

リスト5-3-4 task2.cの追記部

```
40 // キューハンドル定義
41 QueueHandle_t LCDQueue;
42 QueueHandle_t MQTTQueue;
43 QueueHandle_t LOGQueue;
44
45 /** キュー送信用バッファ定義 ***/
46 union {
47     uint8_t Buf[20];
48     struct {
49         uint8_t kind;
50         float pres;
51         float temp;
52         float humi;
53     } mes;
54 } Measure;
55 /* グローバル変数、定数定義 */
56 uint32_t temp_cal, pres_cal, hum_cal, Interval;
57 extern uint8_t LogFlag;
58 portTickType LastTimeI;
```

ハンドル定義

キュー送信バッファの定義

```
133
134
135 case TASK2_STATE_SERVICE_TASKS:
136 {
137     /** センサ計測実行とキュー送信 **/
138     bme_getdata(); // データ取得
139     temp_cal = calib_temp(temp_raw); // 温度較正
140     pres_cal = calib_pres(pres_raw); // 気圧較正
141     hum_cal = calib_hum(hum_raw); // 湿度較正
142     /** 計測値実際の値にスケール変換 *****/
143     Measure.mes.kind = 'M'; // データ区別セット
144     Measure.mes.temp = (float)(temp_cal/100.0); // 温度変換
145     Measure.mes.pres = (float)(pres_cal/100.0); // 気圧変換
146     Measure.mes.humi = (float)(hum_cal/ 1024.0); // 湿度変換
147     /** キューへ送信 **/
148     xQueueSend(LCDQueue, Measure.Buf, 0); // キューへ送信
149     if(LogFlag == 1)
150     | xQueueSend(LOGQueue, Measure.Buf, 0);
151     // 5秒待ち // 5秒ごとに繰り返し
152     LastTimeI = xTaskGetTickCount();
153     vTaskDelayUntil(&LastTimeI, 5000/portTICK_RATE_MS);
154     /** 1分ごとにMQTT送信 **/
155     Interval++; // 1分ごとに繰り返し
156     if(Interval >= 12){
157         Interval = 0;
158         xQueueSend(MQTTQueue, Measure.Buf, 0); //MQTTに送信
159     }
160     break;
161 }
```