



基于LKS32MC08x的扫地机解决方案



扫地机主吸方案特点

- 采用单电阻无感FOC控制；
- 启动可靠稳定，基于特有启动算法，零速闭环启动，各电压下启动稳定，100%启动成功率；
- 高转速，运行电转速可达22万转/分钟；
- 保护机制齐全，过欠压、硬件过流、软件过流、缺相、堵转、限速；
- 芯片集成了3P3N预驱，5VLDO、两路运放、两路比较器，集成程度高，节省BOM成本；



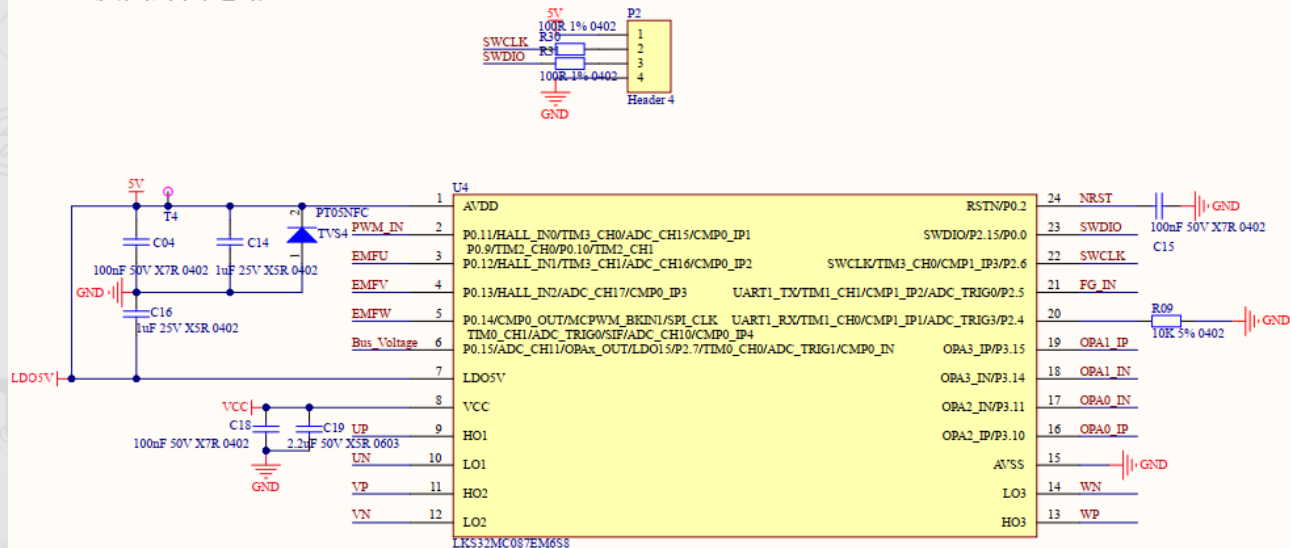
扫地机主吸方案硬件电路设计 - 1

凌鸥创芯

围绕产业链 · 打造创新链

- 芯片采用24Pin的LKS32MC087EM6S8芯片。

MCU及其外围电路

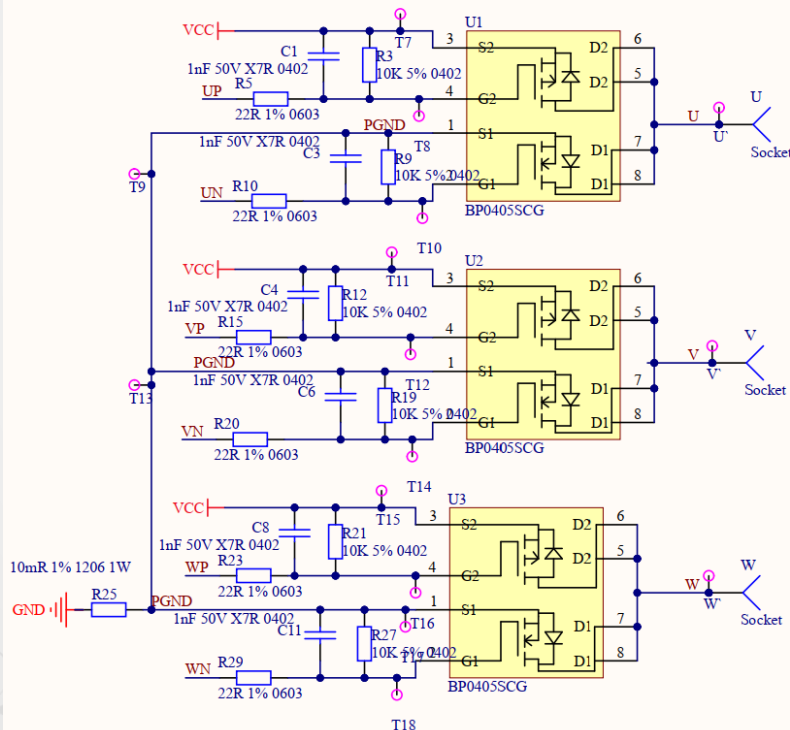




扫地机主吸硬件电路设计 - 2

- 因为扫地机主吸的电机功率一般100W以内，驱动用3P3N的功率管可以满足。驱动电路如下。

三相全桥功率部分

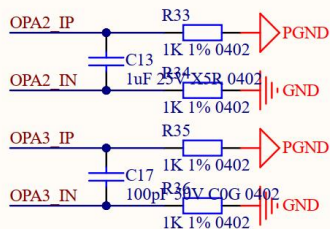




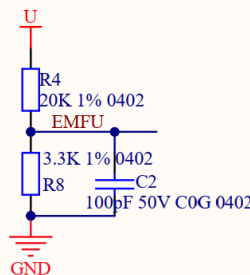
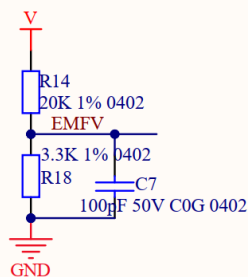
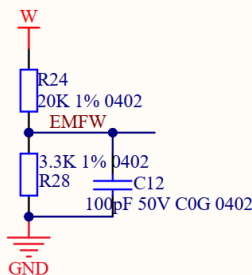
- 两路电流采样，一路不加滤波电容用作母线电流的峰值采样，一路加了1uF的滤波电容用作母线电流的均值采样，做恒功率控制。
- 三路反电动势检测电路，用于判断电机状态是否静止，计算电机滑行状态下的运行角度和速度。

电流采样电路

靠近MCU端放置



反电动势过零点估算角度





扫地机主吸方案软件框架

硬件模块初始化: GPIO、ADC、MCPWM、
定时器、比较器、DAC、运放

软件初始化: 观测器、电流环、功率环、
速度环、全局变量

1ms时基

22Khz的ADC中断时基

获取:
母线电压、
母线电流、
输出功率、
电机转速、
PWM占空比

各种保护逻辑:
过欠压、
短路过流、
过温、
超速、
堵转、
缺相

状态机:
电机状态检测、
功率环、
速度环、
弱磁控制、
停机刹车、
堵孔状态检测、
升降功率处理

FOC算法:
电流采样、
坐标变换、
电流环运算、
角度估算、
PWM输出

快速响应事件:
软件过流保护、
快速欠压保护、
FG信号输出

➤ 匹配硬件设置：PWM载波频率、死区时间、ADC电流采样稳定时间、电压电流采样通道、采样电阻值、电机参数、电机额定电压电流设定。

```

59  MC_Parameter.h hardware_config.h
100 /* ADC相电流采样时序。硬件相关 */
101 /* Porting Application Notice 注意采样序列 */
102 #define ADC0_CURRENT_A_CHANNEL (ADC0_CHANNEL_OPA2)
103 #define ADC0_CURRENT_B_CHANNEL (ADC0_CHANNEL_OPA2)
104 #define ADC0_CURRENT_C_CHANNEL (ADC0_CHANNEL_OPA2)
105
106 #define ADC_1SHUNT_CURR_CH (ADC0_CHANNEL_OPA3) /* 单电阻采样电流通道 */
107
108 #define ADC_BUS_VOL_CHANNEL (ADC_CHANNEL_11) /* 母线电压ADC采样通道 */
109 #define M0_ADC_BUS_CURR_CH (ADC0_CHANNEL_OPA3) /* 母线电流ADC采样通道 */
110
111 #define ADC_IBUS_CHANNEL (ADC0_CHANNEL_OPA2) /* 母线电流采样通道 */
112
113 #define ADC0_TEMP_CHANNEL (ADC_CHANNEL_7) /* 温度检测 */
114 #define ADC0_NTC_CHANNEL (ADC_CHANNEL_12) /* NTC温度检测 */
115
116 #define BEMF_CH_A ADC_CHANNEL_10/ADC_CHANNEL_10 /* A相反电势检测ADC通道 */
117 #define BEMF_CH_B ADC_CHANNEL_16/ADC_CHANNEL_17 /* B相反电势检测ADC通道 */
118 #define BEMF_CH_C ADC_CHANNEL_17/ADC_CHANNEL_16 /* C相反电势检测ADC通道 */
119
120 /* 电流内环相电流采样，ADC通道采样结果宏定义 */
121 #if (CURRENT_SAMPLE_TYPE == CURRENT_SAMPLE_1SHUNT)
122 #define GET_ADC0_DATA(value) (value.nData0 == (s16)ADC0_DAT0;)
123 #define GET_ADC1_DATA(value) (value.nData0 == (s16)ADC0_DAT2;)
124 #else

```



扫地机主吸方案程序介绍 - 2

凌鸥创芯

围绕产业链 · 打造创新链

- 状态机说明：主要用到IDLE空闲状态，BEMF_CHECK电机顺逆风检测状态，OPEN_RUN电机低速运行状态，RUN电机高速运行和功率环控制状态，FAULT故障状态。

```
89 void Sys_State_Machine(void)
90 {
91     switch (struFOC_CtrProc.eSysState)
92     {
93         case IDLE: // 空闲状态
94             /* 开机指令使能，判断母线电压在启动电压范围内，执行变量初始化后，进入下个状态 */
95             break;
96         case CHARGE:
97             break;
98         case BEMF_CHECK:
99             StateBemfDirCheck(); // 反电势检测函数
100             break; // 判断电机状态，选择进行静止启动还是顺风启动模式
101         case DIR_CHECK: // 顺逆风检测状态
102             break;
103         case INIT:
104             /* 初始化状态 */
105             break;
106         case POS_SEEK: // 初始位置检测状态
107             break;
108         case ALIGN: // 预定位状态
109             break;
110         case OPEN_RUN: // 低速时运行的状态，只跑电流环 // 开环强拖状态
111             break;
112         case RUN: // 运行状态
113             /* 电机速度起来后运行的状态，会运行功率环和速度环 */
114             break;
115         case HALL_RUN:
116             break;
117         case HALL_LEARN:
118             break;
119         case BRAKE:
120             /* 电机刹车状态 */
121             break;
122         case BEMF_BRAKE: /* 反电势刹车处理 */
123             break;
124         case WAIT:
125             /* 等待状态 */
126             break;
127         case FAULT: // 故障状态
128             /* 发生故障进入的状态 */
129             break;
130     }
131 }
```



扫地机主吸方案程序介绍 - 3

凌鸥创芯

围绕产业链 · 打造创新链

➤ 静止启动和顺风启动参数:

启动电流一般设置: 3A~20A;

观测器切换速度一般设置: 50~200Hz;

启动PLL_KP一般设置: 3~100;

启动PLL_KI一般设置: 3~100

```
103 /* 启动参数 */
104 #define START_CURRENT (7.0) /* 单位: A, 启动电流 */
105 #define START_CURRENT2 (1.5) /* 单位: A, 启动电流 */
106 #define FREQ_SWITCH (130.0)/(4.0) /* 单位: Hz, 观测器切换速度 */
107 #define FREQ_BRAKE (300.0) /* 单位: Hz, 刹车处理 */
108
109 #define PLL_KP_GAIN_START (7.5)/(1.1) /* 启动PLL_KP */
110 #define PLL_KI_GAIN_START (16)/(20)/(20) /* 启动PLL_KI */
111
112 #define OBS_MIN_OUT_SPEED (50)/(45) /* 观测器最小输出电频率, 单位: Hz */
113 #define OBS_MAX_OUT_SPEED (18000) /* 观测器最大输出电频率, 单位: Hz */
114
115 #define START_DIVISOR (6.0) /* 启动参数--移位 */
116 #define START_OBSFACTOR (40.0) /* 启动参数--系数 */
117
118 #define OBS_COEF (0.2) /* 观测器滤波系数 0~1 0.11 */
119 #define OPEN_RUN_STATUS (FALSE) /* 开环状态 TRUE = 开环运行 */
120
121 #define STATE04_WAITE_TIME (80) /* Unit: ms 速度变量初始化时间 */
122
123 /* ----- 环路选择 ----- */
124 #define CURRENT_LOOP (0) /* 电流环 */
125 #define SPEED_LOOP (1) /* 速度环 */
126 #define POWER_LOOP (2) /* 功率环 */
127 #define VOLT_LOOP (3) /* 电压环 */
128 #define TORQUE_LOOP (4) /* 转矩环 */
129 #define CLOSE_LOOP (2) /* 环路选择 */
130
131 /* ----- 观测器PLL PI参数 ----- */
132 #define PLL_KP_GAIN (40)/(40) /* PLL Kp 估算器Kp 60 */
133 #define PLL_KI_GAIN (100)/(100) /* PLL Ki 估算器Ki 120 */
134
135 /* ----- 反电势检测参数 ----- */
136 #define BEMF_CHECK_TIME (50) /* 单位: ms 反电势检测时间 */
137 #define BEMF_TRACK_FREQ (1400) /* 单位: Hz 反电势检测频率 */
138 #define BEMF_TRACK_FREQ1 (300) /* 单位: Hz 反电势检测频率 */
139 #define SPEED_TRACK_ON_FREQ_THH (250) /* 单位: Hz 顺风切闭环频率 */
140
141 #define BEMF_TRACK_VOLTAGE (2.0) /* 单位: V 顺逆风检测电压 */
142 #define BEMF_TRACK_VOLTAGE2 (4.0) /* 单位: V 顺逆风检测电压 */
143 #define BEMF_TIME (400) /* 单位: ms 反电势检测时间 */
144 #define BEMF_DUTY (FRAC16(0.3)) /* 刹车占空比 */
145
146 /* ----- 顺逆风检测参数 ----- */
147 #define SPEED_TRACK_DELAYTIME (50) /* 单位: ms 顺逆风检测时间 */
148 #define SPEED_TRACK_ON_FREQ_THH (60) /* 单位: Hz 顺风切闭环频率 */
149 #define EBRK_ON_FREQ_THH (80) /* 单位: Hz 逆风刹车频率 */
150
151 #define MOTOR_STOP_CUR_THD (120) /* 电机停止检测电流阈值 */
152 #define MOTOR_STOP_CUR_DIF_THD (12) /* 电机停止检测电流差阈值 */
153
154 #define STOP_TIME (10) /* 单位: ms 电机停止检测时间 */
155 #define STOP_DELAY_TIME (20) /* 单位: ms 电机停止后延迟时间 */
```



- 环路控制：电流环作为内环，功率环和速度环作为外环。

```
218 修改日期      版本号      修改人      修改内容
219 2020/8/5      V1.0        Howlet Li  创建
220
221 *****/
222 void ModuCircle_Limitation3(void); 电流环给定      电流环反馈
223 inline void CurrentLoopReg(stru_FOC_CurrLoopDef *this)
224 {
225     this->mPI_Torque.wInError = this->nQCur_Reference - this->mStatCurrDQ.nAxisQ; /* Q轴输入误差 */
226     this->mPI_Flux.wInError = this->nDCur_Reference - this->mStatCurrDQ.nAxisD; /* D轴输入误差 */
227
228     /* 误差限幅, 可根据实际负载调整, 影响PI响应的快慢 */
229     this->mPI_Torque.wInError = sat(this->mPI_Torque.wInError, -0x800, 0x800);
230     this->mPI_Flux.wInError = sat(this->mPI_Flux.wInError, -0x800, 0x800);
231
232     this->mStatVoltDQ.nAxisQ = CurrentPIRegulator(&this->mPI_Torque); /* Q轴电流PI计算, 输出Vq */
233     this->mStatVoltDQ.nAxisD = CurrentPIRegulator(&this->mPI_Flux); /* D轴电流PI计算, 输出Vd */
234
235     ModuCircle_Limitation3(); /* 电压极限限制 */
236 }
237
238 *****
239 函数名称:      void SpeedLoopReg(MechanicalQuantity *this)
240 功能描述:      速度环PI计算
241 输入参数:      MechanicalQuantity *this 结构体指针
242 输出参数:      无
243 返回值:      无
244 其它说明:
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304 2020/8/5      V1.0        Howlet Li  创建
305 *****/
306 void PowerLoopReg(stru_PowerBasic_t *this)
307 {
308     this->struPowerPI.wInError = this->wPowerRef - this->wPowerValue;
309
310     /* 误差限幅, 可根据实际负载调整, 影响PI响应的快慢 */
311     this->struPowerPI.wInError = sat(this->struPowerPI.wInError, -1000, 1000);
312
313     struFOC_CurrLoop.nQCurrentSet = PowerPIRegulator(&this->struPowerPI);
314 }
315
316 *****
317 * 函数名 : void PowerLoopInit(void)
318 * 说明 : 功率计算变量初始化
319 * 设计思路 : 1. 变量初始化
320 * 参数 : 无
321 * 返回值 : 无
```



- 控制板输入信号和输出信号：一般用PWM占空比作为输入信号，FG信号频率作为电机转速输出信号。

```
Project: LK_StdPeriph
  Linko 08x Project
    Include
      function_config.h
      hardware_config.h
      MC_Parameter.h
    AppFunction
      Global_Variable.c
      hardware_init.c
      Task Scheduler.c
      interrupt.c
      System_Init.c
      state_machine.c
      fault_detection.c
      Process_Control.c
      PowerCalculation.c
      SpeedScan.c
      KeyScan.c

state_machine.c
MC_Parameter.h
FOC_Drive.c
PowerCalculation.c
Task Scheduler.c

82 struFOC_CtrProc.eSysState = FAULT;
83 Recover_FaultCnt = 0;
84
85 else if( stru_Faults.R == 0 )
86 {
87     if( ( Recover_FaultCnt++ ) >= 120000 ) // 2min无故障则清零5次计数
88     {
89         Recover_FaultCnt = 0;
90         Error_CNT.SoftCurrentErrorC = 0;
91         Error_CNT.HardCurrentErrorC = 0;
92         Error_CNT.SpeedErrorC = 0;
93         Error_CNT.PhaseLossErrorC = 0;
94     }
95
96     PWMSpeedScan2(); //调整转速
97     Sys_State_Machine(); /* 电机0状态机调度 */
98     MotorRapSwitch();
99     /*****转速获取*****/
100     if( struFOC_CtrProc.bObs_Type == 1 )
101     {
102         MotorRap = Core2AppFreqTrans(struMotorSpeed.wSpeedEst) * 3 / 5 / U_MOTOR_PP; //
```

```
137 if(FG_VERSION_FLAG == 0)
138 {
139     if(struFOC_CtrProc.bMC_RunFlg)
140     {
141         if( MotorRap > 1000 )
142         {
143             FGScan( struFluxOB_Param.wElectAngle >> 16 );
144             nTheta_simu = struFluxOB_Param.wElectAngle >> 16;
145         }
146         else if( run_flag == 1 )
147         {
148             nTheta_simu = nTheta_simu + 218; //100hz*65535/(1s/50us)=327.675-100hz*65535/(1s/35us)=234
149             FGScan(nTheta_simu);
150         }
151     }
152 }
```



扫地机主吸方案程序介绍 - 6

凌鸥创芯

围绕产业链 · 打造创新链

- 故障检测保护、恢复、掉电保存：硬件过流、软件过流、快速欠压保护、过欠压保护、缺相保护、堵转保护。

```
241 void DAC_Init(void)
242 {
243     SYS_AnalogModuleClockCmd(SYS_AnalogModule_DAC, ENABLE);
244     SYS_WB_PROTECT = 0x7a83;
245     SYS_AFE_REG1 |= ~(0x3210x03 << 6);
246     SYS_AFE_REG1 |= (DAC_RANGE_1V2 << 6); /* 设置DAC量程程为1.2V: 00:3V1 01:1.2V 10:4
247     if(((SYS_AFE_REG1 >> 6) & 0x03) == DAC_RANGE_1V2)
248     {
249         /* 加载DAC 1.2V量程校正值 */
250         SYS_AFE_DAC_DC = Read_Trim(0x00000344);
251         SYS_AFE_DAC_AMC = Read_Trim(0x00000334);
252     }
253     else if(((SYS_AFE_REG1 >> 6) & 0x03) == DAC_RANGE_1V0)
254     {
255         /* 加载DAC 3.0V量程校正值 */
256         SYS_AFE_DAC_DC = Read_Trim(0x00000340);
257         SYS_AFE_DAC_AMC = Read_Trim(0x00000330);
258     }
259     else if(((SYS_AFE_REG1 >> 6) & 0x03) == DAC_RANGE_4V85)
260     {
261         /* 加载DAC 1.0V量程校正值 */
262         SYS_AFE_DAC_DC = Read_Trim(0x00000340);
263         SYS_AFE_DAC_AMC = Read_Trim(0x00000330);
264     }
265     SYS_AFE_DAC_30 = 0.01/1.2*4096; /* 1.2*850/4096/0.05 = 3.5A, 其中0.05为母线采样电阻
266     }
267 }
268
221 /* 过欠压检测参数 */
222 #define U_DCB_OVERVOLTAGE_FAULT (22.0) /* 单位: V 过压检测设定值 */
223 #define U_DCB_OVERVOLTAGE_RECOVER (22.0) /* 单位: V 过压恢复设定值 */
224 #define U_DCB_UNDERVOLTAGE_FAULT (9.0) /* 单位: V 欠压检测设定值 */
225 #define U_DCB_UNDERVOLTAGE_RECOVER (7.5) /* 单位: V 欠压恢复设定值 */
226 #define U_DCB_UNDERVOLTAGE_FAULT (10.0) /* 单位: V 欠压恢复设定值 */
227
228 /* 温度检测参数 */
229 #define TEMP_FAULT (5936) /* 过温检测设定值 */
230 #define TEMP_RECOVER (8000) /* 过温恢复设定值 */
231 #define TEMP_BREAK (32000) /* NTC开路设定值 */
232
233 /* 堵转检测参数 */
234 #define STALL_TIME_FAULT (200) /* 单位: Sms 堵转检测时间为5*3000=1.5s */
235 #define STALL_SPEED_MAX (2500.0) /* 单位: Hz 堵转口检测转速 */
236 #define STALL_SPEED_MIN (50.0) /* 单位: Hz 堵转口检测转速 */
237
619 // FaultStartCheck(&struFaultStartTime) //启动失败检测
620
621 case 4:
622 {
623     if((struFOC_CtrProc.eSysState == RUN) || (struFOC_CtrProc.eSysState == OPEN_RUN))
624     {
625         FaultPhaseCheck(&struImaxCurrent, &struCurrentAmplitude, &struFaultPhaseTime); //缺相检测
626     }
627     break;
628 }
629 case 5:
630 {
631     FaultSpeedCheck(MotorRap, &struFaultSpeedTime); //低速检测、堵转
632     break;
633 }
634 default:
635     break;
636 }
637 if(stru_Faults.R != 0)
638 {
639 }
```



➤ 测试硬件是否能正常输出PWM模式

```
29
30 /* -----存储地址相关定义----- */
31 #define HALL_LEARN_ADDR 0x10000 /* Hall学习存储地址 */
32
33 /* -----FOC功能相关定义----- */
34 #define MAX_MODULE_VALUE 21000//18918 /* PWM调制比 1/sqrt(3)*32767 */ //18918
35
36 /* -----功能相关定义----- */
37 #define TEST_ON 1
38 #define TEST_OFF 0
39
40 #define DEBUG_PWM_OUTPUT TEST_OFF /* PWM在调试的时候输出一定占空比 */
41
42 #define ALTERNATE_OUT_PWM TEST_OFF /* PWM输出调试, 交替输出Ch012 CH3 */
43
44 #define FUNCTION_ON 1
45 #define FUNCTION_OFF 0
46
47 #define FMSTR_FUNCTION FUNCTION_OFF /* FreeMaster 调试功能 */
48
49 #define RTT_FUNCTION FUNCTION_OFF /* RTT 调试功能 */
50
51 #endif /* __FUNCTION_CFG_PARAM_H */
52
53 /***** (C) COPYRIGHT LINKO SEMICONDUCTOR *****/
54 /* -----END OF FILE----- */
55
```



江苏省南京市经济技术开
发区兴智科技园B栋15层
<http://www.linkosemi.com>

為天地立心
為控制塑魂

创芯驱动，领航电控未来！

正直诚信！ 利他共赢！ 成长超越！