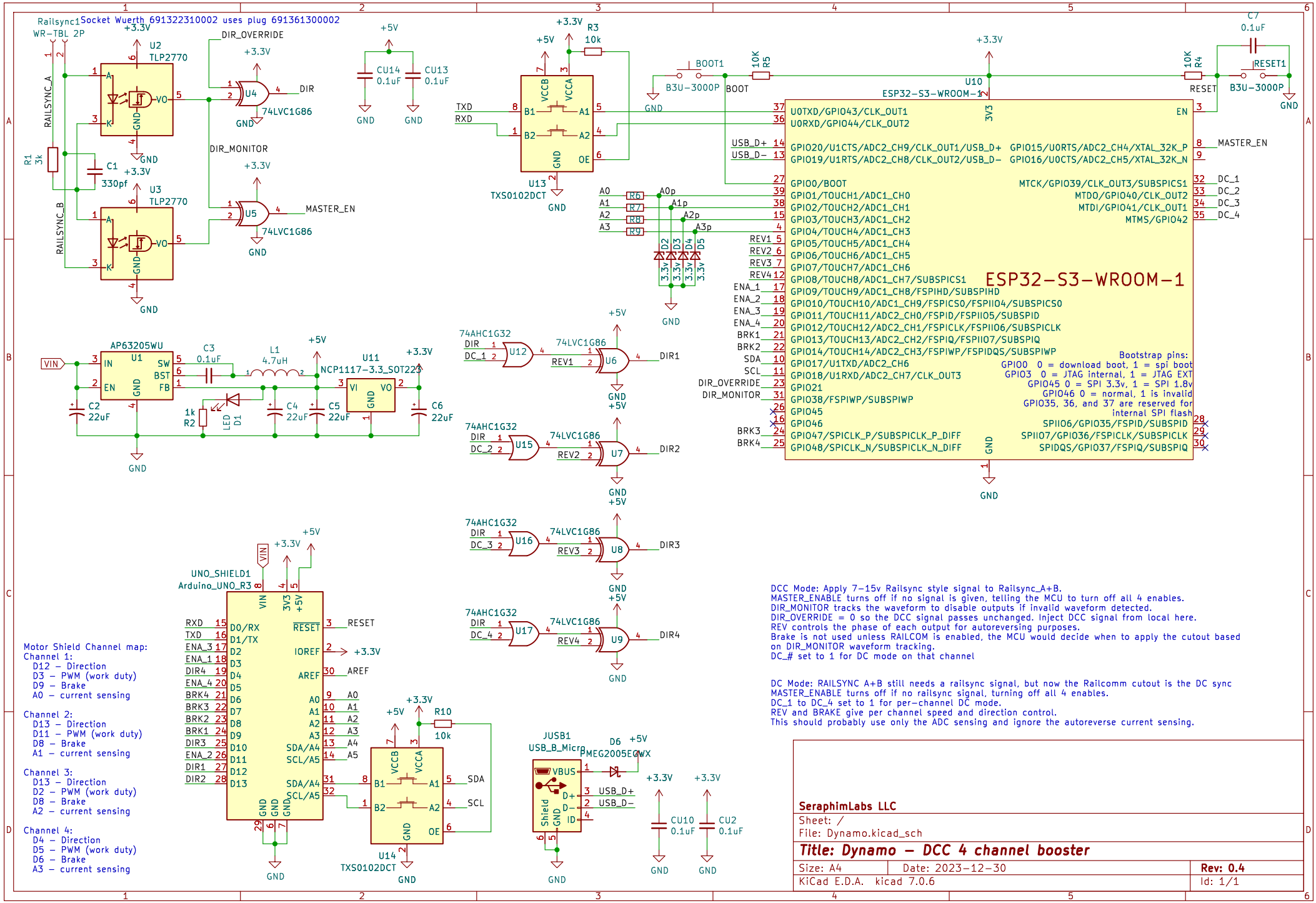
Railsync1 Socket Wuerth 691322310002 uses plug 691361300002
WR-TBL 2P
RAILSYNC_A
RAILSYNC_B
R1 3k
C1 330pf
U2 TLP2770
U3 TLP2770
DIR_OVERRIDE
DIR_MONITOR
U4 74LVC1G86
U5 74LVC1G86
DIR
MASTER_EN
CU14 0.1uF
CU13 0.1uF
R3 10k
U13 TXS0102DCT
TXD
RXD
BOOT1
B3U-3000P
BOOT
R5 10K
R4 10K
RESET
RESET1
C7 0.1uF
U10 ESP32-S3-WROOM-1
ESP32-S3-WROOM-1
U0TXD/GPIO43/CLK_OUT1
U0RXD/GPIO44/CLK_OUT2
GPIO20/U1CTS/ADC2_CH9/CLK_OUT1/USB_D+
GPIO19/U1RTS/ADC2_CH8/CLK_OUT2/USB_D-
GPIO15/U0RTS/ADC2_CH4/XTAL_32K_P
GPIO16/U0CTS/ADC2_CH5/XTAL_32K_N
GPIO0/BOOT
GPIO1/TOUCH1/ADC1_CH0
GPIO2/TOUCH2/ADC1_CH1
GPIO3/TOUCH3/ADC1_CH2
GPIO4/TOUCH4/ADC1_CH3
GPIO5/TOUCH5/ADC1_CH4
GPIO6/TOUCH6/ADC1_CH5
GPIO7/TOUCH7/ADC1_CH6
GPIO8/TOUCH8/ADC1_CH7/SUBSPICS1
GPIO9/TOUCH9/ADC1_CH8/FSPIHD/SUBSPIHD
GPIO10/TOUCH10/ADC1_CH9/FSPICS0/FSPIIO4/SUBSPICS0
GPIO11/TOUCH11/ADC2_CH0/FSPID/FSPIIO5/SUBSPID
GPIO12/TOUCH12/ADC2_CH1/FSPICLK/FSPIIO6/SUBSPICLK
GPIO13/TOUCH13/ADC2_CH2/FSPIQ/FSPIIO7/SUBSPIQ
GPIO14/TOUCH14/ADC2_CH3/FSPIWP/FSPIDQS/SUBSPIWP
GPIO17/U1TXD/ADC2_CH6
GPIO18/U1RXD/ADC2_CH7/CLK_OUT3
GPIO21
GPIO38/FSPIWP/SUBSPIWP
GPIO45
GPIO46
GPIO47/SPICLK_P/SUBSPICLK_P_DIFF
GPIO48/SPICLK_N/SUBSPICLK_N_DIFF
MTCK/GPIO39/CLK_OUT3/SUBSPICS1
MTDO/GPIO40/CLK_OUT2
MTDI/GPIO41/CLK_OUT1
MTMS/GPIO42
EN
USB_D+
USB_D-
REV1
REV2
REV3
REV4
ENA_1
ENA_2
ENA_3
ENA_4
BRK1
BRK2
SDA
SCL
DIR_OVERRIDE
DIR_MONITOR
BRK3
BRK4
MASTER_EN
DC_1
DC_2
DC_3
DC_4
Bootstrap pins:
GPIO0 0 = download boot, 1 = spi boot
GPIO3 0 = JTAG internal, 1 = JTAG EXT
GPIO45 0 = SPI 3.3v, 1 = SPI 1.8v
GPIO46 0 = normal, 1 is invalid
GPIO35, 36, and 37 are reserved for internal SPI flash
SPIIO6/GPIO35/FSPID/SUBSPID
SPIIO7/GPIO36/FSPICLK/SUBSPICLK
SPIDQS/GPIO37/FSPIQ/SUBSPIQ
A0 R6 A0p
A1 R7 A1p
A2 R8 A2p
A3 R9 A3p
D2 D3 D4 D5
AP63205WU U1
VIN
C2 22uF
C3 0.1uF
L1 4.7uH
NCP1117-3.3_SOT223 U11
C4 22uF
C5 22uF
C6 22uF
1k R2
LED D1
74AHC1G32 U12 DIR DC_1 REV1 74LVC1G86 U6 DIR1
74AHC1G32 U15 DIR DC_2 REV2 74LVC1G86 U7 DIR2
74AHC1G32 U16 DIR DC_3 REV3 74LVC1G86 U8 DIR3
74AHC1G32 U17 DIR DC_4 REV4 74LVC1G86 U9 DIR4
UNO_SHIELD1 Arduino_UNO_R3
RXD D0/RX
TXD D1/TX
ENA_3 D2
ENA_1 D3
DIR4 D4
ENA_4 D5
BRK4 D6
BRK3 D7
BRK2 D8
BRK1 D9
DIR3 D10
ENA_2 D11
DIR1 D12
DIR2 D13
RESET
IOREF
AREF
A0 A1 A2 A3 A4 A5
SDA/A4 SCL/A5
R10 10k
U14 TXS0102DCT
SDA
SCL
JUSB1 USB_B_Micro
VBUS
D+ D- ID
Shield GND
D6 PMEG2005EGWX
CU10 0.1uF
CU2 0.1uF
Motor Shield Channel map:
Channel 1:
D12 – Direction
D3 – PWM (work duty)
D9 – Brake
A0 – current sensing
Channel 2:
D13 – Direction
D11 – PWM (work duty)
D8 – Brake
A1 – current sensing
Channel 3:
D13 – Direction
D2 – PWM (work duty)
D8 – Brake
A2 – current sensing
Channel 4:
D4 – Direction
D5 – PWM (work duty)
D6 – Brake
A3 – current sensing
DCC Mode: Apply 7–15v Railsync style signal to Railsync_A+B.
MASTER_ENABLE turns off if no signal is given, telling the MCU to turn off all 4 enables.
DIR_MONITOR tracks the waveform to disable outputs if invalid waveform detected.
DIR_OVERRIDE = 0 so the DCC signal passes unchanged. Inject DCC signal from local here.
REV controls the phase of each output for autoreversing purposes.
Brake is not used unless RAILCOM is enabled, the MCU would decide when to apply the cutout based on DIR_MONITOR waveform tracking.
DC_# set to 1 for DC mode on that channel
DC Mode: RAILSYNC A+B still needs a railsync signal, but now the Railcomm cutout is the DC sync
MASTER_ENABLE turns off if no railsync signal, turning off all 4 enables.
DC_1 to DC_4 set to 1 for per-channel DC mode.
REV and BRAKE give per channel speed and direction control.
This should probably use only the ADC sensing and ignore the autoreverse current sensing.
SeraphimLabs LLC
Sheet: /
File: Dynamo.kicad_sch
Title: Dynamo – DCC 4 channel booster
Size: A4
Date: 2023-12-30
Rev: 0.4
KiCad E.D.A. kicad 7.0.6
Id: 1/1