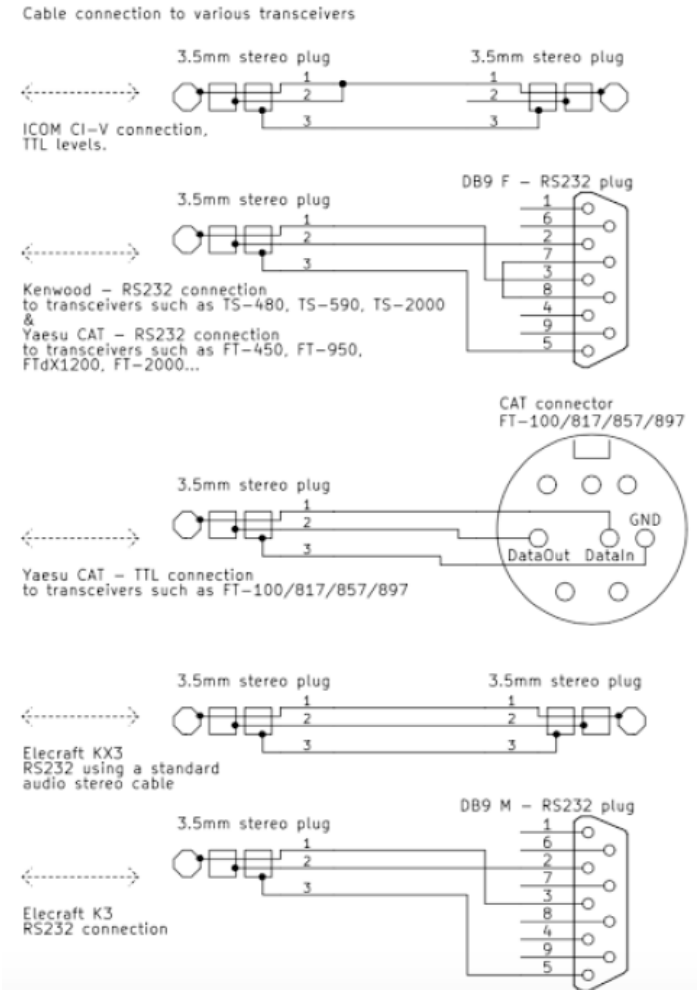
OSSERVAZIONI:

bisogna implementare il protocollo CAT per qualsiasi famiglia di apparati
(al tempo del progetto non erano disponibili in rete librerie per IC, Ele, KW)

Esiste un progetto che realizza questo tipo di interfaccia di TF3LJ / VE2LJX:

https://sites.google.com/site/lofturj/to-automatically-tune-a-magnetic-loop-antenna

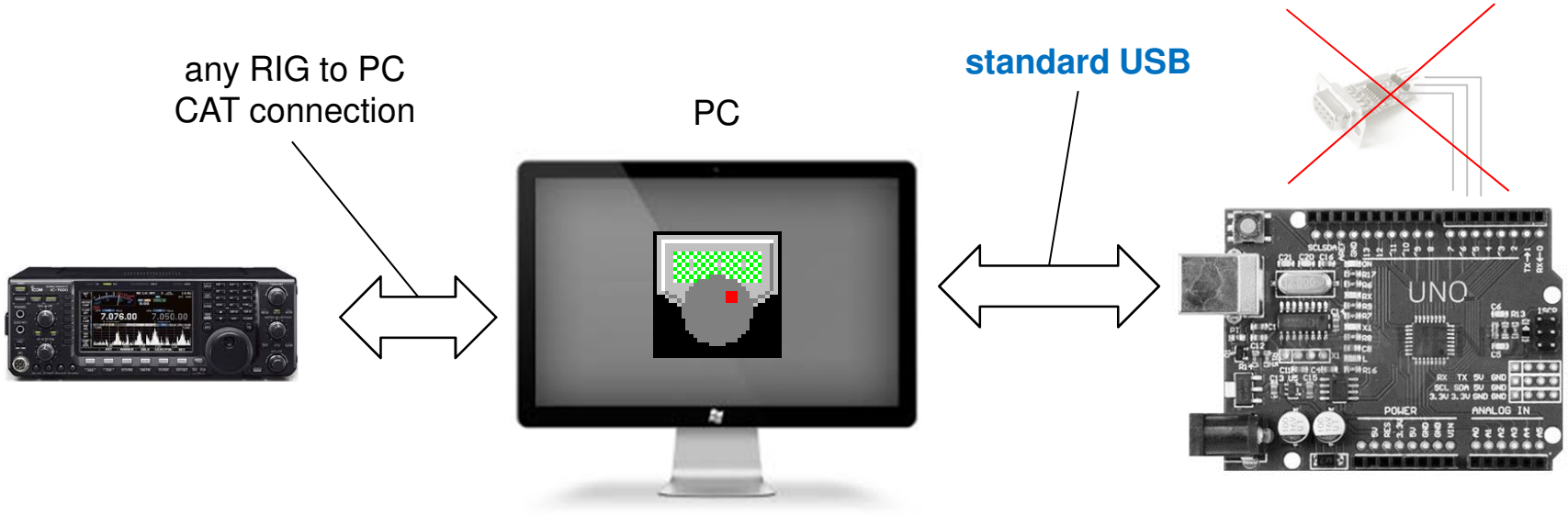
CAT: opzione 2, any RIG RS232



Fonte: TF3LJ / VE2LJX:
https://sites.google.com/site/lofturj/to-automatically-tune-a-magnetic-loop-antenna

CAT: opzione 3, any USB

La soluzione più semplice e veloce è stata quella di utilizzare OMNIRIG di VE3NEA:



Il controller/Arduino esegue un polling al PC che ritorna la frequenza del VFO:

```
Serial.readString();
```

Il cavo virtuale (software su PC) tra RTX ed Arduino è stato scritto in Visual Basic.

CAT: opzione 3, implementazione

```
1   Imports System.IO.Ports
2
3   Public Class Form1
4
5       Dim Rig As OmniRig.RigX
6
7
8       'Form LOAD events
9       Private Sub Form1_Load(sender As Object, e As EventArgs) Handles Me.Load
10          RadioButton1.Checked = True
11          StartOmniRig()
12          GetSerialPortNames()
13
14      End Sub
15
16      Private Sub Form1_FormClosed(sender As Object, e As FormClosedEventArgs)
17          StopOmniRig()
18      End Sub
19
```

Creazione oggetto OMNIRIG
(RTX <-> PC)

```
138
139     Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
140
141         controllerConnected = ""
142         SerialPort1.Close()
143         SerialPort1.PortName = ComboBox1.SelectedItem.ToString
144         Try
145             SerialPort1.Open()
146             SerialPort1.Write("ik0pul")
147         Catch ex As Exception
148             MsgBox("impossibile connettere il controller")
149             Exit Sub
150         End Try
151         MsgBox("OK")
152     End Sub
```

Apertura SERIALE su USB
(PC <-> Arduino)

CAT: opzione 3, implementazione

POLLING:

- il controller/Arduino chiede al PC via SERIALE la frequenza dell'RTX
- SW sul PC interroga l'RTX tramite OMNIRIG
- SW sul PC restituisce la frequenza al controller/Arduino

```
Private Sub SerialPort1_DataReceived(sender As Object, e As SerialDataReceivedEventArgs) Handles
    Dim Incoming As String = SerialPort1.ReadLine()

    If Rig.Status = ST_ONLINE Then
        SerialPort1.Write(Rig.GetRxFrequency / 10)
    Else
        SerialPort1.Write("0")
    End If
```

CAT: opzione 3, implementazione

OmniRig_VB_DEMO (In esecuzione) - Microsoft Visual Studio Express 2015 per Windows Desktop

File Modifica Visualizza Progetto Compila Debug Team Strumenti Test Finestra ?

Release | Any CPU | Continua

Processo: [2132] IK0PUL dinamic loop v00.vs | Eventi ciclo di vita | Thread: | Stack frame:

Origini dati

Form1.vb | Form1.vb [Progettazione]

OmniRig_VB_DEMO | SerialPort1

```vb
Private Sub RadioButton1_Click(sender As Object, e As EventArgs) Handles RadioButton1.Click
    SelectRig(1)
End Sub

Private Sub RadioButton2_Click(sender As Object, e As EventArgs) Handles RadioButton2.Click
    SelectRig(2)
End Sub

Private Sub SerialPort1_DataReceived(sender As Object, e As SerialDataReceivedEventArgs) Handles SerialPort1.DataReceived
    Dim Incoming As String = SerialPort1.ReadLine()

    If Rig.Status = ST_ONLINE Then
        SerialPort1.Write(Rig.GetRxFrequency / 10)
    Else
        SerialPort1.Write("0")
    End If
```

100 %

IK0PUL dinamic loop v00

CONTROLLER | Open | RIG1 RIG2 | 182650 | Settings

Omni-Rig Settings

RIG 1 | RIG 2 | About

Rig type	IC-7600
Port	COM 3
Baud rate	19200
Data bits	8
Parity	None
Stop bits	1
RTS	Low
DTR	Low
Poll int., ms	1000
Timeout, ms	100

OK | Cancel

Output

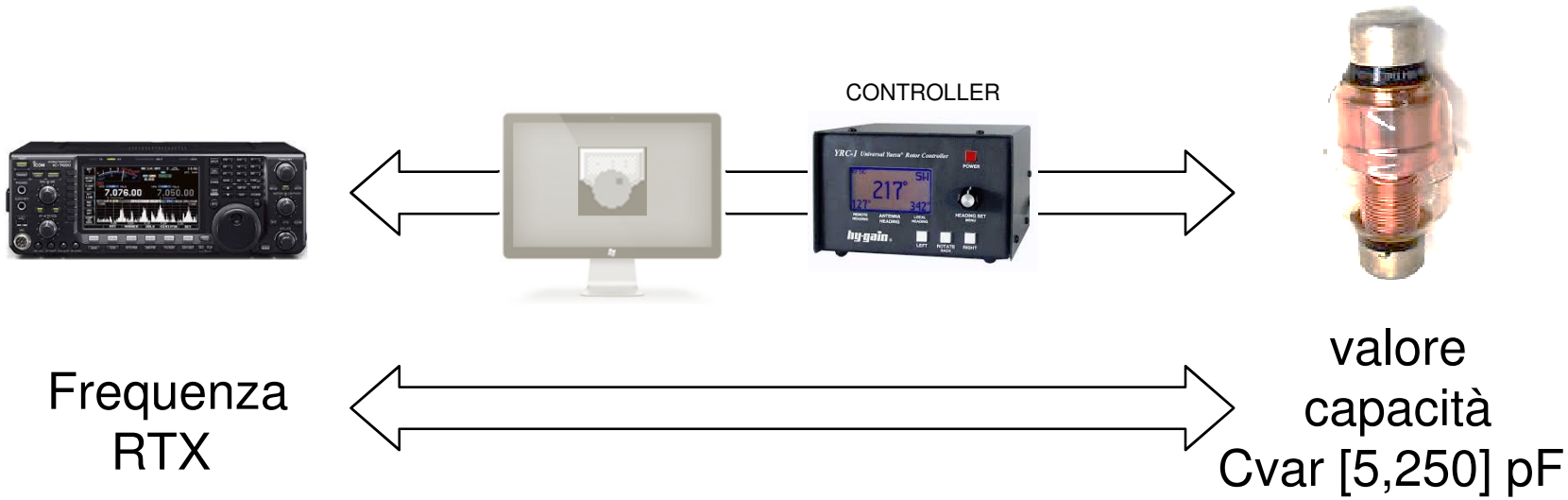
Mostra output di: Debug

```
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.v  \WINDOWS\Microsoft.Net\assembly\GA
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.v  \WINDOWS\assembly\GAC_MSIL\Microsc
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.v  \WINDOWS\Microsoft.Net\assembly\GA
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.v  \WINDOWS\Microsoft.Net\assembly\GA
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.v  \WINDOWS\Microsoft.Net\assembly\GA
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.v  \WINDOWS\assembly\GAC_MSIL\Microsc
'IK0PUL dinamic loop v00.vshost.exe' (CLR v4.0.30319: IK0PUL dinamic loop v00.vshost.exe): caricamento di 'C:\WINDOWS\assembly\GAC_MSIL\Microsc
```

Riepilogo

REQUISITI implementati:

1. controller per il tuning a distanza del loop (posizionamento della capacità)
2. interfaccia *CAT* tra controller ed RTX (qualsiasi RTX)
3. tuning *automatico* (o semi automatico) del loop



Realizzazioni sperimentate

A.R.I. Associazione Radioamatori Italiani Sezione di Perugia "IQ0VN Mario Benevenga"

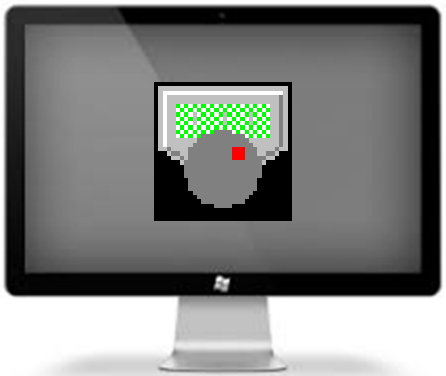
Opz.3

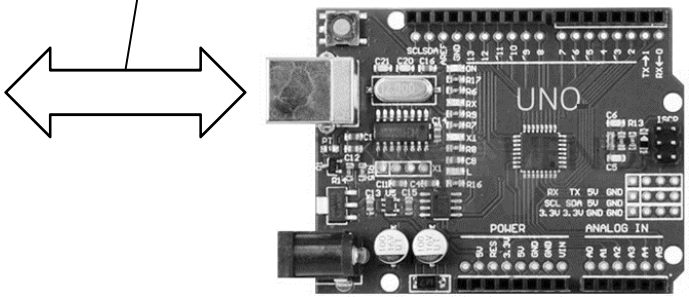
any RIG to PC
CAT connection

PC

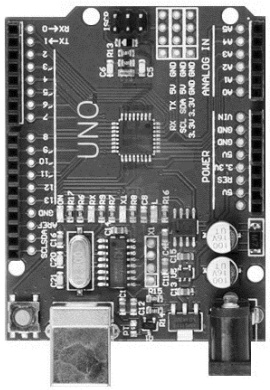
standard USB

Firmware 2
(demo)





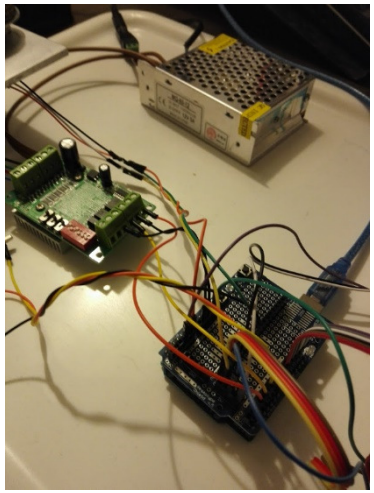
Opz.1

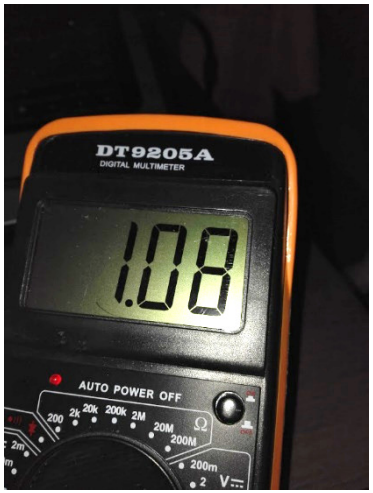


Firmware 1

CALIBRAZIONE
MANUALE

Prototipo




DT9205A
DIGITAL MULTIMETER
1.08
AUTO POWER OFF

MIC LOOP V01
ing...1.08V

MANUAL TUNED
14050.0MHz

Note

I seguenti aspetti sono soggetti a revisione dopo sperimentazione:

motore passo passo, in full-step oppure 1/8 di step

PRECISIONE del posizionamento (set-point, gioco generale attuatore)

BANCHI di MEMORIA (necessità di estendere in sottobande)

RF Choke / Ferriti nell'attuatore per evitare rientri di RF al controller

NOISE dell'alimentatore switching all'interno del controller
(risolto con due condensatori agli ingressi analogici di Arduino)

Possibili sviluppi:

implementazione della Calibrazione automatica

supporto Any Rig (ma senza PC/OMNIRIG)

DEMO

Riferimenti

Alan De Carolis, NY
https://www.alain.nyc/2012/02/10/ft-817-automatic-loop/

VE3BUX:
http://ve3bux.com/?page_id=501

GM4WZG:
http://www.gm4wzg.co.uk/antennas.html

TF3LJ / VE2LJX:
https://sites.google.com/site/lofturj/to-automatically-tune-a-magnetic-loop-antenna

I6IBE:
http://www.radioamatoripeligni.it/i6ibe/loop/loop.htm

IU3BRK:
http://www.aripadova.it/autocostruzione/36-antenne/275-una-antenna-loop-magnetica-da-premio.html

THANK YOU

A.R.I.
Associazione
Radioamatori Italiani
Sezione di
Perugia
"IØWN Mario Bencivenga"

IKØPUL, Andrea