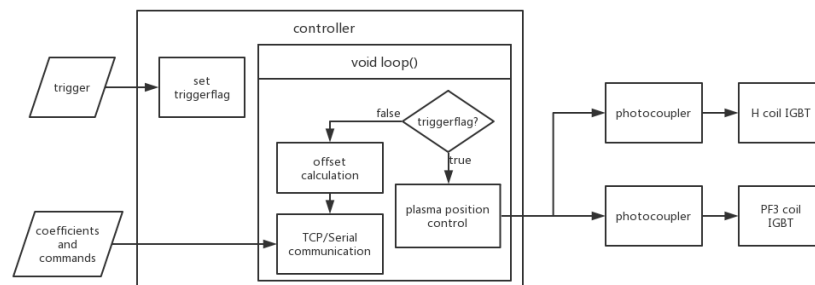


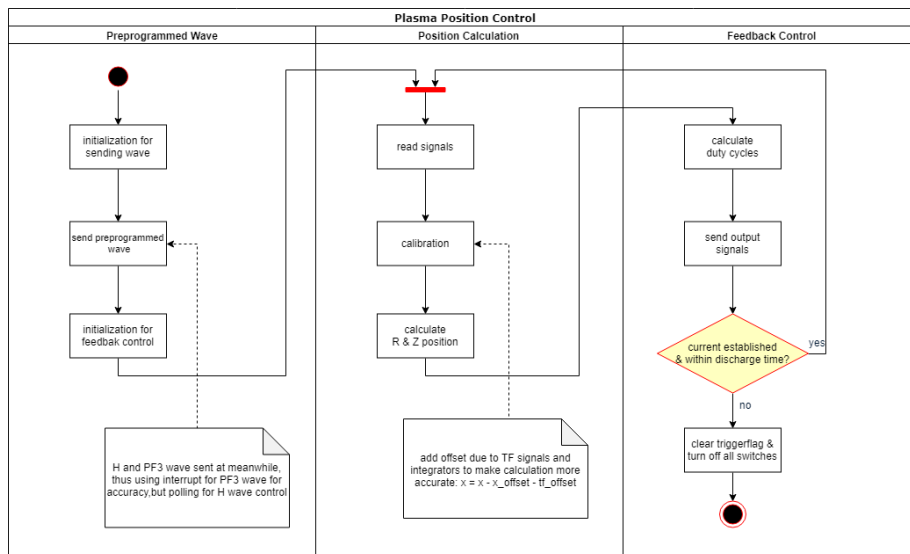
PF3 & H coil controller notes

In order to reduce manual work to control plasma position and get a more stable position, a new controller for PF3 and H coil was developed in TST-2 spherical tokamak. The new controller is featured with feedback control of plasma position in both horizontal and vertical dimensions, which will perform a real-time calculation of plasma position and give feedback control signals according to the error from the target position.

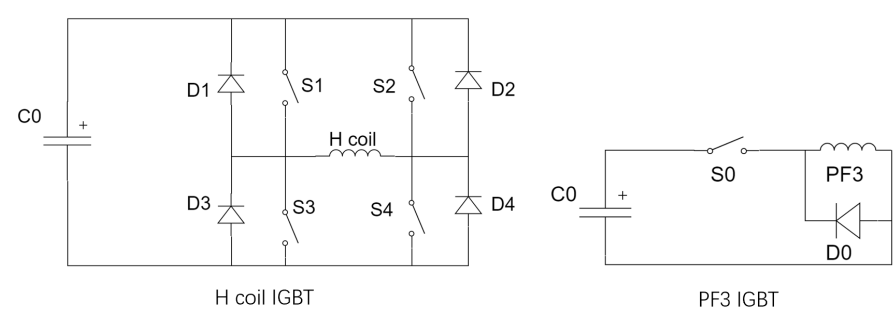
Overview



A piece of [Arduino Mega 2560](#) board is used as the processing unit of controller, which is based on [ATmega2560](#) and with a maximum CPU frequency of 16MHz. The controller is working on two states. When in idle time, it calculate signal offsets due to integrator baselines. And in this period we can send configurations of discharge via TCP-Serial communication. When trigger comes in, trigger flag is set by interrupt, the controller will enter into controlling process in next loop.



Before the formation of plasma current, preprogrammed waveforms are generated by the two coils. When the plasma current is established, it repeats the process of position calculation and feedback control, until time is over (100 loops) or plasma current disappeared (op signal is below the limit).



Signals controlling the on and off of IGBT are output from feedback control process. For **H coil**, turn on **H1(S1)** and **H4(S4)(PWM)** makes the current flow in one direction, turn on **H2(S2)(PWM)** and **H3(S3)** makes the current flow in another direction. For **PF3**, only **PF(S0)(PWM)** is output to control the IGBT.

Usage Instructions

Before discharge, edit targets, preprogrammed waves and other configurations in `config.txt`, run `./configure.sh` to establish communication, sending configurations and waiting for data to be sent back. The controller address is `192.168.1.31:10001`. Communication log will be wrote to `test_shotNo.txt`. **Rerun the script every time before discharge.**

If it's Ohmic discharge, run `./configureOH.sh` to send all the calibration coefficients for Ohmic discharge, which is enough to run for only once after turning on the controller. **The default settings is for LH discharge.**

Pin Mapping

PIN	TAG	SIGNAL	MODE
A1	0	SO0	INPUT
A2	1	SO1	INPUT
A3	2	SO2	INPUT
A4	3	SO3	INPUT
A5	4	SO4	INPUT
A6	5	SO5	INPUT
A7	o0	o0	INPUT
A8	c0	c0	INPUT
A9	BP COIL	OP	INPUT
A10	TF	TF	INPUT
2	R	PF	OUTPUT
3	Z0	H4	OUTPUT
4	Z1	H1	OUTPUT
5	Z2	H2	OUTPUT
6	Z3	H3	OUTPUT

5 PIN	Z2 TAG	H2 SIGNAL	OUTPUT MODE
6	Z3	H3	OUTPUT
20	TRIGGER	TRIGGER	INPUT

Notes:

- TF signal input is divided by a resistor divider, the original divided coefficient is designed of 2. However, due to the input impedance, the actual coefficient is 3, which means real TF signal is $TF \times 3$
- o0 is the flux loop above the six saddle loop, c0 is the flux loop located at the center horizon of solenoid.

Commands

COMMAND	TARGET	TYPE	USE
A	c_op	double	op calibration coefficient
B	c_c	double	c0 calibration coefficient
C	c_o	double	o0 calibration coefficient
D	c_so[]	double	so0-so5 calibration coefficient
E	tffo_so[]	double	coefficient of tf offset on so0-so5
F	tffo_c	double	coefficient of tf offset on c0
G	tffo_op	double	coefficient of tf offset on op
H	tffo_o	double	coefficient of tf offset on o0
I	c_bt	double	tf calibration coefficient
r	r_t	double	R target [m]
z	z_t	double	z target [m]
p	Kz_p	double	P coefficient on z control
i	Kz_i	double	I coefficient on z control
x	Kr_p	double	P coefficient on R control
y	Kr_i	double	I coefficient on R control
s	delayfromTrigger	long	time between trigger and first PF3 preprogrammed wave [us]
f	PF_default	int	the beginning duty cycle of PF3 after preprogrammed wave (0-255)
m	pre_r[]	int	PF3 preprogrammed wave
n	pre_z[]	int	H coil preprogrammed wave
l	pre_z_state[]	int	H coil preprogrammed wave state
o	op_limit	double	op signal limit to stop control

Usage: When TCP communication established, using `'command'+'value'` to send value to variables, using `'command'+'value1'+'value2'+...` to send values to array. There should be space between command and value, and among different values. For example, use `D -0.0104563 -0.0125658 -0.0107368 -0.00741706 0.0115751 0.0132428` to set `c_so[]`

Notes:

- Set PI coefficients to be negative means to change direction of H coil current, but doesn't mean it will change PF3 current direction.
- For `c_so[]` and `tfoff_so[]`, the element number in array is fixed on 6, but for `pre_r[]` and `pre_z[]`, the number is not fixed, which is required to be sent after the command and before waveform values. For example, use `m 3 1000 2000 3000` to set PF3 preprogrammed wave.
- PF3 preprogrammed wave value is the time of on and off in turns, the first value is on. The unit is us. For example, `m 3 1000 2000 3000` means to turn on 1ms, turn off 2ms, turn on 3ms, and then off. PF3 will always turn off after preprogrammed wave ends.
- For H coil preprogrammed wave, because the wave has five kinds of state, `pre_z_state[]` is required to sent after `pre_z[]`, but it doesn't need to send wavenumber. The wavenumber is the same as `pre_z[]`. For example, sent `n 2 1000 1000` and then `1 1 5` to set z preprogrammed wave.

`pre_z_state[]` only have 5 legal values, corresponding to 5 states of H coil wave, where the 4 switches of H bridge are as follows:

PRE_Z_STATE	H1	H2	H3	H4
1	on	off	off	on
2	on	off	off	off
3	off	on	on	off
4	off	off	on	off
5	off	off	off	off