

```

1  /*
2  * File:   menu.h
3  * Author: ricch
4  *
5  * Created on August 30, 2023, 10:17 AM
6  */
7
8  #ifndef MENU_H
9  #define MENU_H
10
11  #ifdef __cplusplus
12  extern "C" {
13  #endif
14
15      #include <stdbool.h>
16      #include <stdint.h>
17      #include <stepperDriver.h>
18      #include "Mc32NVMUtil.h"
19
20      #define RIGHT_ARROW 0x10
21
22      // Enumerations
23
24      /* All menus */
25      typedef enum
26      {
27          MAIN_MENU = 0,
28          CAPTURE_MODE_MENU,
29          SETTINGS_MENU,
30          ABOUT_MENU,
31          MOTOR_MENU,
32          MANUAL_MODE_MENU,
33          LIGHT_MENU,
34          BACKLIGHT_MENU,
35          CAMERA_MENU,
36          SAVE_DATA_MENU,
37          AUTO_HOME_MENU,
38          AUTOMATIC_MODE_MENU,
39
40      } MENU_STATE;
41
42
43
44
45      typedef enum{
46
47          RETURN_SEL = 0,
48
49      } COMMON;
50
51      typedef enum{
52
53          CAPTURE_MODE_SEL = 0,
54          SETTINGS_SEL,
55          ABOUT_SEL,
56
57      } MAIN_MENU_LIST;
58
59      typedef enum{
60
61          LIGHT_INTENSITY_SEL = 1,
62          LIGHT_TIME_SEL,
63          TIME_BW_PICTURES,
64
65      }LEDS_MENU_LIST;
66
67      typedef enum{
68
69          AUTO_HOME_START_SEL = 1,

```

```

70
71     } AUTO_HOME_MENU_LIST;
72
73     typedef enum{
74
75         MANUAL_MODE_SEL = 1,
76         AUTOMATIC_MODE_SEL,
77
78     } CHOICE_SEQ_MENU_LIST;
79
80     typedef enum{
81
82         AUTO_HOME_SEL = 1,
83         ANGLE_SEL,
84
85     } MANUAL_MODE_MENU_LIST;
86
87     typedef enum{
88
89         AUTOMATIC_MODE_START_SEL = 1,
90
91     } AUTO_MODE_MENU_LIST;
92
93     typedef enum{
94
95         MOTOR_SEL = 1,
96         LEDS_SEL,
97         BACKLIGHT_SEL,
98         CAMERA_SEL,
99         SAVE_DATA_SEL,
100
101     } SETTINGS_MENU_LIST;
102
103     typedef enum{
104
105         SPEED_SEL = 1,
106         GEAR_SEL,
107         STEP_PER_TURN_SEL,
108         POWER_SEL,
109
110     } MOTOR_MENU_LIST;
111
112     typedef enum{
113
114         BACKLIGHT_INTENSITY_SEL = 1,
115
116     } BACKLIGHT_MENU_LIST;
117
118     typedef enum{
119
120         EXPOSURE_TIME_SEL = 1,
121         TIME_BW_PICTURES_SEL,
122
123     } CAMERA_MENU_LIST;
124
125
126
127     typedef enum{
128
129         ANGLE_MODIF = 0,
130         SPEED_MODIF,
131         GEAR_MODIF,
132         STEP_PER_TURN_MODIF,
133         POWER_MODIF,
134         BL_INTENSITY_MODIF,
135         LIGHT_INTENSITY_MODIF,
136         LIGHT_TIME_MODIF,
137         EXPOSURE_TIME_MODIF,
138         TIME_BW_PICTURES_MODIF,

```

```

139
140     SAVE_DATA_START,
141     AUTO_HOME_START, // INTERACT
142     AUTOMATIC_MODE_START,
143
144 } MODIF_LIST;
145
146
147
148 // Structures
149 typedef struct{
150
151     uint8_t menuPage;
152     uint8_t menuSize;
153     MENU_STATE menuState;
154     MODIF_LIST modifState;
155
156 } MENU;
157
158
159
160
161
162 // Prototypes
163 void printLcdInit(void);
164 void printMainMenu(void);
165 void printParameterMenuPage0(void);
166 void printParameterMenuPage1();
167 void printMotorMenu0(STEPPER_DATA *pStepperData);
168 void printMotorMenu1(STEPPER_DATA *pStepperData);
169 void printLedsMenu(void);
170 void printChoiceSeqMenu(void);
171 void printAboutMenu(void);
172 void printManualModeMenu(STEPPER_DATA *pStepperData);
173 void printAutoModeMenu(STEPPER_DATA *pStepperData);
174 void printAutoHomeMenu(void);
175 void printBackLightMenu(void);
176 void printCameraMenu(void);
177 void printSaveDataMenu(void);
178
179 void menuManagementProcess(void);
180 void menuActionProcess(int32_t pec12RotationValue);
181 void menuDataProcess(int32_t *pec12RotationValue, STEPPER_DATA *pStepperData);
182 void menuPrintProcess(STEPPER_DATA *pStepperData);
183
184 void clearFirstRow(void);
185 void printCursor(int32_t cursor);
186
187 bool saveDataInEeprom(STEPPER_DATA *pStepperData);
188 bool readDataFromEeprom(STEPPER_DATA *pStepperData);
189
190
191 #ifdef __cplusplus
192 }
193 #endif
194
195 #endif /* MENU_H */
196
197

```