Controls and Monitoring Program for the Cooldown of Hall B's Superconducting Solenoid

Pablo Campero, Mary Ann Antonioli, Peter Bonneau, Brian Eng, Amanda Hoebel, George Jacobs, Mindy Leffel, Tyler Lemon, Marc McMullen, and Amrit Yegneswaran

Physics Division, Thomas Jefferson National Accelerator Facility, Newport News, VA 23606

June 29, 2018

This note presents the details of the controls and monitoring program of Hall B's superconducting Solenoid. The controls and monitoring program, written for the Allen-Bradley PLCs in Rockwell's Logix Designer RSLogix 5000, automates warm-up/cooldown, current ramps, interlock initiations, and transmission of sensors and instrumentation data to EPICS.

The cooldown of the Solenoid is done in phases, each of which has an associated set of cooldown parameters and is defined by the coil temperature, *T*.

$$T \in \begin{cases} [300\text{ K}, 100\text{ K}), & \text{Phase I} \\ [100\text{ K}, 4.5\text{ K}), & \text{Phase II} \\ @\ 4.5\text{ K}, & \text{Phase III} \end{cases}$$

To cool down and hold the Solenoid at 4.5 K, helium is pumped from the end station refrigerator (ESR) to the Distribution Box (DBX), from where it is piped via flexible U-tubes to the Solenoid Service Tower (SST). Located in the SST are the current lead reservoir to cool down the current leads, and the magnet reservoir to provide helium to the Solenoid's thermal shield and coils. After passing through the shield and coils, helium is returned via the DBX back to the ESR, Fig 1.

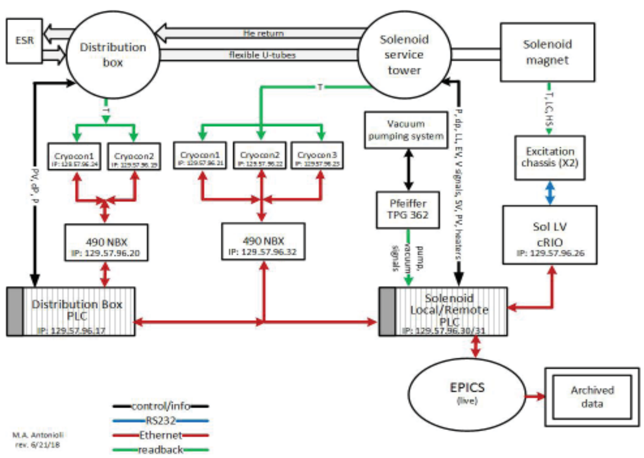

FIG. 1. Cryogenics controls and communication overview.

The Solenoid is controlled and monitored by the DBX and Solenoid PLCs, Fig 1, which share an array of 60 instrumentation and cryogenics process variables (PVs). Based on the outlet helium temperature at the DBX (processed by the DBX PLC's program) and the Solenoid's temperature and pressure, cooldown parameters are calculated by the Solenoid PLC.

PLC programs in the Solenoid PLC, *Fast_DAQ_Processing, Cryocon, Cryo,* and *cPID*, read data and control instrumentation, Fig. 2. Table I lists program details.

Fast_DAQ_Processing reads data from the Sol LV cRIO, which works in conjunction with the Excitation chassis (Fig. 1) to control and monitor 26 Cernox temperature sensors, 15 PT-100s, 16 load cells, and three Hall sensors, all of which are located inside the Solenoid.

Three Cryocon units read 11 PT-100s and 13 Cernox temperature sensors in the SST. Every 250 ms, *Cryocon* sends commands to and receives data from the Cryocon units via the 490NBX module and an Ethernet connection.

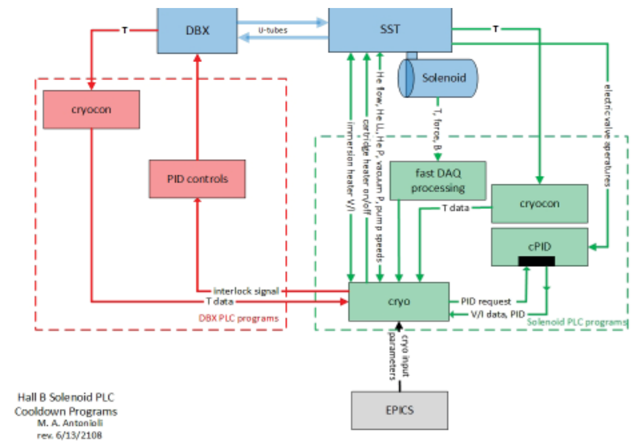

FIG. 2. Controls and monitoring software overview.

Cryo controls and monitors PVs. Vacuum and over-threshold load cell signals monitored by *Cryo* are listed in Tables II and III.

cPID controls the four electric valves and the two heaters used during cooldown. These valves and heaters can be controlled in either manual mode—allows operator to test electric valves by opening and closing the valves from 0% to 100%—or normal mode—requires operator to input PID parameters and run *cPID* control logic. Cascading PID loops in *cPID* allows the operator to input, in the normal PID loop, Max and/or Min values with higher precision.

For many interlock actions, helium supply valves located in the DBX reservoir, controlled by the DBX PLC, are closed to stop supply of warm (300 K) and cold (80 K) helium.

Interlock signals to protect the Solenoid are generated when the measured signals, shared by the Solenoid PLC with the DBX PLC, are higher than established thresholds (Table IV) for 90 s; in this event, DBX PLC stops cooldown by closing the DBX reservoir's helium supply valves.

Prior to cooldown, DBX, SST, and Solenoid instrumentation; pneumatic, electric, solenoid valves; mass flow controllers, and vacuum interlocks are tested.

To conclude, sensors, instrumentation, automated control and monitoring systems, interlock systems, and checkout and operating procedures were designed, developed, and implemented to ensure safe cooldown of the Solenoid.

In July 2017, the Solenoid was cooled down to 4.5 K and has been operating since.

Program	Routine	Function
Cryocon	Scheduler	Sets watchdog counter between Solenoid PLC and Cryocon 18i units. Checks communication errors with 490NBX module and resets communications if needed. Goes to *Crycon* if there are no communication errors.
	Cryocon	Sends commands to 3 Cryocon 18i units every 250 ms to read temperature data. Ensures ASCII commands are sent in correct format. Receives data from Cryocon units in assigned tag that handles string data. Copies responses in placeholder and clears last response's tag for next response from Crycon unit. Sends this received data to *StringParse* for converting string to float. Once data is returned from *StringParse*, displays temperatures individually. Makes data available for use in PLC program.
	StringParse	Splits string data type of response from *Crycon*, converts values to float numbers, and returns temperature as a number.
Cryo	Scheduler	Sets up order/sequence to run each routine in *Cryo*.
	Cooldown_Rate	Calculates cooldown rate of change in coil temperatures and shieldout temperatures in the past 30 minutes, 120 minutes and 600 minutes.
	Cryo_Interlocks	Compares vacuum readouts with set threshold. If any of 16 load cell readout values are above threshold, sends interlock signals to Distribution Box PLC to halt cooldown.
	dP_to_Liquid_Level	Calculates vapor and LHe density for magnet and lead reservoirs based on pressure in each reservoir. Calculates liquid level in each reservoir based on pre-measured height value between 2 fixed points, differential pressure, and previously calculated LHe and He vapor densities.
	Estimated_Coil_ Temperature	Calculates estimated temperature in each of the five coils based on resistance in coils, following R = V/I. Current in each coil considered the same since the five coils are in series. Calculates temperature values by using linear function formula based on coil length.
	Heater_Control	Reads current and voltage signals from heater power supply/controller to calculate power in heaters. Checks liquid level conditions in each reservoir and PID parameters to compare with thresholds set by user from EPICS. If liquid levels ≥ set min level, enables heaters. Checks user's operation selection (manual or normal mode) and sends heater voltage readout to *cPID* to set PID control over heaters. Receives voltage output values and scales these to send them as a voltage percent to control power in heaters.
	Metal_Temps	Calculates all Solenoid cooldown parameters: coil temperature average, max temperatures in all coils, max differential temperatures in coils 1-4, temperature in bobbins and cooling plates, max temperature differential between coils 1-4, bobbins and cooling plates, calculates max temperature between coil 5 and bobbins, max/min temperature of 4 K cold mass, max differential of temperatures in 4 K cold mass, differential of He temperatures between Distribution Box outlet and Solenoid inlet, differential of inlet He temperature and metal temperatures, max temperature in cooling plate, max and average temperatures in shield outlets of Solenoid, max temperature of heat shields, max differential temperature of shields and percent of design cooldown.
	Metal_Temps_Rate	Not used
	Pressure_Deltas	Not used
	PV_Assigner	Searches for process variables that are part of PV array used to transfer data between Solenoid, Torus, and Distribution Box PLCs.
	Temperature_Deltas	Allows user to enter temperature signal names (PV names), looks up value associated with PV name entered, and then calculates delta value between them.

TABLE I. Function list in PLC programs.

	Vacuum_Calcs	Reads and scales vacuum pressure readouts from vacuum controller. Uses constant from vendor gauge specification to calculate vacuum pressure and display vacuum pressure in [Torr]. Enables/disables pneumatic valve used for vacuum from EPICS. Reads and scales turbo and vacuum guard pump speeds signals to display them in % values.
	Vacuum_Rate	Calculates vacuum pressure change in [Torr/hr], stores rate every second up to 1 hr, and then calculates average vacuum rate for past 10 hours, 1hr, and 1 min.
	Vapor_CooledLead_Flow	Checks readback current from MPS; if > 25 A, calculates flow values to send out analog signals to mass flow controllers in lead A and lead B in Solenoid. Allows user to enter values from EPICS to control flow in mass flow controllers.
	VCL_Heater_Control	Based on temperatures measured in lead A and lead B, cartridge heaters are disabled or enabled. If lead temperatures are < 301 K, heaters are enabled. If lead temperatures are >303 K, heaters are disabled.
FastDAQ_Processing	Scheduler	Sets up sequence for all routines running that are part of *FastDaq_Processing*.
	Cernox	Checks counter bit sent by cRIO to determine if data for Cernox sensor has been sent; if not, sets assigned bit to alert comm error. Receives Cernox temperature data from cRIO as whole array, breaks down array into 26 elements and names are assigned as individual temperatures. Scales temperatures to display in [K].
	Cernox_Stats	Calculates Cernox statistics for CCM, T_Max, T_Min, T_Avg, and T_Delta.
	PT100	Checks counter bit sent by cRIO to determine if data for PT100 sensors have been sent; if not, sets assigned bit to alert comm error. Receives PT100 temperature data from cRIO as whole array, breaks down array into 15 elements and names are assigned as individual temperatures. Scales temperatures to display in [K].
	Hall_Sensor	Checks counter bit sent by cRIO to determine if Hall sensors data have been sent; if not, sets assigned bit to alert comm error. Receives magnetic field data from cRIO as whole array, breaks down array into three elements and names are assigned with individual Hall probe sensors. Scales magnetic field readouts to be displayed in [G].
	Load_Cells	Checks counter bit sent by cRIO to determine if Hall sensors data have been sent; if not, sets assigned bit to alert comm error. Receives load cell data from cRIO as whole array, breaks down array into 16 elements, and names are assigned individually for each load cell signal. Scales magnetic field readouts to be displayed in [lbf].
	Comm_Error Status	Verifies communication error for any array transferred from cRIO to PLC. If error is present, waits 60 s before latches error and sends error condition to *Comm_Error_Fill*.
	Comm_Error_Fill	Fills arrays used to transfer data (load cells, Cernox, PT100, and hall sensor) between cRIO and PLC with “555.555”, if there is any communication error.
cPID	Scheduler	Sends to *Cpid_Setup* every 50 ms.
	cPID_Setup	Reads position of valve via LVDT, defines type of valve (electric valve). Sends valve information to *cPID_Overhead* Receives values from *cPID_Overhead*, compares LVDT readouts that give exact position of valve with max and min points (-15 %–115% open). Ensures that valves are not mechanically forced and LVDT are properly calibrated. If position of valves are above range, then valves are driven (open or close) with regards to PID output values. Performs cascading PID with similar logic to allow more precision in control of max and min PID input parameters for valves. Heater PID control performed in similar way, except that output PID control values are sent to *Cryo*, which has other conditions to complete before heaters are energized.

TABLE I cont. Function list in PLC programs.

	cPID_Overhead	Jumps to *PV_Assigner* to check if user changed name of current input parameters for PID control. When value is received from *PV_Assigner*, it updates name and value of PID input parameter. Checks control mode selected for valve (normal or manual). Returns these variables to *cPID_Setup*.
	PV_Assigner	Looks through PV array, which is populated with variables shared between Torus, DBX, and Solenoid PLCs. Each variable has a name with a value assigned. Current input PID parameter name entered for valve is compared with name of each element in array. When exact match in entered name and one of elements in array is found, value is copied in a reserved tag, which then is sent to *cPID_Overhead*. If this routine does not find a match between name entered and PV array elements, routine assigns previous value as input PID parameter.
Cryocon (DBX)	Scheduler	Sets watchdog counter between Solenoid PLC and Cryocon 18i units. Checks communication errors with 490NBX module and resets communications if needed. Jumps to *Crycon* if there is no communication error.
	Cryocon	Sends commands to read temperature data from three Cryocon 18i units every 250 ms. Ensures that ASCII commands are sent in correct format. Receives data from Cryocon units in assigned tag that handles string data. Copies responses in placeholder and clears last response tag for next response from Crycon unit. Sends this data to *StringParse* for converting string to float. Once it receives converted data, displays temperature values individually for each channel. Each temperature readout is available to use in local DBX PLC programs and to send to Solenoid or Torus PLCs as needed.
	StringParse	Takes input response from *Crycon*, splits string data type and converts values to float numbers and returns temperature to be read as numbers in *Cryocon*.
cPID_Control	Sol_Cooldown_ Comparisons	Reads cooldown variables from Solenoid PLC through shared PV array, compares and sets thresholds for each cooldown variable. Reads load cell interlocks enable variable from Solenoid PLC. If any cooldown signal exceeds thresholds, timer delay is enabled and it waits 90 s before enabling boolean tag that is sent to *Valve_Interlocks*.
	Valve_Interlocks	Checks inputs received from *Sol_Cooldown_Comparisons*. If interlock is enabled, sets related valves used in DBX box to warm and cold He to be closed by setting zero as output values for each valve.

TABLE I cont. Function list in PLC programs.

Instrument	Control/monitor	Monitored input signal	Signal qty
combined vacuum gauge	Pfeiffer TPG362/PLC	0–10 V	2
turbo pump speed	Pfeiffer DCU 002/ PLC	0–10 V	1
pneumatic valve	Pfeiffer TPG362/PLC	24 VDC	1
guard vacuum pump	MOOG/PLC	4–20 mA	1
pressure transducer	PLC	4–20 mA	1

TABLE II. Monitored signals of Solenoid vacuum.

Signal name	Description	Threshold limit [lb]
ZS86101US_BR_B	Solenoid US BR axial support lower	1510
ZS86102DS_BR_B	Solenoid DS BR axial support lower	190
ZS86103DS_BR_T	Solenoid DS BR axial support upper	650
ZS86104US_BR_T	Solenoid US BR axial support upper	1750
ZS86105US_BL_T	Solenoid US BL axial support upper	1750
ZS86106DS_BL_T	Solenoid DS BL axial support upper	650
ZS86107DS_BL_B	Solenoid DS BL axial support lower	190
ZS86108US_BL_B	Solenoid US BL axial support lower	1510
RS86101US_BR_B	Solenoid US BR radial support lower	14630
RS86102DS_BR_B	Solenoid DS BR radial support lower	14610
RS86103DS_BR_T	Solenoid DS BR radial support upper	1680
RS86104US_BR_T	Solenoid US BR radial support upper	1690
RS86105US_BL_T	Solenoid US BL radial support upper	1690
RS86106DS_BL_T	Solenoid DS BL radial support upper	1680
RS86107DS_BL_B	Solenoid DS BL radial support lower	14610
RS86108US_BL_B	Solenoid US BL radial support lower	14630

TABLE III. Solenoid load cell thresholds

Signal name	Description	Interlock limit	Interlock action
C14_DT_MAX	maximum differential temp. of coils 1–4	>25 K	Close PV8563C and PV8563W
C14ASY_DT_MAX	maximum differential temp. of coils 1–4, bobbin and cooling plate	>25 K	Close PV8563C and PV8563W
CM_DT_MAX	cold mass differential temp.	>40 K	Close PV8563C and PV8563W
HE_METAL_CD_DT	difference between inlet He temp. (TR8610) and He metal temp. (for cooldown) at SST	>-55 K	Close PV8563C and PV8563W
HE_METAL_WU_DT	difference between inlet He temp. (TR8610) and He metal temp. (for warmup) at SST	>55 K	Close PV8563C and PV8563W
HE_METAL_CD_DT2	difference between inlet He temp. (TD8513S) and He metal temp. (for cooldown) (at DBX reservoir output)	>-55 K	Close PV8563C and PV8563W
HE_METAL_WU_DT2	difference between inlet He temp. (TD8513S) and He metal temp. (for warmup) (at DBX reservoir output)	>55 K	Close PV8563C and PV8563W
LL8620SC	liquid level in lead reservoir	<20%	Disable heater HTR8620
LL8670SC	liquid level in magnet reservoir	<20%	Disable heater HTR8672

TABLE IV. Solenoid cooldown interlock signals.