#include "variant.h"

#include <due\_can.h>

//Leave defined if you use native port

#define Serial SerialUSB

int x = 3000;

void setup()

{

Serial.begin(115200);

// Initialize CAN0 and CAN1, Set the proper baud rates here

Can0.begin(CAN\_BPS\_125K);

Can1.begin(CAN\_BPS\_125K);

//By default there are 7 mailboxes for each device that are RX boxes

//This sets each mailbox to have an open filter that will accept extended

//or standard frames

int filter;

//extended

for (filter = 0; filter < 3; filter++) {

Can0.setRXFilter(filter, 0, 0, true);

Can1.setRXFilter(filter, 0, 0, true);

}

//standard

//for (int filter = 3; filter < 7; filter++) {

//Can0.setRXFilter(filter, 0, 0, false);

//Can1.setRXFilter(filter, 0, 0, false);

//}

Serial.println("BEGIN");

}

void printFrame(CAN\_FRAME &frame) {

Serial.print("ID: 0x");

Serial.print(frame.id, HEX);

Serial.print(" Len: ");

Serial.print(frame.length);

Serial.print(" Data: 0x");

for (int count = 0; count < frame.length; count++) {

Serial.print(frame.data.bytes[count], HEX);

Serial.print(" ");

}

Serial.print("\r\n");

}

void loop(){

CAN\_FRAME incoming;

if (Can0.available() > 0) {

Can0.read(incoming);

Can1.sendFrame(incoming);

//printFrame(incoming); //uncomment line to print frames that are going out

}

if (Can1.available() > 0) {

Can1.read(incoming);

if (incoming.id == 0x09C050B8){

printFrame(incoming);

//Serial.print(incoming.data.bytes[3]);

x = x + 100;

if (x > 7000){x = 0;}

incoming.data.bytes[4] = (byte) (x & 0xFF);

incoming.data.bytes[3] = (byte) ((x >> 8) & 0xFF);

Serial.println(incoming.data.bytes[3] << 8 | incoming.data.bytes[4]);

}

Can0.sendFrame(incoming);

}

}